

REFERENCES

- Brundtland, G.H., (1987). Our common future: The World Commission on Environment and Development. Oxford, Oxford University Press.
- Carvalho, A., Gani, R., Matos, H. (2008). Design of sustainable chemical processes: Systematic retrofit analysis generation and evaluation of alternatives. Process Safety and Environmental Protection, 86, 328-346.
- Hukkerikar, A.S., Kalakul, S., Sarup, B., Young, D.M., Sin, G., Gani, R. (2012). Estimation of Environment-Related Properties of Chemicals for Design of Sustainable Processes: Development of Group-Contribution⁺ (GC⁺) Property Models and Uncertainty Analysis. Chemical Information and Modeling , 56, 2823-2839.
- Mangnimit, S., (2013). Sustainable Process Design of Biofuels :Bioethanol Production from Cassava rhizome. M.S. thesis,The Petroleum and Petrochemical College, Chulalongkorn University, Bangkok, Thailand.
- Piyarak, S., (2012). Development of Software for Life Cycle Assessment. M.S. thesis,The Petroleum and Petrochemical College, Chulalongkorn University, Bangkok, Thailand.
- Saengwirun, P., (2011). Cost calculations and Economic analysis. M.S. thesis,The Petroleum and Petrochemical College, Chulalongkorn University, Bangkok, Thailand.
- Vigon, B.W., Tolle, D.A., Cornaby, B.W. and Latham, H.C. (1993). Life Cycle Assessment: Inventory Guidelines and Principles. Columbus, Ohio: The United States Environmental Protection Agency.
- U.S. Life Cycle Inventory Database. (2012). National Renewable Energy Laboratory (NREL).
<<https://www.lcacommons.gov/nrel/search>> Accessed on August 19, 2012.

APPENDICES

Appendix A LCSoft v 2.0 Manual

A.1 Introduction to life cycle assessment (LCA)

LCA is defined by the ISO 14040 and ISO 14044 as the evaluation of the inputs and outputs and the potential environmental impacts of a product system during the life time of a product or process. For product-process developers, LCA could be used to compare the entire range of the environmental damages that can be attributed to products or processes, and be able to choose the hot spots to develop or choose the least burdensome one among alternatives.

A.1.1 LCA methodology

According to the ISO 14040 and ISO 14044 standards, LCA consists of four steps as follows.

A.1.1.1 Goal and scope definition

This is the step to define the purpose and method of the project also describe the product, process or activity in the life cycle. Establish the context of the assessment and identify system boundaries. In this stage the items that must be determined are the type of information needed, how accurate the results must be and how the results should be interpreted and displayed.

A.1.1.2 Inventory analysis or life cycle inventory (LCI)

This is where all the necessary input and output data for the processes regarding the product system is gathered. These gathered data is related with the reference flow given by the functional unit. LCI data for the different processes is combined over the life cycle and presented as the total emissions of a substance or total resource consumption.

A.1.1.3 Impact assessment or Life cycle impact assessment (LCIA)

This is where the LCI data of input (materials and resources) and output (waste and emissions) are translated into information regarding the impacts the product system has on the environment, human health, and resources.

A.1.1.4 Interpretation

This is where the results from LCIA will be related with the goal of the study. In order to quantify the results sensitivity and uncertainty are also analyzed in this step.

A.2 Aspects concerning LCA

A.2.1 Functional unit

This is the quantified definition of the function of a product to guarantee the comparison between product systems that actually provide the same unit of quantity. For example, the function unit of a bioethanol production process is the production of 1 ton/day of ethanol.

A.2.2 System boundaries

This defines which processes will be included in the system. There are four main options to define the system boundaries as follows;

Cradle to grave: includes the material and energy production chain and all processes from the raw material extraction through the manufacturing, transportation and use phase up to the life time of the product. Life cycle of product and input/output material is shown in Figure 1.

Cradle to gate: includes all processes from the raw material extraction through the manufacturing phase (gate of the factory). This boundary is used to determine the environmental impacts of the production of a product.

Gate to gate: includes the processes from the production phase only. This boundary is used to determine the environmental impacts of a single production step or process.

Gate to grave: includes the processes from the use and end-of-life phases (everything post production of product phase). This boundary is used to determine the environmental impacts of a product once it leaves the factory.

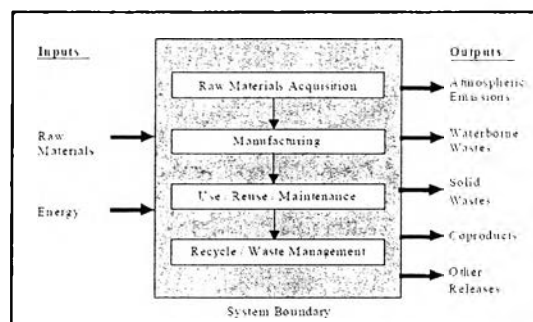


Figure A.1 System boundary of a product.

A.2.3 Allocation

This is defined in this ISO as: partitioning the input and/or output flows of a process to the product system under study, inputs and outputs of a process are allocated to the relevant product and byproducts. In general, products from chemical processes are allocated by mass. Allocation by energy when products of the processes are processes of fuels, the product heating values will be used.

A.3 Conducting LCA in LCSOft

In this chapter, the outline a step by step procedure for conducting an LCA in LCSOft is given below with the case study.

A.3.1 LCA case study: Bioethanol from Cassava rhizome

Bioethanol is a type of biofuel produced from lignocellulosic matter. This case study, lignocelluloses from cassava rhizome is converted to ethanol by hydrolysis via fermentation. The reference for the process is obtained from Wooley et al. (NREL) and simulated using PRO/II simulation program.

The overview of process flow sheet is described as follows and can be divided into 5 main sections (Figure A.2):

1. Pretreatment section, to make the lignocellulosic material enable to be hydrolyzed.
2. Detoxification, the compounds that are toxic for the fermentation microorganisms are treated.
3. Simultaneous saccharification and co-fermentation (SSCF), to convert glucose and other sugars to ethanol.
4. Distillation, to distil the process stream from SSCF section until a mixture of nearly azeotropic water and ethanol is obtained.
5. Dehydration, the ethanol from distillation section is purified using vapor-phase molecular sieves in order to obtain bioethanol within the specifications.

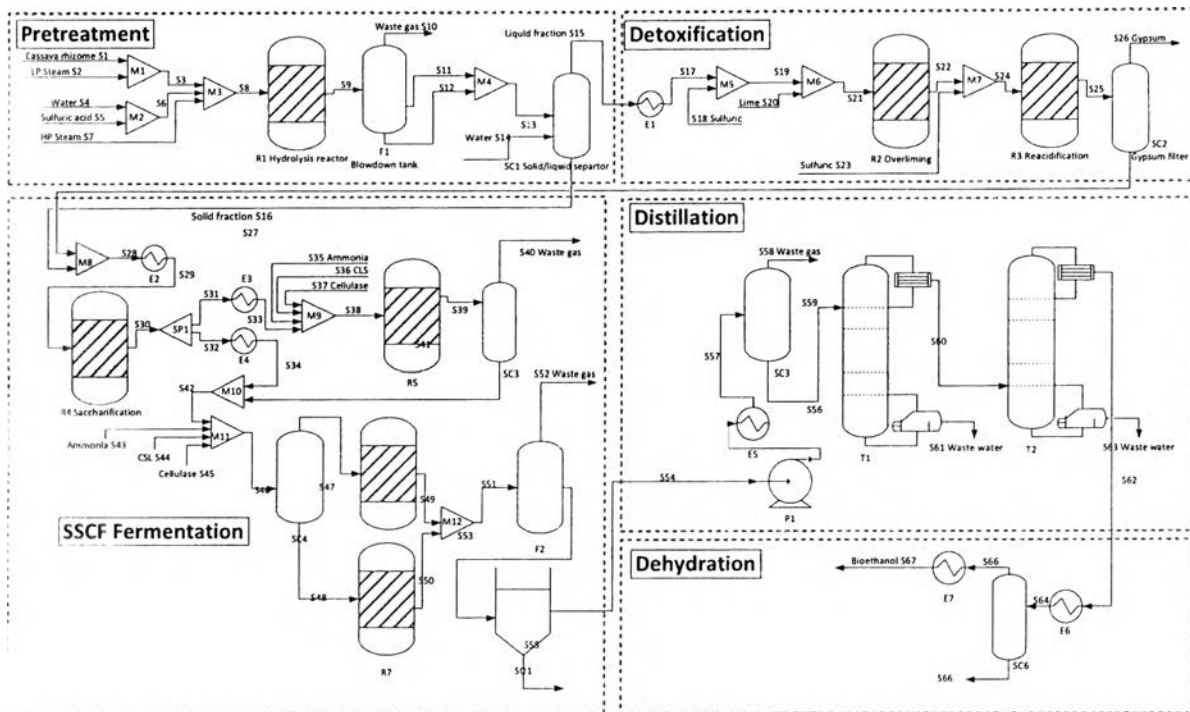


Figure A.2 Flowsheet of the base bioethanol production process implemented in PRO/II

A.4 Getting started with LCSofT

“LCSofT” folder composes of 4 folders and 1 excel file as shown in Figure A.3. “Tool 1” folder is for the user to generate new LCI data. “Projects” folder is where the excel files of LCA project are systematically stored. “LCI KB” folder is where LCSofT LCI databases are stored. “Extended LCI KB” folder is where the LCI data generated by the user is stored. “LC Soft v2.0” program file is the program file that we will use to conduct our LCA project. “USER MANUAL-LCSofT” file is LCSofT user manual.

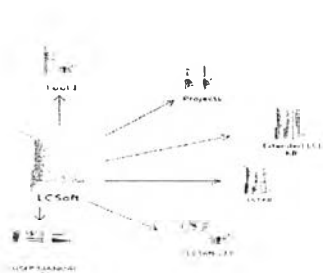


Figure A.3 LCSofT structure

A.4.1 Open/Close LCSoft

Step 1: Open “LCSoft” folder (Figure A.4). Step 2: Open “LCSoft v 2.0” *Excel* file, “Main menu” page will appear. LCSoft will be ready to conduct LCA. Step 3: After conducting LCA, to close LCSoft click “Save and close” button. LCSoft will be automatically saved in “Projects” folder and closed. Step 4: After using additional tool, to close LCSoft click “Save and close” button. LCSoft will be automatically saved in “Projects” folder and closed.

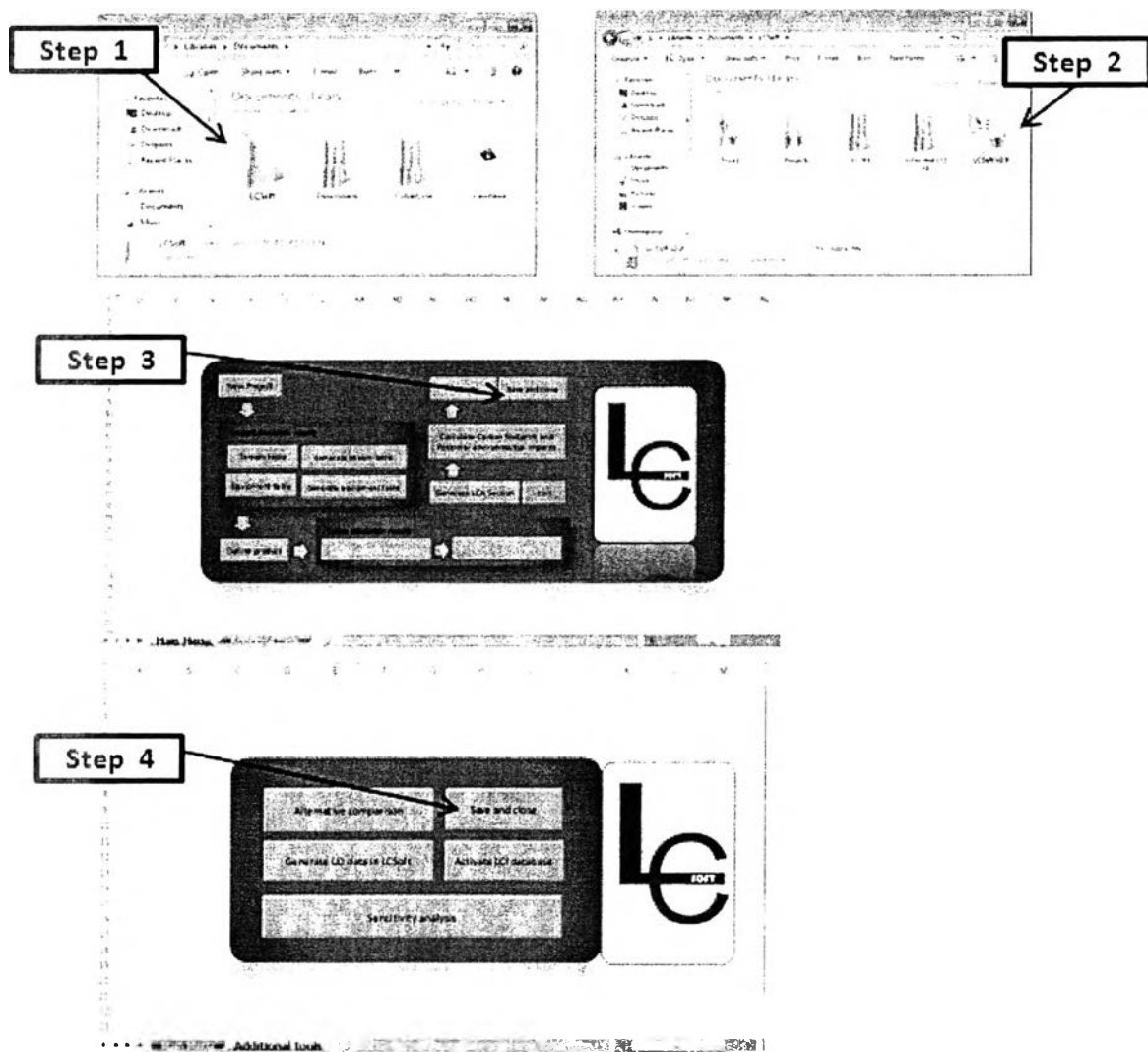


Figure A.4 Steps to open and close LCSoft

A.4.2 Creating new LCA project

Open LCSOFT v 2.0, “Main menu” page will appear as shown in Figure A.5. The user can complete conducting LCA by clicking and filling data following to the blue arrows. After finishing each part, the button will turn to blue color showing that data is complete and ready to do the next step.

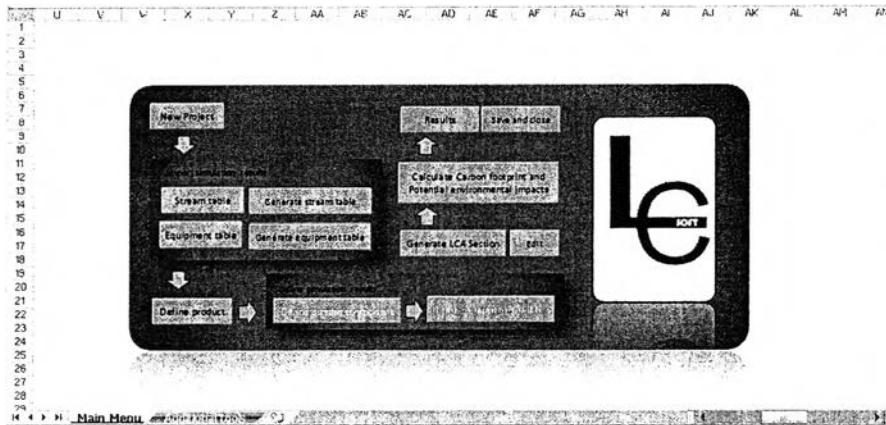


Figure A.5 Main menu page

Step 1: Click on “New project” button in LCSOFT main menu and “New project” form will appear as shown in Figure 6. Step 2: Type name of the project. Step 3: Type the location of the project (Optional). For example, in the case study is “Thailand”. Step 4: Select unit of mass. In the case study the unit of mass is “kg”. Therefore to import stream table the unit of mass must be “kg”. Step 5: Select unit of volume (Optional, just when the user want to do volumetric allocation. In general, allocation method is mass allocation). Step 6: Select unit of time. In the case study the unit of mass is “hr”. Therefore to import equipment table and stream table, the unit of time must be “hr”. Step 7: Select unit of energy. In the case study the unit of energy is “MJ”. Therefore the unit of energy related terms such as duty is MJ/hr. Step 8: Select unit of power. In the case study the unit of power is “kW”. Therefore the unit of power related terms such as work is kW. Step 9: Click “OK” button. The “New project” button will turn to blue color enabling to do the next step of conducting LCA in LCSOFT.

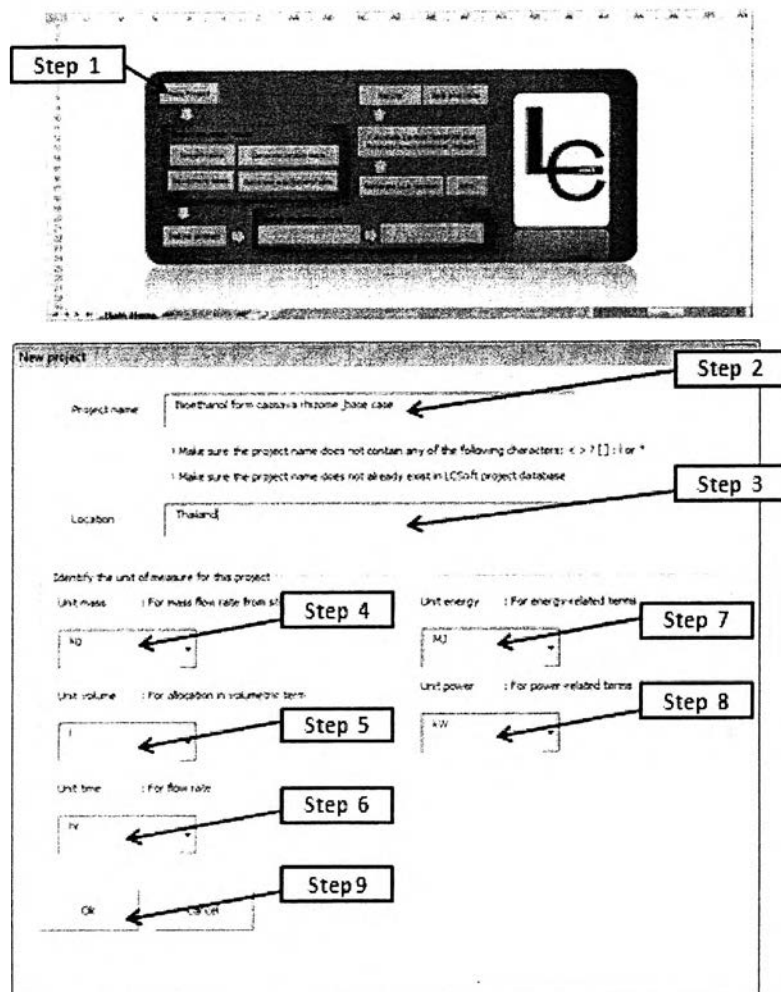


Figure A.6 Steps to create new project

A.4.3 Importing simulation results

LCSOFT provides options for using simulation results which are mass balance from stream table and energy balance from equipment table. (If the user does not have simulation results can skip this step.)

A.4.3.1 Import stream table

The stream table should be the same format with Figure A.7. The heading of stream component must be “Total Weight Comp.Rates”. The first stream name should be located at the first row and the first amount of the mass flow rate (4680.592 kg/hr) must be the same row with the first component (Cellulose).

- To see the stream table, open “LCSOFT” folder > “Projects” folder > “Case study data” folder > “stream table” file.

Stream Name	S1	S2	S3	S4	S5	S6	S7	S8	S9
Stream Description									
Stream Phase	Mixed	Vapor	Mixed	Liquid	Liquid	Liquid	Vapor	Mixed	Mixed
Temperature	30.000	160.000	100.018	25.000	25.000	25.000	268.000	188.002	190.000
Pressure	1.000	6.000	1.000	1.000	1.000	1.000	13.000	12.100	12.100
Total Molecular Weight	103.896	18.015	84.688	18.015	98.079	18.308	18.015	38.223	40.920
Total Weight Comp. Rates (kg/hr)									
Cellulose	4680.592	0.000	4680.592	0.000	0.000	0.000	0.000	4680.592	4320.186
Hemicellulose	6674.090	0.000	6674.090	0.000	0.000	0.000	0.000	6674.090	333.705
Lignin	3653.449	0.000	3653.449	0.000	0.000	0.000	0.000	3653.449	3653.449
Glucose	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	364.047
Xylose	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	6825.731
Cellulbiose	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	34.584
Ethanol	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Water	129.522	785.091	914.613	2972.114	0.000	4972.114	3153.382	9040.110	8273.831
Sulfuric Acid	0.000	0.000	0.000	0.000	99.441	99.441	0.000	99.441	99.441
Furfural	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	242.695
Ammonia	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Oxygen	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Carbon Dioxide	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Glycerol	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Succinic Acid	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Figure A.7 Stream table format

⇒ If the user does not have the stream table that follows the format mentioned above, follow the steps below to generate stream table.

Case study:

- There are 67 streams (S1-S67), 25 components (cellulose, hemicelluloses, lignin, glucose, xylose, cellulbiose, ethanol, water, sulfuric acid, furfural, ammonia, oxygen, carbon dioxide, glycerol, succinic acid, lactic acid, xylitol, acetic acid, corn steep liquor, ZM, cellulase, lime, CASO₄, and ash).
- Mass unit: kg (Figure A.6)
- Time unit: hr (Figure A.6)

Step 1: Click “Generate stream table” button as shown in Figure 8.

Step 2: The software will show the instructions to generate stream table, read it and click “Ok” button. The user will see the example of entering component and mass flow rate in the right hand side of “generate stream table” page.

Step 3: Click “Generate new table” button in “generate stream table” page.

Step 4: Enter the number of streams.

Step 5: Click “Ok” button.

Step 6: Enter the stream components and mass flow rates following to the given example.

- Make sure that the stream components are in the first column and mass flow rates are all numeric.
- The units of mass and time must be corresponding to the units that user have selected when create new project.

Step 7: Click “Done” button in “generate stream table” page to finish.

The figure illustrates the process of generating a steam table through several steps:

- Step 1:** A software interface showing a flowchart of the process. It includes buttons for 'Generate stream table', 'Calculate Carbon Footprint and Generate LCA Section', and 'Generate LCA Section'. The 'LE' logo is visible on the right.
- Step 2:** A Microsoft Excel spreadsheet with a text prompt: 'Please enter streams components and mass flow rates following the given example'. An arrow points to an 'Ok' button.
- Step 3:** A 'Userform' dialog box with 'Generate new table' and 'Done' buttons. An arrow points to the 'Generate new table' button.
- Step 4:** The 'Userform' dialog box with the number of streams set to '0.4'. An arrow points to the 'Ok' button.
- Step 5:** The 'Userform' dialog box with the number of streams set to '0.4'. An arrow points to the 'Ok' button.
- Step 6:** An Excel spreadsheet showing an example of a 'Stream table' with columns for 'Stream Name', 'M', 'W', 'H', 'W', 'M', 'W', 'M', 'W'. The table lists various components like Carbon, Methane, Ethane, etc., with their respective mass flow rates.
- Step 7:** The final 'Stream table' output, showing a detailed table with columns for 'Stream Name', 'M', 'W', 'H', 'W', 'M', 'W', 'M', 'W'. The table lists various components like Carbon, Methane, Ethane, etc., with their respective mass flow rates.

Figure A.8 Steps to generate steam table

⇒ In the case that the user already has stream table that follows the format mentioned above, follow the steps below to import stream table.

Step 1: Click “Import stream table” button as shown in Figure A.9.

Step 2: Select the stream table from the “Open file” window. Select “LCSoft” folder > “Project” folder > “Case study data” folder

Step 3: Select “stream table” file.

Step 4: Click “Open” button in open file window and click “Yes” button to confirm to use the table. The stream table will be imported into LCSoft, the user can check if the stream table is correct by open “Stream table” page as show in Figure A.7. If the user would like to change the table, follow step 1 to step 3 again.

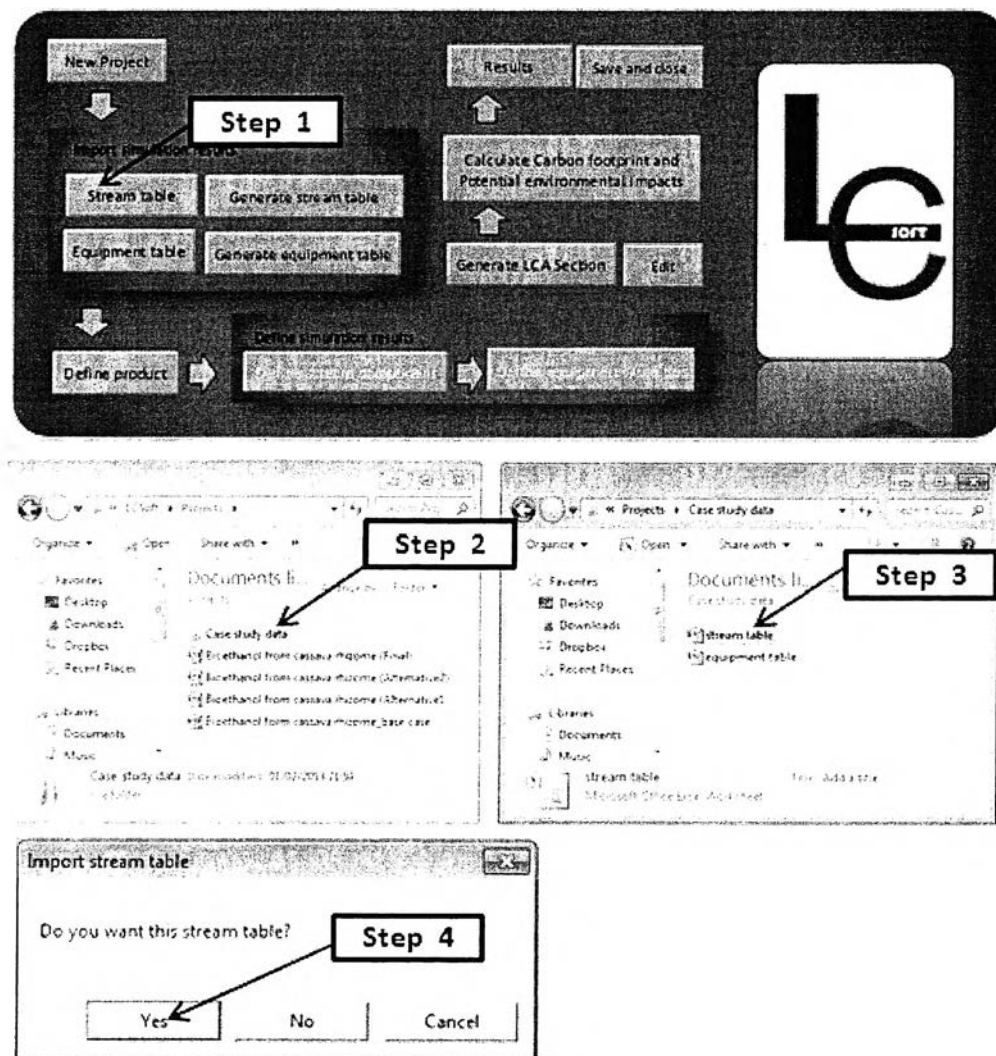


Figure A.9 Steps to import stream table

A.4.3.2 Import equipment table

The format of equipment table must be the same as Figure A.10. To see the stream table, open “LCSoft” folder > “Projects” folder> “Case study data” folder> “equipment table” file.

Unit Operation Summary								
Pump								
Pump Name		P1						
Work	kJ	0.553						
Reactor								
ConReactor Name		R1	R2	R3	R4	R5	R6	R7
Temperature	°C	150	90	30	45	45	45	45
Pressure	ATM	12.100	1.000	1.000	1.000	1.000	1.000	1.000
Duty	MJ/hr	0.700	-388.900	-97.700	1533.200	-242.500	-214.400	-112.900
Heat Of Reaction	MJ/hr	-1.255	-0.351	-0.093	0.372	-0.011	0.350	0.091
Product Inflow	kg/KG	-237.280	1309.560	1339.500	823.184	-86.238	-172.300	8.243
Feed Enthalpy	kJ/KG	-4142.521	1203.700	1310.220	-2582.658	19.703	39.694	31.737
ΔEnthalpy	kJ/KG	1810.080	126.112	15.463	2381.861	-54.216	-112.199	-22.187
ΔH _{298K}	kJ/KG	1.915	0.104	0.015	2.381	-0.026	-0.112	-0.023
Flash								
Flash Name		F1	F2					
Temperature	°C	103.834	41.021					
Pressure	ATM	1.000	1.000					
PH	ATM	12.100	0.000					
Duty	MJ/hr	0.700	0.000					
Stream Calculator								
Stream Calculator Name		SC1	SC2	SC3	SC4	SC5	SC6	
Duty	MJ/hr	0.500	0.000	0.300	0.000	0.000	0.500	
Overhead Product Temperature	°C	52.583	49.227	42.321	41.000	103.000	59.013	
Bottoms Product Temperature	°C	52.583	49.227	42.321	41.000	103.000	59.013	
Heat Exchanger								
Hx Name		E1	E2	E3	E4	E5	E6	E7
Duty	MJ/hr	219.300	1112.000	119.200	2130.000	1571.000	4304.000	4340.000
Column								
Column Name		T1	T2					
Condenser Duty	MJ/hr	-15018.200	-15871.200					
Reboiler Duty	MJ/hr	14529.500	14571.200					
Column Total Molar Feed	kg/DAY	17207.280	1317.017					
Column Total Wt. Feed	kg/DAY	80112.658	184251.108					
Column Condenser Area	ATM	1.720	1.720					
Column Condenser Temp	°C	52.583	51.321					
Column Reboiler Area	kg/DAY	0.500	1611.490					
Column Reboiler Temp	°C	1.200	1.200					

Figure A.10 Equipment table page

⇒ If the user does not have equipment table that follows the format mentioned above, follow the steps below to import stream table.

Case study:

- There are Pump; P1, Reactors; R1, R2, R3, R4, R5, R6, R7, Heat exchangers: E1, E2, E3, E4, E5, E6, E7, Distillation column; T1 condenser, T1 reboiler, T2 condenser, and T2 reboiler.
- Energy unit: MJ (Figure A.6)
- Time unit: hr (Figure A.6)
- Power unit: kW (Figure A.6)

To see the stream table, open “LCSoft” folder > “Projects” folder> “Case study data” folder> “equipment table” file.

Step 1: Click “Generate equipment table” button in “Main menu” page (Figure A.11).

Step 2: The user will get the message “Please enter equipment name, duty and work following the given example”, Click “Ok” button then “generate equipment table” page will appear.

Step 3: Read the instruction from the given example and enter the equipment name, duty and work following the given example. Make sure that the unit of each equipment is the same with the previously defined one. In the case study; unit of energy is MJ, unit of power is kW and unit of time is hr.

- Duties and work are all numeric.
- The units of mass and time must be corresponding to the units that user have selected when create new project.

Step 4: Click “Done” button in “generate equipment table” page to finish.

The figure illustrates the process of generating an equipment table through four sequential steps:

- Step 1:** The user is in the 'Main menu' of the LCSoft application, where the 'Generate equipment table' button is highlighted.
- Step 2:** A dialog box prompts the user to 'Please enter equipment name, duty and work following the given example'. The 'OK' button is highlighted.
- Step 3:** The 'generate equipment table' window displays a table with columns for Unit, Type of unit, Duty/Work, Energy, and Unit. An 'Example' table is also shown for reference. The 'Step 3' label points to the data entry area.
- Step 4:** A 'Utility Entry' dialog box is shown with a 'Done' button highlighted.

Unit	Type of unit	Duty/Work	Energy	Unit
P1	Pump	5 6979 kW		hr
R2	Reactor	-2993.6 MJ		hr
R3	Reactor	-37.2 MJ		hr
R4	Reactor	361.2 MJ		hr
R5	Reactor	-843.3 MJ		hr
R6	Reactor	-9234.4 MJ		hr
R7	Reactor	-322.9 MJ		hr
E1	Heat Exchanger	859.3 MJ		hr
E2	Heat Exchanger	1113 MJ		hr
E3	Heat Exchanger	235.2 MJ		hr
E4	Heat Exchanger	2330 MJ		hr
E5	Heat Exchanger	5670.5 MJ		hr
E6	Heat Exchanger	4804 MJ		hr
E7	Heat Exchanger	4040.4 MJ		hr
T1	Column-Condenser	-18088.2 MJ		hr
T2	Column-Reboiler	24359.6 MJ		hr
T3	Column-Condenser	-19971.2 MJ		hr
T4	Column-Reboiler	14371.4 MJ		hr

Unit	Type of unit	Duty/Work
P1	Pump	5 6979 kW
R2	Reactor	-2993.6 MJ
R3	Reactor	-37.2 MJ
R4	Reactor	361.2 MJ
R5	Reactor	-843.3 MJ
R6	Reactor	-9234.4 MJ
R7	Reactor	-322.9 MJ
E1	Heat Exchanger	859.3 MJ
E2	Heat Exchanger	1113 MJ
E3	Heat Exchanger	235.2 MJ
E4	Heat Exchanger	2330 MJ
E5	Heat Exchanger	5670.5 MJ
E6	Heat Exchanger	4804 MJ
E7	Heat Exchanger	4040.4 MJ
T1	Column-Condenser	-18088.2 MJ
T2	Column-Reboiler	24359.6 MJ
T3	Column-Condenser	-19971.2 MJ
T4	Column-Reboiler	14371.4 MJ

Figure A.11 Steps to generate equipment table

⇒ In the case that the user already has equipment table that follows the format mentioned above, follow the steps below to import stream table.

Step 1: Click “Import equipment table” button as shown in Figure A.12. Step 2: Select the stream table from the “Open file” window. Step 3: Click “Open” button in open file window and click “Yes” button to confirm to use the table. The equipment table will be imported into LCSofT, the user can check if the equipment table is correct by open “Equipment table” page as show in Figure 12. If the user would like to change the table, follow step 1 to step 3 again.

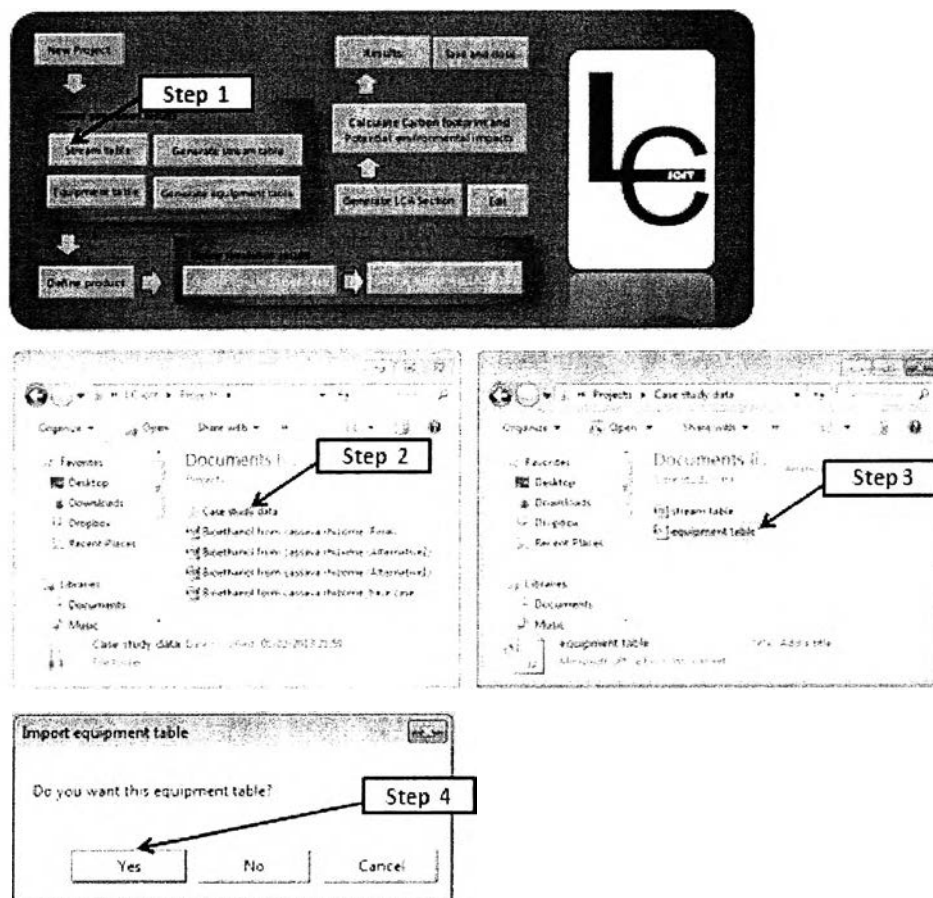


Figure A.12 Equipment table page

A.4.4 Define product/functional unit

Case study:

- use simulation results

Product: Ethanol from stream S67

⇒ *Define product using simulation results*

Step 1: Click “Define product” button as shown in Figure A.13.

Step 2: Click “Yes” button.

Step 3: Select product (Ethanol) and product stream (S67).

Step 4: Click “Ok” button.

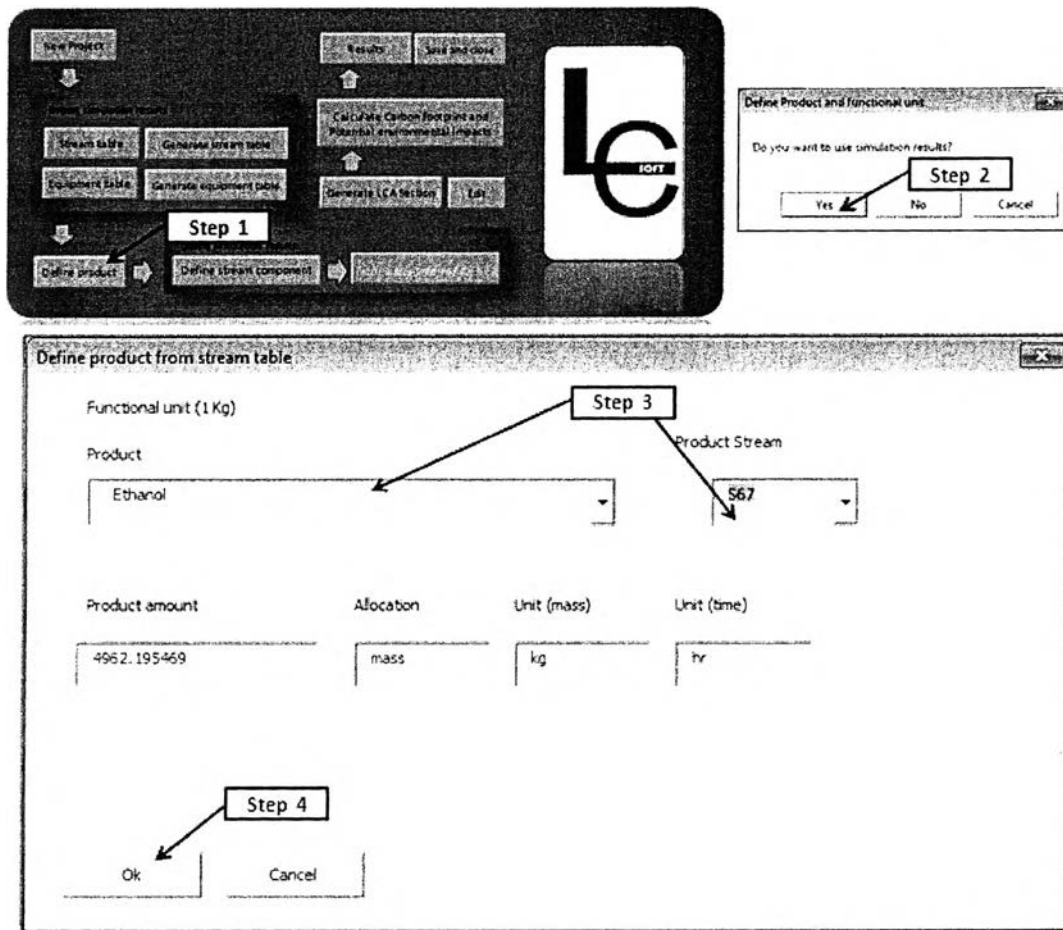


Figure A.13 Steps to define product using simulation results

⇒ *Define product without simulation results*

Step 1: Click “Define product” button as shown in Figure A.14. Step 2: Click “No” button.

Step 3: Enter the name of product. Step 4: Select allocation type, generally allocation type is mass.

Step 5: Select unit of mass of the product. Step 6: Enter amount of the product. Step 7: Click “Ok” button.

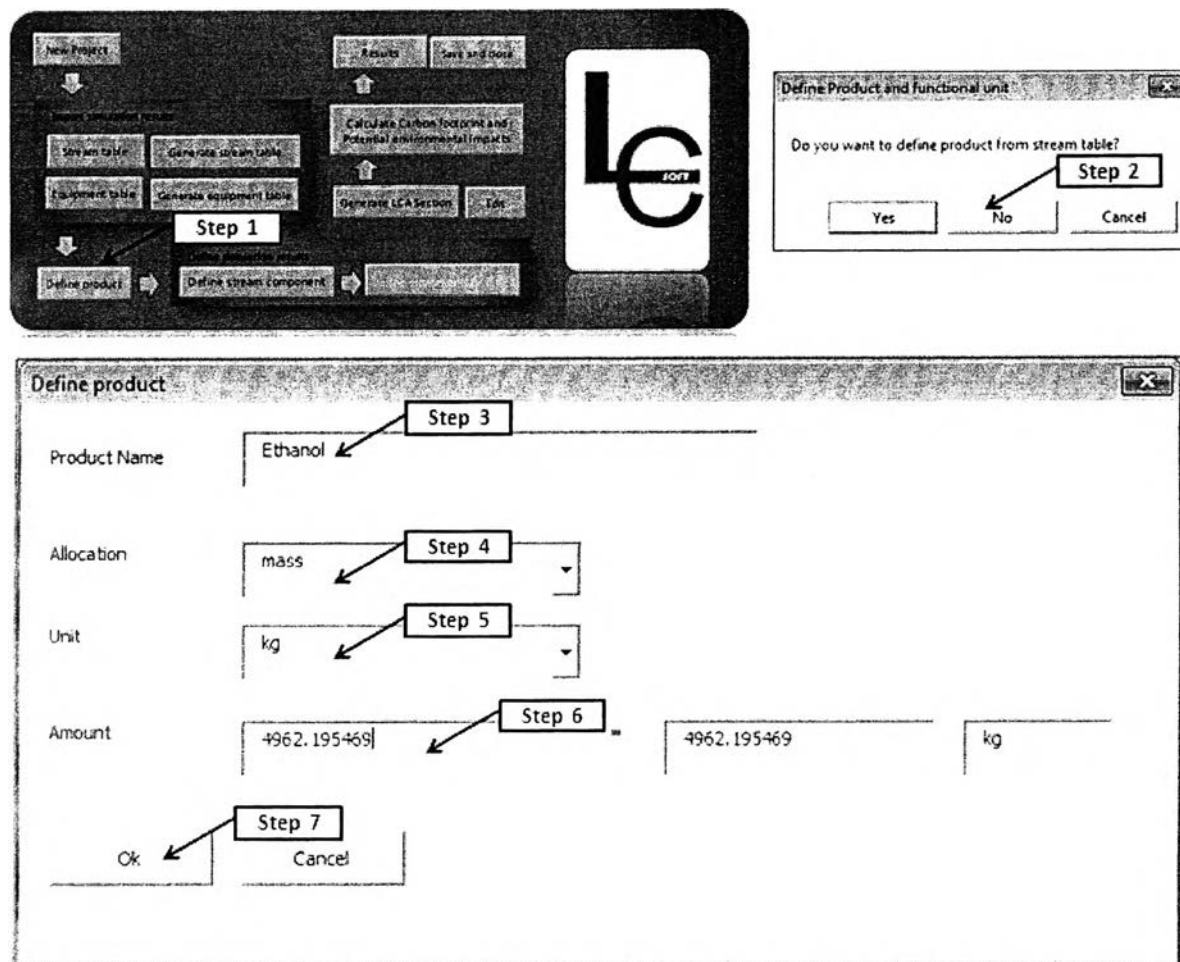


Figure A.14 Steps to define product without simulation results

A.4.5 Define stream component

This step is optional for using simulation results to conduct LCA (*If the user does not have simulation results can skip this step*). To include the environmental impacts from production of raw material, the component entered in the process should be defined.

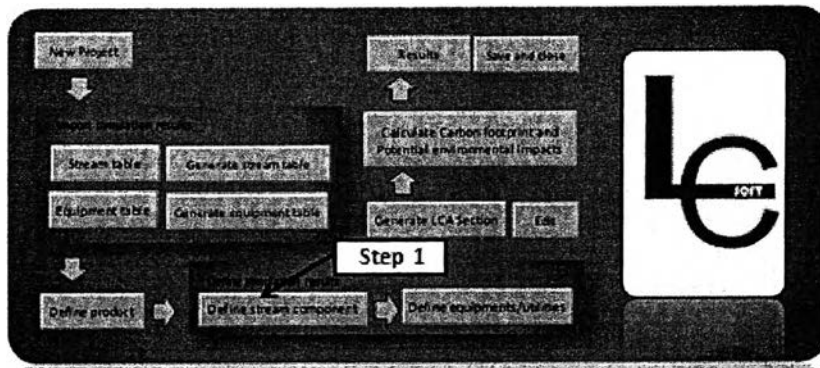
Not all components in the stream table must be defined. The user should define the component that was added into the process because the emissions from production of raw material are necessary to be included in the system to check the effect of using each raw material and ensure that all raw materials will not generate pollution more than emission limits. The product, by product and intermediate are not needed to be defined.

Case study: The components that are added into the process are

- Sulfuric acid
- Ammonia
- Corn steep liquor
- Cellulase
- Cassava rhizome (cellulose, hemicelluloses, and lignin)

(treated water is not included since water is treated and recycled in the process)

Step 1: Click “Define stream component” button (Figure A.15). Step 2: Double click on the gray area cells to define the each component and click “Yes” to confirm to define this component. Step 3: Select type of material which are: Biomass: If the stream component is biomass such as corn stover at conversion plant. Chemical: If the stream component is chemical such as sulfuric acid, ammonia etc. Fuel: If the stream component is fuel such as diesel or anthracite coal. Plant production: If the stream component is plant such as corn stover, at field. Other: Refer to other type of materials that has been added by users. For example; cellulose is the component of cassava rhizome which is plant, cassava rhizome is produced from cassava root. Step 4: Select the name in the list. Step 5: Click “Ok” button. Step 6: Follow step 2 to step 4 to define the next stream component. Step 7: After finishing defining stream components in the stream table, Click “Done” button (Figure A.16). LCSOFT will retrieve the LCI data from LCSOFT database to be used to calculate environmental impacts in a next step. The user can edit the specification of the stream component by clicking “Define stream component” button in “Main menu” page. Click “Done” button in the “Define stream” page after changing the specification.



Component from Stream Table

Microsoft Excel

Do you want to use this component?

Yes No

Stream name		Name	Unit	HS
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				
23				
24				
25				
26				
27				
28				
29				
30				
31				
32				
33				
34				
35				

Select type of material: Plant

Step 3

No.	Name	Unit
1	Cassava root	kg
2	Corn stover, at field	kg
3	Corn, at field	kg
4	Corn, decomposition, 15.5% moisture	kg
5	Corn, production, average	kg
6	Cotton straw, at field	kg
7	Cotton, at field	kg
8	Crude palm kernel oil, at plant	kg
9	fertilizer, corn	ha
10	fertilizer, stover	ha
11	fertilizer, switchgrass	ha
12	fertilizer, winter wheat straw	ha
13	Harvesting, fresh fruit bunch, at farm	kg
14	Lime, agricultural, corn production	ha
15	Potato leaves, at field	kg

Comment

Location: Thailand/Technology: Included processes: Fuel use, fertilizer use, Herbicide use and transport feedstock division in ethanol plant base on Thailand agricultural area. General comment: Cassava cultivation in Lampung province is Kase-Sari species. Land preparation prior to planting is done by diesel tractors and making beds. Cassava from stem cutting is manually planted. During post harvest, stems are not burnt, but utilized to be planted again.

Step 4

Step 5

Ok Cancel

Figure A.15 Steps to define stream component

Component from Stream Table

Done

Step 6

Step 7

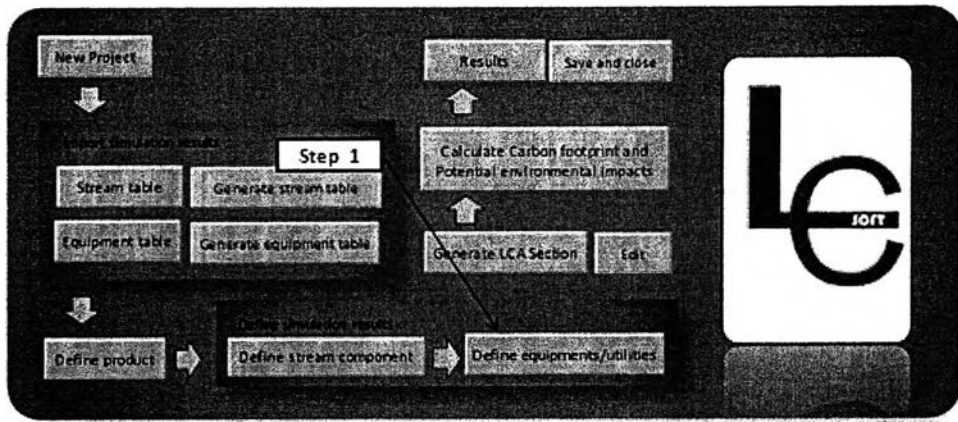
Stream component	Name
Cellulose	Cassava root
Hemicellulose	Cassava root
Lignin	Cassava root
Glucose	(double click here)
Xylose	(double click here)
Cellobiose	(double click here)
Ethanol	Product
Water	(double click here)
Sulfuric Acid	Sulfuric acid, at plant
Furfural	(double click here)
Ammonia	Ammonia, steam reforming, liquid, at plant
Oxygen	(double click here)
Carbon Dioxide	(double click here)
Glycerol	(double click here)
Succinic Acid	(double click here)
Lactic Acid	(double click here)
HMF	(double click here)
Xylitol	(double click here)
Acetic Acid	(double click here)
CornSteep Liquor	corn steep liquor
ZM	(double click here)
Cellulase	enzyme, Cellulase, Novozyme Celluclast
Lime	(double click here)
CASO4	(double click here)
Ash	(double click here)

Figure A.16 Steps to define stream component (Continued)

A.4.6 Define equipments/utilities

This step is optional for using simulation results to conduct LCA (*If the user does not have simulation results can skip this step*). The utility consumption can both directly and indirectly emit substances that effect environmental impacts. The direct emission such as emitted carbon dioxide from combustion of natural gas. The indirect emission such as emitted carbon dioxide from production of natural gas before transportation to the process.

Step 1: Click “Define equipments/utilities” button (Figure A.17). Step 2: Double click on the gray area cells to define the each component and click “Yes”. Step 3: Select utility which are: Hot utility: For the equipment that needs heat. Cold utility: For the equipment that needs coolant. Electricity by fuel: For the equipments that use electricity, type of electricity divided into type of fuel. Electricity by country: For the equipments that use electricity, type of electricity divided into locations or countries. Others: Refer to other type of utility that has been added by user. For example, Pump use electricity, therefore the utility type is “Electricity by fuel”. In the case study the electricity produced from natural gas, therefore the utility name is “natural gas at power plant”. The amount of duty/work is retrieved from the equipment table. Step 4: Select utility name. Step 5: Click “Ok” button. The user can edit the specification of equipments/utilities component by clicking “Define equipments/utilities” button in “Main menu” page. Click “Done” button in the “Define equipment” page after changing the specification. Step 6: Follow step 2-5 to define utility of all equipment (Figure 18) Step 7: After finishing defining equipments/utilities in the stream table, Click “Done” button. LCSOFT will retrieve the LCI data from LCSOFT database to be used to calculate environmental impacts in a next step.



Utility Entry

Step 2

Unit	Type of unit	Qty/Wh	Unit Energy	Unit	Activity	Energy source
P1	Pump		5.679 kwh	kw		Electricity, natural gas steam generation
P2	Pump		229.9 Mj	Mj		Electricity, natural gas steam generation
P3	Pump		13.1 Mj	Mj		Electricity, natural gas steam generation
P4	Pump		811.2 Mj	Mj		Electricity, natural gas steam generation
P5	Pump		541.2 Mj	Mj		Electricity, natural gas steam generation
P6	Pump		15.94 kwh	kw		Electricity, natural gas steam generation
P7	Pump		11.2 kwh	kw		Electricity, natural gas steam generation
E1	Heat Exchanger		874.9 Mj	Mj		Electricity, natural gas steam generation
E2	Heat Exchanger		7770 Mj	Mj		Electricity, natural gas steam generation
E3	Heat Exchanger		1.75 kwh	kw		Electricity, natural gas steam generation
E4	Heat Exchanger		150 Mj	Mj		Electricity, natural gas steam generation
E5	Heat Exchanger		16.78 kwh	kw		Electricity, natural gas steam generation
E6	Heat Exchanger		4874 Mj	Mj		Electricity, natural gas steam generation
E7	Heat Exchanger		4242.4 Mj	Mj		Electricity, natural gas steam generation
T1	Condenser		18083.2 Mj	Mj		Electricity, natural gas steam generation
T2	Reboiler		24911.8 Mj	Mj		Electricity, natural gas steam generation
T3	Condenser		12071.2 Mj	Mj		Electricity, natural gas steam generation
T4	Reboiler		16371.4 Mj	Mj		Electricity, natural gas steam generation

Microsoft Excel dialog box: "Do you want to use (kw) as the utility of this equipment?" with Yes/No buttons.

Select Utility

Electricity by fuel

Step 3

Select energy source

No.	Name	Unit
1	Electricity, alumina refining regions	kWh
2	Electricity, aluminum smelting and ingot casting regions	kWh
3	Electricity, anthracite coal, at power plant	kWh
4	Electricity, at cogem, for natural gas turbine	kWh
5	Electricity, bauxite mining regions	kWh
6	Electricity, biomass, at power plant	kWh
7	Electricity, bituminous coal, at power plant	kWh
8	Electricity, diesel, at power plant	kWh
9	Electricity, lignite coal, at power plant	kWh
10	Electricity, natural gas steam generation	kWh
11	Electricity, nuclear, at power plant	kWh

Step 4

Comment

Location: United States | Technology: Electricity production by natural gas steam generation
 Included processes: unspecified | General comment:
 Important note: although most of the data in the US LCI database has undergone some sort of review, the database as a whole has not yet undergone a formal validation process.
 Please email comments to lci@nrel.gov.

Step 5

Ok Cancel

Figure A.17 Steps to define equipment utilities

Utility Entry

The diagram shows a process flow starting with 'Step 6' and 'Step 7'. An arrow from 'Step 6' points to the 'Activity' column of the table below. Another arrow from 'Step 7' points to a 'Done' button located above the table.

Unit	Type of unit	Duty/Work	Unit		Activity	Energy source
			Energy	time		
P1	Pump	5.6979 kW		hr	Electric usage	Electricity, natural gas, at power plant
R2	Reactor	-399.6 MJ		hr	Cooling	Chilled water, engine-driven chiller using natural gas
R3	Reactor	-97.7 MJ		hr	Cooling	Chilled water, engine-driven chiller using natural gas
R4	Reactor	1613.2 MJ		hr	Heating	Natural gas, combusted in industrial equipment
R5	Reactor	-843.3 MJ		hr	Cooling	Chilled water, engine-driven chiller using natural gas
R6	Reactor	-8214.4 MJ		hr	Cooling	Chilled water, engine-driven chiller using natural gas
R7	Reactor	-322.9 MJ		hr	Cooling	Chilled water, engine-driven chiller using natural gas
E1	Heat Exchanger	859.3 MJ		hr	Heating	Natural gas, combusted in industrial equipment
E2	Heat Exchanger	1113 MJ		hr	Heating	Natural gas, combusted in industrial equipment
E3	Heat Exchanger	235.2 MJ		hr	Heating	Natural gas, combusted in industrial equipment
E4	Heat Exchanger	2180 MJ		hr	Heating	Natural gas, combusted in industrial equipment
E5	Heat Exchanger	5678.5 MJ		hr	Heating	Natural gas, combusted in industrial equipment
E6	Heat Exchanger	4804 MJ		hr	Heating	Natural gas, combusted in industrial equipment
E7	Heat Exchanger	4840.4 MJ		hr	Heating	Natural gas, combusted in industrial equipment
T1condenser	Column-Condenser	-18089.2 MJ		hr	Cooling	Chilled water, engine-driven chiller using natural gas
T1reboiler	Column-Reboiler	24889.6 MJ		hr	Heating	Natural gas, combusted in industrial equipment
T2condenser	Column-Condenser	-19971.2 MJ		hr	Cooling	Chilled water, engine-driven chiller using natural gas
T2reboiler	Column-Reboiler	14371.4 MJ		hr	Heating	Natural gas, combusted in industrial equipment

Figure A.18 Steps to define equipment utilities (continued)

A.4.7 Generate LCA section

A.4.7.1 *Entering data (using simulation results)*

This step is for generate inputs and outputs of the process regarding to the defined system boundaries.

Case study: 5 section (Figure 2)

(1) Pretreatment, (2) Detoxification, (3) SSCF, (4) Distillation, and (5) Dehydration

Step 1: Click on “Generate LCA section” button as shown in Figure A.19.

Step 2: Type the name of the section and the description of the section (optional)

Step 3: Click “OK” button, the section “(1) Pretreatment” page will appear.

Step 4: To define Inputs of this section, double click cells in “INPUTS” bar, “Input” form will appear.

Case study: (1) Pretreatment (use simulation results)

Input streams: S1, S2, S4, S5, S7, and S14

(There is no utility from the equipment in the “Pretreatment” section since the duties of M1, M2, M3, R1, F1, M4, and SC1 are zero.)

Output streams: S10 , compartment: air , sub compartment: unspecified

Step 5: Click “Use simulation results” page.

Step 6: Select “Stream” option and select input stream. Select “Equipment” option if there are equipments in this section and select equipment presented in this section.

Step 7: Click “Ok” button.

Step 8: Do step 4-7 until S1, S2, S4, S5, S7, and S14 are all selected (Figure A.19).

- If the user wants to edit or change the previously selected input, double click on the area shown in Figure 20. The “Input” form will appear, click “Delete” button on “Input” form to delete the previously selected input or change the input.

Step 9: Double click on cells in Figure A.19 then the “Output” form will appear.

Step 10: Click “Use simulation results” page.

Step 11: Select compartment; air, water, and soil. Select sub compartment. Select output stream.

Step 12: Click “Ok” button.

Step 13: Click “Done” button (Figure A.20). The LCI data will be retrieved and stored to be used to calculate potential environmental impacts.

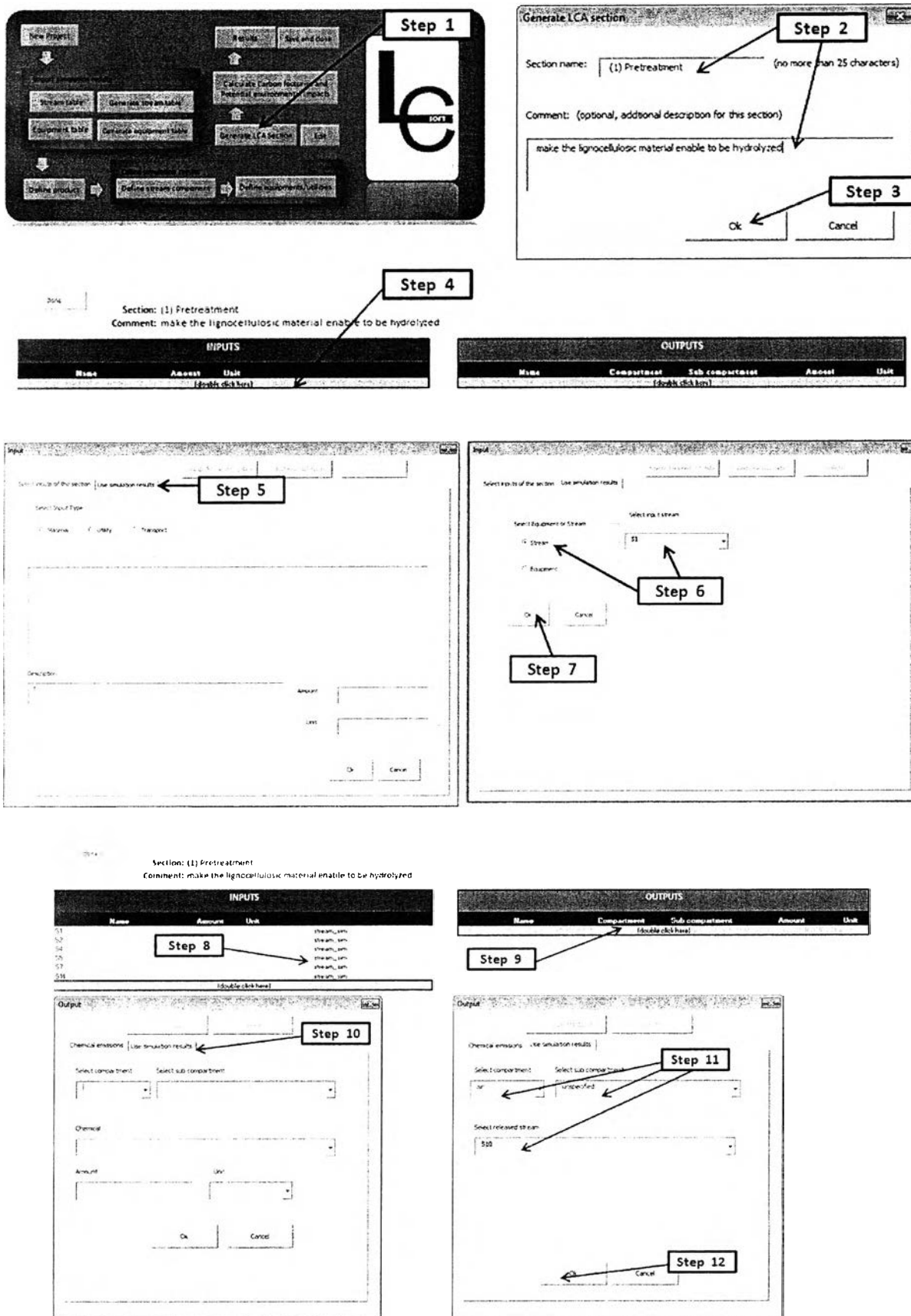


Figure A.19 Steps to generate LCA section

Step 13

Section: (1) Pretreatment
Comment: make the lignocellulosic material enable to be hydrolyzed

INPUTS				
Name	Amount	Unit		
S1			stream_s1m	
S2			stream_s2m	
S4			stream_s4m	
S5			stream_s5m	
S7			stream_s7m	
S14			stream_s14m	

(double click here)

OUTPUTS				
Name	Compartment	Sub compartment	Amount	Unit
S10	air	unspecified		

(double click here)

Figure A.20 Steps to generate LCA section (continued)

Doing

Section: (1) Pretreatment
Comment: make the lignocellulosic material enable to be hydrolyzed

INPUTS				
Name	Amount	Unit		
S1			stream_s1m	
S2			stream_s2m	
S4			stream_s4m	
S5			stream_s5m	
S7			stream_s7m	
S14			stream_s14m	

(double click here)

Double click to edit S1

Select inputs of the section: Use simulation results

Select Equipment or Stream

Stream

Equipment

Select input stream

S1

OK Cancel

Delete S1

Change input stream or input equipment

Figure A.21 Steps to edit inputs

Step 14: Do step 1 to step 13 until the user finish all sections ((1) Pretreatment, (2) Detoxification, (3) SSCF, (4) Distillation, and (5) Dehydration). Figure A.22-A.26 show defining inputs and outputs for each LCA section in the case study.

Section: (1) Pretreatment

Comment: to make the lignocellulosic material enable to be hydrolyzed

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub compartment	Amount	Unit
S1		stream_sim	S10	air	unspecified		
S2		stream_sim	(double click here)				
S4		stream_sim					
S5		stream_sim					
S7		stream_sim					
S14		stream_sim					
(double click here)							

Figure A.22 Inputs and outputs of (1) Pretreatment section

Section: (2) Detoxification

Comment: the compounds that are toxic for the fermentation microorganisms are treated

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub compartment	Amount	Unit
E1		equip_sim	S26	soil	unspecified		
R2		equip_sim	(double click here)				
R3		equip_sim					
S18		stream_sim					
S20		stream_sim					
S23		stream_sim					
(double click here)							

Figure A.23 Inputs and outputs of (2) Detoxification section

Done

Section: (3) SSCF

Comment: to convert glucose and other sugars to ethanol

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub compartment	Amount	Unit
S35		stream_sim	S40	air	unspecified		
S36		stream_sim	S52	air	unspecified		
S37		stream_sim	S55	soil	unspecified		
S43		stream_sim	(double click here)				
S44		stream_sim					
S45		stream_sim					
E2		equip_sim					
R4		equip_sim					
E3		equip_sim					
E4		equip_sim					
R6		equip_sim					
R7		equip_sim					
R5		equip_sim					
(double click here)							

Figure A.24 Inputs and outputs of (3) SSCF section

Section: (4) Distillation

Comment: to distil the process stream from SSCF section until a mixture of nearly azeotropic water and ethanol is obtained

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub compartment	Amount	Unit
E5		equip_sim	S58	air	unspecified		
P1		equip_sim	S63	water	unspecified		
T1condenser		equip_sim	S61	water	unspecified		
T1reboiler		equip_sim	(double click here)				
T2condenser		equip_sim					
T2reboiler		equip_sim					
(double click here)							

Figure A.25 Inputs and outputs of (4) Distillation section

Section: (5) Dehydration

Comment: the ethanol from distillation section is purified using vapor-phase molecular sieves in order to obtain bioethanol within the specifications

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub compartment	Amount	Unit
F6		equip_sim	S66	air	unspecified		
E7		equip_sim	(double click here)				
(double click here)							

Figure A.26 Inputs and outputs of (5) Dehydration section

A.4.7.2 Entering data (without simulation results)

The case study “Bioethanol from cassava rhizome” uses only simulation results. In the case that the user does not have simulation results to enter data for each section follow these steps below:

For example: Sulfuric acid 1000 kg is entered into (1) pretreatment section and carbon dioxide 10 kg is emitted to air from (1) pretreatment section.

Step 1: Double click gray area in “Inputs” bar (Figure A.27).

Step 2: Select type of input which are:

Material: If material is entered to the process such as chemicals, biomass, fuels, and plant.

Utility: If utility is entered to the process such as hot utility, cold utility, and electricity.

Transport: If transport is counted to the process such as truck, train, and aircraft. For transport, the used unit is “tonne-kilometre (tkm)”. For example, 0.00047 tkm is required to carry 0.0021 kg of sodium hypochlorite for 220 km ($2.14 \times 10^{-6} \times 220$).

Step 3: Select sub type of inputs. If the user has never created LCI data these sub type can be selected. In the case that LCI data has been created, the user can select sub type “Others”.

Material: chemicals, biomass, fuels, and plant

Utility: hot utility, cold utility, and electricity

Transport: transport by type of transport

Step 4: Select the name of input (Sulfuric acid, at plant for this example) and enter amount of input (1000 kg).

Step 5: Click “Ok” button.

Step 6: Click gray area in “Outputs” bar (Figure 28).

Step 7: Select compartment: air, water, and soil. Select sub compartment. Select emitted chemical. Enter emitted amount. Select unit of mass. (For this example, 10 kg of carbon dioxide is emitted to unspecified air.)

Step 8: Click “Ok” button

Step 9: Click “Done” button to finish entering data in this section.

Section: (1) Pretreatment
Comment:

Step 1

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub compartment	Amount	Unit
<i>(double click here)</i>			<i>(double click here)</i>				

Select inputs of the section | Use simulation results |

Select Input Type: Material Utility Transport

Select Material Type: Chemicals

Step 2

Step 3

ID	Name	Unit
77	Sodium hydroxide, production mix, at plant	kg
78	Soy biodiesel, production, at plant	kg
79	Soy meal, at plant	kg
80	Soy oil, refined, at plant	kg
81	Soy-based protein, at plant	kg
82	Soy-based resin, at plant	kg
83	Soybean oil, crude, degummed, at plant	kg
84	Sulfur, at plant	kg
85	Sulfuric acid, 18% solution, at plant	kg
86	Toluene dicarboxylate, at plant, CTR	kg
87	Toluene dicarboxylate, at plant	kg

Step 4

Description: location: North America (US and Canada) | Technology: Contact process via stepwise oxidation of sulfur followed by reaction with a sulfuric acid solution | Included processes: Includes material and energy requirements and environmental emissions for the kilogram of sulfuric acid production. General comment: During sulfuric acid production, the burning of sulfur produces heat, which in turn is used to generate steam. This steam is usually used in adjacent processing plants and supplies energy to the sulfuric acid plant. The exported steam is given a credit and shown as recovered energy. Complete inventory data and metadata are available in full in the final report and appendices. Cradle-to-Gate Life Cycle Inventory of Line Polyester Resins and Four Polyurethane Precursors. This report has been

Amount: 1000
Unit: kg

Step 5

Ok Cancel

Figure A.27 Steps to enter data without simulation results

Done

Section: (1) Pretreatment
Comment:

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub-compartment	Amount	Unit
Sulfuric acid anhydrous	1000 kg	kg	Carbon dioxide	air	unspecified	10 kg	kg
(double click here)			(double click here)				

Output

Chemical: Carbon dioxide

Use simulation results

Select compartment: air

Select sub-compartment: unspecified

Chemical: Carbon dioxide

Amount: 10

Unit: kg

Ok Cancel

Done

Section: (1) Pretreatment
Comment:

INPUTS			OUTPUTS				
Name	Amount	Unit	Name	Compartment	Sub-compartment	Amount	Unit
Sulfuric acid anhydrous	1000 kg	kg	Carbon dioxide	air	unspecified	10 kg	kg
(double click here)			(double click here)				

Figure A.28 Steps to enter data without simulation results (continued)

A.4.7.3 Edit LCA section

Step 1: Click “Edit” button (Figure 29)

Step 2: Select section to be edited. For example, “(1) Pretreatment” section. Note that the user can delete the section by clicking “Delete” button.

Step 3: Click “Edit” button. The section that the user want to edit will appear, user can change inputs/outputs of the section by double click on the cells (see step 4 to 12 in 2.2.6.1).

Step 4: For example the user want to delete input stream, S14 from the “(1) Pretreatment” section. Double click any cells in the same row of input stream S14 below “Inputs” bar. “Input” form will appear (Figure A.30).

Step 5: Click delete button or select new specification in “Input” form.

Step 6: Click “Yes” button. Input stream, S14 will be deleted.

Step 7: To edit outputs of the section. For example, the user want to change output stream, S10 to 10 kg of carbon dioxide. Double click on any cells in the same row of output stream S14 below “Outputs” bar. “Output” form will appear.

Step 8: Select compartment, sub compartment, chemical, and unit of mass. Enter amount of emitted chemical (Figure A.31).

Step 9: Click “Ok” button.

Step 10: After the user finish editing in this section, click “Done” button.

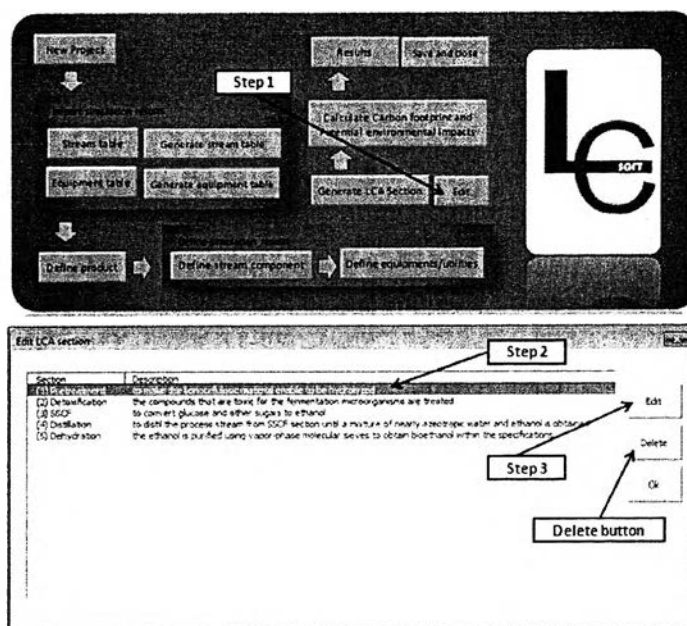


Figure A.29 Steps to edit LCA section

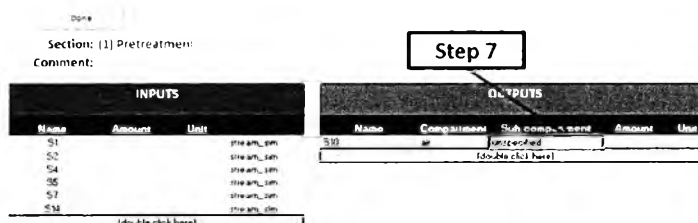
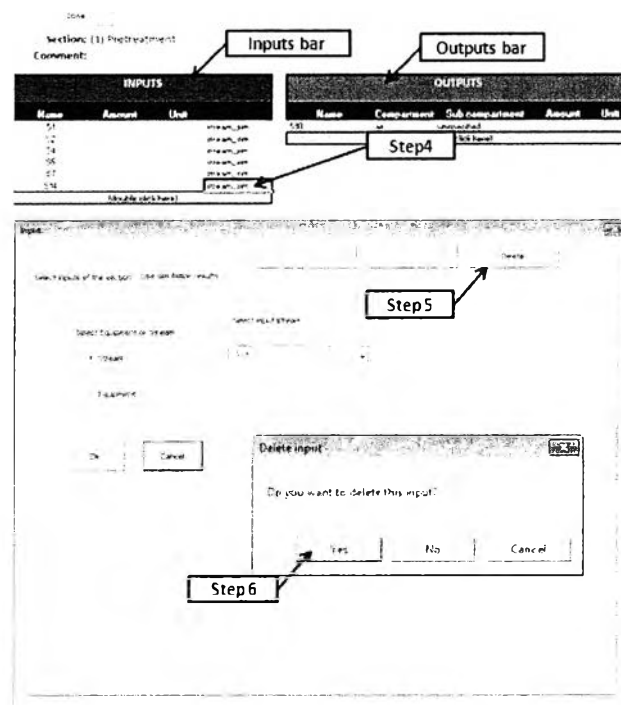


Figure A.30 Steps to edit LCA section (Continued)

Output

Chemical emissions | Use simulation results |

Select compartment | Select sub compartment

air | unspecified

Chemical

Carbon dioxide

Amount | Unit

10 | kg

Step 8

Step 9

Ok | Cancel

Done ← Step 10

Section: (1) Pretreatment

Comment:

INPUTS		
Name	Amount	Unit
S1		stream_sim
S2		stream_sim
S4		stream_sim
S5		stream_sim
S7		stream_sim
S14		stream_sim
(double click here)		

OUTPUTS				
Name	Compartment	Sub compartment	Amount	Unit
Carbon dioxide	air	unspecified	10	kg
(double click here)				

Figure A.31 Steps to edit LCA section (Continued)

A.4.8 Calculate carbon footprint, energy consumption and environmental impacts

LCSOFT provides options for calculation of carbon footprint (CF), energy consumption and environmental impacts which are indicators of environmental performance of the process.

Carbon footprint (CF) is the overall amount of carbon dioxide and other greenhouse gas emissions associated with one kilogram of product. The carbon footprint is quantified using indicators such as the global warming potential (GWP). GWP is an indicator that reflects the relative effect of a greenhouse gas in terms of climate change considering a fixed time period (100 years). In LCSOFT, GWPs are taken from the database of US Environmental Protection Agency (EPA). Emissions of utility consumptions in the process are converted to carbon dioxide emissions (CO₂ eq.) per functional unit of product (such as 1 kg of product, 1 l of product, or 1 MJ of product). This way will be easy to identify which equipment should be improved.

Energy consumption is the amount of energy (MJ) consumed by the resources per functional unit of product (such as 1 kg of product, 1 L of product, or 1 MJ of product) over the life cycle. Resources used in production of raw material phase, energy production phase and manufacturing phase are converted to total energy consumption and % renewable energy.

Environmental impacts are calculated from mass of emitted substance multiply by its characterization factors for a specific compartment (air, water, and soil) per functional unit of product (such as 1 kg of product, 1 L of product, or 1 MJ of product). In LCSOFT, 11 impact categories are calculated.

(1) Human toxicity by ingestion (HTPI, unit 1/LD₅₀): Human toxicity from ingestion and inhalation is used to estimate toxicity potential because they considered all of the primary routes of exposure of a chemical. HTPI were calculated for a chemical if it existed as a liquid or solid at a temperature of 0°C and atmospheric pressure, and an exposure potential. The lethal dose that produced death in 50% of rats by oral ingestion (LD₅₀) was used as an estimate for the HTPI. The value was chosen because of its prevalence in the literature and acceptance as a standard toxicity indicator. For those chemicals for which a rat-oral LD₅₀ value was not available, a value was estimated by molecular methods.

(2) Human toxicity by exposure (HTPE, unit 1/TWA): Human toxicity from dermal exposure used to estimate toxicity potential because they considered all of the

primary routes of exposure of a chemical. To estimate the HTPE, time-weighted averages (TWA) of the threshold limit values (TLV) were used. These values were obtained from OSHA, ACGIH, NIOSH and represent occupational safety exposure limits. This was considered to be an adequate measuring stick for comparison of chemicals that would pose a more thorough and relevant human toxicity value. Once completed, those values will supplant the human toxicity values that are currently stored in the LCSoft database.

(3) Aquatic toxicity (AC, unit $1/LC_{50}$): The ATP was estimated by using toxicological data for a single, representative species of fish, *Pimephales promelas* (fathead minnows). This species was chosen again because of its acceptance as a universal aquatic indicator and its prevalence of data. The data for this assay comes in the form of a LC_{50} , a lethal concentration which causes death in 50% of the test specimens.

(4) Terrestrial toxicity (TTP, unit $1/LD_{50}$): For the time being the LD_{50} values will be used to provide a relative toxicity comparison for both human and terrestrial entities.

(5) Global warming (GWP unit $kg\ CO_2\ eq.$): GWP is determined by comparing the extent to which a unit mass of a chemical absorbs infrared radiation over its atmospheric lifetime to the extent that CO_2 absorbs infrared radiation over its respective lifetimes. The half-lives of each of these chemicals was factored into the calculation for determining the GWP. Since, chemicals have different atmospheric half-lives the length of time over which the comparison is made will change the GWP of a chemical. For this database, 100 years was chosen as the base time frame.

(6) Ozone depletion (ODP, unit $kg\ CFC-11\ eq.$): The ODP is determined by comparing the rate at which a unit mass of chemical reacts with ozone to form molecular oxygen to the rate at which a unit mass of CFC-11 (trichlorofluoromethane) reacts with ozone to form molecular oxygen. For a chemical to have ODP it must exist in the atmosphere long enough to reach the stratosphere, it, also, must contain a chlorine or bromine atom.

(7) Photochemical oxidation (PCOP, $kg\ C_2H_2\ eq.$): The PCOP or smog formation potential is determined by comparing the rate at which a unit mass of chemical

reacts with a hydroxyl radical ($\text{OH}\cdot$) to the rate at which a unit mass of ethylene reacts with $\text{OH}\cdot$.

(8) Acidification (AP, unit H^+ eq.): The AP or acid rain potential is determined by comparing the rate of release of H^+ the atmosphere as promoted by a chemical to the rate of release of H^+ in the atmosphere as promoted by SO_2 .

(9) Carcinogenics (HTC, unit kg benzene eq.): The characterization factors for human toxicity (carcinogenic impacts) for chemical emissions to urban air, rural air, freshwater, sea water, agricultural soil and/or natural soil.

(10) Non carcinogenics (HTNC, unit kg toluene eq.): The characterization factors for human toxicity (non carcinogenic impacts) for chemical emissions to urban air, rural air, freshwater, sea water, agricultural soil and/or natural soil.

(11) Fresh water ecotoxicity (ET, unit kg 2,4-D eq.): The characterization factors for ecotoxicological characterization factor of chemicals include impacts for emissions to urban air, rural air, freshwater and/or agricultural soil.

To calculate CF, energy consumption and environmental impacts follow these steps below: Step 1: Click “Calculate carbon footprint and potential environmental impacts” button on “Main menu” page as shown in Figure 32. Wait until the software finish running. After finishing running, LCSofT will shown the status bare “finished running” that means all results are ready. Step 2: To see the carbon footprint results, click “results” button. Step 3: Click “Carbon footprint” option and click “Go” button. Step 4: Click “Done” button to back to “Main menu” page (Figure A.33). Step 5: To see PEI results, click “results” button. Step 6: Click “Impact Assessment” option and click “Go” button. Environmental impacts and energy consumption results will appear, if there is no chart in this page, click “Chart” button to generate chart of PEI results. Step 7: Click “Done’ button to back to “Main menu” page. After calculation of CF, energy consumption, and Environmental impacts, lists of resources used per one kilogram of product are presented in each section. To see the list of resources, open the sheet named “re_& section name”. For example, list of resources used in “(1) Pretreatment” section are in sheet named “re_(1) Pretreatment” as shown in Figure A.34.

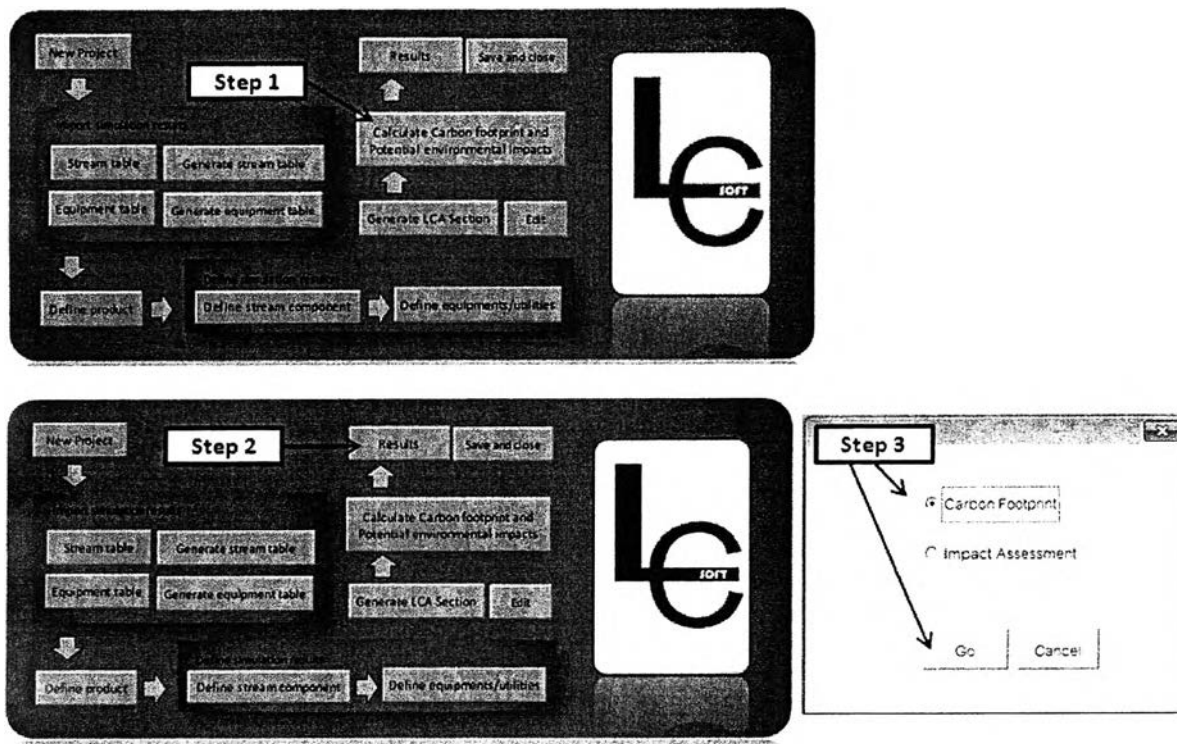


Figure A.32: Steps to calculate carbon footprint and potential environmental impacts

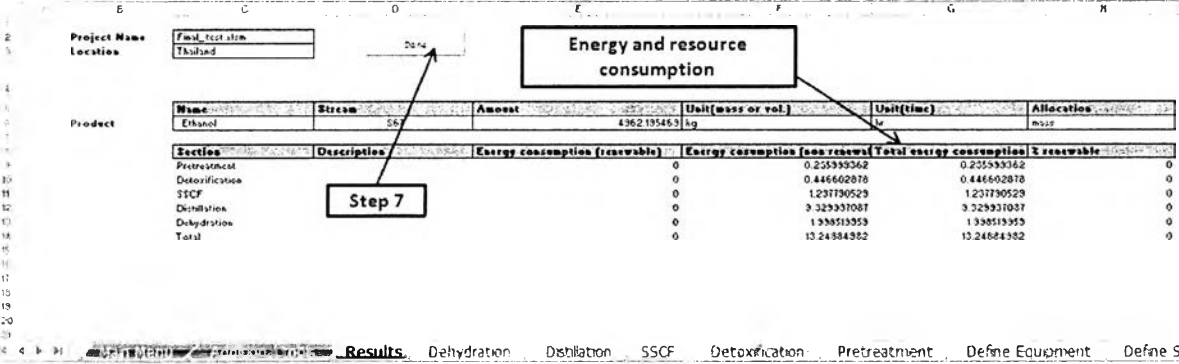
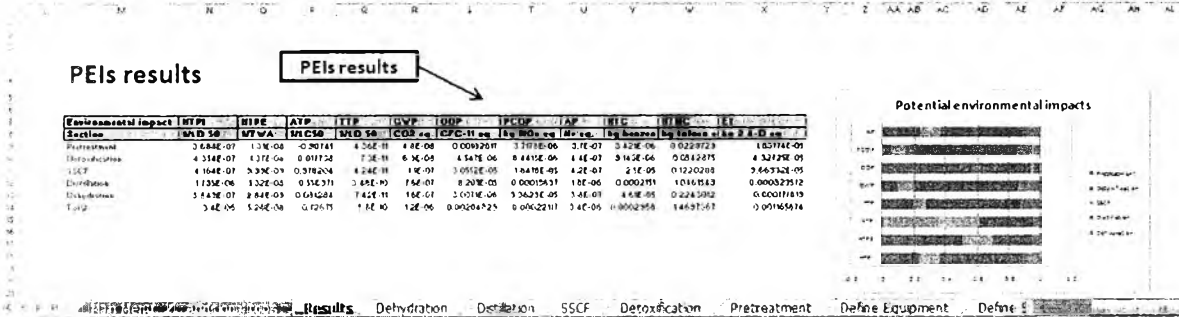
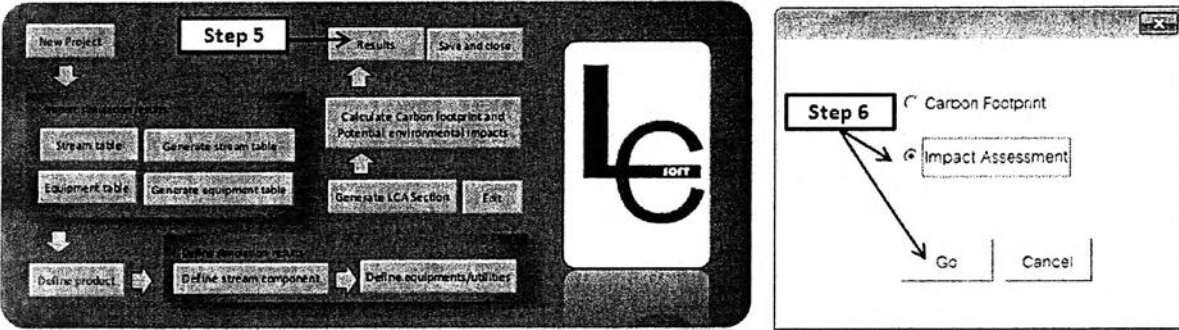
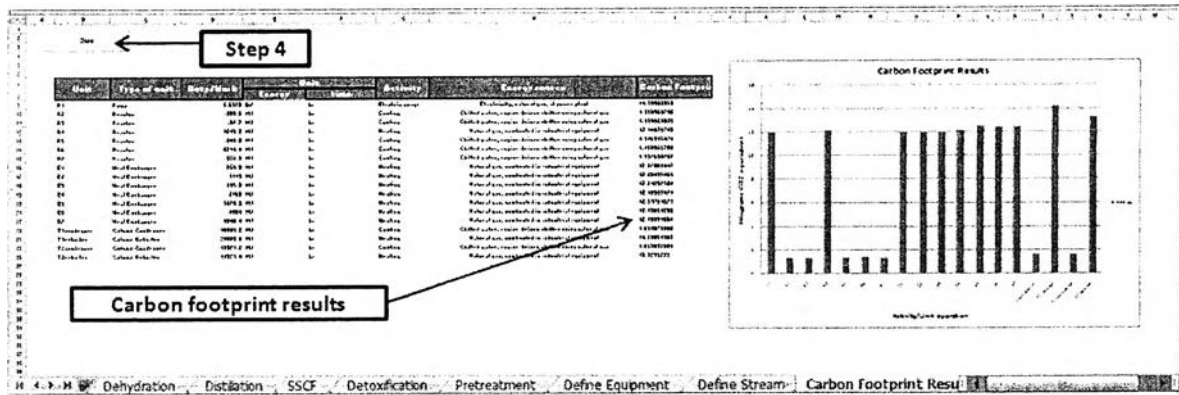


Figure A.33: Steps to calculate carbon footprint and potential environmental impacts

	A	B	C	D	E
1					
33					
39					
40					
41	Name	category	sub category	unit	Total
42	Coal, bituminous, 24.8 MJ per kg	resource	ground-	kg	2.0739
43	Uranium oxide (U3O8), 332 GJ per kg, in ore	resource	ground-	kg	5E-05
44	Carbon dioxide, in air	resource	in air	kg	0.0316
45	Oil, crude	resource	ground-	kg	15.55
46	Nitrogen, in air	resource	ground-	kg	2.508
47	Gas, natural	resource	ground-	m3	0.8621
48	Gas, natural, 49.8 MJ per kg, in ground	resource	fossil-	kg	1.6717
49	Coal, hard, unspecified, in ground	resource	in ground	kg	0.0023
50	Limestone, in ground	resource	in ground	kg	0.0163
51	Gas, natural, in ground	resource	in ground	m3	0.0089
52	Oil, crude, in ground	resource	in ground	kg	0.0035
53	Phosphate ore, in ground	resource	in ground	kg	0.008
54	Potassium oxide	resource	in ground	kg	0.002
55	Uranium, in ground	resource	in ground	kg	6E-08
56	Water, process, unspecified natural origin/kg	resource	in ground	kg	6.9
57	Water, unspecified natural origin/m3	resource	in ground	m3	0.1251
58	Energy, from coal	resource	in ground	MJ	312.46
59	Transformation, from pasture and meadow	resource	land	m2	4.3358
60	Transformation, to industrial area	resource	land	m2	4.3358
61	Gas, natural	resource	ground-	m3	0
62	Oil, crude	resource	ground-	kg	0
63					
64					
65					
66					
67					
68					

Figure A.34: List of resources in (1) Pretreatment section

A.5 Summary and additional tools

A.5.1 Summary

LCSOft, LCA software is exclusively designed for evaluation of chemical processes that provides options to import simulation results that help users to save time in order to conduct LCA. Results from LCSOft are carbon footprint, energy consumption, and potential environmental impacts (Environmental impacts) which are the environmental indicators in order to select the optimal design from various feasible alternatives. Furthermore, in order to design the sustainable process, it is crucial to ensure that the chemical product, feed stocks and energy production can be sustainably manufactured. Therefore, the application of LCSOft for bioethanol process is highlighted in order to evaluate the effects of feedstock and energy sources variation to find the cleanest feed stock and energy sources for bioethanol process.

A.5.2 Additional tools

LCSOft provides 3 additional tools in “Additional tools” page (Figure A.35) for special application as described below: (i) **Alternative comparison** is for comparison LCA results from different design cases (alternatives). (ii) **Sensitivity analysis** is the study of how Environmental impacts and energy consumption will change as a result of variation in raw materials and energy sources used in the process. (iii) **Generate LCI data in LCSOft** is for adding new LCI data which is specific for the process, this LCI data will be stored in the extended LCSOft LCI database.

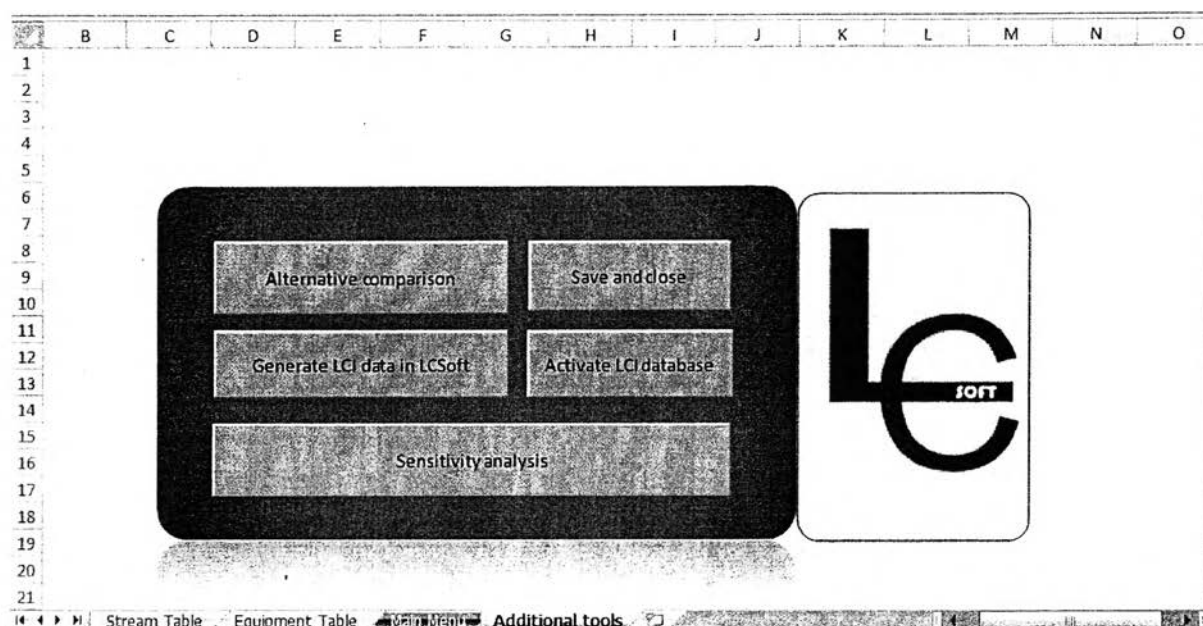


Figure A.35 Additional tools page in LCSOFT

A.5.3 Alternative comparison

Step 1: Click “Alternative comparison” button in “Additional tools” page (Figure A.35), the “Alternatives” page will appear as shown in Figure A.36. The first row of the alternative case is the LCA results of this project which is “Bioethanol from cassava rhizome_base case”. Step 2: Click on “Select alternative case” button. Step 3: Select project to analyze. Step 4: Click “Select” button. Step 5: Click “Yes” button. Do step 2-5 to select more alternatives. Step 6: When finish selection of alternatives, click “Create Chart” button. Step 7: Click “Done” to go back to “Main menu” page.

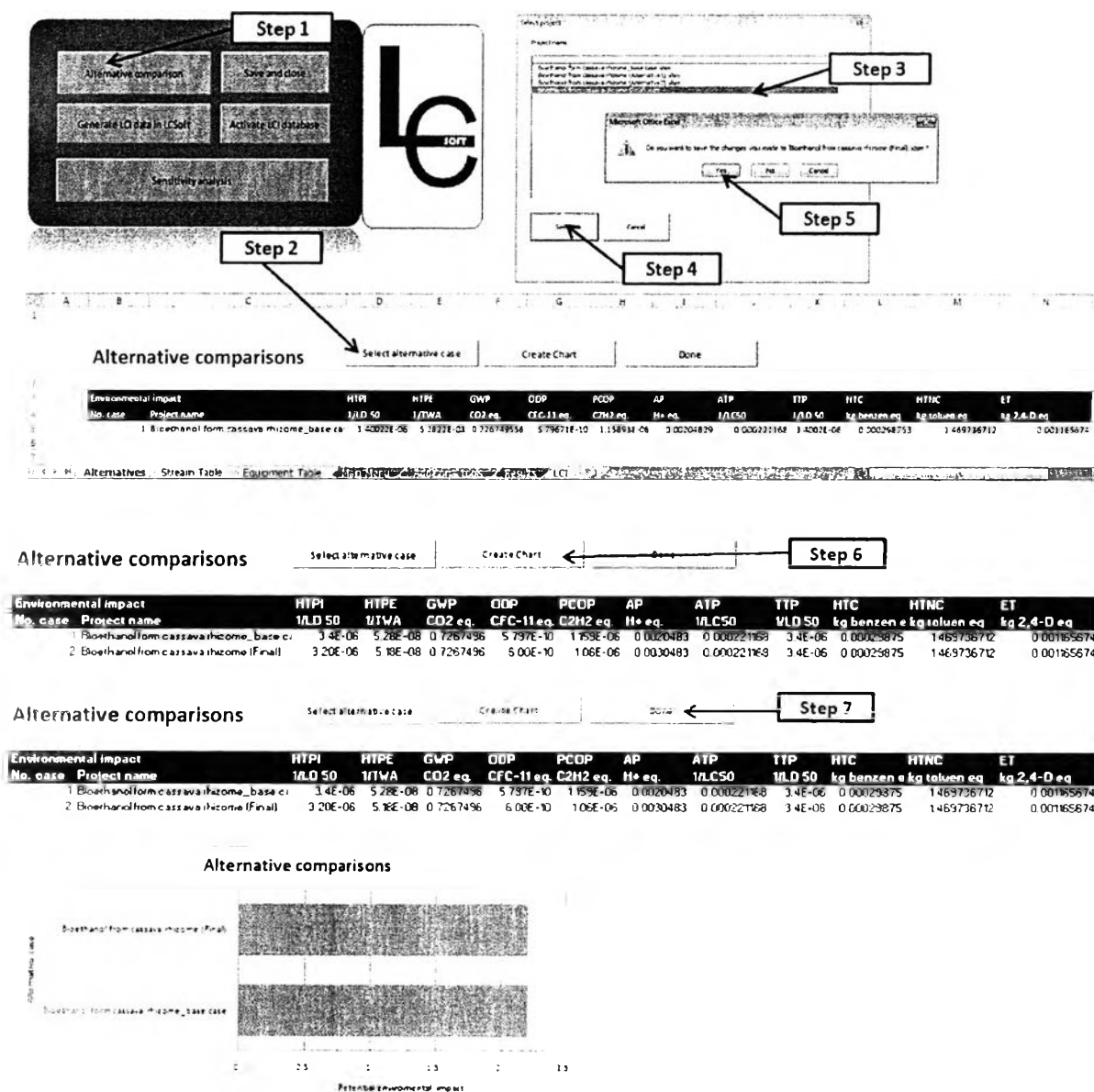


Figure A.36 Alternative comparison page in LCSOft

A.5.4 Sensitivity analysis

The variation of raw materials and energy sources effects Environmental impacts of the process, LCSOft provides this option to analyze how Environmental impacts change when the feed stocks and energy sources are changed.

From the case study “(4) Distillation” section contributes high environmental impacts from emissions from utility consumption. To study the effect of utility type for example hot utility in this case, hot utility is “Natural gas, combusted in industrial equipment” will be changed to “Wood waste, unspecified, combusted in industrial boiler”. To

change the specification of hot utility, there are 2 ways; (1) change from simulation results, and (2) change from section page. Both ways will be discussed.

Change from simulation results (see Figure A.37). Step 1: Click “Define Equipment” button in “Main menu” page. Step 2: Double click cells in H column to change the specification of hot utility, click “Yes” button to confirm. Step 3: Select “hot utility”, “Wood waste, unspecified, combusted in industrial boiler”, and click “Ok” button to confirm. Do step 2-3 until all hot utility specification are changed. Step 4: Click “Done” button to back to “Main menu” page (Figure A.38). Sensitivity analysis steps (see Figure A.39): Step 1: Click “Sensitivity analysis” button in “Additional tools” page. Step 2: Click “Sensitivity analysis” button in “Sensitivity Analysis” page. Step 3: Select section to analyze and Click “Do sensitivity analysis” button. Step 4: Enter case name “(4) Distillation_1” and comment “Natural gas, combusted in industrial equipment” will be changed to “Wood waste, unspecified, combusted in industrial boiler”, then click “Ok” button (see Figure A.40). Step 5: “(4) Distillation_1_sen” section page will appear, click “Done” button. Environmental impacts and energy consumption will presented in “Sensitivity Analysis” page. List of resources will presented in “re_Distillation_1_sen” page.

Change from section page: Step 1: Click “Sensitivity analysis” button in “Additional tools” page (see Figure A.39). Step 2: Click “Sensitivity analysis” button in “Sensitivity Analysis” page. Step 3: Select section to analyze and Click “Do sensitivity analysis” button. Step 4: Enter case name “(4) Distillation_1” and comment “Natural gas, combusted in industrial equipment” will be changed to “Wood waste, unspecified, combusted in industrial boiler”, then click “Ok” button. Step 5: “(4) Distillation_1_sen” section page will appear (see Figure A.41).

This section consumes, Electricity for equipment P1(5.6979 kWh), Hot utilities for equipment E5 (5678.5 MJ/hr), T1 reboiler (24889.6 MJ/hr), and T2 reboiler (14371.4 MJ/hr), Cold utility for equipment T1 condenser (-18089.2 MJ/hr), and T2 condenser (-19971.2 MJ/hr).

To change hot utility from “Natural gas, combusted in industrial equipment” to “Wood waste, unspecified, combusted in industrial boiler”, Double click any cells in row 9 (Figure A.41). Step 6: Select “Section inputs of the section” page on “Input” form. Step 7: Select “Utility” option button, Select “Hot utility” in utility list. Step 8: Select “Wood waste, unspecified, combusted in industrial boiler”, Enter amount of energy 5678.5 MJ (In this case duty of E5 is 5678.5 MJ/hr in order to produce product (ethanol) 4962.195469 kg/hr), Click “Ok” button. Do step 5-8, Double click on cells in row 12 to change T1 reboiler (Figure

A.42) and row 14 to change T2 reboiler (Figure A.43). Step 9: Click “Done” button (Figure A.41). Environmental impacts and energy consumption will be presented in “Sensitivity Analysis” page. List of resources will be presented in “re_Distillation_1_sen” page.

The figure illustrates the steps to change hot utility specification from simulation results. It consists of three main parts:

Step 1: A screenshot of the LEPORT software main menu. The process flow is: Define product → Define streams components → Define equipment layout → Calculate Carbon footprint and Potential environmental impacts → Generate ECA Section → Generate equipment table → Results → Save and close. A box labeled "Step 1" points to the "Define equipment layout" step.

Step 2: A screenshot of the "Utility Entry" spreadsheet. The table lists various utilities with columns for Unit, Type of unit, Daily flow, Energy, Cost, Utility, and Process number. A box labeled "Step 2" points to the "Process number" column.

Unit	Type of unit	Daily flow	Energy	Cost	Utility	Process number
R1	Reactor	56073 m ³	0	0	Electric energy	Electricity, natural gas, at source (heat)
R2	Reactor	789.6 MJ	0	0	Cooling	Unkilled steam engine, steam (chiller), steam (natural gas)
R3	Reactor	107.7 MJ	0	0	Cooling	Colded steam engine, steam (chiller), steam (natural gas)
R4	Reactor	9813.2 MJ	0	0	Heating	Treated gas, combusted in industrial boiler
R5	Reactor	943.2 MJ	0	0	Cooling	Colded steam engine, steam (chiller), steam (natural gas)
R6	Reactor	6276.4 MJ	0	0	Cooling	Colded steam engine, steam (chiller), steam (natural gas)
R7	Reactor	1122.3 MJ	0	0	Cooling	Unkilled steam engine, steam (chiller), steam (natural gas)
R8	Heat Exchanger	273.3 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R9	Heat Exchanger	100 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R10	Heat Exchanger	235.2 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R11	Heat Exchanger	280 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R12	Heat Exchanger	56.75 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R13	Heat Exchanger	6084 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R14	Heat Exchanger	8160.4 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R15	Heat Exchanger	2000.9 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R16	Heat Exchanger	1917.1 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R17	Heat Exchanger	1471.4 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R18	Condenser	3000.2 MJ	0	0	Cooling	Unkilled steam engine, steam (chiller), steam (natural gas)
R19	Condenser	2000.9 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R20	Condenser	1917.1 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)
R21	Condenser	1471.4 MJ	0	0	Heating	Unkilled steam engine, steam (chiller), steam (natural gas)

Step 3: A screenshot of the "Select Utility" dialog box. The "Hot utility" dropdown is set to "Hot utility". The "Select energy source" list shows various energy sources with their units. A box labeled "Step 3" points to the "Hot utility" dropdown.

No.	Name	Unit
10	Heat, onsite boiler, softwood mill, average	MJ
11	Lignite coal, combusted in industrial boiler	MJ
12	Liquefied petroleum gas, combusted in industrial boiler	MJ
13	Natural gas, combusted in industrial boiler, at hydrocracker, for petadene	MJ
14	Natural gas, combusted in industrial boiler, at hydrocracker, for ethylene	MJ
15	Natural gas, combusted in industrial boiler, at hydrocracker, for propylene	MJ
16	Natural gas, combusted in industrial boiler, at hydrocracker, for pyrolysis gas	MJ
17	Natural gas, combusted in industrial boiler	MJ
18	Natural gas, combusted in industrial boiler	MJ
19	Residual fuel oil, combusted in industrial boiler	MJ
20	Wood waste, combusted in industrial boiler	MJ

The "Comment" field contains the following text:

Location: US (Technology: Typical) Included processes: Unspecified (General comment: Converted previous version from the assumption of 50% MC green basis; 100% MC od basis to OD basis to align with CORRIM calculations. Important note: although most of the data in the US LCI database has undergone some sort of review, the database as a whole has not yet undergone a formal validation process. Please email comments to lo@rrim.gov.)

Figure A.37 Steps to change hot utility specification from simulation results

Utility Entry

Step 4

Unit	Type of unit	Duty/Work	Unit Energy	Unit time	Activity	Energy source
P1	Pump	5.6379 kW	hr	hr	Electric usage	Electricity, natural gas, air power plant
R2	Reactor	-395.6 MJ	hr	hr	Cooling	Chilled water, engine-driven chiller using natural gas
R3	Reactor	-97.7 MJ	hr	hr	Cooling	Chilled water, engine-driven chiller using natural gas
R4	Reactor	1613.2 MJ	hr	hr	Heating	Natural gas, combusted in industrial equipment
R5	Reactor	-843.3 MJ	hr	hr	Cooling	Chilled water, engine-driven chiller using natural gas
R6	Reactor	-3214.4 MJ	hr	hr	Cooling	Chilled water, engine-driven chiller using natural gas
R7	Reactor	-322.9 MJ	hr	hr	Cooling	Chilled water, engine-driven chiller using natural gas
E1	Heat Exchanger	853.3 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
E2	Heat Exchanger	1113 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
E3	Heat Exchanger	235.2 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
E4	Heat Exchanger	2180 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
E5	Heat Exchanger	5678.5 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
E6	Heat Exchanger	4804 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
E7	Heat Exchanger	4840.4 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
T1condenser	Column-Condenser	-18033.2 MJ	hr	hr	Cooling	Chilled water, engine-driven chiller using natural gas
T1boiler	Column-Reboiler	24893.6 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler
T2condenser	Column-Condenser	-13971.2 MJ	hr	hr	Cooling	Chilled water, engine-driven chiller using natural gas
T2boiler	Column-Reboiler	14371.4 MJ	hr	hr	Heating	Wood waste, unspecified, combusted in industrial boiler

Sensitivity Analysis (5) Dehydration (4) Distillation (3) SSCF (2) Detoxification (1) Pretreatment Stream Table Equipment

Figure A.38 Steps to change hot utility specification from simulation results (continued)

Sensitivity Analysis

Step 1

Alternative comparison Save and close
Generate LCI data in LCSoft Activate LCI database
Sensitivity analysis

LCSoft

Step 2

Sensitivity Analysis

Section Name Description Total energy consumption % renewable

Step 3

Sensitivity analysis

Section	Description
(1) Pretreatment	to make the lignocellulosic material enable to be hydrolyzed.
(2) Detoxification	the compounds that are toxic for the fermentation microorganisms are treated.
(3) SSCF	to convert glucose and other sugars to ethanol.
(4) Distillation	to distill the process stream from SSCF section into a mixture of near azeotropic water and ethanol is obtained.
(5) Dehydration	the ethanol from distillation section is purified using vapor-phase molecular sieves in order to obtain bioethanol within the specifications.

Do sensitivity analysis Cancel

Figure A.39 Steps to perform sensitivity analysis by changing hot utility specification from simulation results

Sensitivity Analysis

Analyze: (4) Distillation

Case name: (4) Distillation_1 (no more than 22 characters)

Comment: (optional, additional description for this section)

"Natural gas, combusted in industrial equipment" will be changed to "Wood waste, unspecified, combusted in industrial boiler"

Ok Cancel

Step 4

Step 5

Section: (4) Distillation_1_sen
 Comment: "Natural gas, combusted in industrial equipment" will be changed to "Wood waste, unspecified, combusted in industrial boiler"

INPUTS

Name	Amount	Unit
E3		558
P1		563
T1condenser		561
T1reboiler		
T2condenser		
T2reboiler		

(double click here)

Sensitivity Analysis

Result

Section name	Description	EE GHG emissions (scope 1+2)	Scope 3
(4) Distillation_sen	"Natural gas, combusted in industrial equipment"	5.073512854	35.80570461

Environmental Impact	HTPI	HTPE	GWP	ODP	PODP	AP	ATP	TTP	HTC	HTNE	ET
Section	WLO 50	WVA	CO2-eq	CFC-11	CFC-12	HFC	WLO 50	WLO 50	kg benz	kg tolu	kg 2,4-D
(4) Distillation_sen	0.00033369	0.05	0.99%	5E-06	7E-05	7E-05	5E-05	0.0001	0.0007	0.0021	2.007E-05

Figure A.40 Steps to perform sensitivity analysis by changing hot utility specification from simulation results (continued)

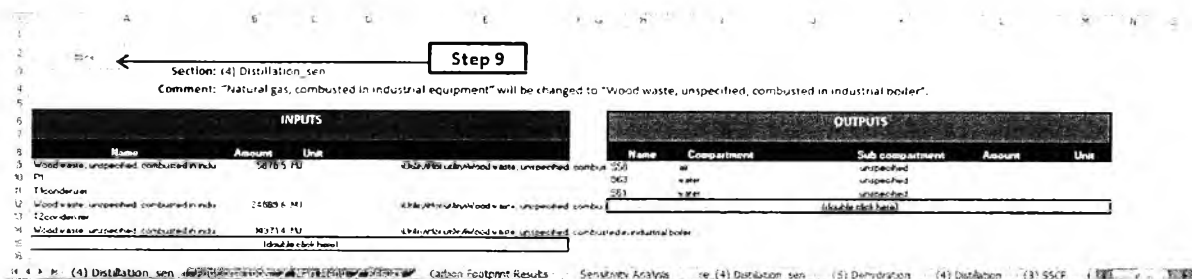
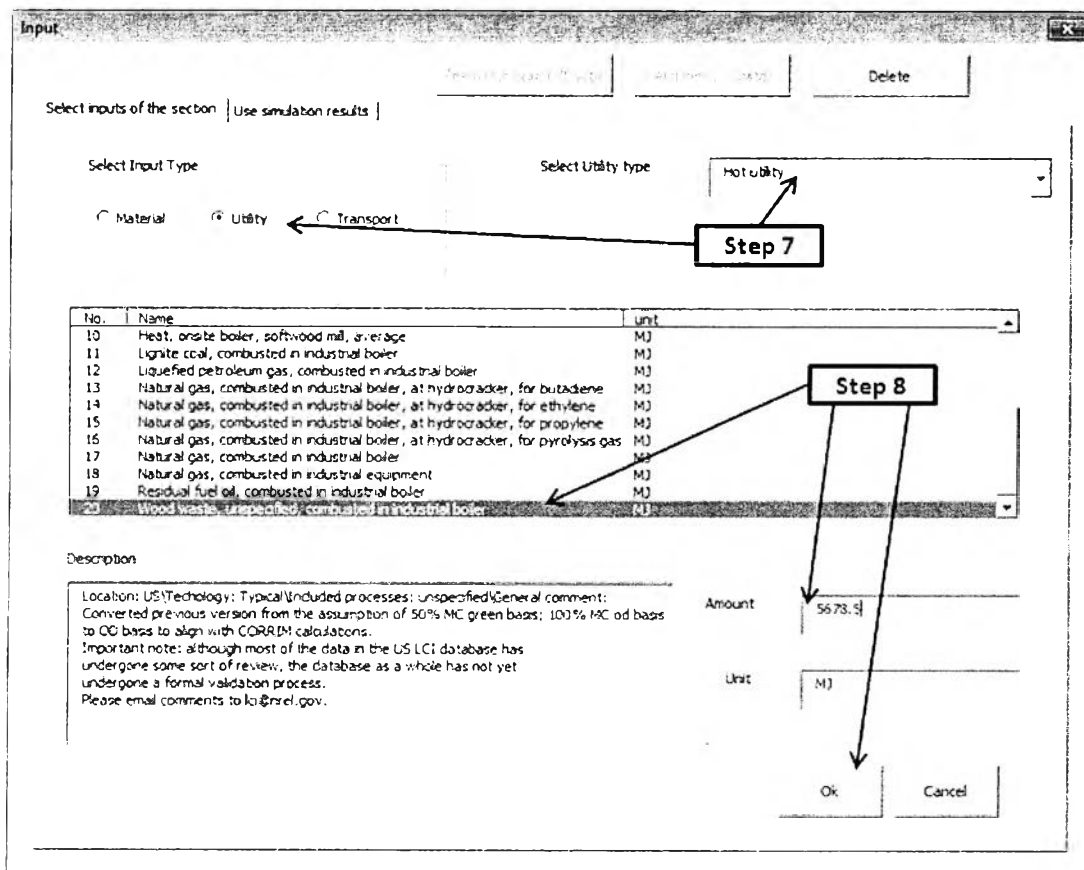
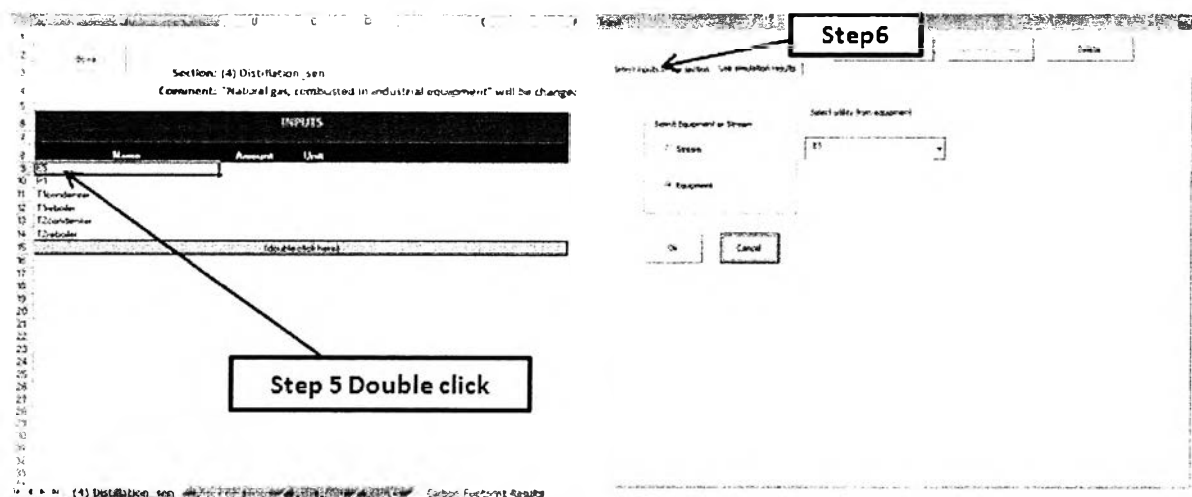


Figure A.41 Steps to change hot utility specification from section page

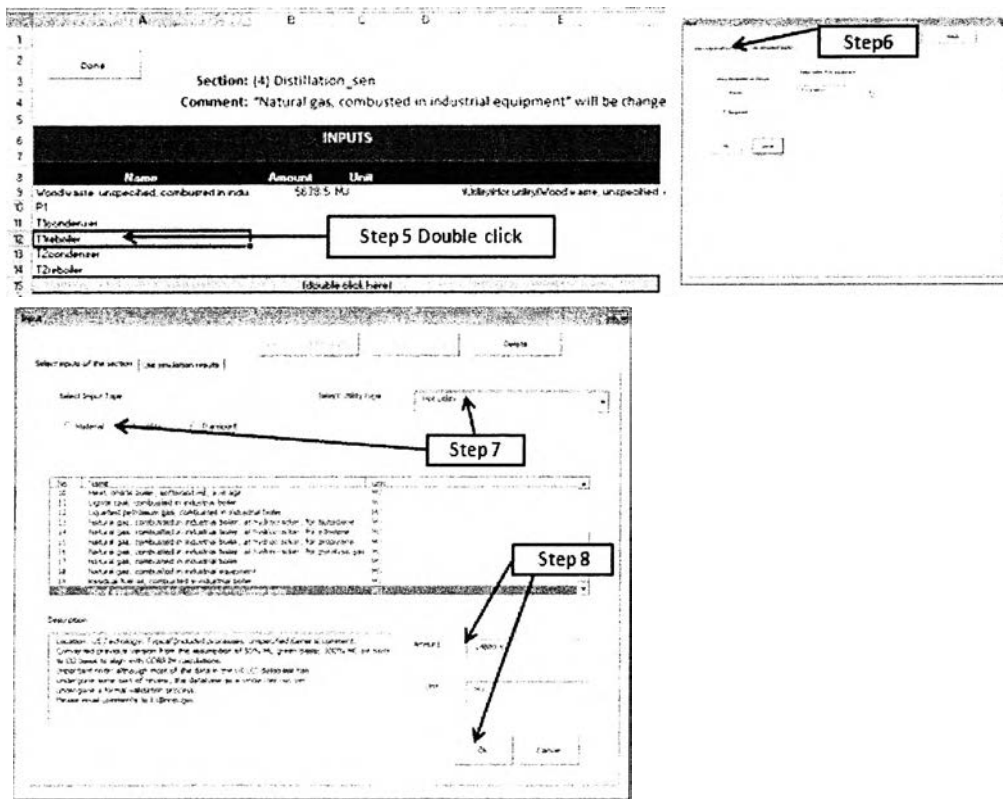


Figure A.42 Repeat step 5-8 to change hot utility specification of T1 reboiler

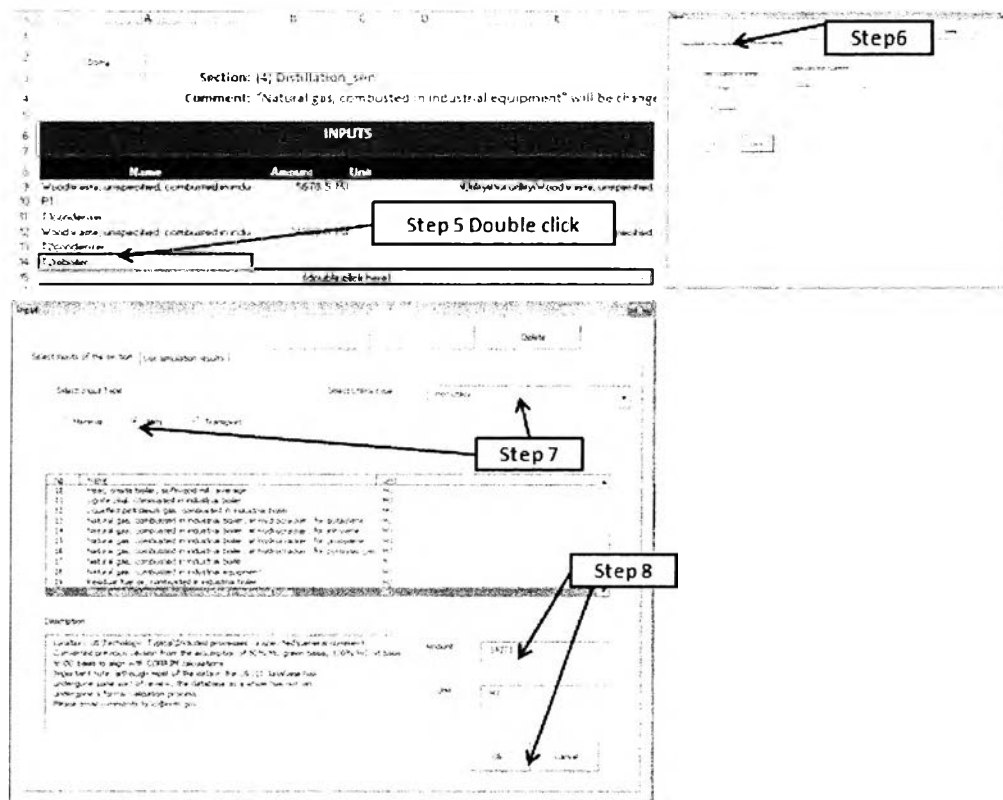


Figure A.43 Repeat step 5-8 to change hot utility specification of T2 reboiler

A.5.5 Generate LCI data in LCSof

LCSof provides a tool to generate LCI database. The following example is for generation LCI database of 1kg of Novozymes Spirizyme. Novozymes Spirizyme is the saccharification enzyme which is used for the extraction of the fermentable sugars from starch in saccharification and fermentation processes.

LCI data

Product: Novozymes Spirizyme 1 kg

Category: Material

Allocation: Mass

Unit: kg

Amount: 1

Location: US

Source of LCI data: NREL

Technology: current

Included processes: The assessment addresses environmental impacts potentials associated with enzyme production in a 'cradle-to-gate' perspective, i.e. all processes from production of raw materials to the final enzyme product at Novozymes' gate.

General comment:

Technology: current, Included processes: The assessment addresses environmental impacts potentials associated with enzyme production in a 'cradle-to-gate' perspective, i.e. all processes from production of raw materials to the final enzyme product at Novozymes' gate.

Process: -

Resources:

Water, unspecified natural origin; compartment resource; sub compartment in ground 0.029 m³

Energy from coal; compartment resource; sub compartment in ground 65.7 MJ (Heating value = 1 MJ/MJ from coal)

Emissions:

Carbon dioxide, fossil; compartment air; sub compartment unspecified 4.4 kg

Sulfur dioxide; compartment air; sub compartment unspecified 0.009 kg

Ethylene; compartment air; sub compartment unspecified 0.0016 kg
 Phosphate; compartment soil; sub compartment unspecified 0.026 kg

Step 1: Open LCSoft folder (Figure A.44)

Step 2: Select “Tool 1” folder.

Step 3: Open “Generate LCI data” file.

Step 4: Click “Define Product” button in “Main menu” page.

Step 5: Enter product name, select product type (material, utility, transport), select allocation method, select unit of product, and enter amount of product.

Step 6: Select “Descriptions” page in “Define product” form.

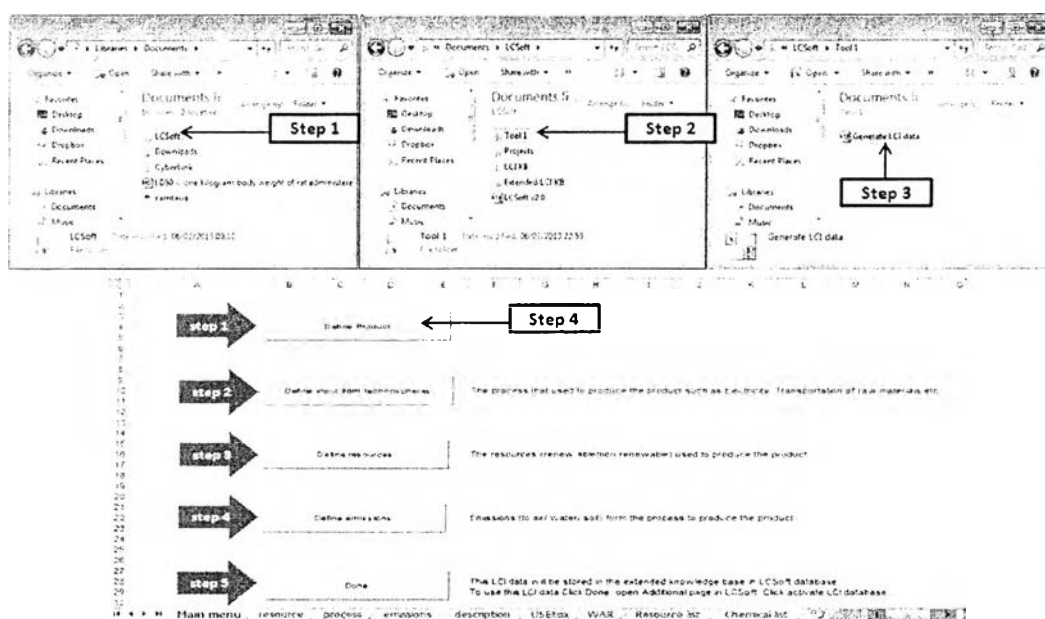


Figure A.44 Steps to generate LCI data

Step 7: Enter location, source of LCI data, technology, included processes, and general comment (Figure 45).

Step 8: Click “Ok” button.

Step 9: Click “Define input from technosphere” button in “Main menu” page.

Input from technosphere is the process that used to produce the product such as Electricity, Transportation of raw materials etc. The “process” page will appear.

Step 10: In the case study, there is no data about input from technosphere so “process” page can be closed. Click “Done” button in “process” page.

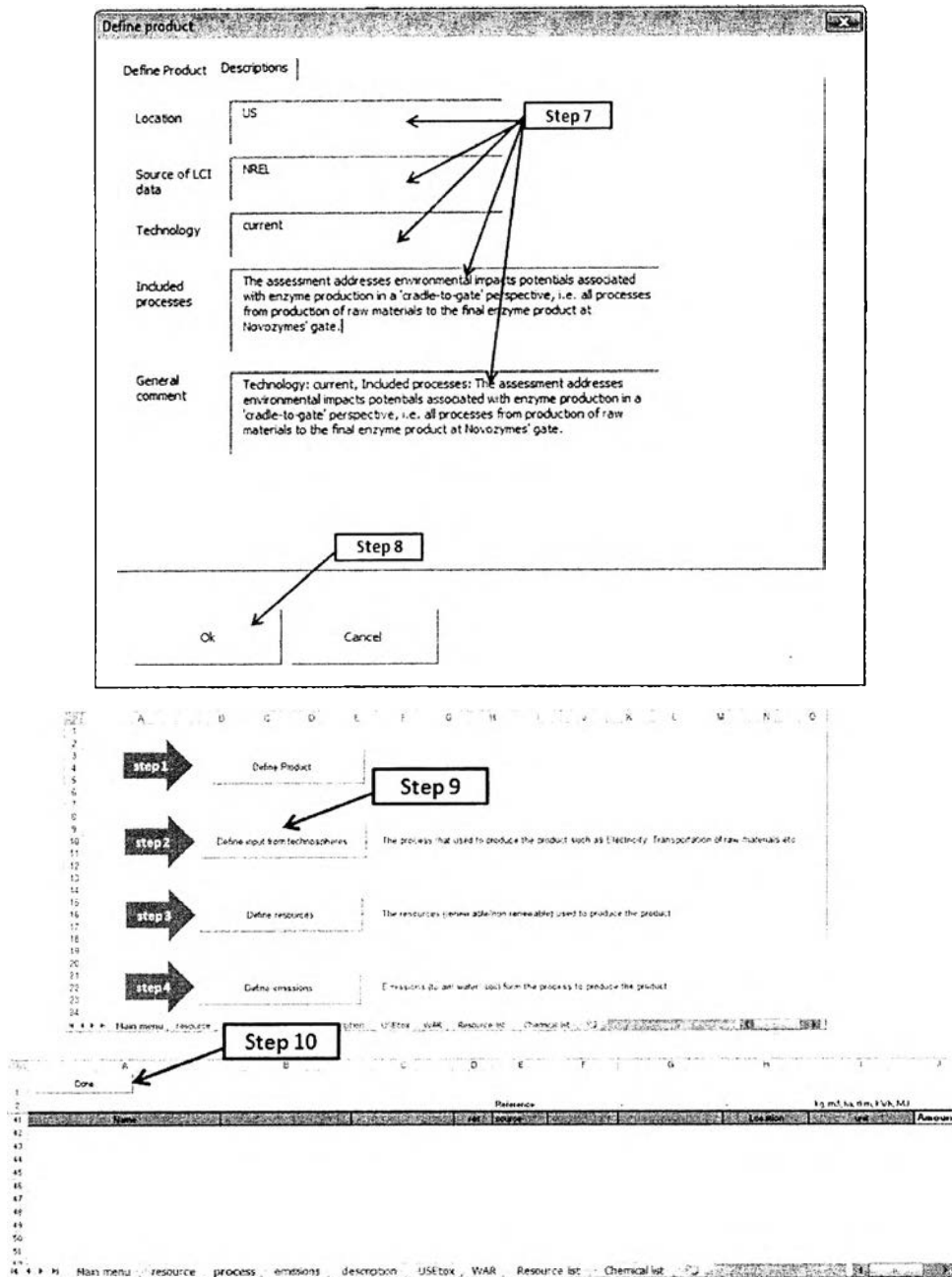


Figure A.45 Steps to generate LCI data (continued)

Step 11: Click "Defines resources" button in "Main menu" page (Figure 46).

Step 12: Click "Add resource" in "resource" page.

Step 13: Select resource which is "Water, unspecified natural origin" and unit is "m³", enter amount 0.029, click "Ok" button.

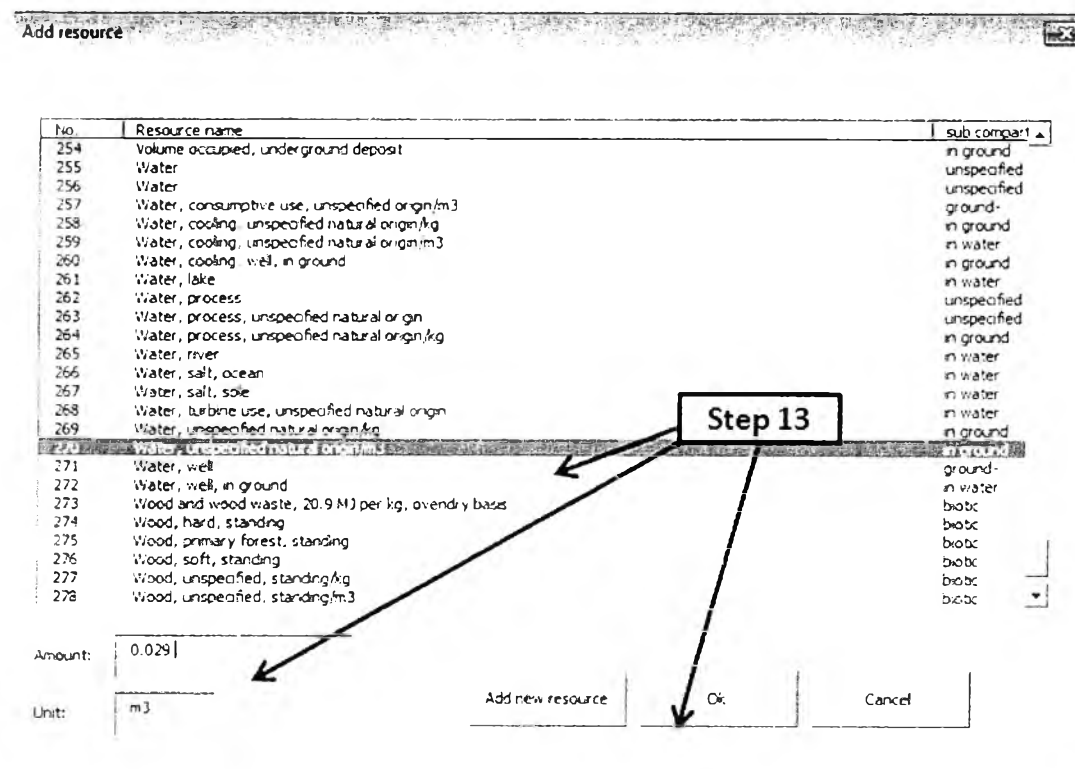
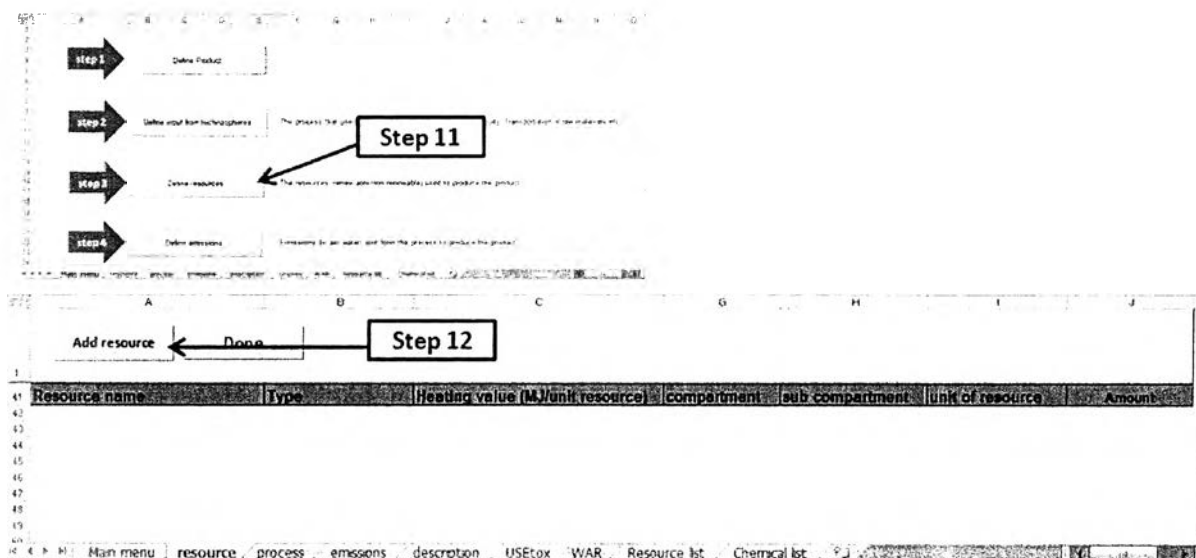


Figure A.46 Steps to generate LCI data (continued)

Step 14: Add resource which is Energy from coal; compartment resource; sub compartment in ground 65.7 MJ (Heating value = 1 MJ/MJ from coal). Click “Add resource” in “resource” page (Figure A.47)

Step 15: This time, there is no list of the resource, the user has to add new resource list by clicking “Add new resource” button on “Add resource” form.

Step 16: Enter resource name, type (renewable, non-renewable), amount, unit (Note that type unit follow the example in the message in “Add resource” form. For example the unit “MJ”, the user should type “MJ” not “mj” or “mJ”). Click “Ok” button.

Step 17: Click “Done” button to finish adding resources.

Step 18: Click “Define emissions” button in “Main menu” page.

Step 19: Click “Add emissions” button in “emissions” page.

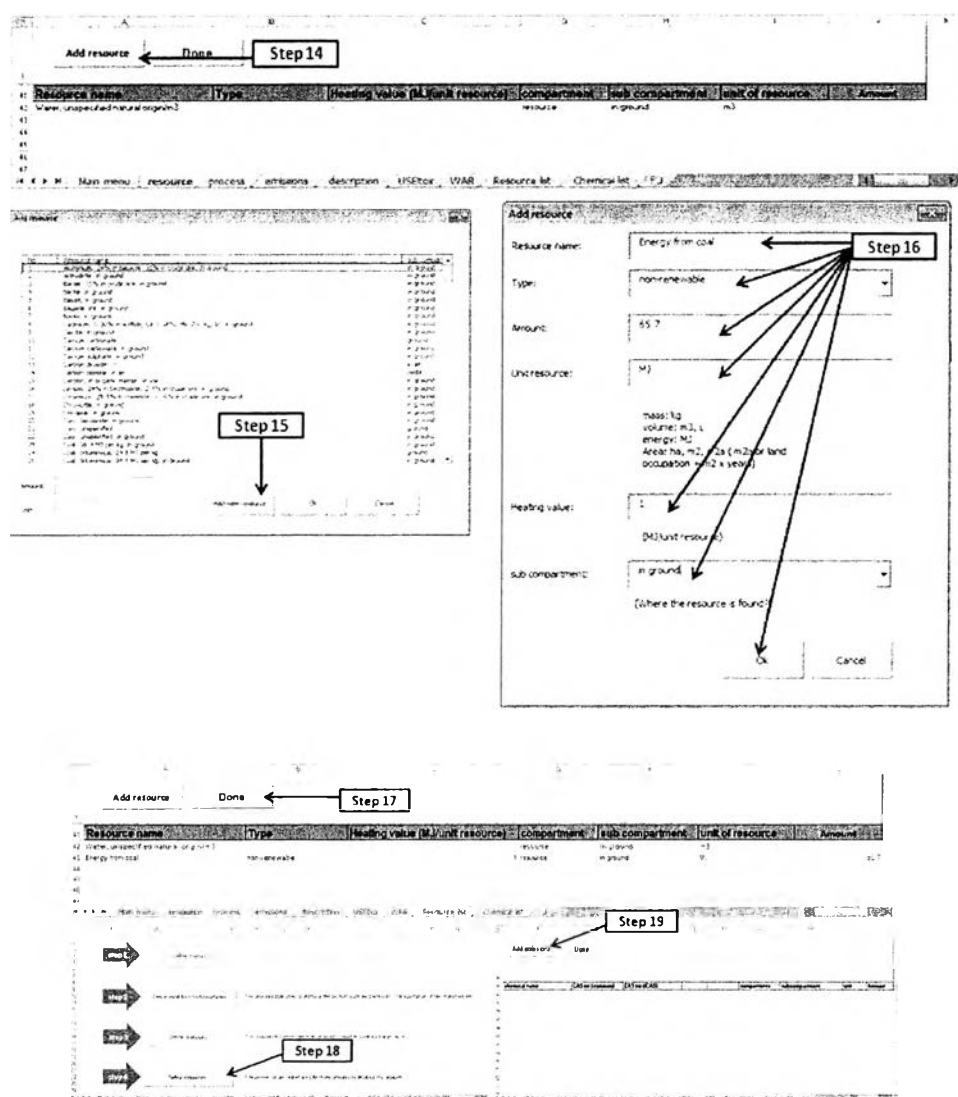


Figure A.47 Steps to generate LCI data (continued)

Step 20: “Add emissions” form will appear, select “common chemical” option button, select compartment (air, water, and soil), select sub compartment,

enter amount of the chemical, select unit of the chemical, and click “Ok” button (Figure A.48). In this case, carbon dioxide 4.4 kg is emitted to air, unspecified.

Do step 19-20 by changing chemical to Sulfur dioxide; compartment air; sub compartment unspecified 0.009 kg, Ethylene; compartment air; sub compartment unspecified 0.0016 kg, and Phosphate; compartment soil; sub compartment unspecified 0.026 kg (Figure A.49-A.51).

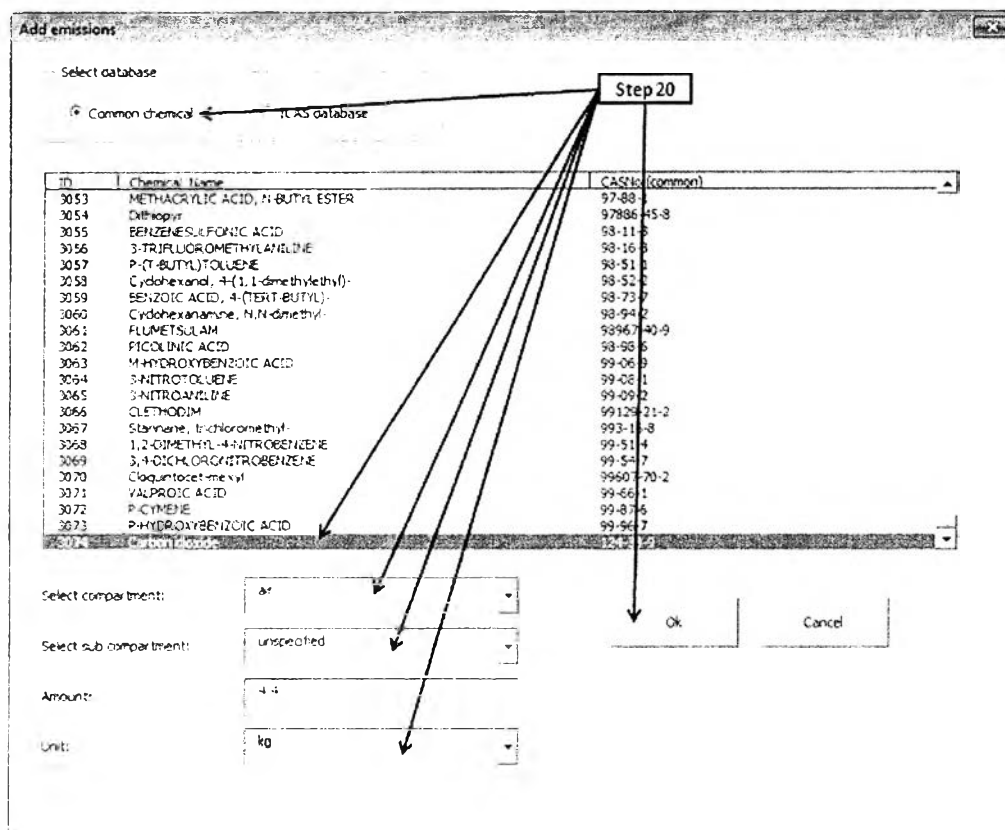


Figure A.48 Steps to generate LCI data (continued)

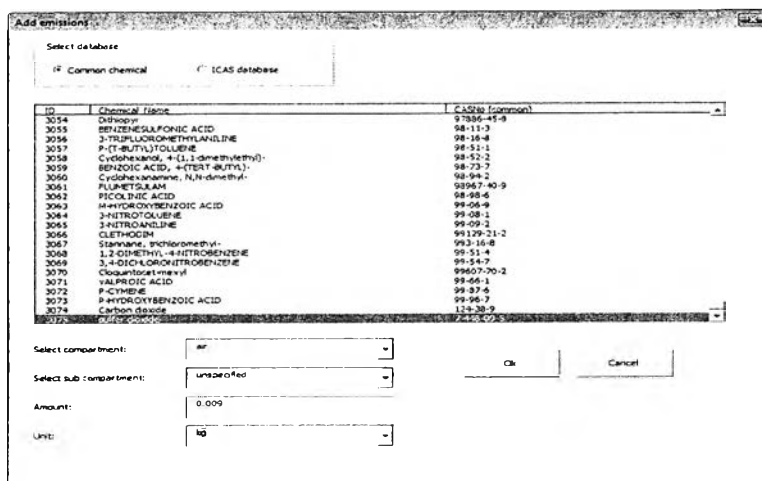


Figure A.49 Adding emitted chemical (Sulfur dioxide)

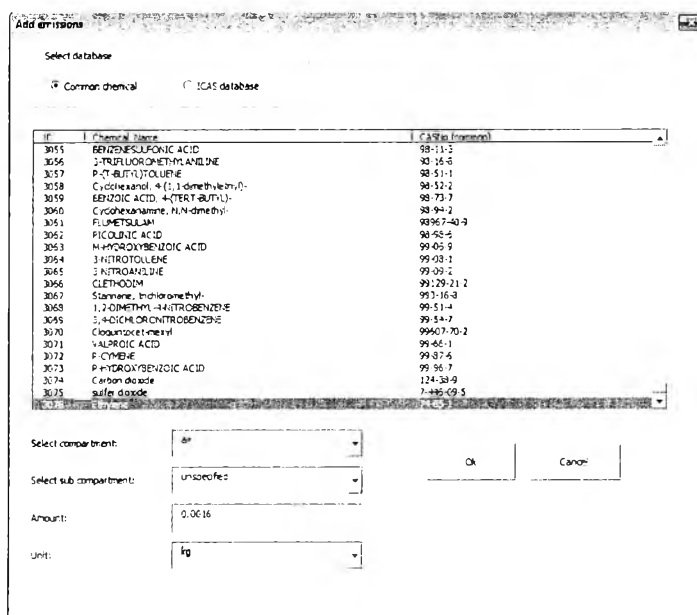


Figure A.50 Adding emitted chemical (Ethylene)

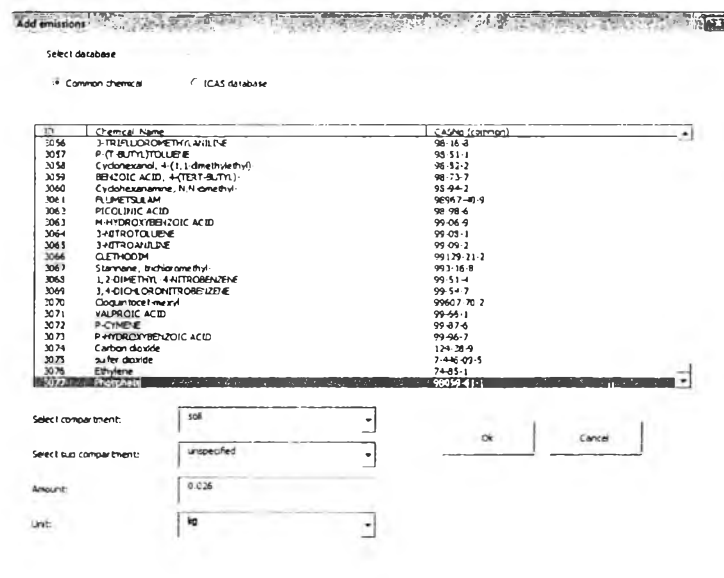


Figure A.51 Adding emitted chemical (Phosphate)

Step 21: Click “Done” button to finish adding emissions (Figure A.52).

Step 22: Click “Done” button in “Main menu” page to finish generation of LCI data.

Step 23: To use this LCI data, open LCSofT program file, select “Additional tools” page, and click “Activate LCI database” button.

Step 24: Click “Ok” to finish activation of LCI database.

To use this data, open “Input” form, select “Material”, select “Others” as shown in Figure A.53.

Screenshot of a software interface showing a table of chemical data and a step-by-step guide.

chemical name	CAE no (coems)	CAE no (ICAE)	compartment	techcompartment	unit	Amount
Carbon dioxide	160700		air	unspecified	kg	4.4
nitric dioxide	160701		air	unspecified	kg	0.03
nitrogen	160702		air	unspecified	kg	0.0016
Phosphorus	160703		soil	unspecified	kg	0.026

Navigation: Main menu | resource | process | emissions | description | USEtox | WAR | Resource list | Chemical list

Step 21: Done

Step 22: extended knowledge base in LCI database open Additional page in LCI database (Click activate LCI database)

Step 23: Microsoft Excel - Activation LCI database is complete

Step 24: OK

Diagram illustrating the steps to generate LCI data:

Step 21: Define Product

Step 22: Define inputs from technospheres (The process that used to produce the product such as Electricity, Transportation of raw materials etc)

Step 23: Define resources (The resources (renewable/non-renewable) used to produce the product)

Step 24: Define emissions (Emissions (to air, water, soil) from the process to produce the product)

Step 23: Microsoft Excel - Activation LCI database is complete

Step 24: OK

Figure A.52: Steps to generate LCI data (continued)

Input

Select inputs of the section | Use simulation results |

Select Input Type

Material
 Utility
 Transport

Select Material type: Others

No.	Name	unit
1	Novozymes Sorzyme	kg

Description

Location: US\Technology: current\Included processes: The assessment addresses environmental impacts potentials associated with enzyme production in a 'cradle-to-gate' perspective, i.e. all processes from production of raw materials to the final enzyme product at Novozymes' gate. \General comment: Technology: current, Included processes: The assessment addresses environmental impacts potentials associated with enzyme production in a 'cradle-to-gate' perspective, i.e. all processes from production of raw materials to the final enzyme product at Novozymes' gate.

Amount: _____

Unit: kg

Figure A.53 Input form

APPENDICES

Appendix B Manual of Interface of SustainPro, LCSOFT, and ECON

The purposes of this manual is to introduce the user to use this interface in order to perform process sustainability analysis, life cycle assessment (LCA), and economic evaluation throughout process design tools, SustainPro, LCSOFT, and ECON, respectively. Chapter 1 provides an introduction to interface of SustainPro, LCSOFT, and ECON, and the case study, chapter 2 provides outline a step by step procedure for conducting process sustainability analysis using SustainPro, Chapter 3 provides a step by step procedure for conducting LCA using LCSOFT, and Chapter 4 provides a step by step procedure for conducting economic evaluation using ECON. And appendix-A provides the activity diagram of the interface and data of the case study.

B.1 Introduction to Interface of SustainPro, LCSOFT, and ECON (Case study: Bioethanol from cassava rhizome)

Bioethanol is a type of biofuel produced from lignocellulosic matter. This case study, lignocelluloses from cassava rhizome is converted to ethanol by hydrolysis via fermentation. The reference for the process is obtained from Wooley et al. (NREL) and simulated using PRO/II simulation program.

The overview of process flow sheet is described as follows and can be divided into 5 main sections (Figure B.1):

1. Pretreatment section, to make the lignocellulosic material enable to be hydrolyzed.
2. Detoxification, the compounds that are toxic for the fermentation microorganisms are treated.
3. Simultaneous saccharification and co-fermentation (SSCF), to convert glucose and other sugars to ethanol.
4. Distillation, to distil the process stream from SSCF section until a mixture of nearly azeotropic water and ethanol is obtained.
5. Dehydration, the ethanol from distillation section is purified using vapor-phase molecular sieves in order to obtain bioethanol within the specifications.

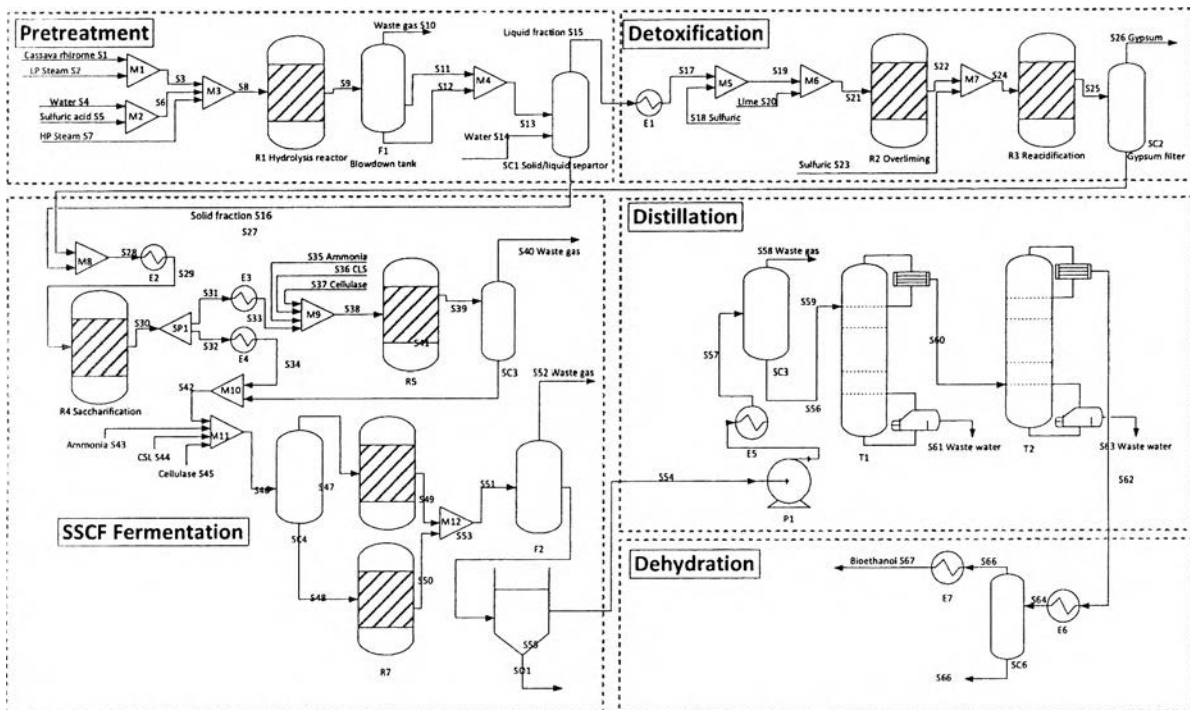


Figure B.1: Flowsheet of the base bioethanol production process implemented in PRO/II

B.2 Getting started with interface

‘Interface’ folder composes of sub folders; ‘Project Library’ folder is where interface data of the process of project is stored after performing process sustainability analysis, life cycle assessment (LCA), and economic evaluation; ‘Inteface.xlsm’ is the software for conducting process evaluation; ‘Software’ folder is where process design tools (SustainPro, LCSof, and ECON) are stored with the case study files and manual for each tool; ‘USER MANUAL_interface’ file is the manual to use this interface (see Figure B.2).

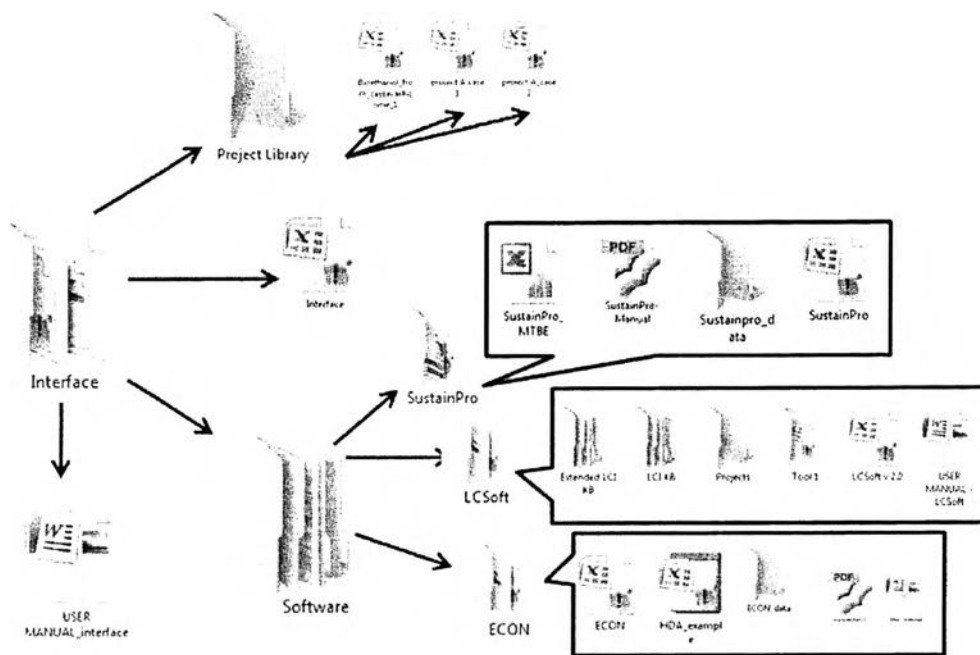


Figure B.2 The structure of process design tool interface.

B.3 Open and close interface

Step 1: Open “Interface” folder (Figure B.3).

Step 2: Open “Interface.xlsm” *Excel* file, “Start menu” page will appear.

Step 3: After conducting process evaluation, to close this interface click “Save and close” button. Interface data of this process or project will be automatically saved in “Project Library” folder and closed.

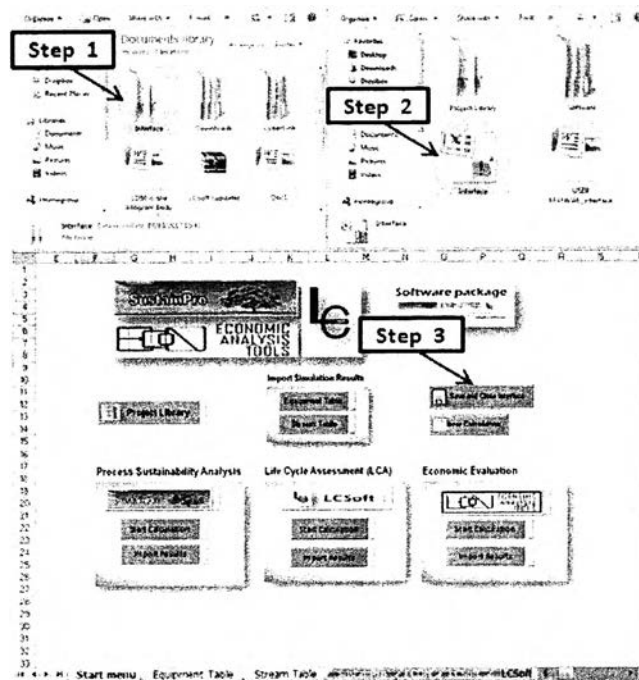


Figure B.3 Steps to open and close Interface of SustainPro, LCSoft, and ECON

B.4 Perform process evaluation thought out interface

Interface of SustainPro, LCSoft, and ECON composes of 5 main parts as shown in Figure B.4. (To see the example: Open “Interface” folder > “Project Library” folder > “Bioethanol_from_cassavarhizome_1.xlsm”)

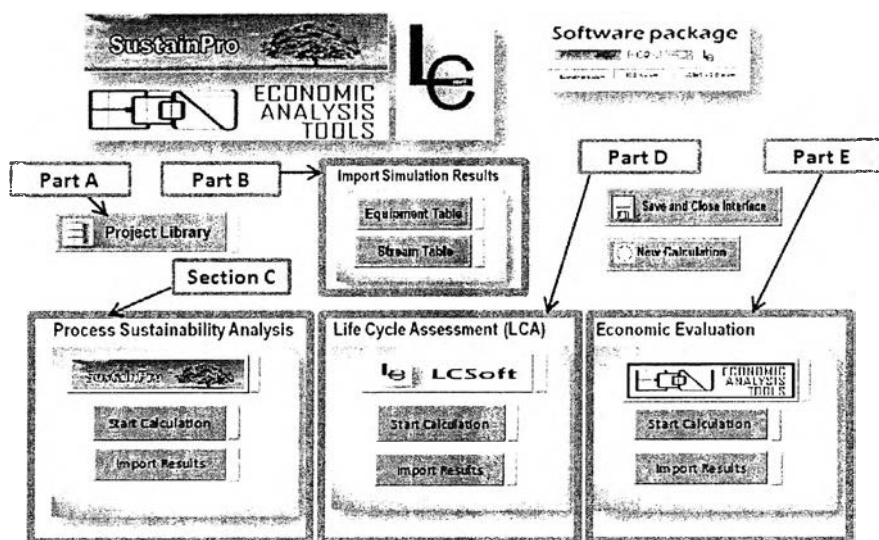


Figure B.4 Sections in interface of SustainPro, LCSoft, and ECON

Part A: Project Library is the first step to create the project before conducting other process design tools.

A1: Steps to create new project

Step 1: Open “Interface.xlsm” file, “Start menu” page will appear then click “Project Library” button as shown in Figure B.5.

Step 2: Click “New Project” button to create your own project.

Step 3: Enter project name (Bioethanol_from_cassavarhizome), case number (1), and select process type (continuous), then click “Ok” button. The blue tab near “Project Library” will appear that means your project name

“Bioethanol_from_cassavarhizome_1” has been created.

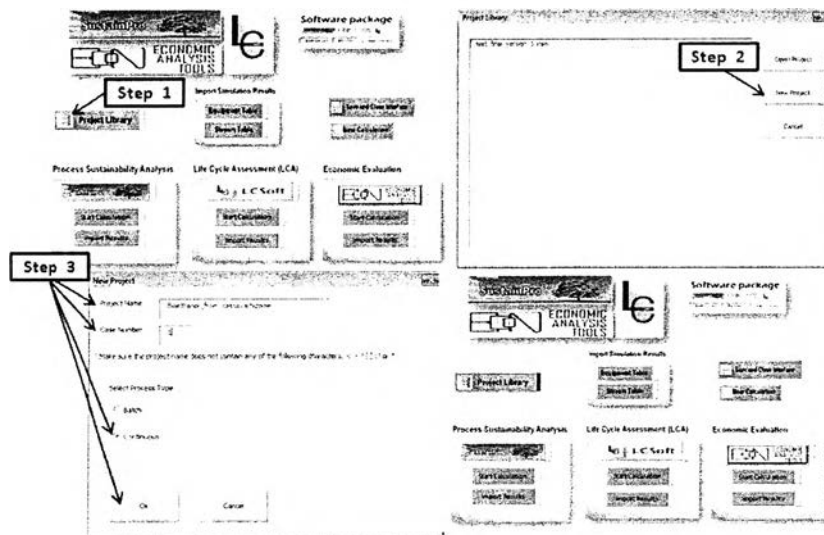


Figure B.5 Steps to create new project

Part B: Import simulation results which are stream table and equipment table that represent the mass and energy balance of the process, respectively, this interface provides options for import simulation results and the user should finish this section before conducting section C, D, and E.

B1: Steps to generate equipment table

Step 1: Click “Equipment table” button as shown in Figure B.6, “Equipment table” page will appear.

Step 2: Enter number of equipments in the process and click “Generate equipment table” button.

Step 3: Enter equipment name in row 8, equipment type in row 9, and identify the utility of the equipment.

- The type of equipment that should be inserted in the table are:

Reactor => “Reactor”

Compressor => “Pump”

Pump => “Pump”

Distillation Column => “Column”

Heat Exchanger => “HE”

Evaporator => “Evap”

Condenser => “Cond”

- The utility consumed by the equipment can be:

Heating duty (HP) in the unit “GJ/hr”

Heating duty (LP) in the unit “GJ/hr”

Cooling duty in the unit “GJ/hr”

Electrical work in the unit “kW”

Step 4: Click “Done” button to go back to the “Start menu” page.

(To see the example of equipment table: Open

“Bioethanol_from_cassavarhizome_1.xlsm” file > “Equipment table” page)

The screenshot displays the software interface for creating a new project. It features a 'Software package' section with logos for 'SustainPro' and 'ECONOMIC ANALYSIS TOOLS'. Below this, there are three main analysis modules: 'Process Sustainability Analysis', 'Life Cycle Assessment (LCA)', and 'Economic Evaluation'. The 'Equipment table' window is open, showing a 'Number of equipments' field set to 33. The 'Stream table' window is also open, showing a table with columns for equipment names and various energy flows. Arrows labeled 'Step 1', 'Step 2', 'Step 3', and 'Step 4' indicate the sequence of actions.

Equipment name	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14	E15	E16	E17	E18	E19	E20	E21	E22	E23	E24	E25
Type	Reactor	HE	Reactor	HE	HE	Reactor	Reactor	Reactor	Reactor	Pump	HE	HE	HE	Column	Column										
Heating duty (HP) (GJ/hr)	0													3475	6308										
Heating duty (LP) (GJ/hr)																									
Cooling duty (GJ/hr)	2.0000	0.4	0.001	1.00	1.00	1.00	1.00	2.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
Electrical work (kW)																									

Figure B.6: Steps to create new project

B2: Steps to generate stream table

Step 1: Click “Stream table” button and “Stream table” page will appear as shown in Figure B.7.

Step 2: Enter number of stream (67), and number of component (25) then click “Generate stream table”.

(To see the example of stream table: Open

“Bioethanol_from_cassavarhizome_1.xlsm” file > “Stream table” page)

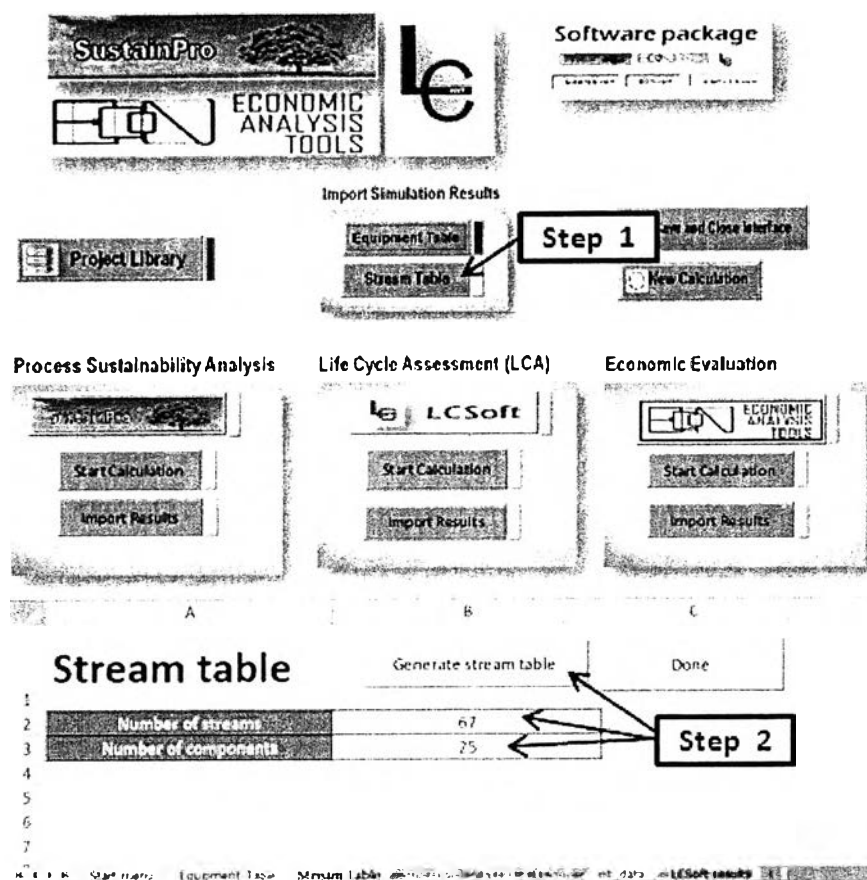


Figure B.7: Steps to generate stream table

Step 3: Enter the component name in column A, molecular weight, hazardous (“Yes” if the component is hazardous), type of each component follow the given key; Raw Material: RM; Final Product: P; By-product: BP; Inert: I; Solvent: S, and price of each component as shown in Figure B.8.

Stream table		Generate stream table		Done		
1						
2	Number of streams	67	Step 3			
3	Number of component	25				
4						
5						
6	Stream name				\$1	\$2
7	Initial Unit				0	0
8	Final Unit					
Component name	Molecular weight (g/mol)	Hazardous (Yes)	Type (RM/BP/HS)	Price (\$/kg)	M1	M1
9 Cellulose	162.14	Yes	RM	0.017	4680.592	0.0
10 Hemicellulose	132.11	Yes	RM	0.017	4474.040	0.0
11 Lignin	122.43	Yes	RM	0.017	3653.449	0.0
12 Glucose	180.16	Yes	BP	0	0.000	0.0
13 Xylose	150.13	Yes	BP	0	0.000	0.0
14 Cellulose	342.3	Yes	BP	0	0.000	0.0
15 Ethanol	46.06	Yes	P	1.14	0.000	0.0
16 Water	18.02	Yes	RM	0.001	129.522	785.0
17 Sulfuric Acid	98.08	Yes	RM	0.6	0.000	0.0
18 Furfural	96.08	Yes	BP	0	0.000	0.0
19 Ammonia	17.03	Yes	RM	0.565	0.000	0.0
20 Oxygen	31.99	Yes	BP	0	0.000	0.0
21 Carbon Dioxide	44	Yes	BP	0	0.000	0.0
22 Glycerol	92.09	Yes	BP	0	0.000	0.0
23 Succinic Acid	118.09	Yes	BP	0	0.000	0.0
24 Lactic Acid	90.08	Yes	BP	0	0.000	0.0
25 HMF	126.11	Yes	BP	0	0.000	0.0
26 Xylitol	152.14	Yes	BP	0	0.000	0.0
27 Acetic Acid	60.05	Yes	BP	0	0.000	0.0
28 Corn Steep Liquor	18.02	Yes	RM	0.8	0.000	0.0
29 2M	24.63	Yes	RM	0	0.000	0.0
30 Cellulose	75.66	Yes	RM	5	0.000	0.0
31 Lime	74.09	Yes	BP	0.4	0.000	0.0
32 CASO4	136.14	Yes	BP	0	0.000	0.0
33 Ash	75.32	Yes	BP	0.0018	578.255	0.0
34	Total Mass Flow (kg/hr)				15715.849	785.0
35	Temperature (degrees Celsius)				30.000	160.0
36	Pressure (kPa)				1.000	4.0

Figure B.8 Steps to generate stream table (Continued)

Step 4: Input initial units and final units. Type the name of the unit (equipment name) where the stream starts exactly with the same name as the name specified in the equipment table in row 6. If the stream starts outside the process (fresh feed) the number zero should be written instead of a unit name. Type in row 7 the name of the unit where the stream ends, exactly, with the same name as the name specified in the equipment table. If the stream ends outside the process (demand/exit stream) the number zero should be written instead of a unit name (see Figure B.9).

Step 5: Enter mass flow rate (kg/hr) of each component in the specific stream, total mass flow rate (kg/hr), and enthalpy flow (GJ/hr) for each stream (see Figure B.9).

Step 6: Click "Done" after finishing adding all data. In "Start menu" page, the blue line near "Stream table" button will appear that means the user has completed input stream and equipment table and ready to do the next part.

Stream table

Number of streams: 23
Number of components: 18

	S1	S2	S3	S4	S5	S6	S7	S8	S9
Component name									
Cellulose	4182.53	1.03E-006	4600.53	0.000	0.000	0.000	0.000	4600.53	1.279 E-006
Hexachlorobenzene	66.74 E-06	1.03E-006	66.74 E-06	0.000	0.000	0.000	0.000	66.74 E-06	2.11 E-06
Lignin	3851.44	1.03E-006	3851.44	0.000	0.000	0.000	0.000	3851.44	3.851 E-006
Cellulose	0.000	1.03E-006	0.000	0.000	0.000	0.000	0.000	0.000	364.647
Air	0.000	1.03E-006	0.000	0.000	0.000	0.000	0.000	0.000	4823.72

Stream name	S1	S2	S3	S4	S5	S6	S7	S8	S9
Initial Unit	0	0	0	0	0	0	0	0	0
Final Unit	141	141	141	141	141	141	141	141	141
Component name									
Cellulose	32.14	Yes	PSM	0.017	44.74 E-06	0.000	66.74 E-06	0.000	0.000
Hexachlorobenzene	132.43	Yes	PSM	0.017	36.82 E-06	0.000	66.74 E-06	0.000	0.000
Lignin	130.11	Yes	BP	0	3.400	3.500	0.000	0.000	0.000
Cellulose	342.3	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Ethanol	46.06	Yes	P	1.14	0.000	0.000	0.000	0.000	0.000
Water	31.02	Yes	FSM	0.1971	579.82	785.45	794.61	6372.76	6.000
Sulfuric Acid	58.08	Yes	FSM	0.6	0.000	0.000	0.000	99.44	33.64
Furfural	36.19	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Ammonia	17.00	Yes	FSM	0.565	0.000	0.000	0.000	0.000	0.000
Oxygen	31.77	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Carbon Dioxide	44	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Glycerol	32.09	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Sulfuric Acid	119.31	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Lactic Acid	30.18	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
IBMF	28.71	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Lignin	152.14	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Sulfuric Acid	60.19	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
1,6-Hexanediol	31.02	Yes	FSM	0.5	0.000	0.000	0.000	0.000	0.000
PM	24.63	Yes	FSM	0	0.000	0.000	0.000	0.000	0.000
Cellulose	75.66	Yes	FSM	5	0.000	0.000	0.000	0.000	0.000
Lena	74.05	Yes	BP	0.4	0.000	0.000	0.000	0.000	0.000
CASDH	1.38.14	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Air	75.32	Yes	BP	0.0018	578.29	0.000	578.29	0.000	0.000
Total Mass Flow (kg/hr)	97.78 E-01				788.89	850.30	4372.76	99.44	5071.55
Temperature (Average Celsius)	25.000				100.000	100.000	25.000	25.000	25.000
Pressure (kPa)	1.013				0.000	0.000	0.000	0.000	0.000
Enthalpy Flow (kJ/hr)	-4.224				-2.163	-24.28	0.522	0.000	8.776

Stream tab

Number of streams: 23
Number of components: 18

Stream name	S1	S2	S3	S4	S5	S6	S7	S8	S9
Component name									
Cellulose	32.14	Yes	PSM	0.017	44.74 E-06	0.000	66.74 E-06	0.000	0.000
Hexachlorobenzene	132.43	Yes	PSM	0.017	36.82 E-06	0.000	66.74 E-06	0.000	0.000
Lignin	130.11	Yes	BP	0	3.400	3.500	0.000	0.000	0.000
Cellulose	342.3	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Ethanol	46.06	Yes	P	1.14	0.000	0.000	0.000	0.000	0.000
Water	31.02	Yes	FSM	0.1971	579.82	785.45	794.61	6372.76	6.000
Sulfuric Acid	58.08	Yes	FSM	0.6	0.000	0.000	0.000	99.44	33.64
Furfural	36.19	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Ammonia	17.00	Yes	FSM	0.565	0.000	0.000	0.000	0.000	0.000
Oxygen	31.77	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Carbon Dioxide	44	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Glycerol	32.09	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Sulfuric Acid	119.31	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Lactic Acid	30.18	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
IBMF	28.71	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Lignin	152.14	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Sulfuric Acid	60.19	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
1,6-Hexanediol	31.02	Yes	FSM	0.5	0.000	0.000	0.000	0.000	0.000
PM	24.63	Yes	FSM	0	0.000	0.000	0.000	0.000	0.000
Cellulose	75.66	Yes	FSM	5	0.000	0.000	0.000	0.000	0.000
Lena	74.05	Yes	BP	0.4	0.000	0.000	0.000	0.000	0.000
CASDH	1.38.14	Yes	BP	0	0.000	0.000	0.000	0.000	0.000
Air	75.32	Yes	BP	0.0018	578.29	0.000	578.29	0.000	0.000

Software screenshot showing 'Stream tab' and various analysis tools:

- Process Sustainability Analysis
- Life Cycle Assessment (LCA)
- Economic Evaluation
- Project Library
- Stream Table
- Equipment Table

Figure B.9 Steps to generate stream table (Continued)

Part C: Perform process sustainability analysis using SustainPro thought out interface which is related process data is entered in this step and is sent to “SustainPro.xlsxm” to calculate the sustainability factors of the process and the results will be imported from “SustianPro.xlsxm” to this interface.

Part D: Perform LCA using LCSofT thought out interface which related process data is entered in this step and is sent to “LCSofT v2.0.xlsxm” to calculate the potential environmental impacts (Environmental impacts), carbon footprint, and resource and energy consumption and this results will be imported from “LCSofT v2.0.xlsxm” to this interface.

Part E: Perform economic evaluation using ECON thought out interface which related process data is entered in this step and is sent to “ECON.xlsxm” to calculate capital cost, operating cost, and other economic factors, and the results will be imported from “ECON.xlsxm” to this interface.

In this manual, process sustainability analysis using SustainPro, LCA, and economic analysis using ECON thought out this interface will be respectively performed.

B.5 Process sustainability analysis using SustainPro thought out interface:

Steps to enter sustainability data

After stream table and equipment table are generated, interface allows the user to conduct sustainability analysis using SustainPro.

Step 1: Click “SustainPro” button in “Start menu” page (Figure B.10), “SustainPro” page will appear.

Step 2: Enter number of reactions, operating time per year (hr), and number of desired product in [B2], [B3], and [B6], respectively.

Step 3: Click “Generate Table A” button, interface will create tables for entering sustainability related data for conducting sustainability analysis.

Step 4: Look for the blue tap “Reaction data”, enter the name of product stream (S67 for the case study)

Step 5: “Reaction\Stoichiometric Coefficients” table, specify for all the rows in the table (reactions), the components belonging to that reaction writing the stoichiometry coefficients for each component and the respective reaction. If the component is a reactant a minus sign should be inserted before the stoichiometry coefficient, if the component is a product the stoichiometry coefficients should be positive. Regarding to components not involved in the reaction the respective cells should remain empty.

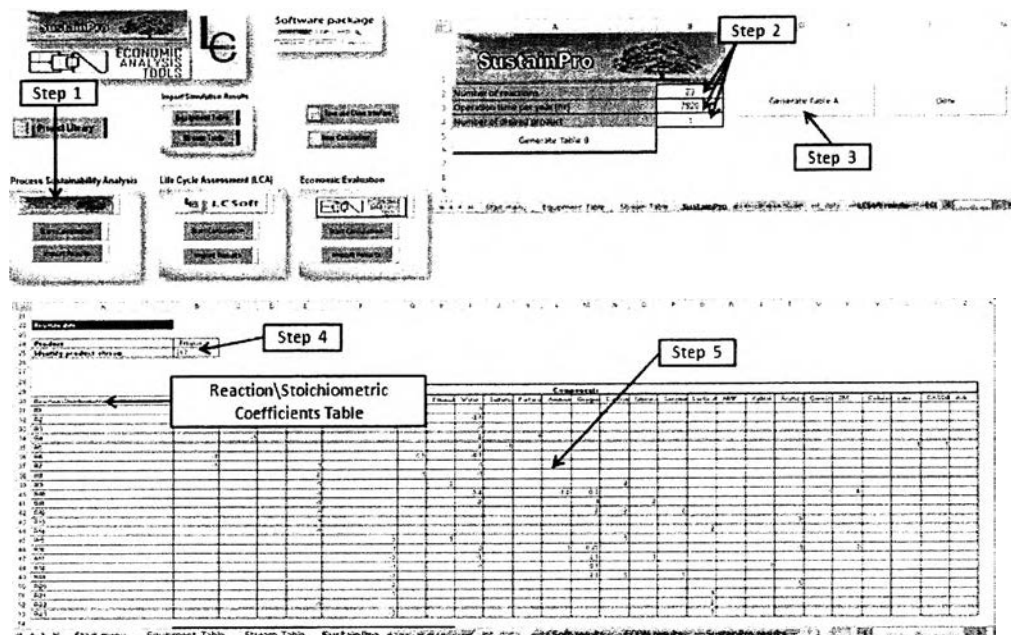


Figure B.10: Steps to perform sustainability analysis (Continued)

Step 6: Look for “Reaction/Reactor” table (Figure B.10). The figure below is the example of entering reaction data in “Reaction/Reactor” table.

	Units
R 1	R1
Component	TOLUENE
R 2	0.75
Component	TOLUENE

C1. Specify the conversion of the reaction in the correspondent reactor.

$\%C =$

$$\frac{M_{in}^{(c)} - M_{out}^{(c)}}{M_{in}^{(c)}} \quad (C.1)$$

Where $M_{in}^{(c)}$ is the mass flow rate of compound c entering the reactor and $M_{out}^{(c)}$ is the mass flow rate of compound c leaving the reactor

C2. Specify the name of the component c , which was considered in the conversion calculation.

Step 7: If fuel credit can be given to any of the demand streams present in the process, type the word “Fuel” in the table’s row called type. With fuel credit we mean a stream that might be burn. Do not insert the components prices in those

columns defined with Fuel type. Those columns should become empty, since they are going to be calculated by the software.

Step 8: Type the sale price for the demand/exit streams that do not have a fuel credit in the respective cells. If the demand stream requires treatments or any kind of other expenses, the price of those streams should be specified with a minus signal before the cost price.

Step 9: Click “Generate Table B” button.

The image shows two screenshots from the SustanPro software interface. The left screenshot displays the 'Reaction/Reactor Table' with columns for Unit, R1, R2, R3, R4, R5, R6, and R7. A box labeled 'Step 6' points to the 'Units' column. The right screenshot displays the 'Demand stream price tab' with columns for Type, S10, S20, S40, S52, S53, S5F, S61, S62, S65, and S67. A box labeled 'Step 7' points to the 'Generate Table B' button, and a box labeled 'Step 8' points to the price cells. A box labeled 'Step 9' also points to the 'Generate Table B' button.

Figure B.11: Steps to perform sustainability analysis (Continued)

Step 10: Look for “Utility price tap”, insert the prices for all the utilities specified in the table (Figure B.12).

Step 11: Specify the different properties for each component in the respective unit in the empty cells that appear below the name of the property. The properties are determined for the inlet conditions, except in the cases of units with phase exchange. In those cases the conditions used to determine the properties should be the ones present in the outlet stream. The properties names are displayed automatically by the program, depending on the type of process involved in each unit.

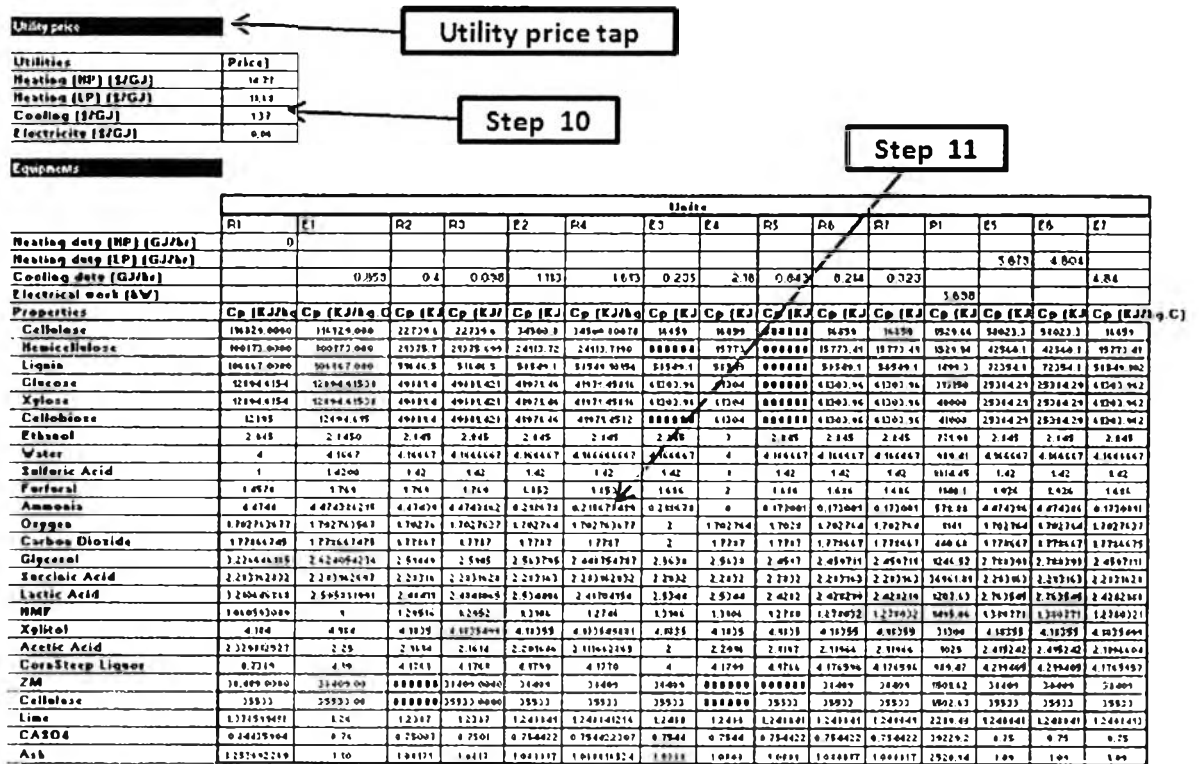


Figure B.12: Steps to perform sustainability analysis (Continued)

Step 12: Specify the outlet streams from the reboiler and the condenser in the distillation columns table. Remind that the name of the stream should be written exactly in the same way as specified before in streams (Figure B.13).

Step 13: Specify the different properties for each component in the empty cells that appear below the name of the property. The properties are determined for the inlet conditions, except in the cases of units with phase exchange. In those cases the conditions used to determine the properties should be the ones present in the outlet stream. The properties names are displayed automatically by the program, depending on the type of process involved in each unit.

Step 14: Look for "Safety" tap, fill out the table with the information specified in the table. Use MSDS sheets.

Step 15: Click "Done" button in "SustainPro" page to go back to "Start menu" page.

Distillation column Table

Step 12

Distillation column	T1	T2	Reboiler (HP)	24.89	14.371
Condenser	16.089	19.971	Reboiler (LP)		
Streams	S60	S62	Streams	S61	S63

Hu (kJ/kg)

Cellulose			Cellulose		
Hemicellulose			Lignin		
Lignin			Glucose	388.203	388.203
Glucose	371.128	371.128	Xylose	388.203	388.203
Xylose	371.128	371.128	Cellobiose	388.203	388.203
Cellobiose	371.128	371.128	Ethanol	772.542	783.235
Ethanol	808.345	809.721	Water	2,216.20	2,235.47
Water	0.000	0.000	Sulfuric Acid	205.671	279.022
Sulfuric Acid	455.385	460.707	Furfural	591.595	595.082
Furfural	603.467	603.720	Ammonia	512.69	603.19
Ammonia	768.34	776.57	Oxygen		
Oxygen			Carbon Dioxide		
Carbon Dioxide			Glycerol	931.11	935.22
Glycerol	945.03	945.60	Succinic Acid	514.569	509.177
Succinic Acid	496.213	495.822	Lactic Acid	705.435	696.021
Lactic Acid	673.386	672.703	HMF	364.31	366.91
HMF	0.000	0.000	Xylitol	1152.291	1132.904
Xylitol	1086.290	1084.883	Acetic Acid	448.706	455.235
Acetic Acid	470.933	471.406	CornSteep Liquor	2,216.20	2,235.47
CornSteep Liquor	2,282.65	2,282.65	ZM		
ZM			Cellulase		
Cellulase			Lime		
Lime			CASO4		
CASO4			Ash		
Ash					

Step 13

Safety tap

Step 14

Components	Flash Point	Boiling Point (°C)	LEL (%vol)	LEL (%vol)	Toxic limit (ppm)
Cellulose	No	552.25	Non-explosive	Non-explosive	Asphyxiant
Hemicellulose	No	552.25	Non-explosive	Non-explosive	Asphyxiant
Lignin	No	552.25	Non-explosive	Non-explosive	Asphyxiant
Glucose	No	552.25	Non-explosive	Non-explosive	Asphyxiant
Xylose	No	552.25	Non-explosive	Non-explosive	Asphyxiant
Cellobiose	No	552.25	Non-explosive	Non-explosive	Asphyxiant
Ethanol	13	78.29	19	3.3	1000
Water	No	100	Non-explosive	Non-explosive	Asphyxiant
Sulfuric Acid	No	336.85	Non-explosive	Non-explosive	0.25
Furfural	60	161.7	19.3	2.1	2
Ammonia	No	-33.43	28	15	25
Oxygen	No	182.96	Non-explosive	Non-explosive	Asphyxiant
Carbon Dioxide	No	-78.48	Non-explosive	Non-explosive	Asphyxiant
Glycerol	199	287.85	Non-explosive	0.9	133
Succinic Acid	No	317.85	Non-explosive	Non-explosive	Asphyxiant
Lactic Acid	213	216.85	Non-explosive	Non-explosive	Asphyxiant
HMF	No	259.55	Non-explosive	Non-explosive	Asphyxiant
Xylitol	No	216	Non-explosive	Non-explosive	Asphyxiant
Acetic Acid	42.77	117.9	16	4	10
CornSteep Liquor	No	100	Non-explosive	Non-explosive	Asphyxiant
ZM	No		Non-explosive	Non-explosive	Asphyxiant
Cellulase	No		Non-explosive	Non-explosive	Asphyxiant
Lime	No		Non-explosive	Non-explosive	4.94
CASO4	No	163	Non-explosive	Non-explosive	10
Ash	No		Non-explosive	Non-explosive	Asphyxiant

Step 15

SustainPro

Number of reactions	23
Operation time per year (hr)	7920
Number of desired product	1

Generate Table A Done

Start menu Equipment Table Stream Table SustainPro xsoft_data xsoft_results ECO

Figure B.13: Steps to perform sustainability analysis (Continued)

B.6 Steps to calculate flowsheet decomposition in SustainPro

SustainPro will perform decomposition of the process/operation flowdiagram in terms of: MCP (mass closed-path); MOP (mass opened-path); ECP (energy closed-path); and EOP (energy open-path). For batch processes, the following AP (accumulation-paths) will also appear. To see details of any of these, click on the respective “boxes” and click on “Go to Main Menu” to return to the main menu page.

Step 1: Click “Start Calculation” button in “Start menu” page below “SustainPro” button. Interface will send sustainability data to SustainPro and “Main menu” page of SustainPro will appear (Figure B.14).

Step 2: Click “General data” button in “Main menu” page.

Step 3: Check data in this sheet. Click “Go to Main Menu” button to back to “Main Menu” page.

Step 4: Click “Streams” button (Figure B.15).

Step 5: Click “Reactive units” button in “Streams” page. Enter heat of the reaction in the new generated column (SustainPro-Manual, page 19-21).

Step 6: Click “Go to Main Menu” button to back to “Main Menu” page.

Step 7: Do step 3 and step 4 with “MCP”, “MOP”, “ECP”, “EOP”, “AP” buttons, respectively.

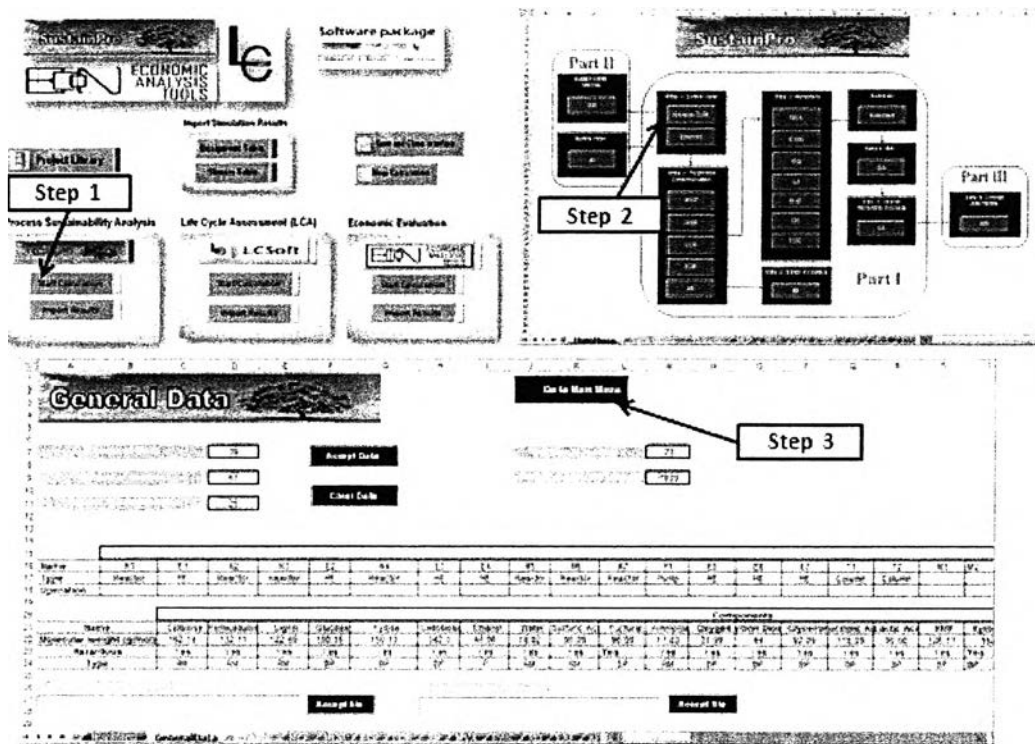


Figure B.14: Steps to perform sustainability analysis (Continued)

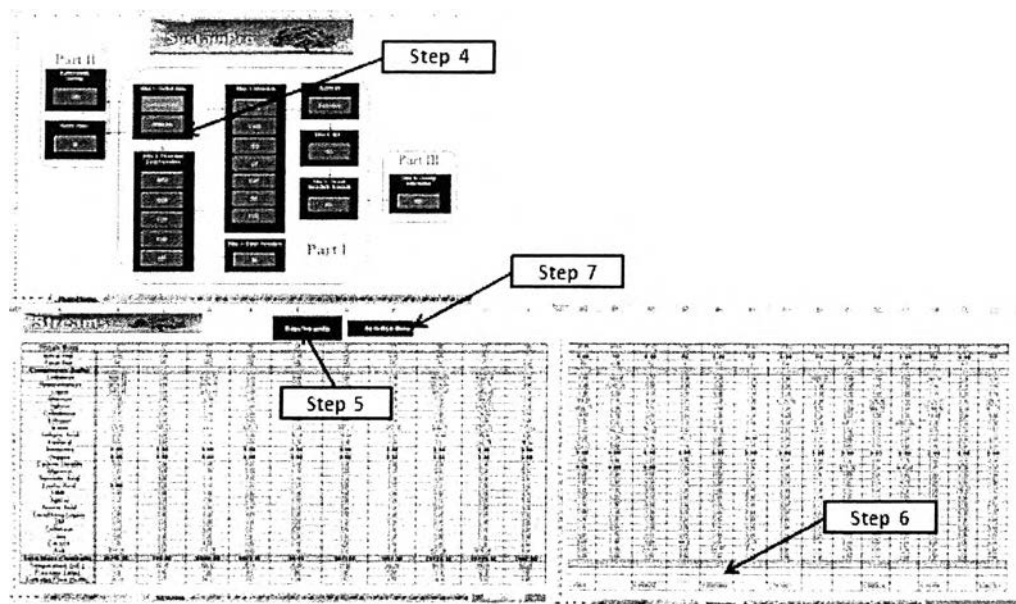


Figure B.15: Steps to perform sustainability analysis (Continued)

B.7 Steps to calculate indicators in SustainPro

The following indicator will be calculated without further addition of data, AF (Accumulation Factor). The following indicators will need additional data to calculate them, MVA (Material Value Added), EWC (Energy Waste Cost) and RQ (Reaction Quality). Click on the respective “orange” box for each indicator. The following indicators, EAF (Energy Accumulation Factor), DC (Demand Cost), and TDC (Total Demand Cost) are calculated automatically after the calculations of EWC.

Step 1: Click “MVA” button in “MVA” page (Figure B.16).

Step 2: Click “MVA” button in “MVA” page.

Step 3: Click “Go to Main Menu” page to back to “Main Menu” page.

Step 4: Do step 1 to step 3 with “EWC”, “RQ”, “AF”, “EAF”, “DC”, and “TDC” buttons.

- For “RQ” button, click “Reactive units” button in “RQ” page. The table for entire open- and closed-path will appear. Fill out all information in the table. It is necessary to specify for the entire open- and closed-path the value of E (see see SustainPro-Manual, page 42-44).
- For “BI” button, This Excel sheet is only applied for batch processes . Here all operational and compound indicators are calculated (see SustainPro-Manual, page 49-50).

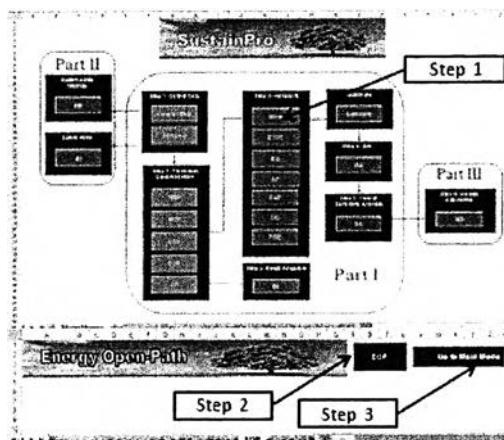


Figure B.16: Steps to perform sustainability analysis (Continued)

- Step to calculate summary results, indicator sensitivity analysis (ISA), sensitivity analysis (SA), new design (ND), sustainability metrics (SM), and safety index (SI) , see SustainPro-Manual, p 51-65.

B.8 Steps to import SustianPro results to interface

Step 1: Open “Bioethanol_from_cassavarhizome_1” window (FigureB.17).

Step 2: Select “Start menu” page and click “Import Results” button.

Step 3: Select results from SustianPro.

Step 4: Click “Ok” button. Results from SustianPro will be imported.

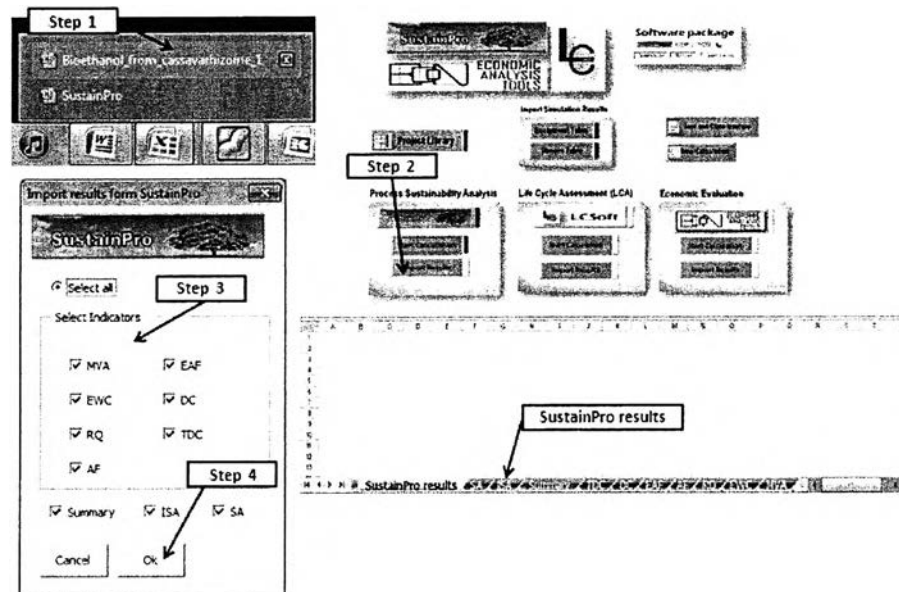


Figure B.17: Steps to perform sustainability analysis (Continued)

B.9 LCA using LCSof thought out interface:

Steps to enter LCA data

LCA data of the case study mentioned in USER MANUAL – LCSof will be used.

Step 1: Click “LCSof” button in “Start menu” page (Figure B.18).

Step 2: Click “Step 1” button in “LCSof” page.

Step 3: Select product name, product stream, and number of sections then click “Ok” button.

Step 4: Enter the name and description (optional) for each LCA section.

Step 5: Click “Step 2” button. Table of input and output for each section will appear.

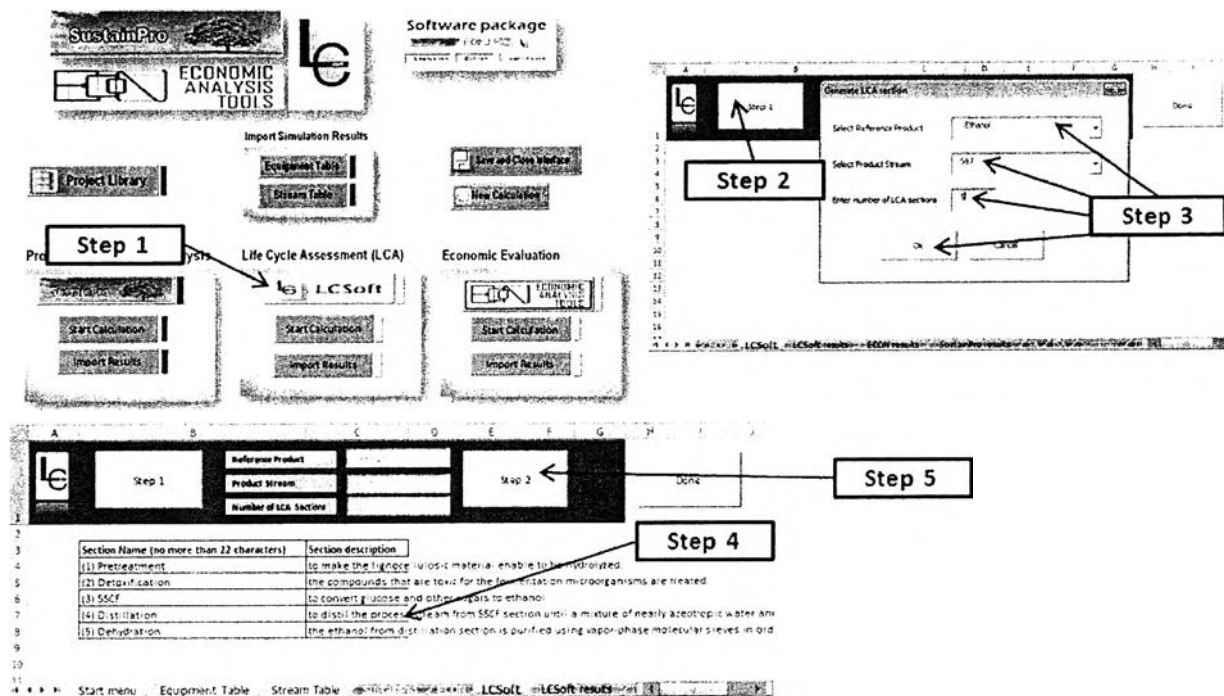


Figure B.18: Steps to perform LCA

Step 6: Enter input stream names, input equipment names and output streams names for each section (Figure B.20). The names should be the same with the name that appear in the table of stream and equipment names as shown below.

Input Streams:	S1	S2	S4	S5	S7	S14	S18	S20	S23	S35	S36	S37	S43	S44	S45
Input Equipments:	E3	E2	E5	E2	E4	E3	E4	E5	E6	E7	E1	E5	E4	E2	E1
Output Streams:	S10	S26	S40	S52	S55	S58	S61	S63	S66	S67					

Table of stream and equipment name

Figure 19: the table of stream and equipment names

For each output stream, compartment and its sub compartment must be specified.

Compartment: air Sub compartment: unspecified, urban air, and continental rural air

Compartment: water Sub compartment: unspecified, and continental fresh water

Compartment: soil Sub compartment: unspecified, continental natural soil, continental, and agricultural soil

Type exactly these words into the table of each section!

Section:	(1) Pretreatment						
Comment:	to make the lignocellulosic material enable to be hydrolyzed.						
Input streams:	S1	S2	S4	S5	S7	S14	
Input equipments:							
Output streams:	S10						
Compartment (air/water/soil)	air						
Sub compartment	unspecified						
Section:	(2) Detoxification						
Comment:	the compounds that are toxic for the fermentation microorganisms are treated.						
Input streams:	S18	S20	S23				
Input equipments:	E1	R2	R3				
Output streams:	S26						
Compartment (air/water/soil)	soil						
Sub compartment	unspecified						
Section:	(3) SSCF						
Comment:	to convert glucose and other sugars to ethanol.						
Input streams:	S43	S44	S45	S35	S36	S37	
Input equipments:	E2	R4	E3	E4	R5	R6	R7
Output streams:	S40	S52	S55				
Compartment (air/water/soil)	air	air	soil				
Sub compartment	unspecified	unspecifi	unspecified				
Section:	(4) Distillation						
Comment:	to distil the process stream from SSCF section until a mixture of nearly azeotropic water						
Input streams:							
Input equipments:	P1	E5	T1-condens	T2-condenser			
Output streams:	S58	S61	S63				
Compartment (air/water/soil)	air	water	water				
Sub compartment	unspecified	unspecifi	unspecified				
Section:	(5) Dehydration						
Comment:	the ethanol from distillation section is purified using vapor-phase molecular sieves in c						
Input streams:							
Input equipments:	E6	E7					
Output streams:	S66						
Compartment (air/water/soil)	air						
Sub compartment	unspecified						

Figure B.20: Steps to perform LCA (Continued)

B.10 Steps to calculate LCA results

Step 1: Click “Start Calculation” button below “LCSoft” button (Figure B.21).

Step 2: LCSoft will shown the messages, click “Ok” button for every messages.

Step 3: Click “Define stream component” button.

Step 4: Define each component in the process (see USER MANUAL – LCSoft, p 18-p 20).

Step 5: Click “Done” button to back to “Main menu” page.

Step 6: Click “Define equipments/utilities” button.

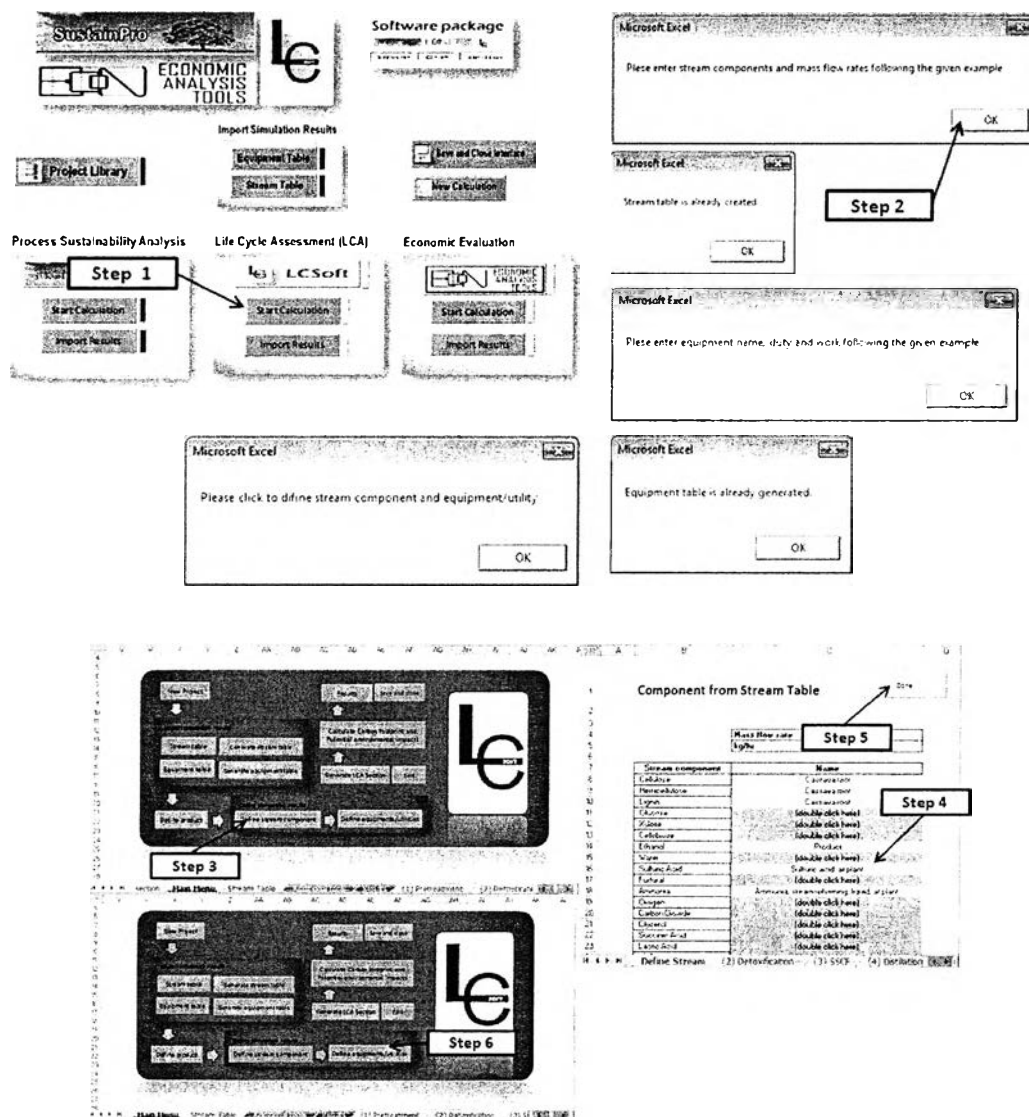


Figure B.21: Steps to perform LCA (Continued)

Step 7: Click define utility of each equipment (see USER MANUAL – LCSOFT, p 21- p 23) (Figure B.22). Environmental impacts, carbon footprint, energy consumption, and resource lists will be calculated.

The screenshot displays the 'Utility Entry' table with the following data:

Unit	Type of unit	Duty/Work	Unit	Energy	Time	Activity	Energy source
E1	HE		0.552	GJ	hr	Cooling	(double click here)
E2	Reactor		0.41	GJ	hr	Cooling	(double click here)
E3	Reactor		0.251	GJ	hr	Cooling	(double click here)
E4	HE		1.111	GJ	hr	Cooling	(double click here)
E5	Reactor		1.811	GJ	hr	Cooling	(double click here)
E6	HE		0.285	GJ	hr	Cooling	(double click here)
E7	HE		0.78	GJ	hr	Cooling	(double click here)
E8	Reactor		0.843	GJ	hr	Cooling	(double click here)
E9	Reactor		0.234	GJ	hr	Cooling	(double click here)
E10	Reactor		0.223	GJ	hr	Cooling	(double click here)
E11	Pump		5.036	kWh	hr	Flare gas usage	(double click here)
E12	HE		8.173	GJ	hr	Heating	(double click here)
E13	HE		4.804	GJ	hr	Heating	(double click here)
E14	HE		4.84	GJ	hr	Cooling	(double click here)
E15	Condenser	Column	10.065	GJ	hr	Cooling	(double click here)
E16	Condenser	Column	19.313	GJ	hr	Cooling	(double click here)

The main menu below the table includes buttons for 'New Project', 'Results', 'Save and close', 'Stream table', 'Generate stream table', 'Calculate Carbon footprint and Potential environmental impacts', 'Equipment table', 'Generate equipment table', 'Generate LCA Section', 'Edit', 'Define product', 'Define stream component', and 'Define equipment/utility'. The 'LCSOFT' logo is also visible.

Figure B.22: Steps to perform LCA (Continued)

B.11 Steps to import LCSOFT results

Step 1: Open interface which is “Bioethanol_from_cassavarhizome_1” window (Figure B.23).

Step 2: Click “Import Results” button.

Step 3: Select results and click “Ok” button. LCSOFT results will be imported. Select “LCSOFT results” page to see LCSOFT results.

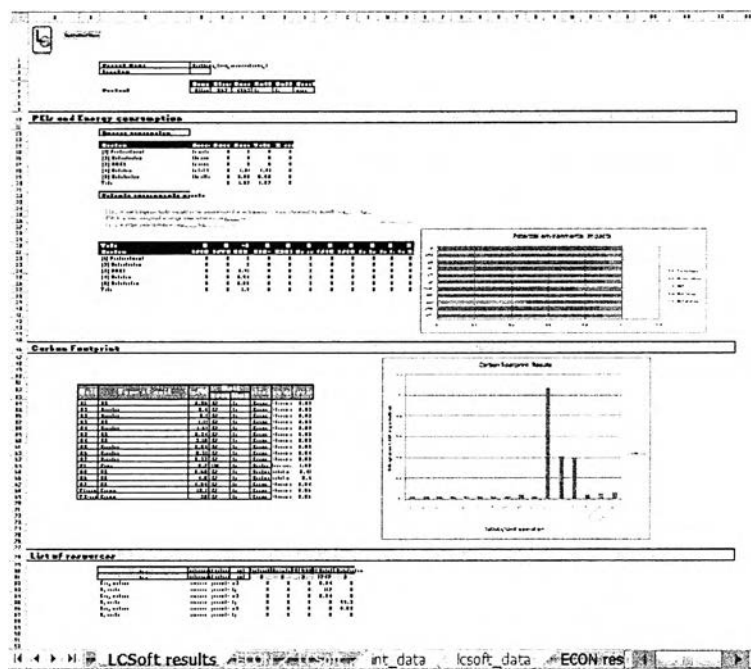
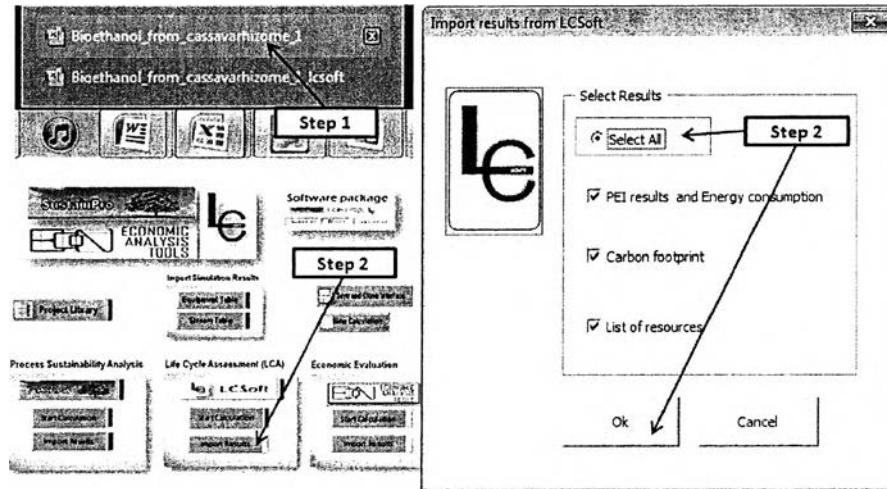


Figure B.23: Steps to perform LCA (Continued)

B.12 Economic evaluation using ECON thought out interface:

Steps to enter economic data

Step 1: Click “ECON” button in “Start menu” page (Figure B.24).

Step 2: Enter number of equipments in the table in “ECON” page (see Table B.1).

Step 3: Fill out all information of each equipment in the gray area in “ECON” page (see equipment specification table). Type exactly the words in the table.

Step 4: Click “Done” button to back to “Start menu” page.

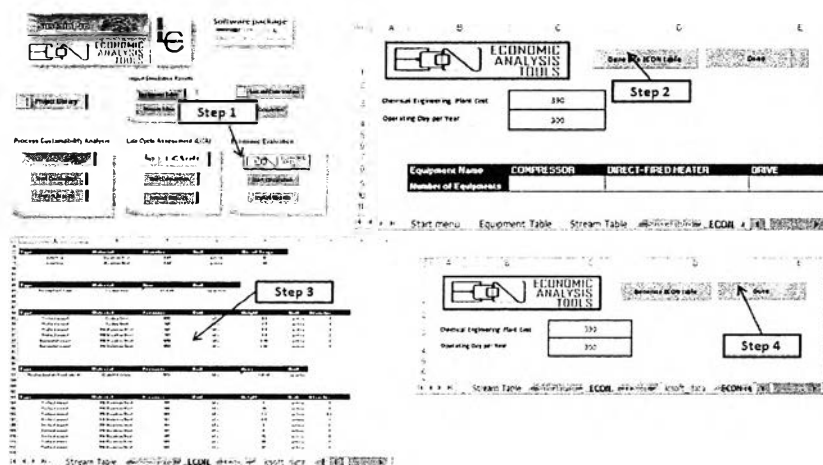


Figure B.24: Steps to perform economic evaluation

B.13 Steps to calculate economic results

Step 1: Click “Start Calculation” button below “ECON” button (Figure B.25).

Step 2: Interface will send economic data to ECON software.

Steps to calculate capital cost, operating cost & total product cost, economic evaluation, pie charts, sensitivity analysis, and alternative case are mentioned in “manual200211, section 8 – 13”.

(To open manual: Open “Interface” folder > “Software” folder > “ECON” folder > “manual200211” file)

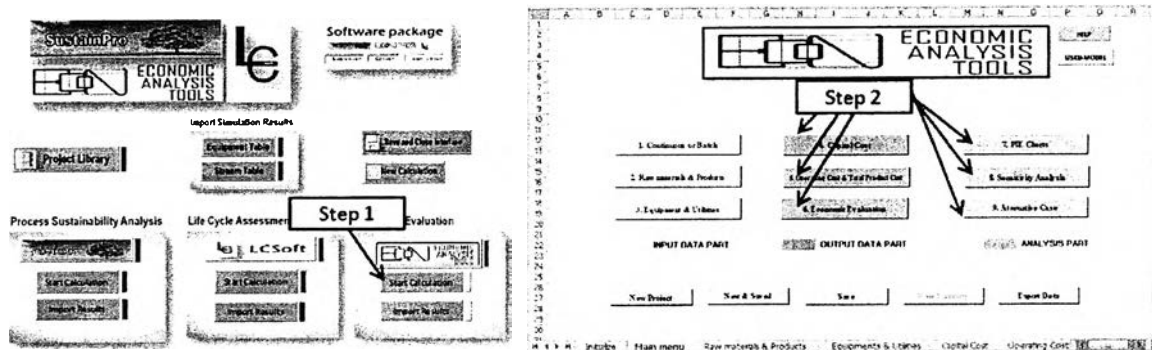


Figure B.25: Steps to perform economic evaluation (Continued)

Table B.1: Equipment list of the case study

EQUIPMENT NAME	NUMBER OF EQUIPMENTS
COMPRESSOR	
DIRECT-FIRED HEATER	
DRIVE	
FURNACE	
HEAT EXCHANGER	11
PUMP INCLUDE DRIVE	
PUMP	1
TOWER UNIT	2
TRAY	2
STORAGE	1
VESSEL	6
PFR	1
REACTOR	9
TURBINE	
COOLING TOWER	1
MIXER	12
OTHER EQUIPMENT	4

B.14 Steps to import economic results

Step 1: Open interface which is “Bioethanol_from_cassavarhizome_1” window (Figure B.26).

Step 2: Click “Import Results” button.

Step 3: Select results and click “Ok” button. ECON results will be imported. Select “ECON results” page to see ECON results.

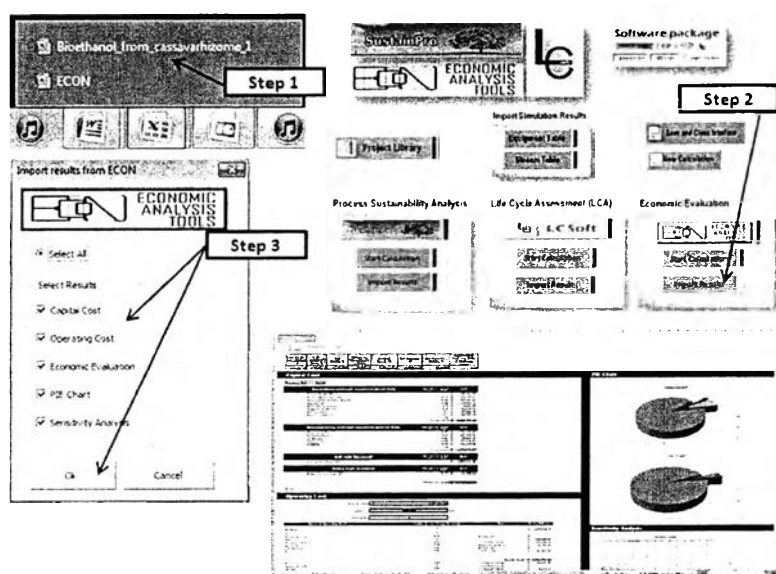


Figure B.26 Steps to perform economic evaluation (Continued)

Appendix C Bioethanol conversion process flowsheet and data for process evaluation

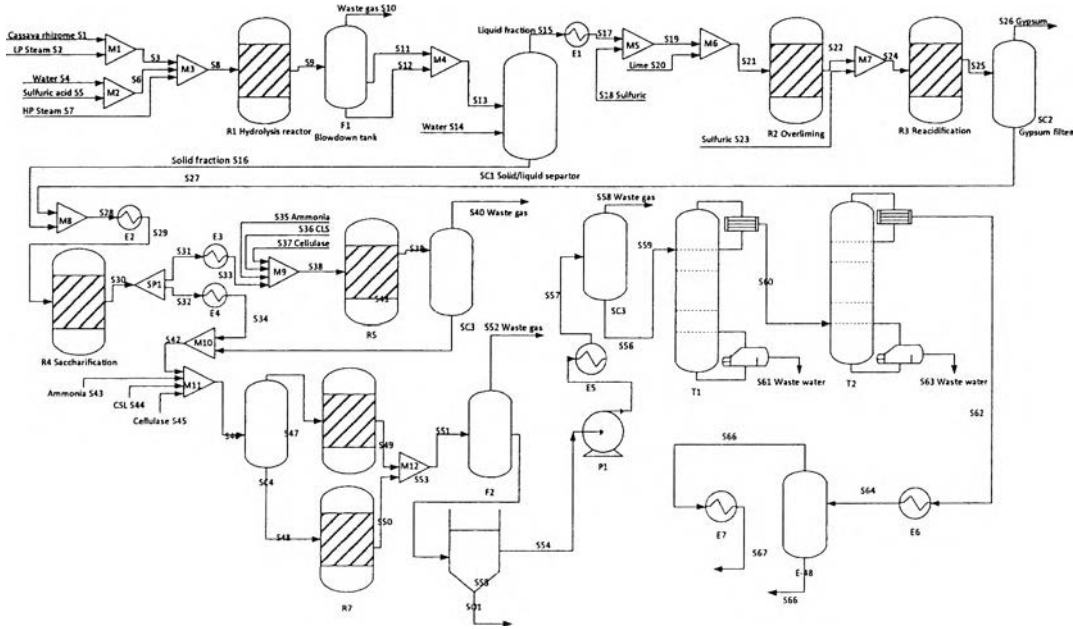


Figure C.1.1 Flowsheet of the case study.

Table C1.1 Stream table of the bioethanol process from cassava rhizome for the case study

Stream Name	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
Stream Phase	Mixed	Vapor	Mixed	Liquid	Liquid	Liquid	Vapor	Mixed	Mixed	Vapor	Liquid	Solid	Mixed	Liquid	Mixed	Mixed	Liquid	Liquid	Mixed	Solid
Temperature	30	160	100	25	25	25	268	186	190	124	124	124	124	25	63	63	50	25	50	25
Pressure	1	6	1	1	1	1	13	12	1	1	1	1	1	1	1	1	1	1	1	1
Total Molecular Weight	104	15	65	18	96	16	18	36	41	19	35	35	47	18	22	46	23	95	23	74
Cellulose	4661	0	4561	0	0	0	0	4661	4320	0	0	4320	4320	0	0	4320	22	0	22	0
Hemicellulose	6674	0	6574	0	0	0	0	6674	334	0	0	334	334	0	0	334	0	0	0	0
Lignin	3653	0	3653	0	0	0	0	3653	3652	0	0	3653	3653	0	0	3653	19	0	19	0
Glucose	0	0	0	0	0	0	0	0	0	0	364	0	364	0	268	76	268	0	268	0
Xylose	0	0	0	0	0	0	0	0	0	0	6826	0	6826	0	4437	2389	4437	0	4437	0
Cellulase	0	0	0	0	0	0	0	0	35	0	35	0	35	0	27	7	27	0	27	0
Ethanol	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Water	120	785	915	4972	0	4972	3153	9040	8274	2372	5902	0	5902	12695	14852	3948	14852	0	14852	0
Sulfuric Acid	0	0	0	0	92	99	0	99	99	0	99	0	99	0	73	21	73	112	192	0
Formal	0	0	0	0	0	0	0	0	243	116	127	0	127	0	100	27	100	0	100	0
Ammonia	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Oxygen	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Carbon Dioxide	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Glycerol	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Succinic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lactic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HMF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Xylitol	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Acetic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Corn/Steep Liquor	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ZM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Cellulase	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lime	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	179
CaSO4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ash	578	0	578	0	0	0	0	578	578	0	0	0	578	578	0	0	0	0	0	0

Table C1.1 Stream table of the bioethanol process from cassava rhizome for the case study (continue)

Stream Name	S41	S42	S43	S44	S45	S46	S47	S48	S49	S50	S51	S52	S53	S54	S55	S56	S57	S58	S59	S60
Stream Phase	Mixed	Mixed	Liquid	Mixed	Mixed	Mixed	Mixed	Mixed	Mixed	Mixed	Mixed	Mixed	Mixed	Mixed	Vapor	Liquid	Mixed	Mixed	Mixed	Vapor
Temperature	50	50	25	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
Pressure	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Total Molecular Weight	22	23	89	23	43	138	23	28	28	50	30	30	30	30	17	18	18	18	28	44
Cellulose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Hemicellulose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lignin	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Glucose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Xylose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Cellulose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ethanol	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Water	14822	14822	0	14822	14842	0	14842	14858	0	14858	14873	1543	14873	1543	14873	1543	14873	1543	14873	1543
Sulfuric Acid	180	100	0	47	47	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Furfural	100	100	0	100	100	0	100	127	127	127	127	127	127	127	127	127	127	127	127	127
Ammonia	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Oxygen	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Carbon Dioxide	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Glycerol	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Succinic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lactic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HMF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Xylool	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Acetic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Corsteeo Liqueur	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ZM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Cellulase	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lime	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CASO4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ASH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table C1.1 Stream table of the bioethanol process from cassava rhizome for the case study (continue)

Stream Name	S41	S42	S43	S44	S45	S46	S47	S48	S49	S50	S51	S52	S53	S54	S55	S56	S57	S58	S59	S60
Stream Phase	Mixed	Mixed	Vapor	Liquid	Mixed	Mixed	Mixed	Mixed	Mixed	Liquid	Mixed	Vapor	Mixed	Liquid	Solid	Liquid	Liquid	Vapor	Liquid	Vapor
Temperature	41	41	29	29	29	41	41	41	41	41	41	41	41	41	41	41	41	41	41	41
Pressure	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Total Molecular Weight	28	28	17	18	21	29	29	28	28	50	30	30	30	30	17	18	18	18	28	44
Cellulose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Hemicellulose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lignin	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Glucose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Xylose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Cellulose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Ethanol	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Water	1893	1843	0	0	1839	1843	0	1843	1850	0	1843	1843	1843	1843	1843	1843	1843	1843	1843	1843
Sulfuric Acid	2	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Furfural	18	18	0	0	18	18	0	18	22	22	22	22	22	22	22	22	22	22	22	22
Ammonia	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Oxygen	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Carbon Dioxide	24	24	0	0	24	24	0	24	24	0	24	24	24	24	24	24	24	24	24	24
Glycerol	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Succinic Acid	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lactic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HMF	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Xylool	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Acetic Acid	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Corsteeo Liqueur	60	60	0	0	60	60	0	60	60	0	60	60	60	60	60	60	60	60	60	60
ZM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Cellulase	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Lime	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CASO4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ASH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table C1.1 Stream table of the bioethanol process from cassava rhizome for the case study (continue)

Stream Name	S61	S62	S63	S64	S65	S66	S67
Stream Phase	Liquid	Liquid	Liquid	Vapor	Vapor	Vapor	Liquid
Temperature	117	93	110	100	100	100	40
Pressure	1	2	2	2	1	1	1
Total Molecular Weight	46	42	42	46	46	46	46
Cellulose	0	0	0	0	0	0	0
Hemicellulose	0	0	0	0	0	0	0
Lignin	0	0	0	0	0	0	0
Glucose	0	0	0	0	0	0	0
Xylose	0	0	0	0	0	0	0
Cellulase	0	0	0	0	0	0	0
Ethanol	61	4562	15	4562	4562	0	4562
Water	17612	1518	172	318	2	315	2
Sulfuric Acid	21	0	0	0	0	0	0
Furfural	122	0	0	0	0	0	0
Ammonia	0	0	0	0	0	0	0
Oxygen	0	0	0	0	0	0	0
Carbon Dioxide	0	0	0	0	0	0	0
Glycerol	0	0	0	0	0	0	0
Succinic Acid	14	0	0	0	0	0	0
Lactic Acid	318	0	0	0	0	0	0
HMF	0	0	0	0	0	0	0
Xylool	48	0	0	0	0	0	0
Acetic Acid	17	0	0	0	0	0	0
Corsteeo Liqueur	147	0	0	0	0	0	0
ZM	0	0	0	0	0	0	0
Cellulase	0	0	0	0	0	0	0
Lime	0	0	0	0	0	0	0
CASO4	0	0	0	0	0	0	0
ASH	0	0	0	0	0	0	0

Table C1.2 Equipment data

Pump Name	Unit	P1							
Work	KW	5.598							
ConReactor Name	Unit	R1	R2	R3	R4	R5	R6	R7	
Temperature	C	190	50	50	55	41	41	41	
Pressure	ATM	12.100	1.000	1.000	1.000	1.000	1.000	1.000	
Duty	M*KJ/HR	0.000	-0.400	-0.093	1.513	-0.843	-8.214	-0.323	
Heat Of Reaction	M*KJ/HR	-1.258	-0.351	-0.093	-0.575	-0.011	-0.350	0.092	
Product Enthalpy	KJ/KG	-227.731	1309.957	1335.907	821.194	-36.535	-272.505	8.343	
Feed Enthalpy	KJ/KG	-4142.821	1205.755	1310.420	-1562.569	19.703	39.594	31.737	
ΔEnthalpy	KJ/KG	3915.090	104.212	25.488	2383.563	-55.239	-312.199	-23.353	
	GJ/KG	3.915	0.104	0.025	2.384	-0.055	-0.312	-0.023	
Flash Name	Unit	F1	F2						
Temperature	C	103.854	41.021						
Pressure	ATM	1.000	1.000						
DP	ATM	11.100	0.000						
Duty	M*KJ/HR	0.000	0.000						
Stream Calculator Name	Unit	SC1	SC2	SC3	SC4	SC5	SC6		
Duty	M*KJ/HR	0.000	0.000	0.000	0.000	0.000	0.000		
Overhead Product Temperature	C	52.553	49.537	42.531	41.000	100.000	100.015		
Bottoms Product Temperature	C	52.553	49.537	42.531	41.000	100.000	100.015		
Hx Name	Unit	E1	E2	E3	E4	E5	E6	E7	
Duty	M*KJ/HR	0.859	1.113	0.235	2.150	5.579	4.504	4.840	
Column Name	Unit	T1	T2						
Condenser Duty	M*KJ/HR	-18.059	-19.971						
Reboiler Duty	M*KJ/HR	24.590	14.371						
Column Total Molar Feed	KG-MOL/DAY	27507.250	3517.017						
Column Total Wt. Feed	KG/DAY	502124.559	*****						
Column Condenser Pres	ATM	1.770	1.770						
Column Condenser Temp	C	93.531	93.344						
Column Reflux Rate	KG-MOL/DAY	0.000	9528.490						
Column Reflux Ratio		3.200	3.200						

C.2 Data for process sustainability analysis

C.2.1 Stream data

Table C2.1 Stream data (Mangnimit, 2012)

Component name	Molecular weight (g/mol)	Hazardous	Type	Price (\$/kg)
Cellulose	162.14	No	RM	0.51*
Hemicellulose	152.12	No	RM	0.61*
Lignin	122.49	No	RM	0.31*
Glucose	180.16	No	BP	0
Xylose	150.13	No	BP	0
Cellobiose	342.30	No	BP	0
Ethanol	46.07	Yes	P	1.14
Water	18.02	No	RM	0.591
Sulfuric Acid	98.08	Yes	RM	0.5
Formic Acid	96.09	Yes	BP	0
Ammonia	17.03	Yes	RM	0.556
Oxygac.	32.00	No	BP	0
Carbon Dioxide	44.01	Yes	BP	0
Glycerol	92.09	No	BP	0
Succinic Acid	118.09	Yes	BP	0
Lactic Acid	90.08	Yes	BP	0
HMF	126.11	Yes	BP	0
Xylitol	152.13	No	BP	0
Acetic Acid	60.05	Yes	BP	0
Corn Steep Liquor	18.02	No	RM	0.5
ZM	24.63	No	BP	0
Cellulose	22.84	No	RM	0
Lignin	74.09	Yes	RM	0.4
CaSO4	136.14	Yes	BP	0
Asb	105.99	Yes	RM	0.6618

C2.2 Reaction data

	Reaction	Conversion	Modeled
1	Cellulose _n + nWater \longrightarrow nGlucose C ₅ H ₁₀ O ₅ + H ₂ O \longrightarrow C ₆ H ₁₂ O ₆	Cellulose	0.070
2	Cellulose _n + n/2Water \longrightarrow n/2Cellobiose C ₆ H ₁₀ O ₅ + 1/2H ₂ O \longrightarrow 1/2C ₁₂ H ₂₂ O ₁₁	Cellulose	0.007
3	Hemicellulose _n + nWater \longrightarrow nXylose C ₅ H ₈ O ₄ + H ₂ O \longrightarrow C ₅ H ₁₀ O ₅	Hemicellulose	0.900
4	Hemicellulose _n \longrightarrow nFurfural + 2nWater C ₅ H ₈ O ₄ \longrightarrow C ₅ H ₄ O ₂ + 2H ₂ O	Hemicellulose	0.050
5	Sulfuric Acid + Calcium Hydroxide \longrightarrow Gypsum H ₂ SO ₄ + Ca(OH) ₂ \longrightarrow CaSO ₄ + 2H ₂ O	Sulfuric Acid	1.000
6	Cellulose _n + n/2Water \longrightarrow n/2Cellobiose C ₅ H ₁₀ O ₅ + 1/2H ₂ O \longrightarrow 1/2C ₁₂ H ₂₂ O ₁₁	Cellulose	0.012
7	Cellulose _n + nWater \longrightarrow nGlucose C ₅ H ₁₀ O ₅ + H ₂ O \longrightarrow C ₆ H ₁₂ O ₆	Cellulose	0.900
8	Cellobiose _n + nWater \longrightarrow 2nGlucose C ₁₂ H ₂₂ O ₁₁ + H ₂ O \longrightarrow 2C ₆ H ₁₂ O ₆	Cellobiose	1.000
9	Glucose \longrightarrow 2Ethanol + 2Carbon Dioxide C ₆ H ₁₂ O ₆ \longrightarrow 2C ₂ H ₅ O + 2CO ₂	Glucose	0.900
10	Glucose + 1.2Ammonia \longrightarrow 6Z. mobilis + 2.4Water + 0.3Oxygen C ₆ H ₁₂ O ₆ + 1.2NH ₃ \longrightarrow 6C _{1.8} H _{6.3} O _{2.2} + 2.4H ₂ O + 0.3O ₂	Glucose	0.040
11	Glucose + 2Water \longrightarrow 2Glycerol + Oxygen C ₆ H ₁₂ O ₆ + 2H ₂ O \longrightarrow 2C ₃ H ₅ O ₃ + O ₂	Glucose	0.004
12	Glucose + 2Carbon Dioxide \longrightarrow 2Succinic Acid + Oxygen C ₆ H ₁₂ O ₆ + 2CO ₂ \longrightarrow 2C ₄ H ₄ O ₄ + O ₂	Glucose	0.006
13	Glucose \longrightarrow 3Acetic Acid C ₆ H ₁₂ O ₆ \longrightarrow 3CH ₃ COOH	Glucose	0.015
14	Glucose \longrightarrow 2Lactic Acid C ₆ H ₁₂ O ₆ \longrightarrow 2CH ₃ CHOHCOOH	Glucose	0.002
15	3Xylose \longrightarrow 5Ethanol + 5Carbon Dioxide 3C ₅ H ₈ O ₄ \longrightarrow 5C ₂ H ₆ O + 5CO ₂	Xylose	0.800
16	Xylose + Ammonia \longrightarrow 5Z. mobilis + 2Water + 0.25Oxygen	Xylose	0.040

17	$3\text{Xylose} + 5\text{Water} \longrightarrow 5\text{Glycerol} + 2.5\text{Oxygen}$ $3\text{C}_5\text{H}_{10}\text{O}_5 + 5\text{H}_2\text{O} \longrightarrow 5\text{C}_3\text{H}_8\text{O}_3 + 2.5\text{O}_2$	Xylose	0.003
18	$\text{Xylose} + \text{Water} \longrightarrow \text{Xylitol} + 0.5\text{Oxygen}$ $\text{C}_5\text{H}_{10}\text{O}_5 + \text{H}_2\text{O} \longrightarrow \text{C}_5\text{H}_{12}\text{O}_5 + 0.5\text{O}_2$	Xylose	0.046
19	$3\text{Xylose} + 5\text{Carbon Dioxide} \longrightarrow 5\text{Succinic Acid} + 2.5\text{Oxygen}$ $3\text{C}_5\text{H}_{10}\text{O}_5 + 5\text{CO}_2 \longrightarrow 5\text{C}_4\text{H}_5\text{O}_4 + 2.5\text{O}_2$	Xylose	0.009
20	$2\text{Xylose} \longrightarrow 5\text{Acetic Acid}$ $2\text{C}_5\text{H}_{10}\text{O}_5 \longrightarrow 5\text{CH}_3\text{COOH}$	Xylose	0.014
21	$3\text{Xylose} \longrightarrow 5\text{Lactic Acid}$ $3\text{C}_5\text{H}_{10}\text{O}_5 \longrightarrow 5\text{CH}_3\text{CHOHCOOH}$	Xylose	0.002

Appendix D LCSoft version 2.0 and interface source codes

D.1 LCSoft version 2.0 source codes

D.1.1 Main menu

```
Private Sub CommandButton2_Click()
Application.ScreenUpdating = False
If Sheets("results").Cells(9, 3) <> "" Then
    For j = 9 To Sheets("results").Range("c" & Rows.Count).End(xlUp).Row
    If Sheets("results").Cells(j, 3) <> "Total" Then
        section_name = Sheets("results").Cells(j, 3)
        Sheets(section_name).Select
        On Error Resume Next
        Call step2.edit_LCA_section(section_name)
        Call step2.create_LCA_section(section_name)
        Call step2.retrieve_LCI_input_section(section_name)
        Call step2.retrieve_LCI_output_section(section_name)
    Else
        Sheets("results").Cells(j, 3).EntireRow.Clear
    End If
    Next j
    Sheets("main menu").Activate
    Sheets("main menu").CommandButton2.BackColor = &H8000000A
    Call step4.call_LCIA
    Call step4.LCA_results_chart
    If Sheets("main menu").CommandButton4.Enabled = True Then
        Call step4.carbon_footprint
    End If
    Call step3.cal_energy_resource
    Call step3.resource_list
    Sheets("main menu").CommandButton2.BackColor = &HFFFF00
    Sheets("main menu").Activate
Else
```

```

MsgBox "Please generate LCA section"
End If
End Sub
Private Sub CommandButton3_Click()
ActiveWorkbook.Save
ActiveWorkbook.Close
End Sub
Private Sub CommandButton4_Click()
Sheets("main menu").CommandButton4.BackColor = &HFFFFFF00
Sheets("define equipment").Visible = True
Sheets("define equipment").Activate
    With Sheets("define equipment").[k:k].Font
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With
End Sub
Private Sub CommandButton5_Click()
Sheets("main menu").CommandButton5.BackColor = &HFFFFFF00
Sheets("define stream").Visible = True
Sheets("define stream").Activate
Sheets("define stream").[e:e].Font.ThemeColor = xlThemeColorDark1
End Sub
Private Sub CommandButton7_Click()
Application.DisplayAlerts = False
For Each Worksheet In ThisWorkbook.Worksheets
If Worksheet.Name = "Stream Table" Or Worksheet.Name = "generate stream table"
Then
    Application.DisplayAlerts = False
    Worksheet.Delete
End If
Next
Sheets("gen_stream_t").Visible = True

```



```

Sheets("gen_stream_t").Copy After:=Sheets("main menu")
ActiveSheet.Name = "generate stream table"
MsgBox "Plese enter stream components and mass flow rates following the given
example"
Sheets("generate stream table").Cells(2, 2) =
Sheets("LCI").Range("unit_mass").Value & "/" &
Sheets("LCI").Range("unit_time").Value
Sheets("gen_stream_t").Visible = False
Sheets("LCI").Visible = False
End Sub

Private Sub CommandButton9_Click()
Application.DisplayAlerts = False
For Each Worksheet In ThisWorkbook.Worksheets
If Worksheet.Name = "Equipment Table" Or Worksheet.Name = "generate
equipment table" Then
    Application.DisplayAlerts = False
    Worksheet.Delete
End If
Next
Sheets("gen equip_t").Visible = True
Sheets("gen equip_t").Copy After:=Sheets("main menu")
ActiveSheet.Name = "generate equipment table"
MsgBox "Plese enter equipment name, duty and work following the given example"
Sheets("gen equip_t").Visible = False
Sheets("LCI").Visible = False
End Sub

Private Sub Image1_Click()
    ActiveSheet.Shapes("Image1").ShapeRange.ZOrder msoSendToBack
End Sub

Private Sub Sh1_cmb_newproject_Click()
wbpath = ThisWorkbook.Path

```

```

Application.ScreenUpdating = False
F1_newproject.Show
Sheets("LCI").Visible = False
End Sub

Private Sub CommandButton1_Click()
F10_results.Show
'CommandButton1
End Sub

Private Sub Sh1_cmb_functionalunit_Click()
If Left(Sheets("results").Range("product_stream"), 1) = "S" And
Sheets("results").Range("product_name") <> "" Then
mge_product = "The product is " & Sheets("results").Range("product_name") _
& " from stream: " & Sheets("results").Range("product_stream") _
& ", Do you want to change?"
use_product = MsgBox(mge_product, vbYesNoCancel, "Define Product and
functional unit")
    If use_product = 6 Then
        product_sim = MsgBox("Do you want to define product from stream table?",
vbYesNoCancel, "Define Product and functional unit")
        If product_sim = 6 Then
            F6_Product_sim.Show
        Else
            F6_Product.Show
        End If
    End If
ElseIf Sheets("results").Range("product_stream") = "-" Then
mge_product = "The product is " & Sheets("results").Range("product_name") _
& ", " & Sheets("results").Range("product_amount") & " " &
Sheets("results").Range("product_unit_mass") _
& ", Do you want to change?"
use_product = MsgBox(mge_product, vbYesNoCancel, "Define Product and
functional unit")

```

```

    If use_product = 6 Then
        product_sim = MsgBox("Do you want to define product from stream table?",
vbYesNoCancel, "Define Product and functional unit")
        If product_sim = 6 Then
            F6_Product_sim.Show
        Else
            F6_Product.Show
        End If
    End If
ElseIf Sheets("results").Range("product_name") = "" Then
mge_product = "Do you want to use simulation results? "
use_product = MsgBox(mge_product, vbYesNoCancel, "Define Product and
functional unit")
    If use_product = 6 Then
        F6_Product_sim.Show
    Else
        F6_Product.Show
    End If
End If
End Sub
Private Sub sh1_editsection_Click()
F9_edit_section.Show
End Sub
Private Sub sh1_cmb_imstream_Click()
Application.ScreenUpdating = False
On Error Resume Next
Set wSheet = ThisWorkbook.Sheets("Stream Table")
If wSheet Is Nothing Then
    Do
        import_table = Application.GetOpenFilename(Title:="Import stream table")
        If import_table = False Then Exit Sub
    
```

```

    use = MsgBox("Do you want this stream table?", vbYesNoCancel, "Import
stream table")
    If use = 6 Then
        Sheets("main menu").Sh1_cmb_functionalunit.BackColor = &H8000000A
        Sheets("main menu").CommandButton5.BackColor = &H8000000A
        Sheets("main menu").CommandButton2.BackColor = &H8000000A

        Sheets.Add Type:=import_table
        ActiveSheet.Name = "Stream Table"
        Call step1.stream_table_reader
        Sheets("main menu").Sh1_cmb_functionalunit.BackColor = &H8000000A
        Sheets("main menu").CommandButton4.BackColor = &H8000000A
        Sheets("LCI").Visible = False
        Sheets("main menu").Activate
        Exit Do
    End If
    Loop Until use = 2
Else
    Change = MsgBox("Stream table already exists. Do you want to add new stream
table?", vbYesNo)
    If Change = 6 Then
        Do
            import_table = Application.GetOpenFilename(Title:="Import stream table")
            If import_table = False Then Exit Sub
            use = MsgBox("Do you want this Stream table?", vbYesNoCancel, "Import
stream table")
            If use = 6 Then
                Application.DisplayAlerts = False
                Worksheets("Stream Table").Delete
                Sheets.Add Type:=import_table
                ActiveSheet.Name = "Stream Table"
                Call step1.stream_table_reader
            End If
        Loop
    End If

```

```

Sheets("main menu").Sh1_cmb_functionalunit.BackColor = &H8000000A
Sheets("main menu").CommandButton4.BackColor = &H8000000A
Sheets("main menu").CommandButton2.BackColor = &H8000000A
Sheets("main menu").CommandButton5.BackColor = &H8000000A
Sheets("LCI").Visible = False
Sheets("main menu").Activate
Exit Do
End If
Loop Until use = 2
End If
End If
On Error Resume Next
Set StreamSheet = ActiveWorkbook.Sheets("Stream Table")
If StreamSheet Is Nothing Then
sh1_cmb_imstream.BackColor = &H8000000A
Else
sh1_cmb_imstream.BackColor = &HFFFF00
End If
End Sub
Private Sub sh1_cmb_imEquip_Click()
Application.ScreenUpdating = False
On Error Resume Next
Set wSheet = ActiveWorkbook.Sheets("Equipment Table")
If wSheet Is Nothing Then
Do
import_table = Application.GetOpenFilename(Title:="Import equipment table")
If import_table = False Then Exit Sub
use = MsgBox("Do you want this equipment table?", vbYesNoCancel, "Import
equipment table")
If use = 6 Then
Sheets.Add Type:=import_table
ActiveSheet.Name = "Equipment Table"

```

```

Call step1.equipment_table_reader
Sheets("main menu").Sh1_cmb_functionalunit.BackColor = &H8000000A
Sheets("main menu").CommandButton5.BackColor = &H8000000A
Sheets("main menu").Activate
Sheets("LCI").Visible = False
Exit Do
End If
Loop Until use = 2
Else
    Change = MsgBox("Equipment table already exists. Do you want to add new
equipment table?", vbYesNo)
    If Change = 6 Then
        Do
            import_table = Application.GetOpenFilename(Title:="Import equipment table")
            If import_table = False Then Exit Sub
            use = MsgBox("Do you want this equipment table?", vbYesNoCancel, "Import
equipment table")
            If use = 6 Then
                Application.DisplayAlerts = False
                Worksheets("Equipment Table").Delete
                Sheets.Add Type:=import_table
                ActiveSheet.Name = "Equipment Table"
                Call step1.equipment_table_reader
                Sheets("main menu").Sh1_cmb_functionalunit.BackColor = &H8000000A
                Sheets("main menu").CommandButton5.BackColor = &H8000000A
                Sheets("main menu").Activate
                Sheets("LCI").Visible = False
                Exit Do
            End If
        Loop Until use = 2
    End If
End If

```

```

'updating equipment table step
On Error Resume Next
Set equipmentsheet = ActiveWorkbook.Sheets("Equipment Table")
If equipmentsheet Is Nothing Then
sh1_cmb_imEquip.BackColor = &H8000000A
Else
sh1_cmb_imEquip.BackColor = &HFFFFFF00
End If
End Sub

Private Sub sh1_cmb_lcasection_Click()
F5_create_sec.Show
'sh1_cmb_lcasection
If Sheets("Results").[b9] = "" Then
sh1_cmb_lcasection.BackColor = &H8000000A
Else
sh1_cmb_lcasection.BackColor = &HFFFFFF00
End If
End Sub

Private Sub worksheet_activate()
If Sheets("Results").[b9] = "" Then
sh1_cmb_lcasection.BackColor = &H8000000A
Else
sh1_cmb_lcasection.BackColor = &HFFFFFF00
End If

If sh1_cmb_imstream.BackColor <> &HFFFFFF00 Then
CommandButton5.Enabled = False
Else
CommandButton5.Enabled = True
End If

If sh1_cmb_imEquip.BackColor <> &HFFFFFF00 Then
CommandButton4.Enabled = False
Else

```

```

CommandButton4.Enabled = True
End If
End Sub

```

D.1.2 Software modules

```

Sub unit_convertor(basis_u, enter_u)
  If basis_u <> enter_u Then
    Sheets("units").Visible = True
    Sheets("units").Activate
    Set Match = Sheets("units").[b:b].Find(What:=enter_u, _
    LookIn:=xlValues)
    If Not Match Is Nothing Then
      enter_u_row = Match.Row
    End If
    For j = 2 To 15
      If Sheets("units").Cells(2, j) = basis_u Then
        Sheets("LCI").Range("unit_factor") = Sheets("units").Cells(enter_u_row, j)
      End If
    Next j
    Else
      Sheets("LCI").Range("unit_factor") = 1
    End If
    Sheets("units").Visible = False
  End Sub

```

```

Sub equipment_table_reader()
  'clear existing data
  Sheets("Define Equipment").Visible = True
  Sheets("Define Equipment").Activate
  last_row = Range("b" & Rows.Count).End(xlUp).Row
  Application.DisplayAlerts = False

```



```
Range(Cells(7, 2), Cells(last_row, 20)).Delete
```

```
'read new table
```

```
Sheets("equipment table").Activate
```

```
For i = 1 To Range("A" & Rows.Count).End(xlUp).Row
```

```
Cells(i, 1).Select
```

```
If ActiveCell.Value = "Pump" Then
```

```
p = 3
```

```
Do While Cells(i + 1, p) <> ""
```

```
If Cells(i + 2, p) <> 0 Then
```

```
current_row = Sheets("define equipment").[b5000].End(xlUp).Offset(1, 0).Row
```

```
With Sheets("define equipment")
```

```
.Cells(current_row, 2) = Sheets("equipment table").Cells(i + 1, p)
```

```
.Cells(current_row, 3) = "Pump"
```

```
.Cells(current_row, 4) = Sheets("equipment table").Cells(i + 2, p)
```

```
.Cells(current_row, 5) = Sheets("LCI").Range("unit_power")
```

```
.Cells(current_row, 6) = Sheets("LCI").Range("unit_time")
```

```
End With
```

```
End If
```

```
p = p + 1
```

```
Loop
```

```
ElseIf ActiveCell.Value = "Reactor" Then
```

```
unit = Cells(i, 1).Offset(4, 1).Value
```

```
p = 3
```

```
Do While Cells(i + 1, p) <> ""
```

```
If Cells(i + 4, p) <> 0 Then
```

```
current_row = Sheets("define equipment").[b5000].End(xlUp).Offset(1, 0).Row
```

```
With Sheets("define equipment")
```

```
.Cells(current_row, 2) = Sheets("equipment table").Cells(i + 1, p)
```

```
.Cells(current_row, 3) = "Reactor"
```

```
.Cells(current_row, 4) = Sheets("equipment table").Cells(i + 4, p)
```

```

.Cells(current_row, 5) = Sheets("LCI").Range("unit_energy")
.Cells(current_row, 6) = Sheets("LCI").Range("unit_time")
End With
End If
p = p + 1
Loop

ElseIf ActiveCell.Value = "Heat Exchanger" Then
unit = Cells(i, 1).Offset(5, 1).Value
p = 3
Do While Cells(i + 1, p) <> ""
If Cells(i + 2, p) <> 0 Then
current_row = Sheets("define equipment").[b5000].End(xlUp).Offset(1, 0).Row
With Sheets("define equipment")
.Cells(current_row, 2) = Sheets("equipment table").Cells(i + 1, p)
.Cells(current_row, 3) = "Heat Exchanger"
.Cells(current_row, 4) = Sheets("equipment table").Cells(i + 2, p)
.Cells(current_row, 5) = Sheets("LCI").Range("unit_energy")
.Cells(current_row, 6) = Sheets("LCI").Range("unit_time")
End With
End If
p = p + 1
Loop

ElseIf ActiveCell.Value = "Column" Then
unit = Cells(i, 1).Offset(5, 1).Value
p = 3
Do While Cells(i + 1, p) <> ""
If Cells(i + 2, p) <> 0 And Cells(i + 3, p) <> 0 Then
current_row = Sheets("define equipment").[b5000].End(xlUp).Offset(1, 0).Row
With Sheets("define equipment")
.Cells(current_row, 2) = Sheets("equipment table").Cells(i + 1, p) & "condenser"

```

```
.Cells(current_row, 3) = "Column-Condenser"
.Cells(current_row, 4) = Sheets("equipment table").Cells(i + 2, p)
.Cells(current_row, 5) = Sheets("LCI").Range("unit_energy")
.Cells(current_row, 6) = Sheets("LCI").Range("unit_time")
End With
```

```
    With Sheets("define equipment")
.Cells(current_row + 1, 2) = Sheets("equipment table").Cells(i + 1, p) & "reboiler"
.Cells(current_row + 1, 3) = "Column-Reboiler"
.Cells(current_row + 1, 4) = Sheets("equipment table").Cells(i + 3, p)
.Cells(current_row + 1, 5) = Sheets("LCI").Range("unit_energy")
.Cells(current_row + 1, 6) = Sheets("LCI").Range("unit_time")
End With
End If
p = p + 1
Loop
End If
Next i
```

```
lastrow = Sheets("define equipment").Range("e" & Rows.Count).End(xlUp).Row
For i = 7 To lastrow
If Sheets("define equipment").Cells(i, 8) = "" Then
With Sheets("define equipment").Cells(i, 8)
.HorizontalAlignment = xlCenter
.Font.ColorIndex = xlAutomatic
.Font.TintAndShade = 0
.Interior.Pattern = xlSolid
.Interior.PatternColorIndex = xlAutomatic
.Interior.ThemeColor = xlThemeColorDark1
.Interior.TintAndShade = -0.249977111117893
.Interior.PatternTintAndShade = 0
.Value = "(double click here)"
```

```

End With
End If
Next i
Sheets("Define Equipment").Visible = False
End Sub

Sub stream_table_reader()
    Sheets("Define Stream").Visible = True
    last_col = Sheets("Define Stream").Cells(6,
Columns.Count).End(xlToLeft).Column
    last_row = Sheets("Define Stream").Range("b" & Rows.Count).End(xlUp).Row
    Sheets("Define Stream").Select

    If last_row > 7 Then Range(Cells(8, 2), Cells(last_row, 2)).EntireRow.Delete
    If last_col > 3 Then Range(Cells(6, 6), Cells(7, last_col)).Delete
    last_st_col = Sheets("Stream Table").Cells(1,
Columns.Count).End(xlToLeft).Column
    last_st_row = Sheets("Stream Table").Range("a" & Rows.Count).End(xlUp).Row

    Sheets("stream table").Select
    Range(Cells(1, 3), Cells(2, last_st_col)).Copy
    Sheets("Define Stream").Select
    [f6].Select
    ActiveSheet.Paste
    Sheets("stream table").Select

    component_name = 3

    Sheets("define stream").[c4] = "Mass flow rate"

```

```

Sheets("define stream").[c5] = Sheets("LCI").Range("unit_mass") & "/" &
Sheets("LCI").Range("unit_time")
Range(Cells(component_name, 1), Cells(last_st_row, 1)).Copy
Sheets("define stream").Select
[b8].Select
ActiveSheet.Paste
Sheets("stream table").Select
Range(Cells(component_name, 3), Cells(last_st_row, last_st_col)).Copy
Sheets("define stream").Select
[f8].Select
ActiveSheet.Paste

```

```

Dim Pic As Shape
On Error Resume Next
For Each Pic In ActiveSheet.Shapes
If Pic.Type = msoPicture Then
Pic.Cut
End If
Next Pic

```

```

lastrow = Sheets("Define Stream").Range("b" & Rows.Count).End(xlUp).Row
For i = 8 To lastrow
If Sheets("define stream").Cells(i, 3) = "" Then
With Sheets("define stream").Cells(i, 3)
.HorizontalAlignment = xlCenter
.Font.ColorIndex = xlAutomatic
.Font.TintAndShade = 0
.Interior.Pattern = xlSolid
.Interior.PatternColorIndex = xlAutomatic
.Interior.ThemeColor = xlThemeColorDark1
.Interior.TintAndShade = -0.249977111117893
.Interior.PatternTintAndShade = 0

```

```

.Value = "(double click here)"
End With
End If
Next i
Sheets("Define Stream").Visible = False
End Sub
Sub product_identification()
product_name = Sheets("Results").Range("product_name")
product_stream = Sheets("Results").Range("product_stream")
Sheets("define stream").Visible = True
Sheets("define stream").Activate
'clear previously defined product
For i = 8 To [b8].End(xlDown).Row
Cells(i, 3).Select
If ActiveCell = "Product" Then
Range(Cells(i, 3), Cells(i, 5)).Clear
With ActiveCell
.HorizontalAlignment = xlCenter
.Font.ColorIndex = xlAutomatic
.Font.TintAndShade = 0
.Interior.Pattern = xlSolid
.Interior.PatternColorIndex = xlAutomatic
.Interior.ThemeColor = xlThemeColorDark1
.Interior.TintAndShade = -0.249977111117893
.Interior.PatternTintAndShade = 0
.Value = "(double click here)"
End With
End If
Next i
'identify new defined product
For i = 8 To [b8].End(xlDown).Row
Cells(i, 2).Select

```

```

If ActiveCell = product_name Then
    Cells(i, 3) = "Product"
    With Cells(i, 3).Interior
        .Pattern = xlNone
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With
    Range(Cells(i, 4), Cells(i, 5)).Clear
    For j = 6 To [f6].End(xlToRight).Column
        Cells(6, j).Select
        If ActiveCell = product_stream Then
            Sheets("Results").[e6] = Cells(i, j)
        End If
    Next j
End If
Next i
Sheets("define stream").Visible = False
Sheets("main menu").Activate
End Sub

Sub creat_section()
    'section
    Sheets("section").Visible = True
    Sheets("section").Select
    Sheets("section").Copy After:=Sheets("LCI")
    If Sheets("Results").[b9].Value <> 1 Then
        new_section = 1
        ActiveSheet.Name = "section " & new_section
        F5_create_sec.Show
        F5_create_sec.tb_number = new_section
    Else
        section_no = Sheets("Results").[b1000].End(xlUp).Value
    End If
End Sub

```

```

new_section = section_no + 1
ActiveSheet.Name = "section " & new_section
F5_create_sec.Show
F5_create_sec.tb_number.Value = new_section
End If
Sheets("section").Visible = False
ActiveSheet.Select
End Sub

Sub input_eq_table(equipsheet)
Sheets("define equipment").Visible = True
Sheets("define equipment").Select
Range(Cells(7, 2), Cells(500, 500)).Clear
Sheets(equipsheet).Select
last_row = Sheets(equipsheet).Range("b" & Rows.Count).End(xlUp).Row
Range(Cells(7, 2), Cells(last_row, 6)).Copy
Sheets("define equipment").Select
[b7].Select
ActiveSheet.Paste

Dim Pic As Shape
For Each Pic In ActiveSheet.Shapes
If Pic.Type = msoPicture Then
Pic.Cut
End If
Next Pic

lastrow = Sheets("define equipment").Range("b" & Rows.Count).End(xlUp).Row
For i = 7 To lastrow
If Sheets("define equipment").Cells(i, 8) = "" Then
With Sheets("define equipment").Cells(i, 8)
.HorizontalAlignment = xlCenter
.Font.ColorIndex = xlAutomatic
.Font.TintAndShade = 0

```



```

.Interior.Pattern = xlSolid
.Interior.PatternColorIndex = xlAutomatic
.Interior.ThemeColor = xlThemeColorDark1
.Interior.TintAndShade = -0.249977111117893
.Interior.PatternTintAndShade = 0
.Value = "(double click here)"
End With
End If
Next i
Sheets("define equipment").Visible = False
MsgBox "Equipment table is already generated."
Sheets("main menu").Select
Sheets("main menu").sh1_cmb_imEquip.BackColor = &HFFFFFF0
    End Sub
Sub delete_LCA_section(section_name)
Application.ScreenUpdating = False

Application.ScreenUpdating = False
Sheets("results").Visible = True
Sheets("results").Select
On Error Resume Next
    For Each ChartObject In ActiveSheet.ChartObjects

        ChartObject.Delete
    Next
Sheets("sensitivity analysis").Visible = True
Sheets("sensitivity analysis").Select
On Error Resume Next
    For Each ChartObject In ActiveSheet.ChartObjects

        ChartObject.Delete
    Next

```

```

char_sec = Len(section_name)
For Each Worksheet In ThisWorkbook.Worksheets
If Right(Worksheet.Name, char_sec + 1) = "_" & section_name Or Worksheet.Name
= section_name Then
    Application.DisplayAlerts = False
    Worksheet.Delete
End If
Next
If Right(section_name, 4) = "_sen" Then
Sheets("sensitivity analysis").Select
Set sen_match = Sheets("sensitivity analysis").[b:b].Find(What:=section_name, _
    LookIn:=xlValues)
    If Not sen_match Is Nothing Then
        sen_match.EntireRow.Delete Shift:=xlUp
    End If
    last_use_row = Sheets("sensitivity analysis").Range("b" &
Rows.Count).End(xlUp).Row
    If last_use_row > 9 Then
        For k = 10 To last_use_row
            Cells(k, 2) = k - 9
        Next k
    End If
Else
Set Match = Sheets("results").[c:c].Find(What:=section_name, _
    LookIn:=xlValues)
    If Not Match Is Nothing Then
        Match.EntireRow.Delete Shift:=xlUp
    End If
End If
End Sub

Sub edit_LCA_section(section_name)

```

```
Application.ScreenUpdating = False
```

```
Application.ScreenUpdating = False
```

```
Sheets("results").Activate
```

```
On Error Resume Next
```

```
For Each ChartObject In ActiveSheet.ChartObjects
```

```
    ChartObject.Delete
```

```
Next
```

```
char_sec = Len(section_name)
```

```
For Each Worksheet In ThisWorkbook.Worksheets
```

```
If Right(Worksheet.Name, char_sec + 1) = "_" & section_name And
```

```
Worksheet.Name <> section_name Then
```

```
    Application.DisplayAlerts = False
```

```
    Worksheet.Delete
```

```
End If
```

```
Next
```

```
Sheets(section_name).Activate
```

```
End Sub
```

```
Sub stream_retrieval()
```

```
Application.ScreenUpdating = False
```

```
'clear existing sheet
```

```
For Each Worksheet In ThisWorkbook.Worksheets
```

```
    name_sheet = Worksheet.Name
```

```
If Worksheet.Name = "e_st" Or Worksheet.Name = "re_st" Or Worksheet.Name =  
"p_st" Then
```

```
    Application.DisplayAlerts = False
```

```
    Worksheet.Delete
```

```
End If
```

```
Next
```

```
'add new sheet
```

```

For i = 3 To 5
to_add = Sheets("LCI").Cells(i, 1)
Sheets(to_add).Visible = True
Sheets(to_add).Copy After:=Sheets("LCI")
new_name = Sheets("LCI").Cells(i, 2)
ActiveSheet.Name = new_name
Sheets(to_add).Visible = False
Sheets(new_name).Visible = True
Next i

wb_path = Left(ThisWorkbook.Path, Len(ThisWorkbook.Path) - 9)
wb_name = ThisWorkbook.Name
Windows(wb_name).Activate
Sheets("define stream").Visible = True
Sheets("define stream").Select
last_st_row = Range("b" & Rows.Count).End(xlUp).Row
Range(Cells(8, 2), Cells(last_st_row, 2)).Copy
Sheets("e_st").Select
[a42].Select
ActiveSheet.Paste
For i = 8 To Sheets("define stream").Range("b" & Rows.Count).End(xlUp).Row
Sheets("define stream").Select
If Sheets("define stream").Cells(i, 5) <> "" Then
    'common data
    data_type = "data_stream_input"
    data_row = i
    data_col = "-"
    unit_me = Sheets("LCI").Range("unit_mass")
    unit_t = Sheets("LCI").Range("unit_time")
    amount = "-"
    data_name = "-"
    compartment = "-"
    sub_compa = "-"

```

```

'command data
obj_set = Sheets("define stream").Cells(i, 5)
obj_4_path = wb_path & "\LCI KB" & obj_set & ".xlsx"
Application.Workbooks.Open obj_4_path
obj_4_name = ActiveWorkbook.Name
Windows(wb_name).Activate
  For j = 3 To 5
    space_sheet = Sheets("LCI").Cells(j, 2)
    Sheets(space_sheet).Visible = True
    store_sheet = Sheets("LCI").Cells(j, 1)
    Sheets(store_sheet).Visible = True
'command data
  locate_col = Sheets(space_sheet).Cells(32,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
  Sheets(space_sheet).Select
'common data
  Cells(32, locate_col) = data_type
  Cells(23, locate_col) = data_row
  Cells(24, locate_col) = data_col
  Cells(35, locate_col) = unit_me
  Cells(36, locate_col) = unit_t
  Cells(34, locate_col) = amount
  Cells(33, locate_col) = data_name
  Cells(29, locate_col) = compartment
  Cells(30, locate_col) = sub_compa
  Select Case unit_me
    Case "g"
      factor_me = 0.001
    Case "kg"
      factor_me = 1
    Case "oz"
      factor_me = 0.28

```

```
Case "lb"
factor_me = 0.453
Case "grains"
factor_me = 0.0000648
Case "ton"
factor_me = 907.19
Case "mg"
factor_me = 0.0001
Case "MJ"
factor_me = 1
Case "GJ"
factor_me = 1000
Case "Btu"
factor_me = 0.001056
Case "kcal"
factor_me = 0.00419
Case "kWh"
factor_me = 1
End Select

Cells(37, locate_col) = factor_me
Call step2.input_data(space_sheet, locate_col, store_sheet, obj_4_name,
wb_name, amount)

Sheets(space_sheet).Visible = False
Sheets(store_sheet).Visible = False
Next j
Windows(obj_4_name).Activate
Application.CutCopyMode = False
ActiveWorkbook.Close
Windows(wb_name).Activate
End If
```

```

Next i
Sheets("define stream").Visible = False
End Sub

Sub equipment_retrieval()
Application.ScreenUpdating = False
wb_path = Left(ThisWorkbook.Path, Len(ThisWorkbook.Path) - 9)
wb_name = ThisWorkbook.Name
Windows(wb_name).Activate
'clear existing sheet
  For Each Worksheet In ThisWorkbook.Worksheets
    name_sheet = Worksheet.Name
    If Worksheet.Name = "e_eq" Or Worksheet.Name = "re_eq" Or Worksheet.Name
= "p_eq" Then
      Application.DisplayAlerts = False
      Worksheet.Delete
    End If
  Next
'add new sheet
  For i = 6 To 8
    to_add = Sheets("LCI").Cells(i, 1)
    Sheets(to_add).Visible = True
    Sheets(to_add).Copy After:=Sheets("LCI")
    new_name = Sheets("LCI").Cells(i, 2)
    ActiveSheet.Name = new_name
    Sheets(to_add).Visible = False
    Sheets(new_name).Visible = True
  Next i
Sheets("Define Equipment").Visible = True
For i = 7 To Sheets("Define Equipment").Range("b" & Rows.Count).End(xlUp).Row
Sheets("Define Equipment").Select
If Sheets("Define Equipment").Cells(i, 11) <> "" Then

```

```

'common data
data_type = "data_equip_input"
data_row = "-"
data_col = "-"
unit_me = Sheets("LCI").Range("unit_energy")

    If Cells(i, 7) = "Cooling" And Cells(i, 4) < 0 Then
        amount = -Sheets("Define Equipment").Cells(i, 4)
    Else
        amount = Sheets("Define Equipment").Cells(i, 4)
    End If
data_name = Sheets("Define Equipment").Cells(i, 2)
compartment = "-"
sub_compa = "-"
'command data
obj_set = Sheets("Define Equipment").Cells(i, 11)
obj_4_path = wb_path & "\LCI KB" & obj_set & ".xlsx"
Application.Workbooks.Open obj_4_path
obj_4_name = ActiveWorkbook.Name
Windows(wb_name).Activate
For j = 6 To 8
    space_sheet = Sheets("LCI").Cells(j, 2)
    store_sheet = Sheets("LCI").Cells(j, 1)
'command data
    locate_col = Sheets(space_sheet).Cells(32,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
    Sheets(space_sheet).Visible = True
    Sheets(store_sheet).Visible = True
    Sheets(space_sheet).Select
'common data
Cells(32, locate_col) = data_type
Cells(23, locate_col) = data_row

```


Cells(24, locate_col) = data_col

Cells(35, locate_col) = unit_me

Cells(34, locate_col) = amount

Cells(33, locate_col) = data_name

Cells(29, locate_col) = compartment

Cells(30, locate_col) = sub_compa

Select Case unit_me

Case "g"

factor_me = 0.001

Case "kg"

factor_me = 1

Case "oz"

factor_me = 0.28

Case "lb"

factor_me = 0.453

Case "grains"

factor_me = 0.0000648

Case "ton"

factor_me = 907.19

Case "mg"

factor_me = 0.0001

Case "MJ"

factor_me = 1

Case "GJ"

factor_me = 1000

Case "Btu"

factor_me = 0.001056

Case "kcal"

factor_me = 0.00419

Case "kWh"

factor_me = 1

```

        End Select
        Cells(37, locate_col) = factor_me
        Call step2.input_data(space_sheet, locate_col, store_sheet, obj_4_name,
wb_name, amount)
        Sheets(space_sheet).Visible = False
        Sheets(store_sheet).Visible = False
        Next j

        Windows(obj_4_name).Activate
        Application.CutCopyMode = False
        ActiveWorkbook.Close
        Windows(wb_name).Activate

    End If
    Next i
    Sheets("Define Equipment").Visible = False
    End Sub
    Sub input_data(space_sheet, locate_col, store_sheet, obj_4_name, wb_name,
amount)
        Application.ScreenUpdating = False
        Windows(wb_name).Activate
        Sheets("cal").Visible = True
        Sheets("cal").Select
        [a:j].Delete
        Windows(obj_4_name).Activate
        Sheets("description").Select
        Range(Cells(1, 3), Cells(21, 3)).Copy
        Windows(wb_name).Activate
        Sheets(space_sheet).Select
        Cells(1, locate_col).Select
        ActiveSheet.Paste

```

```

Windows(obj_4_name).Activate
Sheets(store_sheet).Select
[a:j].Copy
Windows(wb_name).Activate

Sheets("cal").Select
[a1].Select
ActiveSheet.Paste
last_store_row = Sheets("cal").Range("a" & Rows.Count).End(xlUp).Row

For j = 42 To last_store_row
    Sheets("cal").Select
        If Cells(j, 10) <> 0 And Cells(j, 10) <> "" Then
            name_j = Cells(j, 1)
            re_e_type = Cells(j, 2)
            re_e_HV = Cells(j, 3)

            war_row = Cells(j, 5)
            usetox_row = Cells(j, 6)
            category_j = Cells(j, 7)
            subcat_j = Cells(j, 8)
            unit_j = Cells(j, 9)
            If amount <> "-" Then
                amount_j = Cells(j, 10) * amount * Sheets(space_sheet).Cells(37, locate_col) /
Sheets("cal").[j11]
            Else
                amount_j = Cells(j, 10) * Sheets(space_sheet).Cells(37, locate_col) /
Sheets("cal").[j11]
            End If

            to_find = name_j & "*" & category_j & "#" & subcat_j & "&" & unit_j

```

```

Sheets(space_sheet).Select
Set Match = Sheets(space_sheet).Cells.Find(What:=to_find, _
After:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows,
SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
If Not Match Is Nothing Then
    locate_row = Match.Row

    Sheets(space_sheet).Cells(locate_row, locate_col) =
Sheets(space_sheet).Cells(locate_row, locate_col) + amount_j

Else
    next_locate_row = Sheets(space_sheet).Range("a" &
Rows.Count).End(xlUp).Offset(1, 0).Row

    Sheets(space_sheet).Cells(next_locate_row, locate_col) =
Sheets(space_sheet).Cells(next_locate_row, locate_col) + amount_j
    Sheets(space_sheet).Cells(next_locate_row, 2) = re_e_type
    Sheets(space_sheet).Cells(next_locate_row, 3) = re_e_HV

    Sheets(space_sheet).Cells(next_locate_row, 4) = to_find
    Sheets(space_sheet).Cells(next_locate_row, 1) = name_j
    Sheets(space_sheet).Cells(next_locate_row, 5) = war_row
    Sheets(space_sheet).Cells(next_locate_row, 6) = usetox_row
    Sheets(space_sheet).Cells(next_locate_row, 7) = category_j
    Sheets(space_sheet).Cells(next_locate_row, 8) = subcat_j
    Sheets(space_sheet).Cells(next_locate_row, 9) = unit_j
End If
End If
Next j
Sheets("cal").Visible = False
End Sub

```

```

Sub create_LCA_section(section_name)
Application.ScreenUpdating = False
For s = 10 To 12
sheet_format = Sheets("LCI").Cells(s, 1)
Sheets(sheet_format).Visible = True
sheet_cal = Sheets("LCI").Cells(s, 2) & section_name
Sheets(sheet_format).Copy After:=Sheets("LCI")
ActiveSheet.Name = sheet_cal
Sheets(sheet_cal).Visible = True
Sheets(sheet_format).Visible = False
Next s

For k = 15 To 17
check_sheet = Sheets("LCI").Cells(k, 1)
sheet_cal = Sheets("LCI").Cells(k, 2) & section_name
Application.ScreenUpdating = False
On Error Resume Next
Set wSheet = ThisWorkbook.Sheets(check_sheet)
If Not wSheet Is Nothing Then
    If Sheets(check_sheet).Visible = False Then Sheets(check_sheet).Visible = True
    last_row = Sheets(check_sheet).Range("a" & Rows.Count).End(xlUp).Row
    Sheets(check_sheet).Select
    Range(Cells(42, 1), Cells(last_row, 9)).Copy
    Sheets(sheet_cal).Select
    [a42].Select
    ActiveSheet.Paste
    Sheets(check_sheet).Select
    Application.CutCopyMode = False
    Sheets(check_sheet).Visible = False
End If

```

```

check_sheet2 = Sheets("LCI").Cells(k + 5, 1)
sheet_cal2 = Sheets("LCI").Cells(k + 5, 2) & section_name
Application.ScreenUpdating = False
On Error Resume Next
Set wSheet = ThisWorkbook.Sheets(check_sheet)
If Not wSheet Is Nothing Then
    If Sheets(check_sheet2).Visible = False Then Sheets(check_sheet2).Visible =
True
    last_row = Sheets(check_sheet2).Range("a" & Rows.Count).End(xlUp).Row
    Sheets(check_sheet2).Select
    Range(Cells(42, 1), Cells(last_row, 9)).Copy
    Sheets(sheet_cal2).Select
    last_sheet_row = Range("a" & Rows.Count).End(xlUp).Row
    Cells(last_sheet_row + 1, 1).Select
    ActiveSheet.Paste
    Sheets(check_sheet2).Select
    Application.CutCopyMode = False
    Sheets(check_sheet2).Visible = False
End If

Next k

End Sub

Sub retrieve_LCI_input_section(section_name)
Application.ScreenUpdating = False
wb_path = Left(ThisWorkbook.Path, Len(ThisWorkbook.Path) - 9)
wb_name = ThisWorkbook.Name
For i = 9 To Sheets(section_name).Range("a" & Rows.Count).End(xlUp).Row - 1
    If Sheets(section_name).Cells(i, 5) = "stream_sim" Then
        stream_name = Sheets(section_name).Cells(i, 1)
        Call step2.retrieve_stream_sim(stream_name, section_name)
    End If
Next i
End Sub

```

```

ElseIf Sheets(section_name).Cells(i, 5) = "equip_sim" Then
equip_name = Sheets(section_name).Cells(i, 1)
Call step2.retrieve_equip_sim(equip_name, section_name)
Else
'common data
data_type = "data_input"
data_row = "-"
data_col = "-"
unit_me = Sheets(section_name).Cells(i, 3)
unit_t = "-"
amount = Sheets(section_name).Cells(i, 2)
data_name = Sheets(section_name).Cells(i, 1)
compartment = "-"
sub_compa = "-"
'command data
obj_set = Sheets(section_name).Cells(i, 5)
obj_4_path = wb_path & "\LCI KB" & obj_set & ".xlsx"
Application.Workbooks.Open obj_4_path
obj_4_name = ActiveWorkbook.Name
Windows(wb_name).Activate
For j = 10 To 12
space_sheet = Sheets("LCI").Cells(j, 2) & section_name
store_sheet = Sheets("LCI").Cells(j, 1)
Sheets(space_sheet).Visible = True
Sheets(store_sheet).Visible = True
'command data
locate_col = Sheets(space_sheet).Cells(32,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
Sheets(space_sheet).Select
'common data
Cells(32, locate_col) = data_type
Cells(23, locate_col) = data_row

```

```
Cells(24, locate_col) = data_col
Cells(35, locate_col) = unit_me
Cells(36, locate_col) = unit_t
Cells(34, locate_col) = amount
Cells(33, locate_col) = data_name
Cells(29, locate_col) = compartment
Cells(30, locate_col) = sub_compa
Select Case unit_me
Case "kg"
factor_me = 1
Case "MJ"
factor_me = 1
Case "KWh"
factor_me = 1
Case "tkm"
factor_me = 1
    Case "ha"
factor_me = 1
    Case "m3"
factor_me = 1
End Select
```

```
Cells(37, locate_col) = factor_me
```

```
Call step2.input_data(space_sheet, locate_col, store_sheet, obj_4_name,
wb_name, amount)
Sheets(space_sheet).Visible = False
Sheets(store_sheet).Visible = False
Next j
Windows(obj_4_name).Activate
Application.CutCopyMode = False
```



```

ActiveWorkbook.Close
Windows(wb_name).Activate
End If

Next i
End Sub

Sub retrieve_LCI_output_section(section_name)
Application.ScreenUpdating = False
space_sheet = "e_" & section_name
If Sheets(space_sheet).Visible = False Then Sheets(space_sheet).Visible = True
For i = 9 To Sheets(section_name).Range("h" & Rows.Count).End(xlUp).Row - 1
If Sheets(section_name).Cells(i, 15) = "stream_sim" Then
    data_type = "data_stream_output"
    data_name = Sheets(section_name).Cells(i, 8)
    compartment = Sheets(section_name).Cells(i, 9)
    sub_compa = Sheets(section_name).Cells(i, 10)
    Sheets("define stream").Visible = True
    Sheets("define stream").Select
    Set Match = Sheets("define stream").Cells.Find(What:=data_name, _
After:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows,
SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
    If Not Match Is Nothing Then
        data_col = Match.Column
        last_st_row = Sheets("define stream").Range("b" &
Rows.Count).End(xlUp).Offset(1, 0).Row
        Range(Cells(8, data_col), Cells(last_st_row, data_col)).Copy
        Sheets(space_sheet).Select
        locate_col = Sheets(space_sheet).Cells(32,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
        Cells(42, locate_col).Select

```

```
ActiveSheet.Paste
```

```
Cells(32, locate_col) = data_type
```

```
Cells(29, locate_col) = compartment
```

```
Cells(30, locate_col) = sub_compa
```

```
Else
```

```
data_type = "data_output"
```

```
data_name = Sheets(section_name).Cells(i, 8)
```

```
compartment = Sheets(section_name).Cells(i, 9)
```

```
sub_compa = Sheets(section_name).Cells(i, 10)
```

```
amount = Sheets(section_name).Cells(i, 11)
```

```
unit_me = Sheets(section_name).Cells(i, 12)
```

```
Sheets(space_sheet).Select
```

```
locate_col = Sheets(space_sheet).Cells(32,  
Columns.Count).End(xlToLeft).Offset(0, 1).Column
```

```
Cells(42, locate_col).Select
```

```
ActiveSheet.Paste
```

```
Cells(32, locate_col) = data_type
```

```
Cells(33, locate_col) = data_name
```

```
Cells(29, locate_col) = compartment
```

```
Cells(30, locate_col) = sub_compa
```

```
Cells(34, locate_col) = amount
```

```
Cells(35, locate_col) = unit_me
```

```
    Select Case unit_me
```

```
        Case "g"
```

```
            factor_me = 0.001
```

```
        Case "kg"
```

```
            factor_me = 1
```

```
        Case "oz"
```

```
            factor_me = 0.28
```

```
        Case "lb"
```

```

        factor_me = 0.453
        Case "grains"
            factor_me = 0.0000648
        Case "ton"
            factor_me = 907.19
        Case "mg"
            factor_me = 0.0001
        End Select
        Cells(34, locate_col) = Cells(34, locate_col) * factor_me
    End If
End If
Next i
Sheets(space_sheet).Visible = False
Sheets("define stream").Visible = False
End Sub

Sub retrieve_stream_sim(stream_name, section_name)
    Application.ScreenUpdating = False
    Sheets("define stream").Visible = True
    Sheets("define stream").Select
    Set Match = Sheets("define stream").Cells.Find(What:=stream_name, _
    After:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows, _
    SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
    If Not Match Is Nothing Then
        data_col = Match.Column
        data_name = stream_name

        For k = 15 To 17
            space_sheet = Sheets("LCI").Cells(k, 2) & section_name
            store_sheet = Sheets("LCI").Cells(k, 1)
            Sheets(space_sheet).Visible = True
            Sheets(store_sheet).Visible = True
        Next k
    End If
End Sub

```

```

Sheets(store_sheet).Select
last_store_col = Sheets(store_sheet).Cells(32,
Columns.Count).End(xlToLeft).Column
last_store_raw = Sheets(store_sheet).Range("a" & Rows.Count).End(xlUp).Row
Range(Cells(1, 10), Cells(last_store_raw, last_store_col)).Copy
Sheets(space_sheet).Select
next_col = Sheets(space_sheet).Cells(32, Columns.Count).End(xlToLeft).Offset(0,
1).Column
Cells(1, next_col).Select
ActiveSheet.Paste
    For m = next_col To Sheets(space_sheet).Cells(32,
Columns.Count).End(xlToLeft).Column
        Cells(24, m) = data_col
        Cells(33, m) = data_name
        data_row = Cells(23, m)
        Cells(34, m) = Sheets("define stream").Cells(data_row, data_col)
        amount = Cells(34, m)
        If amount <> 0 Then
            For n = 42 To Sheets(space_sheet).Range("a" & Rows.Count).End(xlUp).Row
                If Cells(n, m) <> "" And Cells(n, m) <> 0 Then
                    Cells(n, m) = Cells(n, m) * amount
                End If
            Next n
        End If
    Next m
Sheets(space_sheet).Visible = False
Sheets(store_sheet).Visible = False
Next k
End If
Sheets("define stream").Visible = False
End Sub

```

```

Sub retrieve equip_sim(equip_name, section_name)
Application.ScreenUpdating = False
For k = 20 To 22
space_sheet = Sheets("LCI").Cells(k, 2) & section_name
store_sheet = Sheets("LCI").Cells(k, 1)
st_sheet = Sheets("LCI").Cells(k, 3)

equip_row = Sheets(st_sheet).Range("a" & Rows.Count).End(xlUp).Row + 1
Sheets(space_sheet).Visible = True
Sheets(store_sheet).Visible = True
Sheets(store_sheet).Select

    v = 10
    Do
    Sheets(store_sheet).Select
    If Cells(33, v) = equip_name Then
    data_col = v
    End If
    v = v + 1
    Loop Until Sheets(store_sheet).Cells(33, v) = ""

'Set Match = Sheets(store_sheet).Cells.Find(What:=equip_name, _
'after:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows,
_
'SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
'If Not Match Is Nothing Then
'data_col = Match.Column
last_store_raw = Sheets(store_sheet).Range("a" & Rows.Count).End(xlUp).Row

Range(Cells(1, data_col), Cells(38, data_col)).Copy
Sheets(space_sheet).Select

```

```
next_col = Sheets(space_sheet).Cells(32, Columns.Count).End(xlToLeft).Offset(0,
1).Column
```

```
Cells(1, next_col).Select
```

```
ActiveSheet.Paste
```

```
Sheets(store_sheet).Select
```

```
Range(Cells(42, data_col), Cells(last_store_raw, data_col)).Copy
```

```
Sheets(space_sheet).Select
```

```
Cells(equip_row, next_col).Select
```

```
ActiveSheet.Paste
```

```
Sheets(space_sheet).Visible = False
```

```
Sheets(store_sheet).Visible = False
```

```
End If
```

```
Next k
```

```
End Sub
```

```
Sub carbon_footprint()
```

```
Application.ScreenUpdating = False
```

```
Application.ScreenUpdating = False
```

```
Application.StatusBar = "Please wait while performing task"
```

```
Sheets("Carbon Footprint Results").Visible = True
```

```
Sheets("results").Visible = True
```

```
Sheets("Carbon Footprint Results").Activate
```

```
[a9:n4000].Delete
```

```
On Error Resume Next
```

```
If Not ActiveSheet.ChartObjects(1) Is Nothing Then
```

```
ActiveSheet.ChartObjects(1).Activate
```

```
Application.CutCopyMode = False
```

```
Selection.Cut
```

```
End If
```

```
For i = 7 To Sheets("define equipment").Range("b" & Rows.Count).End(xlUp).Row
```

```

Sheets("define equipment").Visible = True
Sheets("define equipment").Select
If Cells(i, 11) <> "" Then
Range(Cells(i, 2), Cells(i, 12)).Copy
Sheets("Carbon Footprint Results").Visible = True
Sheets("Carbon Footprint Results").Select
next_row = Range("e" & Rows.Count).End(xlUp).Offset(1, 0).Row
Cells(next_row, 2).Select
ActiveSheet.Paste
End If
Next i
Sheets("e_eq").Visible = True
last_col = Sheets("e_eq").Cells(32, Columns.Count).End(xlToLeft).Column
last_row = Sheets("e_eq").Range("a" & Rows.Count).End(xlUp).Row

For n = 42 To last_row
Sheets("e_eq").Select
If Cells(n, 7) = "air" Then
    war_row = Cells(n, 5)
    For m = 10 To last_col
        If Sheets("e_eq").Cells(n, m) <> 0 And Sheets("e_eq").Cells(n, m) <> "" Then
            amount = Sheets("e_eq").Cells(n, m)
            gwp = amount * Sheets("WAR").Cells(war_row, 28) / Sheets("results").[e6]
            *****
            Sheets("e_eq").Cells(last_row + 1, m) = Sheets("e_eq").Cells(last_row + 1, m) +
gwp
        End If
    Next m
End If
Next n

Sheets("e_eq").Select

```

```

Range(Cells(last_row + 1, 10), Cells(last_row + 1, last_col)).Copy
Sheets("Carbon Footprint Results").Select
Range("I9").Select
Selection.PasteSpecial Paste:=xlPasteAll, Operation:=xlNone, SkipBlanks:=
False, Transpose:=True
'chart

```

```

k = Sheets("Carbon Footprint Results").Range("b" &
Rows.Count).End(xlUp).Row
l = Sheets("Carbon Footprint Results").Range("C" &
Rows.Count).End(xlUp).Row

```

```

Set mgY = Sheets("Carbon Footprint Results").Cells(k, 9)
Set mgX = Sheets("Carbon Footprint Results").Cells(k, 2)

```

```

ValChart = Sheets("Carbon Footprint Results").Range(Cells(9, 9), mgY)
XVal = Sheets("Carbon Footprint Results").Range(Cells(9, 2), mgX)

```

```

Set Z = Sheets("Carbon Footprint Results").ChartObjects.Add(1500, 90, 750, 450)
With Z

```

```

    .Chart.ChartType = xlColumnClustered
    .Chart.SeriesCollection.NewSeries
    .Chart.SeriesCollection(1).XValues = XVal
    .Chart.SeriesCollection(1).Values = ValChart
    .Chart.SeriesCollection(1).Name = "CO2 eq."
    .Chart.ChartTitle.Caption = "Carbon Footprint Results"
    .Chart.Axes(xlCategory, xlPrimary).HasTitle = True
    .Chart.Axes(xlCategory, xlPrimary).AxisTitle.Caption = "Activity/Unit
operation"
    .Chart.Axes(xlCategory, xlPrimary).AxisTitle.Font.Size = 14
    .Chart.Axes(xlCategory, xlPrimary).AxisTitle.Font.Bold = True

```



```

.Chart.Axes(xlValue, xlPrimary).HasTitle = True
.Chart.Axes(xlValue, xlPrimary).AxisTitle.Caption = "Kilograms CO2
equivalent"
.Chart.Axes(xlValue, xlPrimary).AxisTitle.Font.Size = 14
.Chart.Axes(xlValue, xlPrimary).AxisTitle.Font.Bold = True
End With
Sheets("e_eq").Visible = False
Application.StatusBar = "finished running"
Sheets("define equipment").Visible = False
End Sub

Sub call_LCIA()
Application.ScreenUpdating = False
Application.ScreenUpdating = False
Application.StatusBar = "Please wait while performing task"

Sheets("results").Visible = True
last_row_todel = Sheets("results").Range("c" & Rows.Count).End(xlUp).Row
Sheets("results").Select
If last_row_todel > 9 Then Range(Cells(9, 13), Cells(last_row_todel, 24)).Clear

If Sheets("results").Cells(last_row_todel, 3) = "Total" Then
Range(Cells(last_row_todel, 3), Cells(last_row_todel, 10)).Clear
End If

For i = 9 To Sheets("results").Range("c" & Rows.Count).End(xlUp).Row
If Cells(i, 3) <> "Total" Then
section_name = Sheets("results").Cells(i, 3)
cal_sheet = "e_" & section_name
impact_row = i

```

```

Sheets(cal_sheet).Visible = True
Sheets(cal_sheet).Select
last_col = Sheets(cal_sheet).Cells(32, Columns.Count).End(xlToLeft).Column
last_row = Sheets(cal_sheet).Range("a" & Rows.Count).End(xlUp).Row
    For l = 2 To 12
        impact_category = Sheets("LCI").Cells(l, 6)
        Cells(41, last_col + 1) = impact_category
    Next l

Call LCIA(section_name, cal_sheet, impact_row)
Sheets(cal_sheet).Visible = False
End If
Next i

last_use_row = Sheets("results").Range("c" & Rows.Count).End(xlUp).Row
Sheets("results").Cells(last_use_row + 1, 13) = "Total"
    For k = 14 To 24
        Sheets("results").Cells(last_use_row + 1, k) = Application.Sum(Range(Cells(9, k),
Cells(last_use_row, k)))
    Next k

Sheets("results").Visible = False
For i = 9 To Sheets("results").Range("c" & Rows.Count).End(xlUp).Row
resource_sheet = "re_" & Sheets("results").Cells(i, 3)
Sheets(resource_sheet).Visible = True
Next i

Application.StatusBar = "finished running"
Sheets("main menu").Select

End Sub

```

```

Sub LCIA(section_name, cal_sheet, impact_row)
Application.ScreenUpdating = False
last_col = Sheets(cal_sheet).Cells(32, Columns.Count).End(xlToLeft).Column
last_row = Sheets(cal_sheet).Range("a" & Rows.Count).End(xlUp).Row
usetox_col = last_col + 10
Sheets(cal_sheet).Visible = True
For m = 10 To last_col
Sheets(cal_sheet).Select
-----
If Cells(32, m) = "data_output" Then
    chemical = Cells(33, m)
    compartment = Cells(29, m)
    sub_compa = Cells(30, m)
    amount = Cells(34, m)
    Sheets("WAR").Activate
    Set Match = Sheets("WAR").Cells.Find(What:=chemical, _
After:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows,
-----
SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
If Not Match Is Nothing Then
    data_row = Match.Row
    For l = 2 To 9
        impact_cat_col = Sheets("LCI").Cells(l, 7)
        If Sheets("WAR").Cells(data_row, impact_cat_col) <> 0 And
Sheets("WAR").Cells(data_row, impact_cat_col) <> "" Then
            impact = Sheets("WAR").Cells(data_row, impact_cat_col) * amount
            Sheets(cal_sheet).Cells(last_row + 2, last_col + 1) =
Sheets(cal_sheet).Cells(last_row + 2, last_col + 1) + impact

```

```

    End If
Next l
End If

Sheets("USEtox").Activate
Set Match = Sheets("USEtox").Cells.Find(What:=chemical, _
After:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows,
SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
If Not Match Is Nothing Then
    data_row = Match.Row
    For l = 10 To 12
        impact_col = usetox_col + (l - 10)
        Call step4.usetox_datacol(compartment, sub_compa, l)
        data_col = Sheets("LCI").Range("data_col").Value
        CF = Sheets("USEtox").Cells(data_row, data_col)
        If IsNumeric(CF) And CF > 0 And CF <> "#VALUE!" And CF <> "#DIV/0!"
Then
            impact = CF * amount
            Sheets(cal_sheet).Cells(last_row + 2, impact_col) =
Sheets(cal_sheet).Cells(last_row + 2, impact_col) + impact
        End If
    Next l
End If

-----
ElseIf Cells(32, m) = "data_stream_output" Then
    locate_col = m
    last_st_row = 42 + Sheets("Define Stream").Range("b" &
Rows.Count).End(xlUp).Row
    For n = 42 To last_st_row
        chemical = Sheets(cal_sheet).Cells(n, 1)
        compartment = Sheets(cal_sheet).Cells(29, m)

```

```

sub_compa = Sheets(cal_sheet).Cells(30, m)
amount = Sheets(cal_sheet).Cells(n, m)
Sheets("WAR").Activate
    Set warMatch = Sheets("WAR").Cells.Find(What:=chemical, _
    After:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows,
    SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
    If Not warMatch Is Nothing Then
        data_row = warMatch.Row
        For l = 2 To 9
            impact_cat_col = Sheets("LCI").Cells(l, 7)
            If Sheets("WAR").Cells(data_row, impact_cat_col) <> 0 And
Sheets("WAR").Cells(data_row, impact_cat_col) <> "" Then
                impact = Sheets("WAR").Cells(data_row, impact_cat_col) * amount
                Sheets(cal_sheet).Cells(n, last_col + l) = Sheets(cal_sheet).Cells(n, last_col + l)
+ impact
            End If
        Next l
    End If

Sheets("USEtox").Activate
Set Match = Sheets("USEtox").Cells.Find(What:=chemical, _
After:=ActiveCell, LookIn:=xlValues, LookAt:=xlPart, SearchOrder:=xlByRows,
SearchDirection:=xlNext, MatchCase:=False, SearchFormat:=False)
If Not Match Is Nothing Then
    data_row = Match.Row
    For l = 10 To 12
        impact_col = usetox_col + (l - 10)
        Call step4.usetox_datacol(compartment, sub_compa, l)
        data_col = Sheets("LCI").Range("data_col").Value
        CF = Sheets("USEtox").Cells(data_row, data_col)

```

```

    If IsNumeric(CF) And CF > 0 Then
        impact = CF * amount
        Sheets(cal_sheet).Cells(n, impact_col) = Sheets(cal_sheet).Cells(n, impact_col)
+ impact
    End If
Next l
End If

Next n
-----
Else
locate_col = m
    For n = 42 To last_row
        compartment = Cells(n, 7)
        sub_compa = Cells(n, 8)
        data_war_row = Cells(n, 5)
        data_use_row = Cells(n, 6)
        If data_war_row <> "" Then
            amount = Cells(n, m)
            If amount <> "" And amount <> 0 Then
                For l = 2 To 9
                    impact_cat_col = Sheets("LCI").Cells(l, 7)
                    If Sheets("WAR").Cells(data_war_row, impact_cat_col) <> 0 _
                        And Sheets("WAR").Cells(data_war_row, impact_cat_col) <> ""
Then
                        impact = Sheets("WAR").Cells(data_war_row, impact_cat_col) *
amount
                            Sheets(cal_sheet).Cells(n, last_col + 1) = Sheets(cal_sheet).Cells(n,
last_col + 1) + impact
                                End If
                                    Next l
                                        End If

```

```

End If
If data_use_row <> "" Then
amount = Cells(n, m)
  If amount <> "" And amount <> 0 Then
    For l = 10 To 12
      impact_col = usetox_col + (l - 10)
      Call step4.usetox_datacol(compartment, sub_compa, l)
      data_col = Sheets("LCI").Range("data_col").Value
      CF = Sheets("USEtox").Cells(data_use_row, data_col)
      If IsNumeric(CF) And CF > 0 Then
        impact = CF * amount
        Sheets(cal_sheet).Cells(n, impact_col) = Sheets(cal_sheet).Cells(n,
impact_col) + impact
      End If
    Next l
  End If
Next n
End If
Next m
'-----
Next m

For m = last_col + 2 To last_col + 12
Sheets(cal_sheet).Select
Cells(last_row + 2, m) = Application.Sum(Range(Cells(42, m), Cells(last_row, m))) /
Sheets("results").[e6]

Next m

Range(Cells(last_row + 2, last_col + 2), Cells(last_row + 2, last_col + 12)).Copy
If Right(section_name, 4) = "_sen" Then
Sheets("Sensitivity Analysis").Visible = True

```

```
Sheets("Sensitivity Analysis").Activate
Cells(impact_row, 11).Select
ActiveSheet.Paste

Cells(impact_row, 2) = section_name
Cells(impact_row, 10) = section_name
Cells(impact_row, 3) = Sheets(section_name).[b4]

Else
Sheets("results").Select
Cells(impact_row, 14).Select
ActiveSheet.Paste
Cells(impact_row, 13) = Cells(impact_row, 3)
End If
End Sub

Sub usetox_datacol(compartment, sub_compa, l)
Application.ScreenUpdating = False
    Select Case compartment
    Case "air"
        compa_col = Sheets("LCI").Cells(l, 7)
        Select Case sub_compa
        Case "urban air"
            sub_compa_col = 0
        Case "continental rural air"
            sub_compa_col = 1
        Case "unspecified"
            sub_compa_col = 2
        End Select
    Case "water"
        compa_col = Sheets("LCI").Cells(l, 8)
        Select Case sub_compa
```



```

    Case "continental fresh water"
    sub_compa_col = 0
    Case "continental sea water"
    sub_compa_col = 1
    Case "unspecified"
    sub_compa_col = 2
    End Select
Case "soil"
    compa_col = Sheets("LCI").Cells(1, 9)
    Select Case sub_compa
    Case "continental natural soil"
    sub_compa_col = 0
    Case "continental agricultural soil"
    sub_compa_col = 1
    Case "unspecified"
    sub_compa_col = 2
    End Select
End Select
data_col = compa_col + sub_compa_col
Sheets("LCI").Range("data_col") = data_col
End Sub

Sub GenerateLCIA_Chart()
Application.ScreenUpdating = False
Sheets("results").Activate
    For Each Worksheet In ThisWorkbook.Worksheets
        For Each ChartObject In Worksheet.ChartObjects
            ChartObject.Delete
        Next
    Next
Next
For pei = 14 To 24

```

```

last_row = Sheets("results").Range("b" & Rows.Count).End(xlUp).Row
ValChart = Sheets("results").Range(Cells(9, pei), Cells(last_row, pei))
XVal = Sheets("results").Range(Cells(9, 3), Cells(last_row, 3))
pei_name = Sheets("results").Cells(7, pei)
pei_unit = Sheets("results").Cells(8, pei)
Set Z = Sheets("results").ChartObjects.Add(275, 90, 550, 250)

```

With Z

```

    .Chart.ChartType = xlColumnClustered
    .Chart.SeriesCollection.NewSeries
    .Chart.SeriesCollection(1).XValues = Sheets("results").Range(Cells(9, pei),
Cells(last_row, pei))
    .Chart.SeriesCollection(1).Values = ValChart
    .Chart.SeriesCollection(1).Name = pei_unit
    .Chart.ChartTitle.Caption = pei_name
    .Chart.Axes(xlCategory, xlPrimary).HasTitle = True
    .Chart.Axes(xlCategory, xlPrimary).AxisTitle.Caption = "LCA section"
    .Chart.Axes(xlCategory, xlPrimary).AxisTitle.Font.Size = 14
    .Chart.Axes(xlCategory, xlPrimary).AxisTitle.Font.Bold = True
    .Chart.Axes(xlValue, xlPrimary).HasTitle = True
    .Chart.Axes(xlValue, xlPrimary).AxisTitle.Caption = pei_unit
    .Chart.Axes(xlValue, xlPrimary).AxisTitle.Font.Size = 14
    .Chart.Axes(xlValue, xlPrimary).AxisTitle.Font.Bold = True

```

End With

Next pei

```

last_row = Sheets("results").Range("b" & Rows.Count).End(xlUp).Row
Sheets("results").Cells(last_row + 1, 13) = "Total"
For pei = 14 To 24
    Sheets("results").Cells(last_row + 1, pei) = Application.Sum(Range(Cells(9, pei),
Cells(last_row, pei)))

```

```
Next pei
```

```
End Sub
```

```
Sub LCA_results_chart()
```

```
Application.ScreenUpdating = False
```

```
Sheets("results").Activate
```

```
On Error Resume Next
```

```
For Each ChartObject In ActiveSheet.ChartObjects
```

```
    ChartObject.Delete
```

```
Next
```

```
last_row = Sheets("results").Range("m" & Rows.Count).End(xlUp).Row
```

```
Range(Cells(9, 13), Cells(last_row, 13)).Copy
```

```
[z9].Select
```

```
ActiveSheet.Paste
```

```
Application.CutCopyMode = False
```

```
For j = 27 To 37
```

```
total_pei = Cells(last_row, j - 13)
```

```
For k = 9 To last_row
```

```
Cells(k, j) = Cells(k, j - 13) / total_pei
```

```
Next k
```

```
Next j
```

```
Set Z = Sheets("results").ChartObjects.Add(2280, 90, 550, 250)
```

```
With Z
```

```
.Chart.ChartType = xlBarStacked
```

```
.Chart.ApplyLayout (1)
```

```
.Chart.SetSourceData Source:=Range(Cells(9, 26), Cells(last_row - 1, 37))
```

```
.Chart.SeriesCollection(1).XValues = "'Results'!$aa$7:$ak$7"
```

```
.Chart.ChartTitle.Caption = "Potential environmental impacts"
```

```
End With
```

End Sub

D2 Interface source codes

Worksheet "SustianPro_data"

```
Private Sub cmd_AuxData_Click()
```

```
blocks = ActiveSheet.Range("C2")
streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H2")
reactions = ActiveSheet.Range("H3")
```

```
'MVA Information
```

```
'Cycle to insert the name of reactions in the table
```

```
    b = 1
```

```
    Do While b <= reactions
```

```
        ActiveSheet.Cells(34 + components + b, 1) = "R " & b
```

```
        b = b + 1
```

```
    Loop
```

```
'Cycle to insert the name of components in the table
```

```
    a = 0
```

```
    Do While a <= components - 1
```

```
        ActiveSheet.Cells(34 + components, 2 + a) = ActiveSheet.Cells(26 + a, 1)
```

```
        a = a + 1
```

```
    Loop
```

```
'Create a table
```

```
ActiveSheet.Range(Cells(33 + components, 2), Cells(33 + components, 1 + components)).Select
```

```
    With Selection
```

```
        .HorizontalAlignment = xlCenter
```

```
        .VerticalAlignment = xlBottom
```

```
        .WrapText = False
```

```
        .Orientation = 0
```

```
        .AddIndent = False
```

```
        .IndentLevel = 0
```

```

        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Range(Cells(34 + components, 1), Cells(34 + components + reactions, 1 +
components)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    ActiveSheet.Range(Cells(33 + components, 2), Cells(33 + components, 1 +
components)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic

```

```

End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Cells(33 + components, 2) = "Components"
ActiveSheet.Cells(31 + components, 1) = "Reactions"

```

```

ActiveSheet.Cells(31 + components, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With

```

'Demand Prices

```

ActiveSheet.Cells(36 + components + reactions, 1) = "Price"

```

```

ActiveSheet.Cells(36 + components + reactions, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False

```

```

.Subscript = False
.OutlineFont = False
.Shadow = False
.Underline = xlUnderlineStyleNone
.ColorIndex = xlAutomatic
.TintAndShade = 0
.ThemeFont = xlThemeFontNone
End With
ActiveSheet.Cells(40 + components + reactions, 1) = "Type"
ActiveSheet.Cells(38 + components + reactions, 2) = "Demand streams Price ($/kg)"

'Code to insert the names of the demand streams

a = 0
b = 0

Do While a <= streams - 1

    If ActiveSheet.Cells(24, 2 + a) = 0 Then

        ActiveSheet.Cells(39 + components + reactions, 2 + b) = ActiveSheet.Cells(22,
2 + a)
        b = b + 1

    End If

    a = a + 1

Loop

'Cycle to insert the name of components in the table
c = 0
Do While c <= components - 1

    ActiveSheet.Cells(41 + components + reactions + c, 1) = ActiveSheet.Cells(26
+ c, 1)
    c = c + 1

Loop

'Insert table

ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(38 + components +
reactions, 1 + b)).Select

```

With Selection

.HorizontalAlignment = xlCenter

.VerticalAlignment = xlBottom

.WrapText = False

.Orientation = 0

.AddIndent = False

.IndentLevel = 0

.ShrinkToFit = False

.ReadingOrder = xlContext

.MergeCells = False

End With

Selection.Merge

ActiveSheet.Range(Cells(39 + components + reactions, 1), Cells(40 + components + components + reactions, 1 + b)).Select

Selection.Borders(xlDiagonalDown).LineStyle = xlNone

Selection.Borders(xlDiagonalUp).LineStyle = xlNone

With Selection.Borders(xlEdgeLeft)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlEdgeTop)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlEdgeBottom)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlEdgeRight)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlInsideVertical)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlInsideHorizontal)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With


```

ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(38 + components
+ reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone

```

```

ActiveSheet.Range(Cells(39 + components + reactions, 2), Cells(40 + components
+ components + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)

```

```

.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With

```

```

With Selection.Borders(xlInsideHorizontal)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(40 + components + reactions, 1), Cells(40 + components
+ components + reactions, 1)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(39 + components
+ reactions, 1 + b)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Range(Cells(38 + components + reactions, 1), Cells(39 + components
+ components + reactions, 1 + b)).Select
  With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
  End With

'FEED PRICES

'Cycle to insert the name of components in the table
a = 0
Do While a <= components - 1

  ActiveSheet.Cells(44 + components * 2 + reactions, 2 + a) =
ActiveSheet.Cells(26 + a, 1)
  a = a + 1

Loop

ActiveSheet.Cells(43 + components * 2 + reactions, 2) = "Feed streams"
ActiveSheet.Cells(45 + components * 2 + reactions, 1) = "Price ($/kg)"

'Insert table

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(43 +
components * 2 + reactions, 1 + components)).Select
  With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
  End With
Selection.Merge

```

```

ActiveSheet.Range(Cells(44 + components * 2 + reactions, 1), Cells(45 +
components * 2 + reactions, 1 + components)).Select
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With

```

```

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(43 +
components * 2 + reactions, 1 + components)).Select
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium

```

```

        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone

```

```

    ActiveSheet.Range(Cells(44 + components * 2 + reactions, 2), Cells(45 +
components * 2 + reactions, 1 + components)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```

    ActiveSheet.Cells(45 + components * 2 + reactions, 1).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone

```

```

With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(44 +
components * 2 + reactions, 1 + components)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 1), Cells(45 +
components * 2 + reactions, 1 + components)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
    End With

```

'UTILITY PRICES

```

ActiveSheet.Cells(47 + components * 2 + reactions, 1) = "Utility Price"
ActiveSheet.Cells(47 + components * 2 + reactions, 1).Select
    Selection.Font.Bold = True

```

```

With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With

```

```

ActiveSheet.Cells(49 + components * 2 + reactions, 1) = "Utilities"
ActiveSheet.Cells(49 + components * 2 + reactions, 2) = "Price"
ActiveSheet.Cells(50 + components * 2 + reactions, 1) = "Heating(HP)($/GJ)"
ActiveSheet.Cells(51 + components * 2 + reactions, 1) = "Heating(LP)($/GJ)"
ActiveSheet.Cells(52 + components * 2 + reactions, 1) = "Cooling($/GJ)"
ActiveSheet.Cells(53 + components * 2 + reactions, 1) = "Electricity($/kWh)"

```

'Create a table

```

ActiveSheet.Range(Cells(49 + components * 2 + reactions, 1), Cells(53 +
components * 2 + reactions, 2)).Select

```

```

Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic

```

```

End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With

```

```

ActiveSheet.Range(Cells(49 + components * 2 + reactions, 1), Cells(49 +
components * 2 + reactions, 2)).Select

```

```

Selection.Font.Bold = True
Selection.Font.ColorIndex = 11
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous

```



```

    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

'HEAT EXCHANGED

```

c = 0
a = 0
b = 0

```

Do While a <= blocks - 1

```

    If ActiveSheet.Cells(11, 2 + a) = "HE" Or ActiveSheet.Cells(11, 2 + a) =
"Reactor" Or ActiveSheet.Cells(11, 2 + a) = "Filtration" Then

```

```

        ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
ActiveSheet.Cells(10, 2 + a)
        ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) = "Cp(kJ/kg-C)"
        b = b + 1

```

Else

```

    If ActiveSheet.Cells(11, 2 + a) = "Pump" Or ActiveSheet.Cells(11, 2 + a) =
"Comp" Or ActiveSheet.Cells(11, 2 + a) = "Centrifuge" Or ActiveSheet.Cells(11, 2 +
a) = "Mixer" Then

```

```

        ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
ActiveSheet.Cells(10, 2 + a)
        ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) = "Den(kg/m3)"
        b = b + 1

```

Else

```

    If ActiveSheet.Cells(11, 2 + a) = "Column" Then

```

```

        ActiveSheet.Cells(67 + components * 3 + reactions, 2 + c) =
ActiveSheet.Cells(10, 2 + a)
        c = c + 1

```

Else

```

    If ActiveSheet.Cells(11, 2 + a) = "Evap" Or ActiveSheet.Cells(11, 2 + a) =
    "Cond" Then

```

```

        ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
        ActiveSheet.Cells(10, 2 + a)
        ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) =
        "Hvap(kJ/kg)"
        b = b + 1

```

```

    End If

```

```

End If

```

```

End If

```

```

End If

```

```

a = a + 1

```

```

Loop

```

```

'Insert a table

```

```

If b = 0 Then

```

```

Else

```

```

ActiveSheet.Cells(55 + components * 2 + reactions, 1) = "Heats and Component's
Properties"

```

```

ActiveSheet.Cells(55 + components * 2 + reactions, 1).Select

```

```

    Selection.Font.Bold = True

```

```

    With Selection.Font

```

```

        .Name = "Arial"

```

```

        .Size = 12

```

```

        .Strikethrough = False

```

```

        .Superscript = False

```

```

        .Subscript = False

```

```

        .OutlineFont = False

```

```

        .Shadow = False

```

```

        .Underline = xlUnderlineStyleNone

```

```

        .ColorIndex = xlAutomatic

```

```

        .TintAndShade = 0

```

```

        .ThemeFont = xlThemeFontNone
    End With

```

```

ActiveSheet.Cells(61 + components * 2 + reactions, 1) = "Cooling duty (GJ/hr)"

```

```

ActiveSheet.Cells(59 + components * 2 + reactions, 1) = "Heating duty (HP)(GJ/hr)"

```

```

ActiveSheet.Cells(60 + components * 2 + reactions, 1) = "Heating duty (LP)(GJ/hr)"

```

```

ActiveSheet.Cells(62 + components * 2 + reactions, 1) = "Electrical work (kW)"

```

```

ActiveSheet.Cells(63 + components * 2 + reactions, 1) = "Properties"

```

```

ActiveSheet.Cells(57 + components * 2 + reactions, 2) = "Units"

'Cycle to insert the name of components in the table

d = 0
Do While d <= components - 1

    ActiveSheet.Cells(64 + components * 2 + reactions + d, 1) =
ActiveSheet.Cells(26 + d, 1)
    d = d + 1

Loop
ActiveSheet.Range(Cells(57 + components * 2 + reactions, 2), Cells(57 +
components * 2 + reactions, 1 + b)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic

```

```

End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)

```

```

        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```

    ActiveSheet.Range(Cells(58 + components * 2 + reactions, 2), Cells(58 +
components * 2 + reactions, 1 + b)).Select
    Selection.Font.ColorIndex = 11
    Selection.Font.Bold = True

```

```

    ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1)).Select
    Selection.Font.ColorIndex = 11
    Selection.Font.Bold = True

```

```

    ActiveSheet.Range(Cells(63 + components * 2 + reactions, 2), Cells(63 +
components * 3 + reactions, 1 + b)).Select
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

```

```

    ActiveSheet.Range(Cells(58 + components * 2 + reactions, 2), Cells(63 +
components * 3 + reactions, 1 + b)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
    End With

```

```

        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```

    ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(58 +
components * 2 + reactions, 1 + b)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

```

```

With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(63 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(57 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
End With

```

```

End If

```

If c > 0 Then

```
ActiveSheet.Cells(68 + components * 3 + reactions, 1) = "Condenser(GJ/hr)"
ActiveSheet.Cells(72 + components * 4 + reactions, 1) = "Reboiler (HP)(GJ/hr)"
ActiveSheet.Cells(73 + components * 4 + reactions, 1) = "Reboiler (LP)(GJ/hr)"
ActiveSheet.Cells(70 + components * 3 + reactions, 2) = "Hv(kJ/kg)"
ActiveSheet.Cells(75 + components * 4 + reactions, 2) = "Hv(kJ/kg)"
ActiveSheet.Cells(69 + components * 3 + reactions, 1) = "Streams"
ActiveSheet.Cells(74 + components * 4 + reactions, 1) = "Streams"
```

'Cycle to insert the name of components in the table

c1 = 0

Do While c1 <= components - 1

```
    ActiveSheet.Cells(71 + components * 3 + reactions + c1, 1) =
ActiveSheet.Cells(26 + c1, 1)
    c1 = c1 + 1
```

Loop

c2 = 0

Do While c2 <= components - 1

```
    ActiveSheet.Cells(76 + components * 4 + reactions + c2, 1) =
ActiveSheet.Cells(26 + c2, 1)
    c2 = c2 + 1
```

Loop

'insert table

```
ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(75 +
components * 5 + reactions, 1 + c)).Select
```

With Selection

.HorizontalAlignment = xlCenter

.VerticalAlignment = xlBottom

.WrapText = False

.Orientation = 0

.AddIndent = False

.IndentLevel = 0

.ShrinkToFit = False

.ReadingOrder = xlContext

.MergeCells = False

End With

Selection.Borders(xlDiagonalDown).LineStyle = xlNone

Selection.Borders(xlDiagonalUp).LineStyle = xlNone

With Selection.Borders(xlEdgeLeft)


```

        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```

ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(67 +
components * 3 + reactions, 1 + c)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)

```

```

        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(70 + components * 3 + reactions, 2), Cells(70 +
components * 3 + reactions, 1 + c)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    ActiveSheet.Range(Cells(70 + components * 3 + reactions, 1), Cells(70 +
components * 3 + reactions, 1 + c)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)

```

```

        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(71 + components * 4 + reactions, 1), Cells(71 +
components * 4 + reactions, 1 + c)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(75 + components * 4 + reactions, 2), Cells(75 +
components * 4 + reactions, 1 + c)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
    End With

```

```

        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    ActiveSheet.Range(Cells(75 + components * 4 + reactions, 1), Cells(75 +
components * 4 + reactions, 1 + c)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(75 +
components * 5 + reactions, 1)).Select
    Selection.Font.ColorIndex = 11
    Selection.Font.Bold = True

End If

'Reactions information

Dim dp As Integer

```

```
dp = ActiveSheet.Cells(1, 14)
```

```
ActiveSheet.Cells(77 + components * 5 + reactions, 1) = "Reactions"
```

```
ActiveSheet.Cells(77 + components * 5 + reactions, 1).Select
```

```
Selection.Font.Bold = True
```

```
With Selection.Font
```

```
    .Name = "Arial"
```

```
    .Size = 12
```

```
    .Strikethrough = False
```

```
    .Superscript = False
```

```
    .Subscript = False
```

```
    .OutlineFont = False
```

```
    .Shadow = False
```

```
    .Underline = xlUnderlineStyleNone
```

```
    .ColorIndex = xlAutomatic
```

```
    .TintAndShade = 0
```

```
    .ThemeFont = xlThemeFontNone
```

```
End With
```

```
ActiveSheet.Cells(79 + components * 5 + reactions, 1) = "Product"
```

```
ActiveSheet.Cells(80 + components * 5 + reactions, 1) = "Stream"
```

```
ActiveSheet.Range(Cells(79 + components * 5 + reactions, 1), Cells(80 +  
components * 5 + reactions, 1 + dp)).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```

With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

```

With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With

```

```

ActiveSheet.Range(Cells(79 + components * 5 + reactions, 1), Cells(80 +
components * 5 + reactions, 1)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Cells(82 + components * 5 + reactions, 2) = "Units"

```

```

'Insert the name of the reactive units and count them

```

```

inblo = 0
numreact = 0

```

```

Do While inblo <= blocks - 1

```

```

    If ActiveSheet.Cells(11, 2 + inblo) = "Reactor" Then

```

```

        ActiveSheet.Cells(83 + components * 5 + reactions, 2 + numreact) =
ActiveSheet.Cells(10, 2 + inblo)
        numreact = numreact + 1

```

```

    End If

```

```

    inblo = inblo + 1

```

```

Loop

```

```

'Insert the name of the reactions

```

```
Dim fini As Boolean
```

```
a = 0
a1 = 0
fini = False
```

```
Do Until fini = True
```

```
  If ActiveSheet.Cells(35 + components + a, 1) = 0 Then
```

```
    fini = True
```

```
  Else
```

```
    ActiveSheet.Cells(84 + components * 5 + reactions + a1, 1) =
ActiveSheet.Cells(35 + components + a, 1)
    ActiveSheet.Cells(85 + components * 5 + reactions + a1, 1) = "Component"
    a = a + 1
    a1 = a1 + 2
```

```
  End If
```

```
Loop
```

```
ActiveSheet.Range(Cells(83 + components * 5 + reactions, 1), Cells(83 +
components * 5 + reactions * 3, 1 + numreact)).Select
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
```

```

End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
ActiveSheet.Range(Cells(82 + components * 5 + reactions, 2), Cells(82 +
components * 5 + reactions, 1 + numreact)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)

```



```

.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone

```

```

ActiveSheet.Range(Cells(84 + components * 5 + reactions, 1), Cells(84 +
components * 5 + reactions * 3, 1)).Select
Selection.Font.ColorIndex = 11
Selection.Font.Bold = True

```

```

ActiveSheet.Range(Cells(83 + components * 5 + reactions, 2), Cells(83 +
components * 5 + reactions, 1 + numreact)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

'Safety

```

ActiveSheet.Cells(87 + components * 5 + reactions * 3, 1) = "Components"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 2) = "Flash Point(°C)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 3) = "Boiling Point(°C)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 4) = "UEL(%vol)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 5) = "LEL(%vol)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 6) = "Toxic limit(ppm)"

```

```

ActiveSheet.Cells(85 + components * 5 + reactions * 3, 1) = "Safety"
ActiveSheet.Cells(85 + components * 5 + reactions * 3, 1).Select
Selection.Font.Bold = True
With Selection.Font
.Name = "Arial"
.Size = 12
.Strikethrough = False
.Superscript = False
.Subscript = False
.OutlineFont = False
.Shadow = False

```

```

.Underline = xlUnderlineStyleNone
.ColorIndex = xlAutomatic
.TintAndShade = 0
.ThemeFont = xlThemeFontNone
End With

```

```
Dim fg As Integer
```

```
fg = 0
```

```
Do While fg <= components - 1
```

```

    ActiveSheet.Cells(88 + components * 5 + reactions * 3 + fg, 1) =
ActiveSheet.Cells(26 + fg, 1)
    fg = fg + 1

```

```
Loop
```

```

ActiveSheet.Range(Cells(87 + components * 5 + reactions * 3, 1), Cells(87 +
components * 5 + reactions * 3, 6)).Select
    Selection.Font.ColorIndex = 11
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

End With

```
ActiveSheet.Range(Cells(87 + components * 5 + reactions * 3, 1), Cells(87 +
components * 6 + reactions * 3, 6)).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

End With

```
With Selection.Borders(xlEdgeTop)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

End With

```
With Selection.Borders(xlEdgeBottom)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

End With

```
With Selection.Borders(xlEdgeRight)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

End With

```
With Selection.Borders(xlInsideVertical)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

End With

```
With Selection.Borders(xlInsideHorizontal)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

End With

End Sub

```
Private Sub cmd_Stre_Click()
```

```
blocks = ActiveSheet.Range("C2")
```

```
streams = ActiveSheet.Range("C3")
```

```
components = ActiveSheet.Range("H2")
```

```
reactions = ActiveSheet.Range("H3")
```

'Streams information

'Make a table with the right size surrounding the streams datas

```
ActiveSheet.Cells(20, 1) = "Streams"
```

```
Range(Cells(22, 1), Cells(20 + 9 + components, 1 + streams)).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideVertical)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideHorizontal)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

'Subtract the background from the table.

```
Selection.Interior.ColorIndex = xlNone
```

'Make the text in the table all in the same color(black)

```
Selection.Font.ColorIndex = 0
```

```
'Change borders and color of the letters
Range(Cells(22, 1), Cells(20 + 9 + components, 1)).Select
```

```
Selection.Font.ColorIndex = 11
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
```

```
Range(Cells(22, 1), Cells(22, 1 + streams)).Select
```

```
Selection.Font.ColorIndex = 11
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
```

```

With Selection.Borders(xlEdgeRight)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

'Insert the right name to describe each row in the table

Cells(22, 1) = "Stream Name"
Cells(23, 1) = "Initial Unit"
Cells(24, 1) = "Final Unit"
Cells(25, 1) = "Components (kg/h)"

'Cycle to insert the name of components in the table
a = 0
Do While a <= components

    Cells(26 + a, 1) = ActiveSheet.Cells(15, 3 + a)
    a = a + 1

Loop

Cells(26 + components, 1) = "Total Mass Flow(kg/h)"
Cells(27 + components, 1) = "Temperature (°C)"
Cells(28 + components, 1) = "Pressure (atm)"
Cells(29 + components, 1) = "Enthalpy Flow (GJ/h)"

End Sub

Private Sub cmd_gendata_Click()
ActiveSheet.Select

blocks = ActiveSheet.Range("C2")

```

```

streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H2")
reactions = ActiveSheet.Range("H3")

'General Data Tables

Dim a As Integer

'Warning message when the user didn't fill the values of blocks, or components, or
streams.

If Range("C2") = Empty Or Range("C3") = Empty Or Range("H2") = Empty Then

    Dim answer As String
    Phrase = "Please insert the general data"
    Title = "Warning"
    answer = MsgBox(Phrase, vbExclamation, Title)

Else

    'Generate a table with the block size, in order to the user insert the blocks's name
    and type.

    ActiveSheet.Cells(7, 1) = "General Data"
    ActiveSheet.Cells(10, 1) = "Name"
    ActiveSheet.Cells(11, 1) = "Type"
    ActiveSheet.Cells(9, 2) = "Units"
    ActiveSheet.Cells(12, 1) = "Operation"

    Range(Cells(10, 1), Cells(12, 1 + blocks)).Select

    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```

End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Range(Cells(9, 2), Cells(9, 1 + blocks)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)

```



```

.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone

```

```

ActiveSheet.Cells(14, 3) = "Components"
ActiveSheet.Cells(15, 1) = "Name"
ActiveSheet.Cells(16, 1) = "Molecular weight (g/mol)"
ActiveSheet.Cells(17, 1) = "Hazardous"
ActiveSheet.Cells(18, 1) = "Type"

```

'Generate a table with the components size, in order to the user insert the components's name and their molecular weight.

```

Range(Cells(15, 1), Cells(18, 2 + components)).Select
Application.CutCopyMode = False
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
.LineStyle = xlContinuous

```

```

    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Range(Cells(14, 3), Cells(14, 2 + components)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Range(Cells(15, 1), Cells(15, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0

```

```
.ShrinkToFit = False
.ReadingOrder = xlContext
.MergeCells = False
End With
Selection.Merge
Range(Cells(16, 1), Cells(16, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Range(Cells(17, 1), Cells(17, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Range(Cells(18, 1), Cells(18, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
End If
```

End Sub

Private Sub CommandButton1_Click()

```
blocks = ActiveSheet.Range("C2")
streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H2")
reactions = ActiveSheet.Range("H3")
```

```
inblo = 0
numreact = 0
```

```
Do While inblo <= blocks - 1
```

```
If ActiveSheet.Cells(11, 2 + inblo) = "Reactor" Then
```

```
numreact = numreact + 1
```

```
End If
```

```
inblo = inblo + 1
```

```
Loop
```

```
Dim fi As Workbook
Dim fi2 As Workbook
```

```
Set fi2 = ActiveWorkbook
```

```
b = TextBox1
Workbooks.Open Filename:=b
```

```
Set fi = ActiveWorkbook
```

```
fi.Activate
```

```
'Copy components Name
```

```
Dim comp()
```

```
ReDim comp(components - 1)
```

```
hj = 0
```

```
Do While hj <= components - 1
```

```

comp(hj) = ActiveWorkbook.ActiveSheet.Cells(10 + hj, 4)
hj = hj + 1

```

Loop

'Copy flow-rates

```
Dim co()
```

```
ReDim co(components - 1, streams - 1)
```

```
a = 0
```

```
c = 0
```

```
Do While a <= components - 1
```

```
    Do While c <= streams - 1
```

```
        co(a, c) = ActiveWorkbook.ActiveSheet.Cells(10 + a, 6 + c)
```

```
        c = c + 1
```

```
    Loop
```

```
    c = 0
```

```
    a = a + 1
```

Loop

'Copy temperature

```
Dim temp()
```

```
Dim inctem As Integer
```

```
ReDim temp(streams - 1)
```

```
inctemp = 0
```

```
Do While inctemp <= streams - 1
```

```
    temp(inctemp) = ActiveWorkbook.ActiveSheet.Cells(6, 6 + inctemp)
```

```
    inctemp = inctemp + 1
```

Loop

'Copy pressure

```
Dim pre()
```

```
Dim incpre As Integer
```

```

ReDim pre(streams - 1)

incpre = 0

Do While incpre <= streams - 1

    pre(incpre) = ActiveWorkbook.ActiveSheet.Cells(7, 6 + incpre)
    incpre = incpre + 1

Loop

'Copy the streams names

Dim names()
Dim incna As Integer

ReDim names(streams - 1)

incna = 0

Do While incna <= streams - 1

    names(incna) = ActiveWorkbook.ActiveSheet.Cells(3, 6 + incna)
    incna = incna + 1

Loop

fi2.Activate

'Paste flow-rates

a = 0
c = 0

Do While a <= components - 1

    Do While c <= streams - 1

        ActiveWorkbook.ActiveSheet.Cells(26 + a, 2 + c) = co(a, c)
        c = c + 1

    Loop

    c = 0
    a = a + 1

```

Loop

'Paste temperature

inctemp = 0

Do While inctemp <= streams - 1

```
    ActiveWorkbook.ActiveSheet.Cells(27 + components, 2 + inctemp) =
temp(inctemp)
    inctemp = inctemp + 1
```

Loop

'Paste pressure

incpre = 0

Do While incpre <= streams - 1

```
    ActiveWorkbook.ActiveSheet.Cells(28 + components, 2 + incpre) = pre(incpre)
    incpre = incpre + 1
```

Loop

'Paste the streams names

incna = 0

Do While incna <= streams - 1

```
    ActiveWorkbook.ActiveSheet.Cells(22, 2 + incna) = names(incna)
    incna = incna + 1
```

Loop

'inc = 0

'Do While inc <= streams

```
'ActiveWorkbook.activesheet.Cells(6 + components, 2 + inc).FormulaR1C1 =
"=SUM((INDIRECT("'"R[-]"&
GeneralData!R11C4&""']C"',FALSE)):(INDIRECT("'"R[-1]C"',FALSE)))"
'inc = inc + 1
```

'Loop

End Sub

Worksheet "start menu"

```
Private Sub CommandButton10_Click()  
F7_retrieve_sustainpro.Show  
End Sub
```

```
Private Sub CommandButton11_Click()  
F8_retrieve_lcsoft.Show  
End Sub
```

```
Private Sub CommandButton12_Click()  
F6_retrieve_econ.Show  
End Sub
```

```
Private Sub CommandButton1_Click()  
Sheets("Start menu").CommandButton15.BackColor = &HC00000  
Call I1_sustainpro.link_sustainpro_data  
End Sub
```

```
Private Sub CommandButton13_Click()  
Sheets("Start menu").CommandButton20.BackColor = &HC00000  
Call I3_ECON.link_econ_data  
End Sub
```

```
Private Sub CommandButton14_Click()  
Sheets("Start menu").CommandButton18.BackColor = &HC00000  
Call I2_LCSoft.transfer_to_lcsoft  
End Sub
```

```
Private Sub CommandButton18_Click()  
  
End Sub
```

```
Private Sub CommandButton2_Click()  
ActiveWorkbook.Save  
ActiveWorkbook.Close  
End Sub
```

```
Private Sub CommandButton23_Click()  
  
End Sub
```

```
Private Sub CommandButton26_Click()  
Call I0_Interface.reset_all  
End Sub
```

```
Private Sub CommandButton3_Click()  
If Sheets("Start menu").CommandButton21.BackColor = &HC00000 _
```



```

And Sheets("Start menu").CommandButton22.BackColor = &HC00000 Then
Sheets("sustainpro").Select
Else
MsgBox "Please generate equipment table and stream table before performing
sustainability analysis"
End If
End Sub

```

```

Private Sub CommandButton4_Click()
If Sheets("Start menu").CommandButton21.BackColor = &HC00000 _
And Sheets("Start menu").CommandButton22.BackColor = &HC00000 Then
Sheets("econ").Select
Call I3_ECON.first_clickecon
Else
MsgBox "Please generate equipment table and stream table before performing
economic analysis"
End If
End Sub

```

```

Private Sub CommandButton5_Click()
If Sheets("Start menu").CommandButton21.BackColor = &HC00000 _
And Sheets("Start menu").CommandButton22.BackColor = &HC00000 Then
Call I2_LCSoft.gen_sim_tables
Sheets("lcsoft").Select
Sheets("Start menu").CommandButton17.BackColor = &HC00000
Else
MsgBox "Please generate equipment table and stream table before performing
environmental analysis"
End If
End Sub

```

```

Private Sub CommandButton6_Click()
If Sheets("Start menu").CommandButton21.BackColor = &HC00000 Then
    change_table = MsgBox("Equipment table has already generated, Do you want to
change?", vbYesNoCancel)
    If change_table = 6 Then
        Sheets("equipment table").Select
        Sheets("Start menu").CommandButton21.BackColor =
Sheets("int_data").Range("not")
    End If
Else
Sheets("equipment table").Select
End If

```

'equipment table

```
Sheets("equipment table").Select
```

```
End Sub
```

```
Private Sub CommandButton7_Click()
```

```
Sheets("Start menu").CommandButton7.BackColor = &H80FF80
```

```
Call I1_sustainpro.link_sustainpro_data
```

```
End Sub
```

```
Private Sub CommandButton8_Click()
```

```
If Sheets("Start menu").CommandButton8.BackColor = &H80FF80 Then
```

```
    change_table = MsgBox("Stream table has already generated, Do you want to  
change?", vbYesNoCancel)
```

```
    If change_table = 6 Then
```

```
        Sheets("stream table").Select
```

```
        Sheets("Start menu").CommandButton8.BackColor = &H8000000F
```

```
    End If
```

```
Else
```

```
Sheets("stream table").Select
```

```
End If
```

```
End Sub
```

```
Private Sub CommandButton9_Click()
```

```
Call I0_Interface.check_pj_name
```

```
I1_project_data.Show
```

```
End Sub
```

Worksheet "ECON"

```
Private Sub CommandButton1_Click()
```

```
Range(Cells(11, 1), Cells(5000, 5000)).Delete
```

```
Call I3_ECON.clear_data
```

```
money_unit = "$"
```

```
Cells(11, 2) = "RAW MATERIAL"
```

```
Cells(13, 2) = "NAME"
```

```
Cells(13, 3) = "QUANTITY,KG/YR"
```

```
Cells(13, 4) = "PRICE," & " " & money_unit & "/KG"
```

```
Cells(11, 2).Select
```

```
Call Module1.change_coler_A_cell_bold
```

```
Range(Cells(13, 2), Cells(13, 4)).Select
```

```
Call Module1.change_coler_A_cell_bold
```

```
For i = 9 To Sheets("int_data").Range("no_comp") + 8
```

```
    next_row = Range("b" & Rows.Count).End(xlUp).Offset(1, 0).row
```

```

If Sheets("stream table").Cells(i, 4) = "RM" Then
Sheets("econ").Cells(next_row, 2) = Sheets("stream table").Cells(i, 1)
Sheets("econ").Cells(next_row, 4) = Sheets("stream table").Cells(i, 5)
  For j = 6 To Sheets("int_data").Range("no_stream") + 5
  If Sheets("stream table").Cells(6, j) = 0 Then 'check string
  quantity_rm = Sheets("stream table").Cells(i, j) * 24 * Sheets("econ").[c4]
  Sheets("econ").Cells(next_row, 3) = Sheets("econ").Cells(next_row, 3) +
quantity_rm
  End If
  Next j
End If
Next i

last_row = Range("b" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 2) = "PRODUCT"
Cells(last_row + 4, 2) = "NAME"
Cells(last_row + 4, 3) = "QUANTITY,KG/YR"
Cells(last_row + 4, 4) = "PRICE," & " " & money_unit & "/KG"

Cells(last_row + 2, 2).Select
Call Module1.change_coler_A_cell_bold
Range(Cells(last_row + 4, 2), Cells(last_row + 4, 4)).Select
Call Module1.change_coler_A_cell_bold

For i = 9 To Sheets("int_data").Range("no_comp") + 8
next_row = Range("b" & Rows.Count).End(xlUp).Offset(1, 0).row
If Sheets("stream table").Cells(i, 4) = "P" Then
Sheets("econ").Cells(next_row, 2) = Sheets("stream table").Cells(i, 1)
Sheets("econ").Cells(next_row, 4) = Sheets("stream table").Cells(i, 5)
  For j = 6 To Sheets("int_data").Range("no_stream") + 5
  If Sheets("stream table").Cells(7, j) = 0 Then 'check string
  quantity_p = Sheets("stream table").Cells(i, j) * 24 * Sheets("econ").[c4]
  Sheets("econ").Cells(next_row, 3) = Sheets("econ").Cells(next_row, 3) +
quantity_p
  End If
  Next j
End If
Next i

For i = 3 To 19
If Sheets("econ").Cells(9, i) <> "-" Or Sheets("econ").Cells(9, i) <> "" And
IsNumeric(Sheets("econ").Cells(9, i)) Then
no_e = Sheets("econ").Cells(9, i)
Sheets("int_data").Cells(17, i) = no_e
last_row = Sheets("econ").Range("b" & Rows.Count).End(xlUp).row
Cells(last_row + 3, 2) = Sheets("econ").Cells(8, i)
Sheets("int_data").Cells(18, i) = last_row + 3

```

```

    For k = 22 To Sheets("int_data").Cells(22, i).End(xlDown).row
    next_col = Sheets("econ").Cells(last_row + 4,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
    Sheets("econ").Cells(last_row + 4, next_col) = Sheets("int_data").Cells(k, i)
    Next k

    For m = 1 To no_e
    pre_name = Sheets("int_data").Cells(21, i)
    full_no_name = pre_name & m
    Cells(last_row + 4, 2).Offset(m, 0) = full_no_name
    Next m
    no_col = Sheets("int_data").Cells(22, i).End(xlDown).row - 21
    Sheets("int_data").Cells(16, i) = no_col
    Range(Cells(last_row + 4, 2), Cells(last_row + 4 + no_e, 1 + no_col)).Select
    Call Module1.create_grid
    Range(Cells(last_row + 4, 2), Cells(last_row + 4, no_col + 1)).Select
    Call Module1.change_color_A_cell_bold
    Range(Cells(last_row + 5, 3), Cells(last_row + 4 + no_e, no_col + 1)).Select
    Call I3_ECON.fill_out

    End If
    Next i
    Cells(11, 1).Select

End Sub

```

```

Private Sub CommandButton2_Click()
Call I3_ECON.record_path
Sheets("Start menu").Select
Sheets("Start menu").CommandButton19.BackColor = &HC00000
End Sub

```

Worksheet "LCSoft"

```

Private Sub CommandButton1_Click()
Call I2_LCSoft.list_section1
End Sub

```

```

Private Sub CommandButton2_Click()
F9_lcsoft_section.Show
End Sub

```

```

Private Sub CommandButton3_Click()
Call I2_LCSoft.del_lcsoft_results
Call I2_LCSoft.list_section2
Sheets("Start menu").Select
Sheets("Start menu").CommandButton17.BackColor = &HC00000

```

End Sub

Worksheet "SustainPro"

```
Private Sub CommandButton1_Click()
Range(Cells(8, 1), Cells(5000, 5000)).Delete
Call general_data
Call reaction_data
Call demad_stream_price
```

End Sub

Sub general_data()

```
[a9].Select
ActiveCell = "General data"
  With Selection.Font
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = 0
  End With
  With Selection.Interior
    .Pattern = xlSolid
    .PatternColorIndex = xlAutomatic
    .Color = 6299648
    .TintAndShade = 0
    .PatternTintAndShade = 0
  End With
```

```
[a12] = "Name"
[a13] = "Type"
[a14] = "Operation"
```

```
total_unit = Sheets("Equipment Table").[b2].Value
If IsNumeric(total_unit) Then
For i = 2 To total_unit + 1
equip_name = Sheets("Equipment Table").Cells(8, i)
equip_type = Sheets("Equipment Table").Cells(9, i)
Cells(12, i) = equip_name
Cells(13, i) = equip_type
Next i
```

```
lastcol = Sheets("sustainpro").Cells(12, Columns.Count).End(xlToLeft).Column
If Sheets("int_data").Range("process_type") = "Batch" Then
```

```

For i = 9 To lastcol
Sheets("SustainPro").Cells(14, i - 7) = "BO"
Next i
End If

```

```

Range(Cells(12, 1), Cells(14, lastcol)).Select
Call Module1.create_grid

```

```

Sheets("SustainPro").Cells(11, 2) = "Units"
Range(Cells(11, 2), Cells(11, lastcol)).Select
Call Module1.merge_highlight
End If

```

```

'-----
[a17] = "Name"
[a18] = "Molecular weight (g/mol)"
[a19] = "Hazardous"
[a20] = "Type"

```

```

For i = 9 To Sheets("int_data").Range("no_comp") + 8
Cells(17, i - 7) = Sheets("stream table").Cells(i, 1)
Cells(18, i - 7) = Sheets("stream table").Cells(i, 2)
Cells(19, i - 7) = Sheets("stream table").Cells(i, 3)
Cells(20, i - 7) = Sheets("stream table").Cells(i, 4)
Next i

```

```

Sheets("SustainPro").Cells(16, 2) = "Components"

```

```

last_col_st = Sheets("SustainPro").Cells(17, Columns.Count).End(xlToLeft).Column
Range(Cells(16, 2), Cells(16, last_col_st)).Select
Call Module1.merge_highlight

```

```

Range(Cells(17, 1), Cells(20, last_col_st)).Select
Call Module1.create_grid

```

```

End Sub

```

```

Sub for_ap()
last_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 1) = "Accumulation path data"
  With Cells(last_row + 2, 1).Font
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = 0
  End With
  With Cells(last_row + 2, 1).Interior
    .Pattern = xlSolid
  End With

```

```

        .PatternColorIndex = xlAutomatic
        .Color = 6299648
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With

    Cells(last_row + 4, 1) = "Operation"
    Cells(last_row + 4, 2) = "Type"
    Cells(last_row + 4, 3) = "Time (s)"
    Cells(last_row + 4, 4) = "Volume (m3)"
    Cells(last_row + 4, 5) = "Energy"

    Range(Cells(last_row + 4, 1), Cells(last_row + 4, 5)).Select
    Selection.Font.Bold = True
    For i = 2 To Sheets("int_data").Range("no_unit") + 1
        next_row = Range("a" & Rows.Count).End(xlUp).Offset(1, 0).row
        Cells(next_row, 1) = Sheets("equipment table").Cells(8, i)
        Cells(next_row, 2) = Sheets("equipment table").Cells(9, i)
        Cells(next_row, 3) = Sheets("equipment table").Cells(19, i)
        Cells(next_row, 4) = Sheets("equipment table").Cells(20, i)
    Next i
    last_row2 = Range("a" & Rows.Count).End(xlUp).row
    Range(Cells(last_row + 4, 1), Cells(last_row2, 5)).Select
    Call Module1.create_grid

    Cells(last_row2 + 2, 1) = "Compound"
    Cells(last_row2 + 2, 2) = "Density (kg/m3)"
    Cells(last_row2 + 2, 3) = "Delta Hf (kJ/kg)"
    Range(Cells(last_row2 + 2, 1), Cells(last_row2 + 2, 3)).Select
    Selection.Font.Bold = True

    For i = 9 To Sheets("int_data").Range("no_comp") + 8
        last_row3 = Range("a" & Rows.Count).End(xlUp).row
        Cells(last_row3 + 1, 1) = Sheets("stream table").Cells(i, 1)
    Next i

    last_row4 = Range("a" & Rows.Count).End(xlUp).row
    Range(Cells(last_row2 + 2, 1), Cells(last_row4, 3)).Select
    Call Module1.create_grid

    Cells(last_row4 + 2, 1) = "Reactor"
    Cells(last_row4 + 2, 2) = "Desired Product"
    Range(Cells(last_row4 + 2, 1), Cells(last_row4 + 2, 2)).Select
    Selection.Font.Bold = True
    For i = 2 To Sheets("int_data").Range("no_unit") + 1

```

```

If Sheets("equipment table").Cells(9, i) = "Reactor" Then
last_row5 = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row5 + 1, 1) = Sheets("equipment table").Cells(8, i)
End If
Next i

last_row6 = Range("a" & Rows.Count).End(xlUp).row
Range(Cells(last_row4 + 2, 1), Cells(last_row6, 2)).Select
Call Module1.create_grid

Cells(last_row6 + 2, 1) = "Separation"
Cells(last_row6 + 2, 2) = "Recovery compound"
Cells(last_row6 + 2, 3) = "Compound"
Cells(last_row6 + 2, 4) = "Property"
Range(Cells(last_row6 + 2, 1), Cells(last_row6 + 2, 4)).Select
Selection.Font.Bold = True

For i = 2 To Sheets("int_data").Range("no_unit") + 1
If Sheets("equipment table").Cells(9, i) = "Column" Or _
Sheets("equipment table").Cells(9, i) = "Centrifuge" Or _
Sheets("equipment table").Cells(9, i) = "Filtration" Or _
Sheets("equipment table").Cells(9, i) = "Chromatography" Or _
Sheets("equipment table").Cells(9, i) = "Extraction" Or _
Sheets("equipment table").Cells(9, i) = "Dry" Or _
Sheets("equipment table").Cells(9, i) = "Evap" Or _
Sheets("equipment table").Cells(9, i) = "Cristalization" Then
next_sep_row = Range("c" & Rows.Count).End(xlUp).Offset(1, 0).row

Cells(next_sep_row, 1) = Sheets("equipment table").Cells(8, i)
  For p = 9 To Sheets("int_data").Range("no_comp") + 8
    Cells(next_sep_row + p - 9, 3) = Sheets("stream table").Cells(p, 1)
  Next p
End If
Next i
last_row7 = Range("c" & Rows.Count).End(xlUp).row
Range(Cells(last_row6 + 2, 1), Cells(last_row7, 4)).Select
Call Module1.create_grid

End Sub

Sub reaction_data()
last_row_p = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row_p + 2, 1) = "Reaction data"
  With Cells(last_row_p + 2, 1).Font
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = 0
  End With

```



```

With Cells(last_row_p + 2, 1).Interior
    .Pattern = xlSolid
    .PatternColorIndex = xlAutomatic
    .Color = 6299648
    .TintAndShade = 0
    .PatternTintAndShade = 0
End With
Cells(last_row_p + 4, 1) = "Product"
Cells(last_row_p + 5, 1) = "Identify product stream"
Range(Cells(last_row_p + 4, 1), Cells(last_row_p + 5, 1)).Select
Selection.Font.Bold = True

last_com = Cells(20, Columns.Count).End(xlToLeft).Column
For k = 2 To last_com
If Cells(20, k) = "P" Then
product_name = Cells(17, k)
last_use_col = Cells(last_row_p + 4, Columns.Count).End(xlToLeft).Column
Cells(last_row_p + 4, last_use_col + 1) = product_name
End If
Next k

Range(Cells(last_row_p + 4, 1), Cells(last_row_p + 5, Cells(last_row_p + 4,
Columns.Count).End(xlToLeft).Column)).Select
Call Module1.create_grid

last_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 4, 2) = "Components"
Cells(last_row + 5, 1) = "Reaction\Stoichiometric Coefficients"
For i = 9 To Sheets("int_data").Range("no_comp") + 8
Cells(last_row + 5, i - 7) = Sheets("stream table").Cells(i, 1)
Next i

last_col = Cells(last_row + 5, Columns.Count).End(xlToLeft).Column

If Sheets("int_data").Range("no_rxn") > 0 Then
For j = 1 To Sheets("int_data").Range("no_rxn")
    If Cells(last_row + 6, 1) = "" Then
        Cells(last_row + 6, 1) = "R" & j
    Else
        next_row_r = Sheets("SustainPro").Range("a" &
Rows.Count).End(xlUp).Offset(1, 0).row
        Cells(next_row_r, 1) = "R" & j
    End If
Next j

```

```

last_end_row = Range("a" & Rows.Count).End(xlUp).row
Range(Cells(last_row + 5, 1), Cells(last_end_row, last_col)).Select
Call Module1.create_grid

Range(Cells(last_row + 4, 2), Cells(last_row + 4, last_col)).Select
Call Module1.merge_highlight

    Set to_find = [a:a].find(What:="Reaction\Stoichiometric Coefficients",
LookIn:=xlValues, _
    LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
    MatchCase:=False, SearchFormat:=False)
    If Not to_find Is Nothing Then rxn_row = to_find.Offset(1, 0).row

last_rxn_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_rxn_row + 2, 2) = "Units"
Cells(last_rxn_row + 3, 1) = "Reaction/Reactor"
    n = rxn_row
    Do
        last_rxn_no_row = Range("a" & Rows.Count).End(xlUp).Offset(1, 0).row
        Cells(last_rxn_no_row, 1) = Cells(n, 1)
        Cells(last_rxn_no_row + 1, 1) = "Base component"
        n = n + 1
    Loop Until Cells(n, 1) = ""

For r = 2 To Cells(13, Columns.Count).End(xlToLeft).Column
If Cells(13, r) = "Reactor" Then
    last_r_col = Cells(last_rxn_row + 3, Columns.Count).End(xlToLeft).Column
    Cells(last_rxn_row + 3, last_r_col + 1) = Cells(12, r)
End If
Next r

    last_react_col = Cells(last_rxn_row + 3, Columns.Count).End(xlToLeft).Column
    Range(Cells(last_rxn_row + 3, 1), Cells(Range("a" &
Rows.Count).End(xlUp).row, last_react_col)).Select
    Call Module1.create_grid
    Range(Cells(last_rxn_row + 2, 2), Cells(last_rxn_row + 2, last_react_col)).Select
    Call Module1.merge_highlight

End If

End Sub

Sub demad_stream_price()

last_row = Range("a" & Rows.Count).End(xlUp).row

Cells(last_row + 2, 1) = "Demand stream price"

```

```

With Cells(last_row + 2, 1).Font
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = 0
End With
With Cells(last_row + 2, 1).Interior
    .Pattern = xlSolid
    .PatternColorIndex = xlAutomatic
    .Color = 6299648
    .TintAndShade = 0
    .PatternTintAndShade = 0
End With
Cells(last_row + 4, 2) = "Demand streams price (US $/kg)"

For i = 6 To Sheets("int_data").Range("no_stream") + 5
If Sheets("stream table").Cells(7, i) = 0 Then
last_col_s = Sheets("SustainPro").Cells(last_row + 5,
Columns.Count).End(xlToLeft).Column
Cells(last_row + 5, last_col_s + 1) = Sheets("stream table").Cells(5, i)
End If
Next i

last_end_col = Sheets("SustainPro").Cells(last_row + 5,
Columns.Count).End(xlToLeft).Column

Range(Cells(last_row + 4, 2), Cells(last_row + 4, last_end_col)).Select
Call Module1.merge_highlight
Range(Cells(last_row + 5, 2), Cells(last_row + 5, last_end_col)).Select
Call Module1.create_grid

Cells(last_row + 6, 1) = "Type"
For i = 9 To Sheets("int_data").Range("no_comp") + 8
next_row = Sheets("SustainPro").Range("a" & Rows.Count).End(xlUp).row
Cells(next_row + 1, 1) = Sheets("stream table").Cells(i, 1)
Next i
last_end_row = Sheets("SustainPro").Range("a" & Rows.Count).End(xlUp).row
Range(Cells(last_row + 6, 1), Cells(last_end_row, last_end_col)).Select
Call Module1.create_grid

Set to_find = [a:a].find(What:="Demand stream price", LookIn:=xlValues, _
LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
MatchCase:=False, SearchFormat:=False)
If Not to_find Is Nothing Then demand_row = to_find.row

Set rng = Cells(demand_row + 2, 1)
With ActiveSheet.OLEObjects("CommandButton2")

```

```

        .Top = rng.Top
        .Left = rng.Left
        .Width = rng.Width
        .Height = 28
    End With

```

```
End Sub
```

```
Sub hc()
```

```

    Set to_find = [a:a].find(What:="Demand stream price", LookIn:=xlValues, _
    LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
    MatchCase:=False, SearchFormat:=False)
    If Not to_find Is Nothing Then demand_row = to_find.row

```

```

    first_del_row = Cells(demand_row + 4, 1).End(xlDown).Offset(1, 0).row
    Range(Cells(first_del_row, 1), Cells(5000, 5000)).Delete

```

```
last_row = Range("a" & Rows.Count).End(xlUp).row
```

```
Cells(last_row + 2, 1) = "Heat of combustion"
```

```

    With Cells(last_row + 2, 1).Font
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With

```

```

    With Cells(last_row + 2, 1).Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .Color = 6299648
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With

```

```
Cells(last_row + 4, 2) = "Components"
```

```
Cells(last_row + 6, 1) = "Heat of combustion (kJ/kg)"
```

```
For i = 9 To Sheets("int_data").Range("no_comp") + 8
```

```
Cells(last_row + 5, i - 7) = Sheets("stream table").Cells(i, 1)
```

```
Next i
```

```

Range(Cells(last_row + 4, 2), Cells(last_row + 4,
Sheets("int_data").Range("no_comp") + 1)).Select
Call Module1.merge_highlight

```

```

Range(Cells(last_row + 5, 2), Cells(last_row + 6,
Sheets("int_data").Range("no_comp") + 1)).Select

```

```
Call Module1.create_grid
```

```
Cells(last_row + 6, 1).Select
```

```
Call Module1.create_grid
```

```
'-----
```

```

last_row2 = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row2 + 2, 1) = "Fuel credit streams"

With Cells(last_row2 + 2, 1).Font
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = 0
End With
With Cells(last_row2 + 2, 1).Interior
    .Pattern = xlSolid
    .PatternColorIndex = xlAutomatic
    .Color = 6299648
    .TintAndShade = 0
    .PatternTintAndShade = 0
End With

For l = 2 To Cells(demand_row + 3, Columns.Count).End(xlToLeft).Column
    If Cells(demand_row + 4, l) = "Fuel" Then
        Cells(last_row2 + 4, 2) = "Fuel credit streams"
        Cells(last_row2 + 6, 1) = "Stream price"
        Cells(last_row2 + 7, 1) = "Efficiency"
        Cells(last_row2 + 8, 1) = "Hvap (kJ/kg)"
        For m = 2 To Cells(demand_row + 3, Columns.Count).End(xlToLeft).Column
            If Cells(demand_row + 4, m) = "Fuel" Then
                Cells(last_row2 + 5, Columns.Count).End(xlToLeft).Offset(0, 1) =
Cells(demand_row + 3, m)
            End If
        Next m
        Range(Cells(last_row2 + 5, 2), Cells(last_row2 + 5, Cells(last_row2 + 5,
Columns.Count).End(xlToLeft).Column)).Select
        Call Module1.create_grid
        Range(Cells(last_row2 + 6, 1), Cells(last_row2 + 8, Cells(last_row2 + 5,
Columns.Count).End(xlToLeft).Column)).Select
        Call Module1.create_grid
        Range(Cells(last_row2 + 4, 2), Cells(last_row2 + 4, Cells(last_row2 + 5,
Columns.Count).End(xlToLeft).Column)).Select
        Call Module1.merge_highlight

    Exit For
End If
Next l

End Sub

Private Sub CommandButton2_Click()
Call hc
Call feed_st_price

```

```

Call u_price
Call u_cost
Call safty

If Sheets("int_data").Range("process_type") = "Batch" Then Call for_ap

End Sub

Sub feed_st_price()
last_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 1) = "Feed stream price"
Cells(last_row + 6, 1) = "Price ($/kg)"

    With Cells(last_row + 2, 1).Font
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With
    With Cells(last_row + 2, 1).Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .Color = 6299648
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With

Cells(last_row + 4, 2) = "Feed stream"
    For i = 9 To Sheets("int_data").Range("no_comp") + 8
        Cells(last_row + 5, i - 7) = Sheets("stream table").Cells(i, 1)
        Cells(last_row + 6, i - 7) = Sheets("stream table").Cells(i, 5)
    Next i
last_use_col = Cells(last_row + 5, Columns.Count).End(xlToLeft).Column
last_use_row = Range("a" & Rows.Count).End(xlUp).row

Range(Cells(last_row + 4, 2), Cells(last_row + 4, last_use_col)).Select
Call Module1.merge_highlight
Range(Cells(last_row + 5, 2), Cells(last_row + 5, last_use_col)).Select
Call Module1.create_grid
Range(Cells(last_row + 6, 1), Cells(last_row + 5, last_use_col)).Select
Call Module1.create_grid
End Sub

Sub u_price()
last_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 1) = "Utility price"

    With Cells(last_row + 2, 1).Font
        .ThemeColor = xlThemeColorDark1

```

```

        .TintAndShade = 0
    End With
    With Cells(last_row + 2, 1).Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .Color = 6299648
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With

    Cells(last_row + 4, 1) = "Utilities"
    Cells(last_row + 4, 2) = "Price"
    Cells(last_row + 4, 2).Font.Bold = True

    Cells(last_row + 5, 1) = "Heating (HP) ($/GJ)"
    Cells(last_row + 6, 1) = "Heating (LP) ($/GJ)"
    Cells(last_row + 7, 1) = "Cooling ($/GJ)"
    Cells(last_row + 8, 1) = "Electricity ($/GJ)"

    Range(Cells(last_row + 4, 1), Cells(last_row + 8, 1)).Select
    Selection.Font.Bold = True
    Range(Cells(last_row + 4, 1), Cells(last_row + 8, 2)).Select
    Call Module1.create_grid

End Sub

Sub u_cost()
last_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 1) = "Equipments"
    With Cells(last_row + 2, 1).Font
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With
    With Cells(last_row + 2, 1).Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .Color = 6299648
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With
    Cells(last_row + 4, 2) = "Units"
    Cells(last_row + 6, 1) = "Heating duty (HP) (GJ/hr)"
    Cells(last_row + 7, 1) = "Heating duty (LP) (GJ/hr)"
    Cells(last_row + 8, 1) = "Cooling duty (GJ/hr)"
    Cells(last_row + 9, 1) = "Electrical work (kW)"
    Cells(last_row + 10, 1) = "Properties"

```

```

    For n = 2 To Sheets("equipment table").Cells(8,
Columns.Count).End(xlToLeft).Column
    If Sheets("equipment table").Cells(9, n) <> "Column" Then
    next_col = Sheets("sustainpro").Cells(last_row + 5,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
    For p = 10 To 13
    If Sheets("equipment table").Cells(p, n) <> "" And Sheets("equipment
table").Cells(p, n) <> "-" And Sheets("equipment table").Cells(p, n) >= 0 Then
    e_row = p
    add_row = p - 4
    Sheets("sustainpro").Cells(last_row + 5, next_col) = Sheets("equipment
table").Cells(8, n) 'name
    Sheets("sustainpro").Cells(last_row + add_row, next_col) =
Sheets("equipment table").Cells(e_row, n)
    Sheets("sustainpro").Cells(last_row + 10, next_col) = "Cp (KJ/kg.C)"
    Sheets("sustainpro").Cells(last_row + 10, next_col).Font.Bold = True
    End If
    Next p
    End If
    Next n

```

```

For k = 9 To Sheets("int_data").Range("no_comp") + 8
Cells(last_row + k + 2, 1) = Sheets("stream table").Cells(k, 1)
Next k

```

```

last_use_row = Range("a" & Rows.Count).End(xlUp).row
last_use_col = Cells(last_row + 5, Columns.Count).End(xlToLeft).Column
Range(Cells(last_row + 5, 2), Cells(last_use_row, last_use_col)).Select
Call Module1.create_grid
Range(Cells(last_row + 6, 1), Cells(last_use_row, 1)).Select
Selection.Font.Bold = True
Call Module1.create_grid

```

```

Range(Cells(last_row + 4, 2), Cells(last_row + 4, Cells(last_row + 5,
Columns.Count).End(xlToLeft).Column)).Select
Call Module1.merge_highlight

```

```

'-----
last_row2 = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row2 + 4, 1) = "Distillation column"
Cells(last_row2 + 5, 1) = "Condenser"
Cells(last_row2 + 6, 1) = "Streams"
Cells(last_row2 + 7, 1) = "Hv (kJ/kg)"
'----
col_hv_row1 = last_row2 + 7
col_name_row = last_row2 + 4 'name conden row= last+row2+5

```



```

For i = 9 To Sheets("int_data").Range("no_comp") + 8
last_row_com1 = Range("a" & Rows.Count).End(xlUp).Offset(1, 0).row
Cells(last_row_com1, 1) = Sheets("stream table").Cells(i, 1)
Next i

last_row3 = Range("a" & Rows.Count).End(xlUp).row

Cells(last_row3 + 2, 1) = "Reboiler (HP)"
Cells(last_row3 + 3, 1) = "Reboiler (LP)"
Cells(last_row3 + 4, 1) = "Streams"
Cells(last_row3 + 5, 1) = "Hv (kJ/kg)"
For i = 9 To Sheets("int_data").Range("no_comp") + 8
Cells(last_row3 + i - 3, 1) = Sheets("stream table").Cells(i, 1)
Next i
col_hv_row2 = last_row3 + 5
col_re_hp = last_row3 + 2
col_re_lp = last_row3 + 3

    For i = 2 To Sheets("int_data").Range("no_unit") + 1
        If Sheets("equipment table").Cells(9, i) = "Column" Then
            next_col1 = Cells(col_name_row, Columns.Count).End(xlToLeft).Offset(0,
1).Column
            Sheets("sustainpro").Cells(col_name_row, next_col1) = Sheets("equipment
table").Cells(8, i)
            Sheets("sustainpro").Cells(col_name_row, next_col1).Font.Bold = True
            For l = 10 To 13
                If Sheets("equipment table").Cells(l, i) <> "" And Sheets("equipment
table").Cells(l, i) <> "-" And Sheets("equipment table").Cells(l, i) <> 0 Then
                    If Sheets("equipment table").Cells(l, 1) = "Heating duty (HP) (GJ/hr)"
Then
                        Sheets("sustainpro").Cells(col_re_hp, next_col1) =
Sheets("equipment table").Cells(l, i)
                    ElseIf Sheets("equipment table").Cells(l, 1) = "Heating duty (LP)
(GJ/hr)" Then
                        Sheets("sustainpro").Cells(col_re_lp, next_col1) = Sheets("equipment
table").Cells(l, i)
                    ElseIf Sheets("equipment table").Cells(l, 1) = "Cooling duty (GJ/hr)"
Then
                        Sheets("sustainpro").Cells(last_row2 + 5, next_col1) =
Sheets("equipment table").Cells(l, i)
                    ElseIf Sheets("equipment table").Cells(l, 1) = "Electrical work (kW)"
Then
                        Sheets("sustainpro").Cells(last_row2 + 5, next_col1) =
Sheets("equipment table").Cells(l, i)

```

```

        End If
    End If
Next I
End If
Next i

```

```

last_use_col_col = Cells(col_name_row, Columns.Count).End(xlToLeft).Column
last_use_col_row = Range("a" & Rows.Count).End(xlUp).row
Range(Cells(col_name_row, 1), Cells(last_use_col_row, 1)).Select
Selection.Font.Bold = True
Range(Cells(col_name_row, 1), Cells(last_use_col_row, last_use_col_col)).Select
Call Module1.create_grid
Range(Cells(col_hv_row1, 1), Cells(col_hv_row1, last_use_col_col)).Select
Selection.Merge
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = True
    End With
Range(Cells(col_hv_row2, 1), Cells(col_hv_row2, last_use_col_col)).Select
Selection.Merge
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = True
    End With

```

```
End Sub
```

```

Sub safty()
last_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 1) = "Safety"

```

```

    With Cells(last_row + 2, 1).Font

```

```

.ThemeColor = xlThemeColorDark1
.TintAndShade = 0
End With
With Cells(last_row + 2, 1).Interior
.Pattern = xlSolid
.PatternColorIndex = xlAutomatic
.Color = 6299648
.TintAndShade = 0
.PatternTintAndShade = 0
End With

ActiveSheet.Cells(last_row + 4, 1) = "Components"
ActiveSheet.Cells(last_row + 4, 2) = "Flash Point(°C)"
ActiveSheet.Cells(last_row + 4, 3) = "Boiling Point(°C)"
ActiveSheet.Cells(last_row + 4, 4) = "UEL(%vol)"
ActiveSheet.Cells(last_row + 4, 5) = "LEL(%vol)"
ActiveSheet.Cells(last_row + 4, 6) = "Toxic limit(ppm)"

For i = 9 To Sheets("int_data").Range("no_comp") + 8
next_row = Range("a" & Rows.Count).End(xlUp).Offset(1, 0).row
Cells(next_row, 1) = Sheets("stream table").Cells(i, 1)
Next i
last_collect_row = Range("a" & Rows.Count).End(xlUp).row
Range(Cells(last_row + 4, 1), Cells(last_collect_row, 6)).Select
Call Module1.create_grid

```

```
End Sub
```

```

Private Sub CommandButton4_Click()
Call gentable_sustainpro_sub1
Call sustainpro_data
Sheets("Start menu").CommandButton7.BackColor = &HC00000
Sheets("Start menu").CommandButton15.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton23.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").Select
End Sub

```

Worksheet "Stream table"

```

Private Sub CommandButton1_Click()
If IsNumeric([b2].Value) And IsNumeric([b3].Value) Then
Range(Cells(5, 1), Cells(5000, 5000)).Delete
Sheets("int_data").Range("no_comp") = [b3].Value
Sheets("int_data").Range("no_stream") = [b2].Value
num_compo = Sheets("int_data").Range("no_comp")

```

```

num_st = Sheets("int_data").Range("no_stream")
[a5] = "Stream name"
[a6] = "Initial Unit"
[a7] = "Final Unit"
[a8] = "Component name"
For n = 5 To 7
Range(Cells(n, 1), Cells(n, 5)).Select
Call Module1.merge_highlight_color
Next n

[b8] = "Molecular weight (g/mol)"
[c8] = "Hazardous (Yes)"
[d8] = "Type (RM/P/BP/I/S)"
[e8] = "Price ($/kg)"
Range(Cells(8, 1), Cells(8, 5)).Select
Call Module1.bold_highlight_color

Cells(num_compo + 9, 1) = "Total Mass Flow (kg/hr)"
Cells(num_compo + 10, 1) = "Temperature (degrees Celsius)"
Cells(num_compo + 11, 1) = "Pressure (kPa)"
Cells(num_compo + 12, 1) = "Enthalpy Flow (GJ/hr)"
If Sheets("int_data").Range("process_type") = "Batch" Then
Cells(num_compo + 13, 1) = "Time Dependence"
End If

m = num_compo + 9
Do
Range(Cells(m, 1), Cells(m, 5)).Select
Call Module1.merge_highlight_color
m = m + 1
Loop Until Cells(m, 1) = ""

last_use_row = Range("a" & Rows.Count).End(xlUp).row

Range(Cells(5, 6), Cells(last_use_row, num_st + 5)).Select
Call Module1.create_grid

Range(Cells(9, 1), Cells(num_compo + 8, 5)).Select
Call Module1.create_grid

Else
MsgBox "Please enter number of streams and components (numeric value)"
End If

End Sub
Sub add_model()

```

```

Range(Cells(8, 1), Cells(2000, 2000)).Delete
If [b2] <> "" Then
[a8] = "Equipment name"
[b8] = "Type"
[c8] = "Activity"
[d8] = "Amount"
[e8] = "Unit"
Range(Cells(8, 1), Cells(8, 5)).Select
Selection.Font.Bold = True
  If [b3] <> "" Then
    total_unit = [b2].Value + ([b3].Value * 2)
  Else
    total_unit = [b2].Value
  End If

Range(Cells(8, 1), Cells(total_unit + 8, 5)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
  End With
  With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
  End With
  With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
  End With
  With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
  End With

```

```

End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With

For i = 2 To [b2].Value + 1
next_row = Range("a" & Rows.Count).End(xlUp).Offset(1, 0).row
Cells(next_row, 1) = Cells(5, i)
Next i

If [b3].Value <> "" Then
For i = 2 To [b3].Value + 1
next_row = Range("a" & Rows.Count).End(xlUp).Offset(1, 0).row
Cells(next_row, 1) = Cells(6, i) & "-condenser"
next_row = Range("a" & Rows.Count).End(xlUp).Offset(1, 0).row
Cells(next_row, 1) = Cells(6, i) & "-reboiler"
Next i
End If

For i = 9 To total_unit + 8
Cells(i, 2).Select
    With ActiveCell
        .HorizontalAlignment = xlCenter
        .Font.ColorIndex = xlAutomatic
        .Font.TintAndShade = 0
        .Interior.Pattern = xlSolid
        .Interior.PatternColorIndex = xlAutomatic
        .Interior.ThemeColor = xlThemeColorDark1
        .Interior.TintAndShade = -0.249977111117893
        .Interior.PatternTintAndShade = 0
        .Value = "(double click here)"
    End With
Next i

Else
MsgBox "Please enter the number of equipment"
End If

End Sub

Private Sub CommandButton2_Click()
Sheets("Start menu").Select
Sheets("Start menu").CommandButton22.BackColor = &HC00000

```

End Sub

Worksheet "Equipment table"

```
Private Sub CommandButton1_Click()
Sheets("Start menu").Select
Sheets("Start menu").CommandButton21.BackColor = &HC00000
Sheets("Start menu").CommandButton7.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton15.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton23.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton17.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton18.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton24.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton19.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton20.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton25.BackColor =
Sheets("int_data").Range("not")
End Sub
```

```
Private Sub CommandButton2_Click()
Range(Cells(7, 1), Cells(2000, 2000)).Delete
If IsNumeric([b2]) Then
total_unit = [b2].Value
Sheets("int_data").Range("no_unit") = total_unit
Cells(8, 1) = "Equipment name"
Cells(9, 1) = "Type"
Cells(10, 1) = "Heating duty (HP) (GJ/hr)"
Cells(11, 1) = "Heating duty (LP) (GJ/hr)"
Cells(12, 1) = "Cooling duty (GJ/hr)"
Cells(13, 1) = "Electrical work (kW)"
```

```
Range(Cells(8, 1), Cells(13, 1)).Select
Selection.Font.Bold = True
```

```
    If Sheets("int_data").Range("process_type") = "Batch" Then
        Cells(14, 1) = "Time (s)"
        Cells(15, 1) = "Volume (m3)"
        Range(Cells(8, 1), Cells(15, 1)).Select
        Selection.Font.Bold = True
        Range(Cells(8, 1), Cells(15, total_unit + 1)).Select
```

```

    Else
        Range(Cells(8, 1), Cells(13, total_unit + 1)).Select
    End If
    Call Module1.create_grid

    Else
        MsgBox "Please enter the number of equipments and name of each equipment"
    End If

End Sub

Function InRange(Range1 As Range, Range2 As Range) As Boolean
' returns True if Range1 is within Range2
Dim InterSectRange As Range
    Set InterSectRange = Application.Intersect(Range1, Range2)
    InRange = Not InterSectRange Is Nothing
    Set InterSectRange = Nothing
End Function

Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
If [b2] <> "" And IsNumeric([b2]) Then

    lastcol = [b2].Value + 1
    If InRange(ActiveCell, Range(Cells(10, 2), Cells(10, lastcol))) Then

        'form
        UserForm1.Show

    Else
        MsgBox "Out of range"
    End If

End Sub

Module1
Sub merge_highlight()
Selection.Font.Bold = True
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False

```



```

        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone
    Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
End Sub

Sub create_grid()
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlThin
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
    End With

```

```
.Weight = xlThin
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
End Sub

Sub merge_highlight_color()
Selection.Font.Bold = True
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlMedium
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlMedium
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
```

```

        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone
    Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = -0.149998474074526
        .PatternTintAndShade = 0
    End With
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
End Sub

Sub bold_highlight_color()
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlThin
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlThin
    End With

```

```
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Interior
    .PatternColorIndex = xlAutomatic
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = -0.149998474074526
    .PatternTintAndShade = 0
End With
Selection.Font.Bold = True
End Sub
```

```
Sub clear_table()
    With Selection
        .HorizontalAlignment = xlGeneral
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = True
    End With
    Selection.UnMerge
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
Selection.Borders(xlEdgeLeft).LineStyle = xlNone
Selection.Borders(xlEdgeTop).LineStyle = xlNone
Selection.Borders(xlEdgeBottom).LineStyle = xlNone
Selection.Borders(xlEdgeRight).LineStyle = xlNone
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
Selection.ClearContents
```

```
End Sub
```

```
Sub changecolor_A_cell_bold()
```

```
With Selection.Interior
```

```
    .Pattern = xlSolid
```

```
    .PatternColorIndex = xlAutomatic
```

```
    .Color = 6299648
```

```
    .TintAndShade = 0
```

```
    .PatternTintAndShade = 0
```

```
End With
```

```
With Selection.Font
```

```
    .ThemeColor = xlThemeColorDark1
```

```
    .TintAndShade = 0
```

```
End With
```

```
Selection.Font.Bold = True
```

```
End Sub
```

```
Sub wrap_t()
```

```
With Selection
```

```
    .HorizontalAlignment = xlGeneral
```

```
    .VerticalAlignment = xlBottom
```

```
    .WrapText = True
```

```
    .Orientation = 0
```

```
    .AddIndent = False
```

```
    .IndentLevel = 0
```

```
    .ShrinkToFit = False
```

```
    .ReadingOrder = xlContext
```

```
    .MergeCells = False
```

```
End With
```

```
End Sub
```

```
Sub econ_cut()
```

```
With Selection.Font
```

```
    .Name = "Calibri"
```

```
    .Size = 24
```

```
    .Strikethrough = False
```

```
    .Superscript = False
```

```
    .Subscript = False
```

```

.OutlineFont = False
.Shadow = False
.Underline = xlUnderlineStyleNone
.ThemeColor = xlThemeColorLight1
.TintAndShade = 0
.ThemeFont = xlThemeFontMinor
End With
Selection.Font.Bold = True
With Selection.Font
.ThemeColor = xlThemeColorDark1
.TintAndShade = 0
End With
With Selection.Interior
.Pattern = xlSolid
.PatternColorIndex = xlAutomatic
.ThemeColor = xlThemeColorLight1
.TintAndShade = 4.99893185216834E-02
.PatternTintAndShade = 0
End With
End Sub

Sub lcsoft_head()

Selection.Font.Bold = True
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.ColorIndex = 0
.TintAndShade = 0
.Weight = xlThin
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.ColorIndex = 0
.TintAndShade = 0
.Weight = xlThin
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.ColorIndex = 0
.TintAndShade = 0
.Weight = xlThin
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.ColorIndex = 0

```

```

        .TintAndShade = 0
        .Weight = xlThin
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone
    Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = -0.149998474074526
        .PatternTintAndShade = 0
    End With
    With Selection.Font
        .Name = "Calibri"
        .Size = 20
        .Strikethrough = False
        .Superscript = False
        .Subscript = False
        .OutlineFont = False
        .Shadow = False
        .Underline = xlUnderlineStyleNone
        .ThemeColor = xlThemeColorLight1
        .TintAndShade = 0
        .ThemeFont = xlThemeFontMinor
    End With
End Sub

Sub lcsoft_section_head()
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorLight2
        .TintAndShade = -0.499984740745262
        .PatternTintAndShade = 0
    End With
    With Selection.Font
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlThin
    End With

```

```

With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .ColorIndex = 0
    .TintAndShade = 0
    .Weight = xlThin
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
End Sub

```

```

Sub lcsoft_entire_grid()
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlThin
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlThin
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlThin
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .ColorIndex = 0
    End With

```



```

        .TintAndShade = 0
        .Weight = xlThin
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone
    Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
End Sub

```

```

Sub lc_input_st_grid()
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorAccent5
        .TintAndShade = 0.599993896298105
        .PatternTintAndShade = 0
    End With
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone
    Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
End Sub
Sub lc_input_eq()
    With Selection.Interior

```

```

        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorAccent5
        .TintAndShade = -0.249977111117893
        .PatternTintAndShade = 0
    End With
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone
    Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
End Sub
Sub lc_op_st()
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .Color = 13434726
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous

```

```

        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .ColorIndex = 0
        .TintAndShade = 0
        .Weight = xlMedium
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone
    Selection.Borders(xlInsideHorizontal).LineStyle = xlNone
End Sub

```

Module "IO_Interface"

```

Sub check_pj_name()
    Application.ScreenUpdating = False
    wb_path = ThisWorkbook.Path
    If Right(wb_path, 15) = "Project Library" Then
        directory = wb_path & "\"
    Else
        directory = wb_path & "\Project Library\"
    End If
    Sheets("int_data").[x:x].ClearContents

    Set fso = CreateObject("Scripting.FileSystemObject")
    Set folder = fso.GetFolder(directory)
    For Each file In folder.Files
        file_name = file.Name
        If Left(file_name, 2) <> "~$" Then
            i = Sheets("int_data").Range("x" & Rows.Count).End(xlUp).Offset(1, 0).row
            Sheets("int_data").Cells(i, 24) = file_name
        End If
    Next file

```

End Sub

```

Sub reset_all()
Application.ScreenUpdating = False
Application.DisplayAlerts = False
Sheets("Start menu").Select
Sheets("Start menu").cmb_newpj.BackColor = Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton21.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton22.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton7.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton15.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton17.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton18.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton19.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton20.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton23.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton24.BackColor =
Sheets("int_data").Range("not")
Sheets("Start menu").CommandButton25.BackColor =
Sheets("int_data").Range("not")
'equipment table
Sheets("equipment table").Select
[b2] = ""
Range(Cells(8, 1), Cells(20, 1)).EntireRow.Delete

'stream table
Sheets("stream table").Select
[b2] = ""
[b3] = ""
Range(Cells(5, 1), Cells(3000, 3000)).Delete

'sustainpro

```

```

Sheets("sustainpro").Select
[b2].Clear
[b3].Clear
[b4].Clear
Range(Cells(9, 1), Cells(5000, 5000)).Delete

'econ
Sheets("econ").Select
Range(Cells(8, 1), Cells(5000, 5000)).Delete

'lcsoft
Sheets("lcsoft").Select
Range(Cells(2, 1), Cells(5000, 5000)).Delete
Sheets("Start menu").Select
End Sub

Sub newpj_ready()
Sheets("Start menu").Select
ActiveSheet.cmb_newpj.BackColor = &HC00000

End Sub

```

Module "I1_SustainPro"

```

Sub gentable_sustainpro_sub1()
Application.ScreenUpdating = False
Application.DisplayAlerts = False
Sheets("sustainpro_data").Visible = True
Sheets("sustainpro_data").Select
Range(Cells(7, 1), Cells(7000, 7000)).Delete
blocks = ActiveSheet.Range("C1")
streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H1")
reactions = ActiveSheet.Range("H3")

'MVA Information

'Cycle to insert the name of reactions in the table

    b = 1
    Do While b <= reactions

        ActiveSheet.Cells(34 + components + b, 1) = "R " & b
        b = b + 1

    Loop

'Cycle to insert the name of components in the table

```

```

a = 0
Do While a <= components - 1

    ActiveSheet.Cells(34 + components, 2 + a) = ActiveSheet.Cells(26 + a, 1)
    a = a + 1

Loop

'Create a table

ActiveSheet.Range(Cells(33 + components, 2), Cells(33 + components, 1 +
components)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Range(Cells(34 + components, 1), Cells(34 + components + reactions, 1 +
components)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic

```

```

End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
ActiveSheet.Range(Cells(33 + components, 2), Cells(33 + components, 1 +
components)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Cells(33 + components, 2) = "Components"
ActiveSheet.Cells(31 + components, 1) = "Reactions"

```

```

ActiveSheet.Cells(31 + components, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False

```

```

        .OutlineFont = False
        .Shadow = False
        .Underline = xlUnderlineStyleNone
        .ColorIndex = xlAutomatic
        .TintAndShade = 0
        .ThemeFont = xlThemeFontNone
    End With

'Demand Prices

ActiveSheet.Cells(36 + components + reactions, 1) = "Price"

ActiveSheet.Cells(36 + components + reactions, 1).Select
    Selection.Font.Bold = True
    With Selection.Font
        .Name = "Arial"
        .Size = 12
        .Strikethrough = False
        .Superscript = False
        .Subscript = False
        .OutlineFont = False
        .Shadow = False
        .Underline = xlUnderlineStyleNone
        .ColorIndex = xlAutomatic
        .TintAndShade = 0
        .ThemeFont = xlThemeFontNone
    End With
ActiveSheet.Cells(40 + components + reactions, 1) = "Type"
ActiveSheet.Cells(38 + components + reactions, 2) = "Demand streams Price ($/kg)"

'Code to insert the names of the demand streams

a = 0
b = 0

Do While a <= streams - 1

    If ActiveSheet.Cells(24, 2 + a) = 0 Then

        ActiveSheet.Cells(39 + components + reactions, 2 + b) = ActiveSheet.Cells(22,
2 + a)
        b = b + 1

    End If

```



```
a = a + 1
```

Loop

```
'Cycle to insert the name of components in the table
```

```
  c = 0
```

```
  Do While c <= components - 1
```

```
    ActiveSheet.Cells(41 + components + reactions + c, 1) = ActiveSheet.Cells(26
+ c, 1)
```

```
    c = c + 1
```

Loop

```
'Insert table
```

```
ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(38 + components +
reactions, 1 + b)).Select
```

```
  With Selection
```

```
    .HorizontalAlignment = xlCenter
```

```
    .VerticalAlignment = xlBottom
```

```
    .WrapText = False
```

```
    .Orientation = 0
```

```
    .AddIndent = False
```

```
    .IndentLevel = 0
```

```
    .ShrinkToFit = False
```

```
    .ReadingOrder = xlContext
```

```
    .MergeCells = False
```

```
  End With
```

```
  Selection.Merge
```

```
ActiveSheet.Range(Cells(39 + components + reactions, 1), Cells(40 + components +
components + reactions, 1 + b)).Select
```

```
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
  With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
  End With
```

```
  With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
  End With
```

```
  With Selection.Borders(xlEdgeBottom)
```

```

        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(38 + components
+ reactions, 1 + b)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone

```

```

ActiveSheet.Range(Cells(39 + components + reactions, 2), Cells(40 + components
+ components + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(40 + components + reactions, 1), Cells(40 + components
+ components + reactions, 1)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous

```

```

        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(39 + components
+ reactions, 1 + b)).Select
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    ActiveSheet.Range(Cells(38 + components + reactions, 1), Cells(39 + components
+ components + reactions, 1 + b)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
    End With

'FEED PRICES

'Cycle to insert the name of components in the table
a = 0
Do While a <= components - 1

    ActiveSheet.Cells(44 + components * 2 + reactions, 2 + a) =
ActiveSheet.Cells(26 + a, 1)
    a = a + 1

Loop

ActiveSheet.Cells(43 + components * 2 + reactions, 2) = "Feed streams"

```

```
ActiveSheet.Cells(45 + components * 2 + reactions, 1) = "Price ($/kg)"
```

```
'Insert table
```

```
ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(43 + components * 2 + reactions, 1 + components)).Select
```

```
With Selection
```

```
.HorizontalAlignment = xlCenter
```

```
.VerticalAlignment = xlBottom
```

```
.WrapText = False
```

```
.Orientation = 0
```

```
.AddIndent = False
```

```
.IndentLevel = 0
```

```
.ShrinkToFit = False
```

```
.ReadingOrder = xlContext
```

```
.MergeCells = False
```

```
End With
```

```
Selection.Merge
```

```
ActiveSheet.Range(Cells(44 + components * 2 + reactions, 1), Cells(45 + components * 2 + reactions, 1 + components)).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideVertical)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```

End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(43 +
components * 2 + reactions, 1 + components)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone

```

```

ActiveSheet.Range(Cells(44 + components * 2 + reactions, 2), Cells(45 +
components * 2 + reactions, 1 + components)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideHorizontal)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
ActiveSheet.Cells(45 + components * 2 + reactions, 1).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
Selection.Font.Bold = True
```

```
Selection.Font.ColorIndex = 11
```

```
ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(44 +  
components * 2 + reactions, 1 + components)).Select
```

```
Selection.Font.Bold = True
```

```
Selection.Font.ColorIndex = 11
```

```

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 1), Cells(45 +
components * 2 + reactions, 1 + components)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
End With

```

'UTILITY PRICES

```

ActiveSheet.Cells(47 + components * 2 + reactions, 1) = "Utility Price"
ActiveSheet.Cells(47 + components * 2 + reactions, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With

```

```

ActiveSheet.Cells(49 + components * 2 + reactions, 1) = "Utilities"
ActiveSheet.Cells(49 + components * 2 + reactions, 2) = "Price"
ActiveSheet.Cells(50 + components * 2 + reactions, 1) = "Heating(HP)($/GJ)"
ActiveSheet.Cells(51 + components * 2 + reactions, 1) = "Heating(LP)($/GJ)"
ActiveSheet.Cells(52 + components * 2 + reactions, 1) = "Cooling($/GJ)"
ActiveSheet.Cells(53 + components * 2 + reactions, 1) = "Electricity($/kWh)"

```

'Create a table

```

ActiveSheet.Range(Cells(49 + components * 2 + reactions, 1), Cells(53 +
components * 2 + reactions, 2)).Select

Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone

```



```

With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With

```

```

ActiveSheet.Range(Cells(49 + components * 2 + reactions, 1), Cells(49 +
components * 2 + reactions, 2)).Select

```

```

Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

'HEAT EXCHANGED

c = 0

a = 0

b = 0

Do While a <= blocks - 1

If ActiveSheet.Cells(11, 2 + a) = "HE" Or ActiveSheet.Cells(11, 2 + a) =
"Reactor" Or ActiveSheet.Cells(11, 2 + a) = "Filtration" Then

 ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
ActiveSheet.Cells(10, 2 + a)

 ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) = "Cp(kJ/kg-C)"
 b = b + 1

Else

If ActiveSheet.Cells(11, 2 + a) = "Pump" Or ActiveSheet.Cells(11, 2 + a) =
 "Comp" Or ActiveSheet.Cells(11, 2 + a) = "Centrifuge" Or ActiveSheet.Cells(11, 2 +
 a) = "Mixer" Then

ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
 ActiveSheet.Cells(10, 2 + a)
 ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) = "Den(kg/m3)"
 b = b + 1

Else

If ActiveSheet.Cells(11, 2 + a) = "Column" Then

ActiveSheet.Cells(67 + components * 3 + reactions, 2 + c) =
 ActiveSheet.Cells(10, 2 + a)
 c = c + 1

Else

If ActiveSheet.Cells(11, 2 + a) = "Evap" Or ActiveSheet.Cells(11, 2 + a) =
 "Cond" Then

ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
 ActiveSheet.Cells(10, 2 + a)
 ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) =
 "Hvap(kJ/kg)"
 b = b + 1

End If

End If

End If

End If

a = a + 1

Loop

'Insert a table

If b = 0 Then

Else

```
ActiveSheet.Cells(55 + components * 2 + reactions, 1) = "Heats and Component's Properties"
```

```
ActiveSheet.Cells(55 + components * 2 + reactions, 1).Select
```

```
Selection.Font.Bold = True
```

```
With Selection.Font
```

```
.Name = "Arial"
```

```
.Size = 12
```

```
.Strikethrough = False
```

```
.Superscript = False
```

```
.Subscript = False
```

```
.OutlineFont = False
```

```
.Shadow = False
```

```
.Underline = xlUnderlineStyleNone
```

```
.ColorIndex = xlAutomatic
```

```
.TintAndShade = 0
```

```
.ThemeFont = xlThemeFontNone
```

```
End With
```

```
ActiveSheet.Cells(61 + components * 2 + reactions, 1) = "Cooling duty (GJ/hr)"
```

```
ActiveSheet.Cells(59 + components * 2 + reactions, 1) = "Heating duty (HP)(GJ/hr)"
```

```
ActiveSheet.Cells(60 + components * 2 + reactions, 1) = "Heating duty (LP)(GJ/hr)"
```

```
ActiveSheet.Cells(62 + components * 2 + reactions, 1) = "Electrical work (kW)"
```

```
ActiveSheet.Cells(63 + components * 2 + reactions, 1) = "Properties"
```

```
ActiveSheet.Cells(57 + components * 2 + reactions, 2) = "Units"
```

```
'Cycle to insert the name of components in the table
```

```
d = 0
```

```
Do While d <= components - 1
```

```
    ActiveSheet.Cells(64 + components * 2 + reactions + d, 1) =
```

```
    ActiveSheet.Cells(26 + d, 1)
```

```
    d = d + 1
```

```
Loop
```

```
ActiveSheet.Range(Cells(57 + components * 2 + reactions, 2), Cells(57 + components * 2 + reactions, 1 + b)).Select
```

```
With Selection
```

```
.HorizontalAlignment = xlCenter
```

```
.VerticalAlignment = xlBottom
```

```
.WrapText = False
```

```
.Orientation = 0
```

```
.AddIndent = False
```

```
.IndentLevel = 0
```

```
.ShrinkToFit = False
```

```
.ReadingOrder = xlContext
```

```

    .MergeCells = False
End With
Selection.Merge
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic

```

```
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
```

End With

```
ActiveSheet.Range(Cells(58 + components * 2 + reactions, 2), Cells(58 +
components * 2 + reactions, 1 + b)).Select
Selection.Font.ColorIndex = 11
Selection.Font.Bold = True
```

```
ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1)).Select
Selection.Font.ColorIndex = 11
Selection.Font.Bold = True
```

```
ActiveSheet.Range(Cells(63 + components * 2 + reactions, 2), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11
```

```
ActiveSheet.Range(Cells(58 + components * 2 + reactions, 2), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
```

```

.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(58 +
components * 2 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(63 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium

```



```

        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(57 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
    End With

End If

If c > 0 Then

ActiveSheet.Cells(68 + components * 3 + reactions, 1) = "Condenser(GJ/hr)"
ActiveSheet.Cells(72 + components * 4 + reactions, 1) = "Reboiler (HP)(GJ/hr)"
ActiveSheet.Cells(73 + components * 4 + reactions, 1) = "Reboiler (LP)(GJ/hr)"
ActiveSheet.Cells(70 + components * 3 + reactions, 2) = "Hv(kJ/kg)"
ActiveSheet.Cells(75 + components * 4 + reactions, 2) = "Hv(kJ/kg)"
ActiveSheet.Cells(69 + components * 3 + reactions, 1) = "Streams"
ActiveSheet.Cells(74 + components * 4 + reactions, 1) = "Streams"

'Cycle to insert the name of components in the table

    c1 = 0
    Do While c1 <= components - 1

        ActiveSheet.Cells(71 + components * 3 + reactions + c1, 1) =
ActiveSheet.Cells(26 + c1, 1)
        c1 = c1 + 1

    Loop

    c2 = 0
    Do While c2 <= components - 1

```

```

    ActiveSheet.Cells(76 + components * 4 + reactions + c2, 1) =
ActiveSheet.Cells(26 + c2, 1)
    c2 = c2 + 1

```

Loop

'insert table

```

ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(75 +
components * 5 + reactions, 1 + c)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin

```

```

        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```

    ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(67 +
components * 3 + reactions, 1 + c)).Select
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

```

    ActiveSheet.Range(Cells(70 + components * 3 + reactions, 2), Cells(70 +
components * 3 + reactions, 1 + c)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
    End With

```

```

        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    ActiveSheet.Range(Cells(70 + components * 3 + reactions, 1), Cells(70 +
components * 3 + reactions, 1 + c)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(71 + components * 4 + reactions, 1), Cells(71 +
components * 4 + reactions, 1 + c)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium

```

```

    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(75 + components * 4 + reactions, 2), Cells(75 +
components * 4 + reactions, 1 + c)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

ActiveSheet.Range(Cells(75 + components * 4 + reactions, 1), Cells(75 +
components * 4 + reactions, 1 + c)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium

```

```

        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(75 +
components * 5 + reactions, 1)).Select
    Selection.Font.ColorIndex = 11
    Selection.Font.Bold = True

End If

'Reactions information

Dim dp As Integer

dp = ActiveSheet.Cells(1, 14)

ActiveSheet.Cells(77 + components * 5 + reactions, 1) = "Reactions"
ActiveSheet.Cells(77 + components * 5 + reactions, 1).Select
    Selection.Font.Bold = True
    With Selection.Font
        .Name = "Arial"
        .Size = 12
        .Strikethrough = False
        .Superscript = False
        .Subscript = False
        .OutlineFont = False
        .Shadow = False
        .Underline = xlUnderlineStyleNone
        .ColorIndex = xlAutomatic
        .TintAndShade = 0
        .ThemeFont = xlThemeFontNone
    End With

    ActiveSheet.Cells(79 + components * 5 + reactions, 1) = "Product"
    ActiveSheet.Cells(80 + components * 5 + reactions, 1) = "Stream"

```

```

ActiveSheet.Range(Cells(79 + components * 5 + reactions, 1), Cells(80 +
components * 5 + reactions, 1 + dp)).Select
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With

  With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With

  With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
  End With

  ActiveSheet.Range(Cells(79 + components * 5 + reactions, 1), Cells(80 +
components * 5 + reactions, 1)).Select
  Selection.Font.Bold = True
  Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Cells(82 + components * 5 + reactions, 2) = "Units"

'Insert the name of the reactive units and count them

inblo = 0
numreact = 0

Do While inblo <= blocks - 1

    If ActiveSheet.Cells(11, 2 + inblo) = "Reactor" Then

        ActiveSheet.Cells(83 + components * 5 + reactions, 2 + numreact) =
ActiveSheet.Cells(10, 2 + inblo)
        numreact = numreact + 1

    End If

    inblo = inblo + 1

Loop

'Insert the name of the reactions

Dim fini As Boolean

a = 0
a1 = 0
fini = False

Do Until fini = True

    If ActiveSheet.Cells(35 + components + a, 1) = 0 Then

        fini = True

    Else

        ActiveSheet.Cells(84 + components * 5 + reactions + a1, 1) =
ActiveSheet.Cells(35 + components + a, 1)
        ActiveSheet.Cells(85 + components * 5 + reactions + a1, 1) = "Component"
        a = a + 1
        a1 = a1 + 2

    End If

```


Loop

```

ActiveSheet.Range(Cells(83 + components * 5 + reactions, 1), Cells(83 +
components * 5 + reactions * 3, 1 + numreact)).Select
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
  End With

```

```
ActiveSheet.Range(Cells(82 + components * 5 + reactions, 2), Cells(82 +
components * 5 + reactions, 1 + numreact)).Select
```

```
With Selection
```

```
.HorizontalAlignment = xlCenter
```

```
.VerticalAlignment = xlBottom
```

```
.WrapText = False
```

```
.Orientation = 0
```

```
.AddIndent = False
```

```
.IndentLevel = 0
```

```
.ShrinkToFit = False
```

```
.ReadingOrder = xlContext
```

```
.MergeCells = False
```

```
End With
```

```
Selection.Merge
```

```
Selection.Font.Bold = True
```

```
Selection.Font.ColorIndex = 11
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlMedium
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
Selection.Borders(xlInsideVertical).LineStyle = xlNone
```

```
ActiveSheet.Range(Cells(84 + components * 5 + reactions, 1), Cells(84 +
components * 5 + reactions * 3, 1)).Select
```

```
Selection.Font.ColorIndex = 11
```

```
Selection.Font.Bold = True
```

```

ActiveSheet.Range(Cells(83 + components * 5 + reactions, 2), Cells(83 +
components * 5 + reactions, 1 + numreact)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

'Safety

```

ActiveSheet.Cells(87 + components * 5 + reactions * 3, 1) = "Components"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 2) = "Flash Point(°C)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 3) = "Boiling Point(°C)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 4) = "UEL(%vol)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 5) = "LEL(%vol)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 6) = "Toxic limit(ppm)"

```

```

ActiveSheet.Cells(85 + components * 5 + reactions * 3, 1) = "Safety"
ActiveSheet.Cells(85 + components * 5 + reactions * 3, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With

```

Dim fg As Integer

fg = 0

Do While fg <= components - 1

```

    ActiveSheet.Cells(88 + components * 5 + reactions * 3 + fg, 1) =
ActiveSheet.Cells(26 + fg, 1)
    fg = fg + 1

```

Loop

```

ActiveSheet.Range(Cells(87 + components * 5 + reactions * 3, 1), Cells(87 +
components * 5 + reactions * 3, 6)).Select
Selection.Font.ColorIndex = 11

```

```

Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(87 + components * 5 + reactions * 3, 1), Cells(87 +
components * 6 + reactions * 3, 6)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```

With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Sheets("sustainpro_data").Visible = False
End Sub

Sub gentable_sustainpro_sub2()
blocks = ActiveSheet.Range("C1")
streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H1")
reactions = ActiveSheet.Range("H3")

'Streams information

'Make a table with the right size surrounding the streams datas

ActiveSheet.Cells(20, 1) = "Streams"

Range(Cells(22, 1), Cells(20 + 9 + components, 1 + streams)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic

```

```

End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

'Subtract the background from the table.

Selection.Interior.ColorIndex = xlNone

'Make the text in the table all in the same color(black)
Selection.Font.ColorIndex = 0

'Change borders and color of the letters
Range(Cells(22, 1), Cells(20 + 9 + components, 1)).Select

Selection.Font.ColorIndex = 11
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlDouble
    .Weight = xlThick
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)

```

```

        .LineStyle = xlDouble
        .Weight = xlThick
        .ColorIndex = xlAutomatic
    End With

    Range(Cells(22, 1), Cells(22, 1 + streams)).Select

    Selection.Font.ColorIndex = 11
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlDouble
        .Weight = xlThick
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlDouble
        .Weight = xlThick
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlDouble
        .Weight = xlThick
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlDouble
        .Weight = xlThick
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

```

'Insert the right name to describe each row in the table

```

Cells(22, 1) = "Stream Name"
Cells(23, 1) = "Initial Unit"
Cells(24, 1) = "Final Unit"
Cells(25, 1) = "Components (kg/h)"

```

'Cycle to insert the name of components in the table

```

a = 0
Do While a <= components

    Cells(26 + a, 1) = ActiveSheet.Cells(15, 3 + a)
    a = a + 1

Loop

Cells(26 + components, 1) = "Total Mass Flow(kg/h)"
Cells(27 + components, 1) = "Temperature (°C)"
Cells(28 + components, 1) = "Pressure (atm)"
Cells(29 + components, 1) = "Enthalpy Flow (GJ/h)"

```

```
End Sub
```

```
Private Sub cmd_gendata_Click()
ActiveSheet.Select
```

```

blocks = ActiveSheet.Range("C1")
streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H1")
reactions = ActiveSheet.Range("H3")

```

```
'General Data Tables
```

```
Dim a As Integer
```

```
'Warning message when the user didn't fill the values of blocks, or components, or streams.
```

```
If Range("C2") = Empty Or Range("C3") = Empty Or Range("H2") = Empty Then
```

```

    Dim answer As String
    Phrase = "Please insert the general data"
    Title = "Warning"
    answer = MsgBox(Phrase, vbExclamation, Title)

```

```
Else
```

```
'Generate a table with the block size, in order to the user insert the blocks's name and type.
```



```
ActiveSheet.Cells(7, 1) = "General Data"  
ActiveSheet.Cells(10, 1) = "Name"  
ActiveSheet.Cells(11, 1) = "Type"  
ActiveSheet.Cells(9, 2) = "Units"  
ActiveSheet.Cells(12, 1) = "Operation"  
  
Range(Cells(10, 1), Cells(12, 1 + blocks)).Select  
  
Selection.Borders(xlDiagonalDown).LineStyle = xlNone  
Selection.Borders(xlDiagonalUp).LineStyle = xlNone  
With Selection.Borders(xlEdgeLeft)  
    .LineStyle = xlContinuous  
    .Weight = xlThin  
    .ColorIndex = xlAutomatic  
End With  
With Selection.Borders(xlEdgeTop)  
    .LineStyle = xlContinuous  
    .Weight = xlThin  
    .ColorIndex = xlAutomatic  
End With  
With Selection.Borders(xlEdgeBottom)  
    .LineStyle = xlContinuous  
    .Weight = xlThin  
    .ColorIndex = xlAutomatic  
End With  
With Selection.Borders(xlEdgeRight)  
    .LineStyle = xlContinuous  
    .Weight = xlThin  
    .ColorIndex = xlAutomatic  
End With  
With Selection.Borders(xlInsideVertical)  
    .LineStyle = xlContinuous  
    .Weight = xlThin  
    .ColorIndex = xlAutomatic  
End With  
With Selection.Borders(xlInsideHorizontal)  
    .LineStyle = xlContinuous  
    .Weight = xlThin  
    .ColorIndex = xlAutomatic  
End With  
Range(Cells(9, 2), Cells(9, 1 + blocks)).Select  
With Selection  
    .HorizontalAlignment = xlCenter  
    .VerticalAlignment = xlBottom  
    .WrapText = False  
    .Orientation = 0
```

```

.AddIndent = False
.IndentLevel = 0
.ShrinkToFit = False
.ReadingOrder = xlContext
.MergeCells = False
End With
Selection.Merge
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone

```

```

ActiveSheet.Cells(14, 3) = "Components"
ActiveSheet.Cells(15, 1) = "Name"
ActiveSheet.Cells(16, 1) = "Molecular weight (g/mol)"
ActiveSheet.Cells(17, 1) = "Hazardous"
ActiveSheet.Cells(18, 1) = "Type"

```

'Generate a table with the components size, in order to the user insert the components's name and their molecular weight.

```

Range(Cells(15, 1), Cells(18, 2 + components)).Select
Application.CutCopyMode = False
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous

```

```
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
Range(Cells(14, 3), Cells(14, 2 + components)).Select
With Selection
.HorizontalAlignment = xlCenter
.VerticalAlignment = xlBottom
.WrapText = False
.Orientation = 0
.AddIndent = False
.IndentLevel = 0
.ShrinkToFit = False
.ReadingOrder = xlContext
.MergeCells = False
End With
Selection.Merge
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
```

```
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Range(Cells(15, 1), Cells(15, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Range(Cells(16, 1), Cells(16, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Range(Cells(17, 1), Cells(17, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
```

```
.WrapText = False
.Orientation = 0
.AddIndent = False
.IndentLevel = 0
.ShrinkToFit = False
.ReadingOrder = xlContext
.MergeCells = False
End With
Selection.Merge
Range(Cells(18, 1), Cells(18, 2)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge

End If

End Sub

Private Sub CommandButton1_Click()

blocks = ActiveSheet.Range("C2")
streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H2")
reactions = ActiveSheet.Range("H3")

inblo = 0
numreact = 0

Do While inblo <= blocks - 1

If ActiveSheet.Cells(11, 2 + inblo) = "Reactor" Then

    numreact = numreact + 1

End If

inblo = inblo + 1
```

```

    Loop

Dim fi As Workbook
Dim fi2 As Workbook

Set fi2 = ActiveWorkbook

b = TextBox1
Workbooks.Open Filename:=b

Set fi = ActiveWorkbook

fi.Activate

'Copy components Name

Dim comp()

ReDim comp(components - 1)

hj = 0

Do While hj <= components - 1

    comp(hj) = ActiveWorkbook.ActiveSheet.Cells(10 + hj, 4)
    hj = hj + 1

Loop

'Copy flow-rates

Dim co()

ReDim co(components - 1, streams - 1)
a = 0
c = 0

Do While a <= components - 1

    Do While c <= streams - 1

        co(a, c) = ActiveWorkbook.ActiveSheet.Cells(10 + a, 6 + c)
        c = c + 1

    Loop

    a = a + 1

Loop

```

```
c = 0  
a = a + 1
```

```
Loop
```

```
'Copy temperature
```

```
Dim temp()  
Dim inctem As Integer
```

```
ReDim temp(streams - 1)
```

```
inctemp = 0
```

```
Do While inctemp <= streams - 1
```

```
    temp(inctemp) = ActiveWorkbook.ActiveSheet.Cells(6, 6 + inctemp)  
    inctemp = inctemp + 1
```

```
Loop
```

```
'Copy pressure
```

```
Dim pre()  
Dim incpre As Integer
```

```
ReDim pre(streams - 1)
```

```
incpre = 0
```

```
Do While incpre <= streams - 1
```

```
    pre(incpre) = ActiveWorkbook.ActiveSheet.Cells(7, 6 + incpre)  
    incpre = incpre + 1
```

```
Loop
```

```
'Copy the streams names
```

```
Dim names()  
Dim incna As Integer
```

```
ReDim names(streams - 1)
```

```
incna = 0
```

```
Do While incna <= streams - 1
```

```
names(incna) = ActiveWorkbook.ActiveSheet.Cells(3, 6 + incna)
incna = incna + 1
```

Loop

```
fi2.Activate
```

```
'Paste flow-rates
```

```
a = 0
```

```
c = 0
```

```
Do While a <= components - 1
```

```
    Do While c <= streams - 1
```

```
        ActiveWorkbook.ActiveSheet.Cells(26 + a, 2 + c) = co(a, c)
        c = c + 1
```

Loop

```
c = 0
```

```
a = a + 1
```

Loop

```
'Paste temperature
```

```
inctemp = 0
```

```
Do While inctemp <= streams - 1
```

```
    ActiveWorkbook.ActiveSheet.Cells(27 + components, 2 + inctemp) =
temp(inctemp)
    inctemp = inctemp + 1
```

Loop

```
'Paste pressure
```

```
incpre = 0
```

```
Do While incpre <= streams - 1
```

```
    ActiveWorkbook.ActiveSheet.Cells(28 + components, 2 + incpre) = pre(incpre)
    incpre = incpre + 1
```


Loop

'Paste the streams names

incna = 0

Do While incna <= streams - 1

 ActiveWorkbook.ActiveSheet.Cells(22, 2 + incna) = names(incna)
 incna = incna + 1

Loop

'inc = 0

'Do While inc <= streams

'ActiveWorkbook.activesheet.Cells(6 + components, 2 + inc).FormulaR1C1 =
"=SUM((INDIRECT("R[-]"&
GeneralData!R11C4&"]C",FALSE)):INDIRECT("R[-1]C",FALSE))"
'inc = inc + 1

'Loop

End Sub

End Sub

Sub gentable_sustainpro_sub3()

blocks = ActiveSheet.Range("C2")
streams = ActiveSheet.Range("C3")
components = ActiveSheet.Range("H2")
reactions = ActiveSheet.Range("H3")

'MVA Information

'Cycle to insert the name of reactions in the table

 b = 1

 Do While b <= reactions

```

ActiveSheet.Cells(34 + components + b, 1) = "R " & b
b = b + 1

```

Loop

'Cycle to insert the name of components in the table

```
a = 0
```

```
Do While a <= components - 1
```

```

    ActiveSheet.Cells(34 + components, 2 + a) = ActiveSheet.Cells(26 + a, 1)
    a = a + 1

```

Loop

'Create a table

```

ActiveSheet.Range(Cells(33 + components, 2), Cells(33 + components, 1 +
components)).Select

```

```
With Selection
```

```
    .HorizontalAlignment = xlCenter
```

```
    .VerticalAlignment = xlBottom
```

```
    .WrapText = False
```

```
    .Orientation = 0
```

```
    .AddIndent = False
```

```
    .IndentLevel = 0
```

```
    .ShrinkToFit = False
```

```
    .ReadingOrder = xlContext
```

```
    .MergeCells = False
```

```
End With
```

```
Selection.Merge
```

```

Range(Cells(34 + components, 1), Cells(34 + components + reactions, 1 +
components)).Select

```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```

        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    ActiveSheet.Range(Cells(33 + components, 2), Cells(33 + components, 1 +
components)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Cells(33 + components, 2) = "Components"
    ActiveSheet.Cells(31 + components, 1) = "Reactions"

    ActiveSheet.Cells(31 + components, 1).Select
    Selection.Font.Bold = True

```

```

With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With

'Demand Prices

ActiveSheet.Cells(36 + components + reactions, 1) = "Price"

ActiveSheet.Cells(36 + components + reactions, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With
ActiveSheet.Cells(40 + components + reactions, 1) = "Type"
ActiveSheet.Cells(38 + components + reactions, 2) = "Demand streams Price ($/kg)"

'Code to insert the names of the demand streams

a = 0
b = 0

Do While a <= streams - 1

    If ActiveSheet.Cells(24, 2 + a) = 0 Then

```

```

    ActiveSheet.Cells(39 + components + reactions, 2 + b) = ActiveSheet.Cells(22,
2 + a)
    b = b + 1

End If

a = a + 1

Loop

'Cycle to insert the name of components in the table
c = 0
Do While c <= components - 1

    ActiveSheet.Cells(41 + components + reactions + c, 1) = ActiveSheet.Cells(26
+ c, 1)
    c = c + 1

Loop

'Insert table

ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(38 + components +
reactions, 1 + b)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge

ActiveSheet.Range(Cells(39 + components + reactions, 1), Cells(40 + components +
components + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

```

With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(38 + components
+ reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous

```

```

        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone

    ActiveSheet.Range(Cells(39 + components + reactions, 2), Cells(40 + components
+ components + reactions, 1 + b)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With

    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With

    ActiveSheet.Range(Cells(40 + components + reactions, 1), Cells(40 + components
+ components + reactions, 1)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous

```

```

        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    ActiveSheet.Range(Cells(38 + components + reactions, 2), Cells(39 + components
+ reactions, 1 + b)).Select
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    ActiveSheet.Range(Cells(38 + components + reactions, 1), Cells(39 + components
+ components + reactions, 1 + b)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
    End With

'FEED PRICES

'Cycle to insert the name of components in the table
a = 0
Do While a <= components - 1

    ActiveSheet.Cells(44 + components * 2 + reactions, 2 + a) =
    ActiveSheet.Cells(26 + a, 1)

```



```
a = a + 1
```

```
Loop
```

```
ActiveSheet.Cells(43 + components * 2 + reactions, 2) = "Feed streams"
ActiveSheet.Cells(45 + components * 2 + reactions, 1) = "Price ($/kg)"
```

```
'Insert table
```

```
ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(43 +
components * 2 + reactions, 1 + components)).Select
```

```
With Selection
```

```
.HorizontalAlignment = xlCenter
```

```
.VerticalAlignment = xlBottom
```

```
.WrapText = False
```

```
.Orientation = 0
```

```
.AddIndent = False
```

```
.IndentLevel = 0
```

```
.ShrinkToFit = False
```

```
.ReadingOrder = xlContext
```

```
.MergeCells = False
```

```
End With
```

```
Selection.Merge
```

```
ActiveSheet.Range(Cells(44 + components * 2 + reactions, 1), Cells(45 +
components * 2 + reactions, 1 + components)).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
.LineStyle = xlContinuous
```

```
.Weight = xlThin
```

```
.ColorIndex = xlAutomatic
```

```

End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(43 +
components * 2 + reactions, 1 + components)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone

ActiveSheet.Range(Cells(44 + components * 2 + reactions, 2), Cells(45 +
components * 2 + reactions, 1 + components)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

ActiveSheet.Cells(45 + components * 2 + reactions, 1).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11
```

```

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 2), Cells(44 +
components * 2 + reactions, 1 + components)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Range(Cells(43 + components * 2 + reactions, 1), Cells(45 +
components * 2 + reactions, 1 + components)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
End With

```

'UTILITY PRICES

```

ActiveSheet.Cells(47 + components * 2 + reactions, 1) = "Utility Price"
ActiveSheet.Cells(47 + components * 2 + reactions, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With

```

```

ActiveSheet.Cells(49 + components * 2 + reactions, 1) = "Utilities"
ActiveSheet.Cells(49 + components * 2 + reactions, 2) = "Price"
ActiveSheet.Cells(50 + components * 2 + reactions, 1) = "Heating(HP)($/GJ)"
ActiveSheet.Cells(51 + components * 2 + reactions, 1) = "Heating(LP)($/GJ)"
ActiveSheet.Cells(52 + components * 2 + reactions, 1) = "Cooling($/GJ)"
ActiveSheet.Cells(53 + components * 2 + reactions, 1) = "Electricity($/kWh)"

```

'Create a table

```
ActiveSheet.Range(Cells(49 + components * 2 + reactions, 1), Cells(53 +  
components * 2 + reactions, 2)).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideVertical)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideHorizontal)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection
```

```
    .HorizontalAlignment = xlCenter
```

```
    .VerticalAlignment = xlBottom
```

```
    .WrapText = False
```

```
    .Orientation = 0
```

```
    .AddIndent = False
```

```
    .IndentLevel = 0
```

```
    .ShrinkToFit = False
```

```
    .ReadingOrder = xlContext
```

```
    .MergeCells = False
```

```
End With
```

```
ActiveSheet.Range(Cells(49 + components * 2 + reactions, 1), Cells(49 +
components * 2 + reactions, 2)).Select
```

```
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
```

'HEAT EXCHANGED

c = 0

a = 0

b = 0

Do While a <= blocks - 1

```
If ActiveSheet.Cells(11, 2 + a) = "HE" Or ActiveSheet.Cells(11, 2 + a) =
"Reactor" Or ActiveSheet.Cells(11, 2 + a) = "Filtration" Then
```

```
    ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
ActiveSheet.Cells(10, 2 + a)
```

```
    ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) = "Cp(kJ/kg-C)"
```

b = b + 1

Else

If ActiveSheet.Cells(11, 2 + a) = "Pump" Or ActiveSheet.Cells(11, 2 + a) =
"Comp" Or ActiveSheet.Cells(11, 2 + a) = "Centrifuge" Or ActiveSheet.Cells(11, 2 +
a) = "Mixer" Then

ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
ActiveSheet.Cells(10, 2 + a)
ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) = "Den(kg/m3)"
b = b + 1

Else

If ActiveSheet.Cells(11, 2 + a) = "Column" Then

ActiveSheet.Cells(67 + components * 3 + reactions, 2 + c) =
ActiveSheet.Cells(10, 2 + a)
c = c + 1

Else

If ActiveSheet.Cells(11, 2 + a) = "Evap" Or ActiveSheet.Cells(11, 2 + a) =
"Cond" Then

ActiveSheet.Cells(58 + components * 2 + reactions, 2 + b) =
ActiveSheet.Cells(10, 2 + a)
ActiveSheet.Cells(63 + components * 2 + reactions, 2 + b) =
"Hvap(kJ/kg)"
b = b + 1

End If

End If

End If

End If

a = a + 1

Loop

'Insert a table

If b = 0 Then

Else

ActiveSheet.Cells(55 + components * 2 + reactions, 1) = "Heats and Component's Properties"

ActiveSheet.Cells(55 + components * 2 + reactions, 1).Select

Selection.Font.Bold = True

With Selection.Font

.Name = "Arial"

.Size = 12

.Strikethrough = False

.Superscript = False

.Subscript = False

.OutlineFont = False

.Shadow = False

.Underline = xlUnderlineStyleNone

.ColorIndex = xlAutomatic

.TintAndShade = 0

.ThemeFont = xlThemeFontNone

End With

ActiveSheet.Cells(61 + components * 2 + reactions, 1) = "Cooling duty (GJ/hr)"

ActiveSheet.Cells(59 + components * 2 + reactions, 1) = "Heating duty (HP)(GJ/hr)"

ActiveSheet.Cells(60 + components * 2 + reactions, 1) = "Heating duty (LP)(GJ/hr)"

ActiveSheet.Cells(62 + components * 2 + reactions, 1) = "Electrical work (kW)"

ActiveSheet.Cells(63 + components * 2 + reactions, 1) = "Properties"

ActiveSheet.Cells(57 + components * 2 + reactions, 2) = "Units"

⌘Cycle to insert the name of components in the table

d = 0

Do While d <= components - 1

ActiveSheet.Cells(64 + components * 2 + reactions + d, 1) =

ActiveSheet.Cells(26 + d, 1)

d = d + 1

Loop

ActiveSheet.Range(Cells(57 + components * 2 + reactions, 2), Cells(57 + components * 2 + reactions, 1 + b)).Select

With Selection

.HorizontalAlignment = xlCenter

.VerticalAlignment = xlBottom

.WrapText = False

.Orientation = 0


```

.AddIndent = False
.IndentLevel = 0
.ShrinkToFit = False
.ReadingOrder = xlContext
.MergeCells = False
End With
Selection.Merge
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
Selection.Borders(xlInsideVertical).LineStyle = xlNone
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With

```

```
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
```

```

With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 2), Cells(58 +
components * 2 + reactions, 1 + b)).Select
Selection.Font.ColorIndex = 11
Selection.Font.Bold = True

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1)).Select
Selection.Font.ColorIndex = 11
Selection.Font.Bold = True

ActiveSheet.Range(Cells(63 + components * 2 + reactions, 2), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 2), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous

```

```

    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(58 + components * 2 + reactions, 1), Cells(58 +
components * 2 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(63 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic

```

```

End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(57 + components * 2 + reactions, 1), Cells(63 +
components * 3 + reactions, 1 + b)).Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
End With

End If

If c > 0 Then

ActiveSheet.Cells(68 + components * 3 + reactions, 1) = "Condenser(GJ/hr)"
ActiveSheet.Cells(72 + components * 4 + reactions, 1) = "Reboiler (HP)(GJ/hr)"
ActiveSheet.Cells(73 + components * 4 + reactions, 1) = "Reboiler (LP)(GJ/hr)"
ActiveSheet.Cells(70 + components * 3 + reactions, 2) = "Hv(kJ/kg)"
ActiveSheet.Cells(75 + components * 4 + reactions, 2) = "Hv(kJ/kg)"
ActiveSheet.Cells(69 + components * 3 + reactions, 1) = "Streams"
ActiveSheet.Cells(74 + components * 4 + reactions, 1) = "Streams"

'Cycle to insert the name of components in the table

c1 = 0
Do While c1 <= components - 1

    ActiveSheet.Cells(71 + components * 3 + reactions + c1, 1) =
ActiveSheet.Cells(26 + c1, 1)
    c1 = c1 + 1

```

Loop

c2 = 0

Do While c2 <= components - 1

```

    ActiveSheet.Cells(76 + components * 4 + reactions + c2, 1) =
ActiveSheet.Cells(26 + c2, 1)
    c2 = c2 + 1

```

Loop

'insert table

```

ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(75 +
components * 5 + reactions, 1 + c)).Select

```

With Selection

.HorizontalAlignment = xlCenter

.VerticalAlignment = xlBottom

.WrapText = False

.Orientation = 0

.AddIndent = False

.IndentLevel = 0

.ShrinkToFit = False

.ReadingOrder = xlContext

.MergeCells = False

End With

Selection.Borders(xlDiagonalDown).LineStyle = xlNone

Selection.Borders(xlDiagonalUp).LineStyle = xlNone

With Selection.Borders(xlEdgeLeft)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlEdgeTop)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlEdgeBottom)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

End With

With Selection.Borders(xlEdgeRight)

.LineStyle = xlContinuous

.Weight = xlThin

.ColorIndex = xlAutomatic

```
End With
With Selection.Borders(xlInsideVertical)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideHorizontal)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(67 +
components * 3 + reactions, 1 + c)).Select
```

```
Selection.Font.Bold = True
```

```
Selection.Font.ColorIndex = 11
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlMedium
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideVertical)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
ActiveSheet.Range(Cells(70 + components * 3 + reactions, 2), Cells(70 +
components * 3 + reactions, 1 + c)).Select
```

```
With Selection
```

```

.HorizontalAlignment = xlCenter
.VerticalAlignment = xlBottom
.WrapText = False
.Orientation = 0
.AddIndent = False
.IndentLevel = 0
.ShrinkToFit = False
.ReadingOrder = xlContext
.MergeCells = False
End With
Selection.Merge
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

ActiveSheet.Range(Cells(70 + components * 3 + reactions, 1), Cells(70 +
components * 3 + reactions, 1 + c)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(71 + components * 4 + reactions, 1), Cells(71 +
components * 4 + reactions, 1 + c)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic

```



```

End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(75 + components * 4 + reactions, 2), Cells(75 +
components * 4 + reactions, 1 + c)).Select

```

```

With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlBottom
    .WrapText = False
    .Orientation = 0
    .AddIndent = False
    .IndentLevel = 0
    .ShrinkToFit = False
    .ReadingOrder = xlContext
    .MergeCells = False
End With
Selection.Merge
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Range(Cells(75 + components * 4 + reactions, 1), Cells(75 +
components * 4 + reactions, 1 + c)).Select

```

```

Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic

```

```

End With
With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With

```

```

ActiveSheet.Range(Cells(67 + components * 3 + reactions, 1), Cells(75 +
components * 5 + reactions, 1)).Select
Selection.Font.ColorIndex = 11
Selection.Font.Bold = True

```

```
End If
```

```
'Reactions information
```

```
Dim dp As Integer
```

```
dp = ActiveSheet.Cells(1, 14)
```

```

ActiveSheet.Cells(77 + components * 5 + reactions, 1) = "Reactions"
ActiveSheet.Cells(77 + components * 5 + reactions, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With

```

```
ActiveSheet.Cells(79 + components * 5 + reactions, 1) = "Product"
ActiveSheet.Cells(80 + components * 5 + reactions, 1) = "Stream"
```

```
ActiveSheet.Range(Cells(79 + components * 5 + reactions, 1), Cells(80 +
components * 5 + reactions, 1 + dp)).Select
```

```
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
```

```
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
```

```
With Selection.Borders(xlEdgeLeft)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeTop)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeBottom)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlEdgeRight)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection.Borders(xlInsideHorizontal)
```

```
    .LineStyle = xlContinuous
```

```
    .Weight = xlThin
```

```
    .ColorIndex = xlAutomatic
```

```
End With
```

```
With Selection
```

```
    .HorizontalAlignment = xlCenter
```

```
    .VerticalAlignment = xlBottom
```

```
    .WrapText = False
```

```
    .Orientation = 0
```

```
    .AddIndent = False
```

```
    .IndentLevel = 0
```

```
    .ShrinkToFit = False
```

```
    .ReadingOrder = xlContext
```

```
    .MergeCells = False
```

```
End With
```

```

ActiveSheet.Range(Cells(79 + components * 5 + reactions, 1), Cells(80 +
components * 5 + reactions, 1)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11

```

```

ActiveSheet.Cells(82 + components * 5 + reactions, 2) = "Units"

```

```

'Insert the name of the reactive units and count them

```

```

inblo = 0
numreact = 0

```

```

Do While inblo <= blocks - 1

```

```

    If ActiveSheet.Cells(11, 2 + inblo) = "Reactor" Then

```

```

        ActiveSheet.Cells(83 + components * 5 + reactions, 2 + numreact) =
ActiveSheet.Cells(10, 2 + inblo)
        numreact = numreact + 1

```

```

    End If

```

```

    inblo = inblo + 1

```

```

Loop

```

```

'Insert the name of the reactions

```

```

Dim fini As Boolean

```

```

a = 0
a1 = 0
fini = False

```

```

Do Until fini = True

```

```

    If ActiveSheet.Cells(35 + components + a, 1) = 0 Then

```

```

        fini = True

```

```

    Else

```

```

        ActiveSheet.Cells(84 + components * 5 + reactions + a1, 1) =
ActiveSheet.Cells(35 + components + a, 1)
        ActiveSheet.Cells(85 + components * 5 + reactions + a1, 1) = "Component"
        a = a + 1

```

```

    a1 = a1 + 2

End If

Loop

ActiveSheet.Range(Cells(83 + components * 5 + reactions, 1), Cells(83 +
components * 5 + reactions * 3, 1 + numreact)).Select
    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
    End With

```

```

        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    ActiveSheet.Range(Cells(82 + components * 5 + reactions, 2), Cells(82 +
components * 5 + reactions, 1 + numreact)).Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlBottom
        .WrapText = False
        .Orientation = 0
        .AddIndent = False
        .IndentLevel = 0
        .ShrinkToFit = False
        .ReadingOrder = xlContext
        .MergeCells = False
    End With
    Selection.Merge
    Selection.Font.Bold = True
    Selection.Font.ColorIndex = 11

    Selection.Borders(xlDiagonalDown).LineStyle = xlNone
    Selection.Borders(xlDiagonalUp).LineStyle = xlNone
    With Selection.Borders(xlEdgeLeft)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeTop)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    Selection.Borders(xlInsideVertical).LineStyle = xlNone

    ActiveSheet.Range(Cells(84 + components * 5 + reactions, 1), Cells(84 +
components * 5 + reactions * 3, 1)).Select

```

```
Selection.Font.ColorIndex = 11
Selection.Font.Bold = True
```

```
ActiveSheet.Range(Cells(83 + components * 5 + reactions, 2), Cells(83 +
components * 5 + reactions, 1 + numreact)).Select
Selection.Font.Bold = True
Selection.Font.ColorIndex = 11
```

'Safety

```
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 1) = "Components"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 2) = "Flash Point(°C)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 3) = "Boiling Point(°C)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 4) = "UEL(%vol)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 5) = "LEL(%vol)"
ActiveSheet.Cells(87 + components * 5 + reactions * 3, 6) = "Toxic limit(ppm)"
```

```
ActiveSheet.Cells(85 + components * 5 + reactions * 3, 1) = "Safety"
ActiveSheet.Cells(85 + components * 5 + reactions * 3, 1).Select
Selection.Font.Bold = True
With Selection.Font
    .Name = "Arial"
    .Size = 12
    .Strikethrough = False
    .Superscript = False
    .Subscript = False
    .OutlineFont = False
    .Shadow = False
    .Underline = xlUnderlineStyleNone
    .ColorIndex = xlAutomatic
    .TintAndShade = 0
    .ThemeFont = xlThemeFontNone
End With
```

Dim fg As Integer

fg = 0

Do While fg <= components - 1

```
    ActiveSheet.Cells(88 + components * 5 + reactions * 3 + fg, 1) =
    ActiveSheet.Cells(26 + fg, 1)
    fg = fg + 1
```

Loop

```

ActiveSheet.Range(Cells(87 + components * 5 + reactions * 3, 1), Cells(87 +
components * 5 + reactions * 3, 6)).Select
  Selection.Font.ColorIndex = 11
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
  End With

  ActiveSheet.Range(Cells(87 + components * 5 + reactions * 3, 1), Cells(87 +
components * 6 + reactions * 3, 6)).Select
  Selection.Borders(xlDiagonalDown).LineStyle = xlNone
  Selection.Borders(xlDiagonalUp).LineStyle = xlNone
  With Selection.Borders(xlEdgeLeft)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
  End With
  With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous

```



```

        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlEdgeRight)
        .LineStyle = xlContinuous
        .Weight = xlMedium
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideVertical)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
    With Selection.Borders(xlInsideHorizontal)
        .LineStyle = xlContinuous
        .Weight = xlThin
        .ColorIndex = xlAutomatic
    End With
End Sub

Sub sustainpro_data()
Application.DisplayAlerts = False
Application.ScreenUpdating = False

Sheets("sustainpro_data").Visible = True
wb_path = ThisWorkbook.Path
pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
data_file_name = "sus_" & pj_name & "_" & case_number

wb_path = ThisWorkbook.Path
If Right(wb_path, 15) = "\Project Library" Then
directory = Left(wb_path, Len(wb_path) - 15) & "\SustainPro\SustainPro_data_file"
Else
directory = wb_path & "\SustainPro\SustainPro_data_file"
End If

data_file_path = directory & "\sustainpro_data.xls"
new_data_file_path = directory & "\" & data_file_name & ".xls"
Sheets("int_data").Range("path_sustainpro") = new_data_file_path

'check data
no_stream = Sheets("int_data").Range("no_stream")
no_unit = Sheets("int_data").Range("no_unit")
no_comp = Sheets("int_data").Range("no_comp")

```

```

no_rxn = Sheets("int_data").Range("no_rxn")
no_dp = Sheets("int_data").Range("no_dp")
store_sh = "sustainpro"
space_sh = "sustainpro_data"
process_type = Sheets("int_data").Range("process_type")
opt = Sheets("int_data").Range("opt")
Sheets(space_sh).Select

```

```

Range(Cells(7, 1), Cells(5000, 5000)).Delete

```

```

'import data to sustainpro_data sheet
Sheets(space_sh).[c1] = no_unit
Sheets(space_sh).[c3] = no_stream
Sheets(space_sh).[h1] = no_comp
Sheets(space_sh).[h3] = no_rxn
Sheets(space_sh).[n1] = no_dp
Sheets(space_sh).[n2] = process_type
Sheets(space_sh).[n3] = opt

```

```

[a15] = "Name"
[a16] = "Molecular Weight(g / mol)"
[a17] = "Hazardous"
[a18] = "Type"

```

```

Range(Cells(15, 1), Cells(15, 2)).Merge
Range(Cells(16, 1), Cells(16, 2)).Merge
Range(Cells(17, 1), Cells(17, 2)).Merge
Range(Cells(18, 1), Cells(18, 2)).Merge

```

```

Sheets(store_sh).Select
Range(Cells(9, 1), Cells(14, no_unit + 1)).Copy
Sheets(space_sh).Select
[a7].Select
ActiveSheet.Paste

```

```

Sheets(store_sh).Select
Range(Cells(16, 2), Cells(20, no_comp + 1)).Copy
Sheets(space_sh).Select
[c14].Select
ActiveSheet.Paste

```

```

[a20] = "Streams"
[a22] = "Stream Name"
[a23] = "Initial Unit"
[a24] = "Final Unit"
[a25] = "Components (kg/ hr)"

```

```

Sheets("stream table").Select
Range(Cells(9, 1), Cells(no_comp + 8, 1)).Copy
Sheets(space_sh).Select
[a26].Select
ActiveSheet.Paste
Sheets("stream table").Select
last_row = Range("a" & Rows.Count).End(xlUp).row
Range(Cells(5, 6), Cells(last_row, no_stream + 5)).Copy
Sheets(space_sh).Select
[b22].Select
ActiveSheet.Paste
'reactions
[a37] = "Reactions"
  Sheets(store_sh).Select
  Set to_find = [a:a].find(What:="Reaction\Stoichiometric Coefficients",
LookIn:=xlValues, _
  LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
  MatchCase:=False, SearchFormat:=False)
  If Not to_find Is Nothing Then data_row = to_find.row
  Range(Cells(data_row, 1), Cells(no_rxn + data_row, no_comp + 1)).Copy
  Sheets(space_sh).Select
  Cells(34 + no_comp, 1).Select
  ActiveSheet.Paste
  row_one = 38 + no_comp + no_rxn
  row_two = 40 + no_comp + no_rxn + no_comp
  Rows(row_one & ":" & row_two).Select
  Call Module1.clear_table

  Sheets(store_sh).Select
  Set to_find = [a:a].find(What:="Demand stream price", LookIn:=xlValues, _
  LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
  MatchCase:=False, SearchFormat:=False)
  If Not to_find Is Nothing Then data_row = to_find.row
  last_data_col = Cells(data_row + 3, Columns.Count).End(xlToLeft).Column
  Range(Cells(data_row + 2, 1), Cells(data_row + 4 + no_comp,
last_data_col)).Copy
  Sheets(space_sh).Select
  Cells(38 + no_comp + no_rxn, 1).Select
  ActiveSheet.Paste

  Sheets(store_sh).Select
  Set to_find = [a:a].find(What:="Feed stream price", LookIn:=xlValues, _
  LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
  MatchCase:=False, SearchFormat:=False)
  If Not to_find Is Nothing Then data_row = to_find.row
  Range(Cells(data_row + 3, 2), Cells(data_row + 4, no_comp + 1)).Copy

```

```
Sheets(space_sh).Select
Cells(44 + no_comp + no_rxn + no_comp, 2).Select
ActiveSheet.Paste
```

```
Sheets(store_sh).Select
Set to_find = [a:a].find(What:="Utility Price", LookIn:=xlValues, _
LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
MatchCase:=False, SearchFormat:=False)
If Not to_find Is Nothing Then data_row = to_find.row
Range(Cells(data_row + 3, 2), Cells(data_row + 6, 2)).Copy
Sheets(space_sh).Select
Cells(50 + no_comp + no_rxn + no_comp, 2).Select
ActiveSheet.Paste
Cells(55 + no_comp + no_rxn + no_comp, 1) = "Heats and Component's
Properties"
Cells(55 + no_comp + no_rxn + no_comp, 1).Font.Bold = True
```

```
Sheets(store_sh).Select
Set to_find = [a:a].find(What:="Equipments", LookIn:=xlValues, _
LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
MatchCase:=False, SearchFormat:=False)
If Not to_find Is Nothing Then data_row = to_find.row
    last_use_col1 = Cells(data_row + 3, Columns.Count).End(xlToLeft).Column
    last_use_col2 = Cells(data_row + 12 + no_comp,
Columns.Count).End(xlToLeft).Column
    If last_use_col1 > last_use_col2 Then
        Range(Cells(data_row + 2, 1), Cells(data_row + (3 * no_comp) + 20,
last_use_col1)).Copy
    Else
        Range(Cells(data_row + 2, 1), Cells(data_row + (3 * no_comp) + 20,
last_use_col2)).Copy
    End If
Sheets(space_sh).Select
Cells(57 + no_comp + no_rxn + no_comp, 1).Select
ActiveSheet.Paste
'Cells(57 + (5 * no_comp) + no_rxn + 20, 1) = "Reactions"

Cells(57 + (5 * no_comp) + no_rxn + 25, 1).EntireRow.Select
Selection.MergeCells = False
Selection.ClearContents
```

```
Sheets(store_sh).Select
Set to_find = [a:a].find(What:="Reaction Data", LookIn:=xlValues, _
LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
MatchCase:=False, SearchFormat:=False)
```

```

If Not to_find Is Nothing Then data_row = to_find.row
Range(Cells(data_row + 2, 2), Cells(data_row + 3, no_dp + 1)).Copy
Sheets(space_sh).Select
Cells(57 + (5 * no_comp) + no_rxn + 22, 2).Select
ActiveSheet.Paste
Sheets(store_sh).Select
'second table of reaction data
last_use_col = Cells(11 + no_rxn + data_row,
Columns.Count).End(xlToLeft).Column
Range(Cells(10 + no_rxn + data_row, 1), Cells(11 + (3 * no_rxn) + data_row,
last_use_col)).Copy
Sheets(space_sh).Select
Cells(57 + (5 * no_comp) + no_rxn + 25, 1).Select
ActiveSheet.Paste
'Safety

'Safety

ActiveSheet.Cells(87 + no_comp * 5 + no_rxn * 3, 1) = "Components"
ActiveSheet.Cells(87 + no_comp * 5 + no_rxn * 3, 2) = "Flash Point(°C)"
ActiveSheet.Cells(87 + no_comp * 5 + no_rxn * 3, 3) = "Boiling Point(°C)"
ActiveSheet.Cells(87 + no_comp * 5 + no_rxn * 3, 4) = "UEL(%vol)"
ActiveSheet.Cells(87 + no_comp * 5 + no_rxn * 3, 5) = "LEL(%vol)"
ActiveSheet.Cells(87 + no_comp * 5 + no_rxn * 3, 6) = "Toxic limit(ppm)"

ActiveSheet.Cells(85 + no_comp * 5 + no_rxn * 3, 1) = "Safety"
ActiveSheet.Cells(85 + no_comp * 5 + no_rxn * 3, 1).Select
Selection.Font.Bold = True
With Selection.Font
.Name = "Arial"
.Size = 12
.Strikethrough = False
.Superscript = False
.Subscript = False
.OutlineFont = False
.Shadow = False
.Underline = xlUnderlineStyleNone
.ColorIndex = xlAutomatic
.TintAndShade = 0
.ThemeFont = xlThemeFontNone
End With

Dim fg As Integer

fg = 0

Do While fg <= no_comp - 1

```

```

ActiveSheet.Cells(88 + no_comp * 5 + no_rxn * 3 + fg, 1) = ActiveSheet.Cells(26
+ fg, 1)
fg = fg + 1

```

Loop

```

ActiveSheet.Range(Cells(87 + no_comp * 5 + no_rxn * 3, 1), Cells(87 + no_comp *
5 + no_rxn * 3, 6)).Select
Selection.Font.ColorIndex = 11
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeTop)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
.LineStyle = xlContinuous
.Weight = xlThin
.ColorIndex = xlAutomatic
End With

ActiveSheet.Range(Cells(87 + no_comp * 5 + no_rxn * 3, 1), Cells(87 + no_comp
* 6 + no_rxn * 3, 6)).Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
With Selection.Borders(xlEdgeLeft)
.LineStyle = xlContinuous
.Weight = xlMedium
.ColorIndex = xlAutomatic
End With

```

```

With Selection.Borders(xlEdgeTop)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeBottom)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlEdgeRight)
    .LineStyle = xlContinuous
    .Weight = xlMedium
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideVertical)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With
With Selection.Borders(xlInsideHorizontal)
    .LineStyle = xlContinuous
    .Weight = xlThin
    .ColorIndex = xlAutomatic
End With

Set space_find = [a:a].find(What:="Safety", LookIn:=xlValues, _
LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
MatchCase:=False, SearchFormat:=False)
If Not space_find Is Nothing Then data_row_space = space_find.row
Sheets(store_sh).Select
Set to_find = [a:a].find(What:="Safety", LookIn:=xlValues, _
LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
MatchCase:=False, SearchFormat:=False)
If Not to_find Is Nothing Then data_row = to_find.row
Range(Cells(data_row + 3, 1), Cells(data_row + 2 + no_comp, 6)).Copy
Sheets(space_sh).Select
Cells(data_row_space + 3, 1).Select
ActiveSheet.Paste
Sheets("sustainpro_data").Visible = False
Sheets("start menu").Select
End Sub

Sub link_sustainpro_data()
Application.ScreenUpdating = False
Application.DisplayAlerts = False

```

```

wb1_name = ThisWorkbook.Name
wb_path = ThisWorkbook.Path
pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
new_data_file_name = "sus_" & pj_name & "_" & case_number
process_type = ThisWorkbook.Sheets("int_data").Range("process_type")
opt = ThisWorkbook.Sheets("int_data").Range("opt")
wb_path = ThisWorkbook.Path

software_path = Left(wb_path, Len(wb_path) - 16) &
"\Software\SustainPro\sustainpro.xlsm"
data_file_path = Left(wb_path, Len(wb_path) - 16) &
"\Software\SustainPro\sustainpro_data\sustainpro_data.xls"
new_data_file_name = "sus_" & pj_name & "_" & case_number
new_data_file_path = Left(wb_path, Len(wb_path) - 16) &
"\Software\SustainPro\sustainpro_data\" & new_data_file_name & ".xls"
If Len(Dir$(new_data_file_path)) > 0 Then Kill new_data_file_path

'fill out data_file_path
Sheets("int_data").Range("path_sustainpro") = new_data_file_path

no_unit = Sheets("int_data").Range("no_unit")
no_comp = Sheets("int_data").Range("no_comp")
no_stream = Sheets("int_data").Range("no_stream")

If no_unit > no_comp And no_unit > no_stream Then last_col = no_unit + 5
If no_comp > no_unit And no_comp > no_stream Then last_col = no_comp + 5
If no_stream > no_unit And no_stream > no_comp Then last_col = no_stream + 5

Application.Workbooks.Open (data_file_path)
wb2_name = ActiveWorkbook.Name
Windows(wb1_name).Activate
Sheets("sustainpro_data").Visible = True
Sheets("sustainpro_data").Select
last_row = Range("a" & Rows.Count).End(xlUp).row

Range(Cells(1, 1), Cells(last_row, last_col)).Copy
Windows(wb2_name).Activate
Sheets("sheet1").Select
[a1].Select
ActiveSheet.Paste

ActiveWorkbook.SaveAs Filename:=new_data_file_path, _
FileFormat:=xlOpenXMLWorkbookMacroEnabled, CreateBackup:=False
ActiveWorkbook.Close

```



```

ThisWorkbook.Activate
Sheets("sustainpro_data").Visible = False

'open software
Workbooks.Open Filename:=software_path
Sheets("StartMenu").Select
If process_type <> "Batch" Then
    If Sheets("StartMenu").CheckBox1 = False Then CheckBox1 = True
Else
    If Sheets("StartMenu").CheckBox2 = False Then CheckBox2 = True
End If
Sheets("GeneralData").[m9] = opt
Sheets("StartMenu").Select

Sheets("StartMenu").TextBox6.Value = new_data_file_path
Sheets("StartMenu").TextBox6.Text = new_data_file_path
Sheets("StartMenu").CommandButton7 = True

Windows(new_data_file_name).Close
Windows("sustainpro").Activate
Sheets("MainMenu").Select

End Sub
Sub del_im_sustainpro()
Application.DisplayAlerts = False
For i = 1 To 11
sheet_name = Sheets("int_data").Cells(i, 26)
On Error Resume Next
Sheets(sheet_name).Delete
Next i
End Sub

Sub im_sustainpro1(sheet_name)
Application.DisplayAlerts = False
ThisWorkbook.Activate
wb1_name = ThisWorkbook.Name
software_name = Sheets("Start menu").TextBox1.Value
Windows(software_name).Activate
Sheets(sheet_name).Select
Sheets(sheet_name).Copy After:=Workbooks(wb1_name).Sheets("SustainPro
results")
Sheets("Start menu").Select
End Sub
Module "I2_LCSoft"

```

```

Sub transfer_to_lcsoft()
ThisWorkbook.Activate
Application.ScreenUpdating = False
Application.DisplayAlerts = False
'parameter
wb1_name = ThisWorkbook.Name
wb_path = ThisWorkbook.Path
pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
no_comp = Sheets("int_data").Range("no_comp")
no_stream = Sheets("int_data").Range("no_stream")
case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
software_path = Left(wb_path, Len(wb_path) - 16) & "\Software\LCSoft\LCSoft v
2.0.xlsm"
lcsoft_save_path = Left(wb_path, Len(wb_path) - 16) &
"\Software\LCSoft\Projects\" & pj_name & "_" & case_number & "_lcsoft.xlsm"
product_name = Sheets("lcsoft").TextBox1.Value
product_stream = Sheets("lcsoft").TextBox2.Value
no_section = Sheets("lcsoft").TextBox3.Value

'delete lcsoft file
If Len(Dir$(lcsoft_save_path)) > 0 Then Kill lcsoft_save_path

Application.Workbooks.Open (software_path)
Sheets("main menu").Select
ActiveWorkbook.SaveAs Filename:=lcsoft_save_path, _
FileFormat:=xlOpenXMLWorkbookMacroEnabled, CreateBackup:=False
wb2_name = ActiveWorkbook.Name
Sheets("main menu").Sh1_cmb_newproject.BackColor = &HFFFF00
'identify units in lcsoft
Sheets("LCI").Range("unit_mass") = "kg"
Sheets("LCI").Range("unit_volume") = "l"
Sheets("LCI").Range("unit_energy") = "GJ"
Sheets("LCI").Range("unit_time") = "hr"
Sheets("LCI").Range("unit_power") = "kW"
Sheets("results").[c2] = pj_name & "_" & case_number
Sheets("results").[c3] = ""

'input stream table
Sheets("main menu").CommandButton7 = True
Windows(wb1_name).Activate

Sheets("lcsoft_data").Visible = True
Sheets("lcsoft_data").Select
Range(Cells(13, 1), Cells(no_comp + 14, no_stream + 2)).Copy
Windows(wb2_name).Activate
Sheets("generate stream table").Select
[a1].Select

```

```

ActiveSheet.Paste
Sheets("generate stream table").CommandButton1 = True
'input equipment table
Sheets("main menu").CommandButton9 = True
Windows(wb1_name).Activate
Sheets("lcsoft_data").Select
last equip_row = Sheets("lcsoft_data").Range("a" & Rows.Count).End(xlUp).row
Range(Cells(20 + no_comp, 1), Cells(last equip_row, 5)).Copy
Windows(wb2_name).Activate
Sheets("generate equipment table").Select
[b7].Select
ActiveSheet.Paste
'close lcsoft_data
Windows(wb1_name).Activate
Sheets("lcsoft_data").Visible = False
Windows(wb2_name).Activate
Sheets("generate equipment table").Select
Sheets("generate equipment table").CommandButton1 = True
'product identification
Sheets("results").Range("product_name") = product_name
Sheets("results").Range("product_stream") = product_stream
'product amount
For i = 8 To Sheets("Define Stream").[b8].End(xlDown).row
If Sheets("Define Stream").Cells(i, 2) = product_name Then
    For j = 6 To Sheets("Define Stream").[f6].End(xlToRight).Column
    If Sheets("Define Stream").Cells(6, j) = product_stream Then
        amount = Sheets("Define Stream").Cells(i, j)
    End If
    Next j
End If
Next i
Sheets("results").Range("product_amount") = amount
Sheets("results").Range("product_unit_mass") = "kg"
Sheets("results").Range("product_unit_time") = "hr"
Sheets("results").Range("product_allocation") = "mass"

Application.Run wb2_name & "!step1.product_identification"
Sheets("main menu").Sh1_cmb_functionalunit.BackColor = &HFFFF00
Sheets("Define Equipment").Visible = True
Windows(wb1_name).Activate
Sheets("lcsoft_data").Visible = True
Sheets("lcsoft_data").Select
Range(Cells(20 + no_comp, 6), Cells(last equip_row, 6)).Copy
Windows(wb2_name).Activate
Sheets("Define Equipment").Select
[g7].Select
ActiveSheet.Paste

```

```

Sheets("Define Equipment").Visible = False

'transfer lca section
Windows(wb1_name).Activate
For s = 1 To Sheets("lcsoft_data_2").Range("a" & Rows.Count).End(xlUp).row
Sheets("lcsoft_data_2").Visible = True
Sheets("lcsoft_data_2").Select
If Cells(s, 1) = "Section:" Then
s_name = Sheets("lcsoft_data_2").Cells(s, 2)
s_des = Sheets("lcsoft_data_2").Cells(s + 1, 2)
a_row = s + 1
Do
a_row = a_row + 1
Loop Until Sheets("lcsoft_data_2").Cells(a_row, 1) = "Section:"
last_copy_row = a_row - 2

'transfer to lcsoft
Windows(wb2_name).Activate
'check existance of sheet
On Error Resume Next
If Not Sheets(s_name) Is Nothing Then
Sheets("results").Visible = True
Sheets("results").Select
last_results_row = Sheets("results").Range("c" & Rows.Count).End(xlUp).row
If Sheets("results").Cells(last_results_row, 3) = "Total" Then
Sheets("results").Cells(last_results_row, 3).Clear
For c_row = 9 To last_results_row
If Sheets("results").Cells(c_row, 3) = s_name Then
Cells(c_row, 3).EntireRow.Delete
Exit For
End If
Next c_row
End If

Application.Run wb2_name & "!step2.delete_LCA_section (s_name)"
'add new data
Sheets("results").Visible = True
Sheets("results").Select
next_add_row = Sheets("Results").Range("c" & Rows.Count).End(xlUp).Offset(1,
0).row
Cells(next_add_row, 3) = s_name
Cells(next_add_row, 4) = s_des

Sheets("section").Visible = True
Sheets("section").Copy After:=Sheets("LCI")
ActiveSheet.Name = s_name
With [e:e].Font

```

```

        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With

    With [o:o].Font
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With
    [b3] = s_name
    [b4] = s_des
    Sheets("main menu").CommandButton2.BackColor = &H8000000A
    Sheets("LCI").Visible = False

```

```

'retrieve data from interface
Windows(wb1_name).Activate
Sheets("lcsoft_data_2").Visible = True
Sheets("lcsoft_data_2").Select
Range(Cells(s, 1), Cells(last_copy_row, 20)).Copy
Windows(wb2_name).Activate
Sheets(s_name).Select
Application.DisplayAlerts = False
[a3].Select
ActiveSheet.Paste
Windows(wb1_name).Activate
Sheets("lcsoft_data_2").Visible = False
End If
Windows(wb1_name).Activate
Next s
Sheets("lcsoft_data_2").Visible = False

```

```
'input data to lcsoft by user
```

```

Windows(wb2_name).Activate
Sheets("Main Menu").Select
MsgBox "Please click to difine stream component and equipment/utility"

```

```

End Sub
Sub gen_sim_tables()
'stream table
Application.ScreenUpdating = False
Application.DisplayAlerts = False
Sheets("lcsoft_data").Visible = True
store_sh = "stream table"
space_sh = "lcsoft_data"
Sheets(space_sh).Select
Range(Cells(11, 1), Cells(3000, 3000)).Delete
Cells(11, 1).Select

```

```

ActiveCell = "Stream table"
Call Module1.bold_highlight_color
Cells(13, 1) = "Stream Name"
Cells(14, 1) = "Total Weight Comp. Rates"
For i = 9 To Sheets("int_data").Range("no_comp") + 8
next_row = Sheets("lcsoft_data").Range("a" & Rows.Count).End(xlUp).Offset(1,
0).row
Cells(next_row, 1) = Sheets("stream table").Cells(i, 1)
Next i

For i = 6 To Sheets("int_data").Range("no_stream") + 5
next_col = Sheets("lcsoft_data").Cells(13, Columns.Count).End(xlToLeft).Offset(0,
1).Column
If next_col = 2 Then next_col = 3
If next_col < 2 Then next_col = 3
Cells(13, next_col) = Sheets("stream table").Cells(5, i)
Next i

For i = 9 To Sheets("int_data").Range("no_comp") + 8
For j = 6 To Sheets("int_data").Range("no_stream") + 5
Cells(i + 6, j - 3) = Sheets("stream table").Cells(i, j)
Next j
Next i

Range(Cells(13, 1), Cells(Cells(Rows.Count, 1).End(xlUp).row, Cells(13,
Columns.Count).End(xlToLeft).Column)).Select
Call Module1.create_grid

last_row = Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 1) = "Equipment Table"
Cells(last_row + 2, 1).Select

Call Module1.bold_highlight_color
start_equip_row = last_row + 6

Cells(last_row + 4, 1) = "Unit"
Cells(last_row + 4, 2) = "Type of unit"
Cells(last_row + 4, 3) = "Duty/Work"
Cells(last_row + 4, 4) = "Unit"
Cells(last_row + 5, 4) = "Energy"
Cells(last_row + 5, 5) = "time"
Cells(last_row + 5, 6) = "Activity"

Range(Cells(last_row + 4, 1), Cells(last_row + 5, 1)).Select
Selection.Merge
Call Module1.bold_highlight_color
Range(Cells(last_row + 4, 2), Cells(last_row + 5, 2)).Select

```

```

Selection.Merge
Call Module1.bold_highlight_color
Range(Cells(last_row + 4, 3), Cells(last_row + 5, 3)).Select
Selection.Merge
Call Module1.bold_highlight_color
Range(Cells(last_row + 4, 4), Cells(last_row + 4, 5)).Select
Selection.Merge
Call Module1.bold_highlight_color
Cells(last_row + 5, 4).Select
Call Module1.bold_highlight_color
Cells(last_row + 5, 5).Select
Call Module1.bold_highlight_color
Range(Cells(last_row + 4, 6), Cells(last_row + 5, 6)).Select
Call Module1.bold_highlight_color
Selection.Merge

```

```

For i = 2 To Sheets("equipment table").Cells(8,
Columns.Count).End(xlToLeft).Column
next_row = Sheets("lcsoft_data").Range("a" & Rows.Count).End(xlUp).Offset(1,
0).row
If Sheets("equipment table").Cells(9, i) <> "Column" Then
For j = 10 To 13
If Sheets("equipment table").Cells(j, i) <> 0 And Sheets("equipment table").Cells(j,
i) <> "-" And Sheets("equipment table").Cells(j, i) <> "" Then
activity = Sheets("equipment table").Cells(j, 1)
Select Case Left(activity, 4)
Case "Heat"
Sheets("lcsoft_data").Cells(next_row, 6) = "Heating"
Sheets("lcsoft_data").Cells(next_row, 4) = "GJ"
Case "Cool"
Sheets("lcsoft_data").Cells(next_row, 6) = "Cooling"
Sheets("lcsoft_data").Cells(next_row, 4) = "GJ"
Case "Elec"
Sheets("lcsoft_data").Cells(next_row, 6) = "Electric usage"
Sheets("lcsoft_data").Cells(next_row, 4) = "kW"
End Select
Sheets("lcsoft_data").Cells(next_row, 1) = Sheets("equipment table").Cells(8, i)
Sheets("lcsoft_data").Cells(next_row, 2) = Sheets("equipment table").Cells(9, i)
Sheets("lcsoft_data").Cells(next_row, 3) = Sheets("equipment table").Cells(j, i)
Sheets("lcsoft_data").Cells(next_row, 5) = "hr"
End If
Next j
End If
Next i

```

```

For i = 2 To Sheets("equipment table").Cells(8,
Columns.Count).End(xlToLeft).Column
next_row = Sheets("lcsoft_data").Range("a" & Rows.Count).End(xlUp).Offset(1,
0).row
If Sheets("equipment table").Cells(9, i) = "Column" Then
For j = 10 To 13
If Sheets("equipment table").Cells(j, i) <> 0 And Sheets("equipment table").Cells(j,
i) <> "-" And Sheets("equipment table").Cells(j, i) <> "" Then
activity = Sheets("equipment table").Cells(j, 1)
Select Case Left(activity, 4)
Case "Heat"
next_row = Sheets("lcsoft_data").Range("a" & Rows.Count).End(xlUp).Offset(1,
0).row
Sheets("lcsoft_data").Cells(next_row, 1) = Sheets("equipment table").Cells(8, i) & "-
reboiler"
Sheets("lcsoft_data").Cells(next_row, 2) = Sheets("equipment table").Cells(9, i)
Sheets("lcsoft_data").Cells(next_row, 3) = Sheets("equipment table").Cells(j, i)
Sheets("lcsoft_data").Cells(next_row, 6) = "Heating"
Sheets("lcsoft_data").Cells(next_row, 4) = "GJ"
Sheets("lcsoft_data").Cells(next_row, 5) = "hr"
Case "Cool"
Sheets("lcsoft_data").Cells(next_row, 1) = Sheets("equipment table").Cells(8, i) & "-
condenser"
Sheets("lcsoft_data").Cells(next_row, 2) = Sheets("equipment table").Cells(9, i)
Sheets("lcsoft_data").Cells(next_row, 3) = Sheets("equipment table").Cells(j, i)
Sheets("lcsoft_data").Cells(next_row, 6) = "Cooling"
Sheets("lcsoft_data").Cells(next_row, 4) = "GJ"
Sheets("lcsoft_data").Cells(next_row, 5) = "hr"
Case "Elec"
Sheets("lcsoft_data").Cells(next_row, 1) = Sheets("equipment table").Cells(8, i) & "-
condenser"
Sheets("lcsoft_data").Cells(next_row, 2) = Sheets("equipment table").Cells(9, i)
Sheets("lcsoft_data").Cells(next_row, 3) = Sheets("equipment table").Cells(j, i)
Sheets("lcsoft_data").Cells(next_row, 6) = "Electric usage"
Sheets("lcsoft_data").Cells(next_row, 4) = "kW"
Sheets("lcsoft_data").Cells(next_row, 5) = "hr"
End Select
End If
Next j
End If
Next i
Range(Cells(start Equip_row, 1), Cells(Sheets("lcsoft_data").Range("a" &
Rows.Count).End(xlUp).row, 6)).Select
Call Module1.create_grid
Sheets("lcsoft_data").Visible = False

End Sub

```



```

Sub gen_section_table()
Sheets("LCSoft").Select
Application.DisplayAlerts = False
Application.ScreenUpdating = False
Range(Cells(2, 1), Cells(3000, 3000)).Delete
Cells(3, 2) = "Section Name (no more than 22 characters)"
Cells(3, 3) = "Section description"
Range(Cells(3, 2), Cells(3 + Sheets("LCSoft").TextBox3.Value, 3)).Select
Call Module1.create_grid
Selection.Font.Bold = True
Columns("b").AutoFit
Columns("c").AutoFit
End Sub

```

```

Sub list_section1()
Sheets("lcsoft").Select
st_del_row = Sheets("lcsoft").TextBox3.Value + 5
Application.DisplayAlerts = False
Application.ScreenUpdating = False
Range(Cells(st_del_row, 1), Cells(5000, 5000)).Delete

```

```

list_row = Sheets("lcsoft").Range("b" & Rows.Count).End(xlUp).row
Cells(list_row + 2, 2) = "Input Streams:"
Cells(list_row + 2, 2).Select
Call Module1.lc_input_st_grid
Cells(list_row + 3, 2) = "Input Equipments:"
Cells(list_row + 3, 2).Select
Call Module1.lc_input_eq
Cells(list_row + 4, 2) = "Output Streams:"
Cells(list_row + 4, 2).Select
Call Module1.lc_op_st
Range(Cells(list_row + 2, 2), Cells(list_row + 4, 2)).Font.Bold = True
'input streams
For k = 6 To Sheets("int_data").Range("no_stream") + 5
If Sheets("stream table").Cells(6, k) = 0 Then
next_col = Sheets("lcsoft").Cells(list_row + 2,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
Cells(list_row + 2, next_col) = Sheets("stream table").Cells(5, k)
Cells(list_row + 2, next_col).Select
Call Module1.lc_input_st_grid
End If
Next k
'input equipments
start_eq = 20 + Sheets("int_data").Range("no_comp")
end_eq = Sheets("lcsoft_data").Range("a" & Rows.Count).End(xlUp).row

```

```

For k = start_eq To end_eq
next_col = Sheets("lcsoft").Cells(list_row + 3,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
Cells(list_row + 3, next_col) = Sheets("lcsoft_data").Cells(k, 1)
Cells(list_row + 3, next_col).Select
Call Module1.lc_input_eq
Next k
'output streams
For k = 6 To Sheets("int_data").Range("no_stream") + 5
If Sheets("stream table").Cells(7, k) = 0 Then
next_col = Sheets("lcsoft").Cells(list_row + 4,
Columns.Count).End(xlToLeft).Offset(0, 1).Column
Cells(list_row + 4, next_col) = Sheets("stream table").Cells(5, k)
Cells(list_row + 4, next_col).Select
Call Module1.lc_op_st
End If
Next k

col1 = Sheets("lcsoft").Cells(list_row + 2, Columns.Count).End(xlToLeft).Column
col2 = Sheets("lcsoft").Cells(list_row + 3, Columns.Count).End(xlToLeft).Column
col3 = Sheets("lcsoft").Cells(list_row + 4, Columns.Count).End(xlToLeft).Column
If col1 >= col2 And col1 >= col3 Then
hi_col = col1
ElseIf col2 >= col1 And col2 >= col3 Then
hi_col = col2
ElseIf col3 >= col1 And col3 >= col2 Then
hi_col = col3
End If

For i = 4 To Sheets("lcsoft").TextBox3.Value + 3
section_name = Sheets("lcsoft").Cells(i, 2)
section_des = Sheets("lcsoft").Cells(i, 3)
last_row = Sheets("lcsoft").Range("b" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 2) = "Section:"
Cells(last_row + 2, 2).Font.Bold = True
Cells(last_row + 2, 3) = section_name
Cells(last_row + 3, 2) = "Comment:"
Cells(last_row + 3, 3) = section_des
Cells(last_row + 3, 2).Font.Bold = True
Range(Cells(last_row + 2, 2), Cells(last_row + 2, hi_col)).Select
Call Module1.lcsoft_section_head

Cells(last_row + 5, 2) = "Input streams:"
Cells(last_row + 5, 2).Select
Call Module1.lc_input_st_grid
Cells(last_row + 6, 2) = "Input equipments:"
Cells(last_row + 6, 2).Select

```

```

Call Module1.lc_input_eq
Cells(last_row + 7, 2) = "Output streams"
Cells(last_row + 7, 2).Select
Call Module1.lc_op_st
Cells(last_row + 8, 2) = "Compartment (air/water/soil)"
Cells(last_row + 9, 2) = "Sub compartment"
Range(Cells(last_row + 5, 2), Cells(last_row + 9, 2)).Select
Selection.Font.Bold = True
  For m = 5 To 9
    Range(Cells(last_row + m, 3), Cells(last_row + m, hi_col)).Select
    Call Module1.lcsoft_entire_grid
  Next m
Next i

End Sub

Sub list_section2()
Application.ScreenUpdating = False
Application.DisplayAlerts = False
Sheets("int_data").Visible = True
Sheets("lcsoft_data_2").Visible = True
Sheets("lcsoft_data_2").Select
Range(Cells(1, 1), Cells(5000, 5000)).Delete

For i = Sheets("lcsoft").TextBox3.Value + 9 To Sheets("lcsoft").Range("b" &
Rows.Count).End(xlUp).row
If Sheets("lcsoft").Cells(i, 2) = "Section:" Then
section_name = Sheets("lcsoft").Cells(i, 3)
section_des = Sheets("lcsoft").Cells(i + 1, 3)
input_stream = i + 3
input_eq = i + 4
output_stream = i + 5
compa = i + 6
sub_compa = i + 7

Sheets("int_data").Select
[a77:l82].Copy
Sheets("lcsoft_data_2").Select
last_row = Sheets("lcsoft_data_2").Range("a" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 1).Select
ActiveSheet.Paste
Cells(last_row + 2, 2) = section_name
Cells(last_row + 3, 2) = section_des
  'input stream
  For j = 3 To Sheets("lcsoft").Cells(input_stream,
Columns.Count).End(xlToLeft).Column

```

```

    next_input = Sheets("lcsoft_data_2").Range("a" &
Rows.Count).End(xlUp).Offset(1, 0).row
    Sheets("lcsoft_data_2").Cells(next_input, 1) = Sheets("lcsoft").Cells(input_stream,
j)
    Sheets("lcsoft_data_2").Cells(next_input, 5) = "stream_sim"

    With Sheets("lcsoft_data_2").Cells(next_input, 5).Font
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = 0
    End With
    Next j
    'input equipment
    For j = 3 To Sheets("lcsoft").Cells(input_eq,
Columns.Count).End(xlToLeft).Column
        next_input = Sheets("lcsoft_data_2").Range("a" &
Rows.Count).End(xlUp).Offset(1, 0).row
        Sheets("lcsoft_data_2").Cells(next_input, 1) = Sheets("lcsoft").Cells(input_eq, j)
        Sheets("lcsoft_data_2").Cells(next_input, 5) = "equip_sim"
        With Sheets("lcsoft_data_2").Cells(next_input, 5).Font
            .ThemeColor = xlThemeColorDark1
            .TintAndShade = 0
        End With
    Next j
    cut_row = Sheets("lcsoft_data_2").Range("a" & Rows.Count).End(xlUp).Offset(1,
0).row
    Sheets("int_data").Select
    [a84:e84].Copy
    Sheets("lcsoft_data_2").Select
    Cells(cut_row, 1).Select
    ActiveSheet.Paste
    'output
    For j = 3 To Sheets("lcsoft").Cells(output_stream,
Columns.Count).End(xlToLeft).Column
        next_input = Sheets("lcsoft_data_2").Range("h" &
Rows.Count).End(xlUp).Offset(1, 0).row
        Sheets("lcsoft_data_2").Cells(next_input, 8) =
Sheets("lcsoft").Cells(output_stream, j)
        Sheets("lcsoft_data_2").Cells(next_input, 9) = Sheets("lcsoft").Cells(compa, j)
        Sheets("lcsoft_data_2").Cells(next_input, 10) = Sheets("lcsoft").Cells(sub_compa,
j)
        Sheets("lcsoft_data_2").Cells(next_input, 15) = "stream_sim"
        With Sheets("lcsoft_data_2").Cells(next_input, 15).Font
            .ThemeColor = xlThemeColorDark1
            .TintAndShade = 0
        End With
    Next j

```

```

    cut_row = Sheets("lcsoft_data_2").Range("h" & Rows.Count).End(xlUp).Offset(1,
0).row
    Sheets("int_data").Select
    [h84:l84].Copy
    Sheets("lcsoft_data_2").Select
    Cells(cut_row, 8).Select
    ActiveSheet.Paste

```

```

End If
Next i
End Sub

```

```

Sub del_lcsoft_results()
Application.ScreenUpdating = False
Application.DisplayAlerts = False
ThisWorkbook.Activate
Sheets("lcsoft results").Select
Range(Cells(5, 1), Cells(5000, 5000)).Delete
End Sub

```

```

Sub im_peis()
ThisWorkbook.Activate
Application.ScreenUpdating = False
Application.DisplayAlerts = False
wb1_name = ThisWorkbook.Name
pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
no_section = Sheets("lcsoft").TextBox3.Value
wb2_name = pj_name & "_" & case_number & "_lcsoft.xlsm"
wb2_name = "Bioethanol from cassava rhizome_base case_forpic.xlsm"
Sheets("lcsoft results").Select
Cells(10, 1) = "PEIs and Energy consumption"
Range(Cells(10, 1), Cells(10, 29)).Select
Call Module1.lcsoft_head
[c12].Select
ActiveCell = "Energy consumption"
    With Selection.Font
        .Bold = True
        .ThemeColor = xlThemeColorLight1
        .TintAndShade = 0
    End With
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorDark1
        .TintAndShade = -0.149998474074526
        .PatternTintAndShade = 0
    End With

```

Call Module1.create_grid

```
Cells(17 + no_section, 3).Select
ActiveCell = "Potential environmental impacts"
  With Selection.Font
    .Bold = True
    .ThemeColor = xlThemeColorLight1
    .TintAndShade = 0
  End With
  With Selection.Interior
    .Pattern = xlSolid
    .PatternColorIndex = xlAutomatic
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = -0.149998474074526
    .PatternTintAndShade = 0
  End With
Call Module1.create_grid
```

```
Windows(wb2_name).Activate
Sheets("results").Visible = True
Sheets("results").Select
last_results_row = Range("c" & Rows.Count).End(xlUp).row
[b2:c3].Copy
Windows(wb1_name).Activate
Sheets("lcoft results").Select
[c2].Select
ActiveSheet.Paste
[c2:d3].Select
Call Module1.create_grid
Windows(wb2_name).Activate
Sheets("results").Visible = True
Sheets("results").Select
[b5:h6].Copy
Windows(wb1_name).Activate
Sheets("lcoft results").Select
[c5].Select
ActiveSheet.Paste
Windows(wb2_name).Activate
Sheets("results").Visible = True
Sheets("results").Select
Range(Cells(8, 3), Cells(last_results_row, 8)).Copy
Windows(wb1_name).Activate
Sheets("lcoft results").Select
[c14].Select
ActiveSheet.Paste
Windows(wb2_name).Activate
```

```

Sheets("results").Visible = True
Sheets("results").Select
If last_results_row <= 25 Then
Range(Cells(4, 13), Cells(25, 38)).Copy
Else
Range(Cells(4, 13), Cells(last_results_row, 38)).Copy
End If
Windows(wb1_name).Activate
(Cells(22 + Sheets("lcsoft").TextBox3.Value, 3)).Select
ActiveSheet.Paste
    Set rng = Cells(19 + Sheets("lcsoft").TextBox3.Value, 3)
    With ActiveSheet.OLEObjects("Image1")
        .Top = rng.Top
        .Left = rng.Left
        .Width = rng.Width
    End With

Range(Cells(1, 13), Cells(1, 26)).EntireColumn.ColumnWidth = 8
Columns("c").AutoFit

End Sub

Sub im_cf()
ThisWorkbook.Activate
Application.ScreenUpdating = False
Application.DisplayAlerts = False
wb1_name = ThisWorkbook.Name
pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
no_section = Sheets("lcsoft").TextBox3.Value
wb2_name = pj_name & "_" & case_number & "_lcsoft.xlsm"
wb2_name = "Bioethanol from cassava rhizome_base case_forpic.xlsm"
Sheets("lcsoft results").Select
last_use_row = Range("c" & Rows.Count).End(xlUp).row

Cells(last_use_row + 9, 1) = "Carbon Footprint"
Range(Cells(last_use_row + 9, 1), Cells(last_use_row + 9, 29)).Select
Call Module1.lcsoft_head
Windows(wb2_name).Activate
Sheets("Carbon Footprint Results").Visible = True
Sheets("Carbon Footprint Results").Select
last_cf_row = Range("b" & Rows.Count).End(xlUp).row
If last_cf_row <= 35 Then
Range(Cells(4, 2), Cells(35, 28)).Copy
Else
Range(Cells(4, 2), Cells(last_cf_row, 28)).Copy
End If

```

```

Windows(wb1_name).Activate
Cells(last_use_row + 14, 2).Select
ActiveSheet.Paste
Cells(last_use_row + 13, 2).EntireRow.Delete
Cells(last_use_row + 12, 2).EntireRow.Delete

End Sub
Sub im_resource()
ThisWorkbook.Activate
Application.ScreenUpdating = False
Application.DisplayAlerts = False
wb1_name = ThisWorkbook.Name
pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
no_section = Sheets("lcsoft").TextBox3.Value
wb2_name = pj_name & "_" & case_number & "_lcsoft.xlsm"
wb2_name = "Bioethanol from cassava rhizome_base case_forpic.xlsm"
Sheets("lcsoft results").Select
last_use_row = Range("c" & Rows.Count).End(xlUp).row
Cells(last_use_row + 9, 1) = "List of resources"
Range(Cells(last_use_row + 9, 1), Cells(last_use_row + 9, 29)).Select
Call Module1.lcsoft_head
list_r = "re_" & Sheets("lcsoft").Cells(4, 2)
Windows(wb2_name).Activate
Sheets(list_r).Select
last_r_row = Range("a" & Rows.Count).End(xlUp).row
Range(Cells(41, 1), Cells(last_r_row, 4)).Copy
Windows(wb1_name).Activate
Sheets("lcsoft results").Select
Cells(last_use_row + 11, 3).Select
ActiveSheet.Paste
For r = 4 To no_section + 3
ThisWorkbook.Activate
r_sheet = "re_" & Sheets("lcsoft").Cells(r, 2)
s_name = Sheets("lcsoft").Cells(r, 2)
Windows(wb2_name).Activate
Sheets(r_sheet).Visible = True
Sheets(r_sheet).Select
Range(Cells(41, 5), Cells(last_r_row, 5)).Copy
ThisWorkbook.Activate
Sheets("lcsoft results").Select
next_col = Cells(last_use_row + 11, Columns.Count).End(xlToLeft).Offset(0,
1).Column
Cells(last_use_row + 11, next_col).Select
ActiveSheet.Paste
Cells(last_use_row + 11, next_col) = s_name
Cells(last_use_row + 11, next_col).Select

```



```
Call Module1.create_grid
Next r
```

```
End Sub
```

```
Sub tee()
Sheets("Icsoft results").Select
Range(Cells(1, 1), Cells(1, 12)).EntireColumn.ColumnWidth = 9
End Sub
```

Module "I3_ECON"

```
Sub first_clickecon()
Application.ScreenUpdating = False
Application.DisplayAlerts = False
```

```
Sheets("econ").Select
Range(Cells(8, 1), Cells(5000, 5000)).Delete
```

```
Sheets("int_data").Visible = True
Sheets("int_data").Select
[ac3:at4].Copy
Sheets("econ").Select
[b8].Select
ActiveSheet.Paste
Sheets("int_data").Visible = False
Sheets("econ").Select
```

```
End Sub
```

```
Sub link_econ_data()
```

```
wb1_name = ThisWorkbook.Name
pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
```

```
wb_path = ThisWorkbook.Path
software_path = Left(wb_path, Len(wb_path) - 16) &
"\Software\ECON\ECON.xlsm"
data_file_path = Left(wb_path, Len(wb_path) - 16) &
"\Software\ECON\econ_data\econ_data.xls"
new_data_file_name = "econ_" & pj_name & "_" & case_number
```

```

new_data_file_path = Left(wb_path, Len(wb_path) - 16) &
"\Software\ECON\econ_data\" & new_data_file_name & ".xls"
directory = Left(wb_path, Len(wb_path) - 16) & "\Software\ECON\econ_data"

```

```

If Len(Dir$(new_data_file_path)) > 0 Then Kill new_data_file_path

```

```

Application.Workbooks.Open (data_file_path)
wb2_name = ActiveWorkbook.Name
Sheets("project data").Range("PROJECTNAME") = pj_name
Sheets("project data").Range("CASENO") = case_number
Sheets("rawmaterial").Select

```

```

'raw material

```

```

Windows(wb1_name).Activate

```

```

Sheets("econ").Select

```

```

If [b14] <> "" Then

```

```

    Set to_find = [b:b].find(What:="PRODUCT", LookIn:=xlValues, _
    LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
    MatchCase:=False, SearchFormat:=False)

```

```

    If Not to_find Is Nothing Then data_row = to_find.row

```

```

    If data_row > 11 Then

```

```

        Range(Cells(14, 2), Cells(data_row - 2, 4)).Copy

```

```

        Windows(wb2_name).Activate

```

```

        Sheets("RAWMATERIAL").Select

```

```

        [b2].Select

```

```

        ActiveSheet.Paste

```

```

        k = 2

```

```

        Do While Cells(k, 2) <> ""

```

```

            Cells(k, 1) = k - 1

```

```

            k = k + 1

```

```

        Loop

```

```

        End If

```

```

End If

```

```

'Product

```

```

Windows(wb1_name).Activate

```

```

no_dp = Sheets("int_data").Range("no_dp")

```

```

Sheets("econ").Select

```

```

    Set to_find = [b:b].find(What:="PRODUCT", LookIn:=xlValues, _
    LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _
    MatchCase:=False, SearchFormat:=False)

```

```

    If Not to_find Is Nothing Then data_row = to_find.row

```

```

    Range(Cells(data_row + 3, 2), Cells(data_row + 2 + no_dp, 4)).Copy

```

```

    Windows(wb2_name).Activate

```

```

Sheets("PRODUCT").Select
[b2].Select
ActiveSheet.Paste
k = 2
Do While Cells(k, 2) <> ""
Cells(k, 1) = k - 1
k = k + 1
Loop

```

```

'Equipments
Windows(wb1_name).Activate

```

```

For i = 3 To 19
Windows(wb1_name).Activate

```

```

If Sheets("int_data").Cells(18, i) <> "" Then
e_row = Sheets("int_data").Cells(18, i)
no_e = Sheets("int_data").Cells(17, i)
no_col = Sheets("int_data").Cells(16, i)
sheet_equip = Sheets("int_data").Cells(20, i)
Sheets("econ").Select
Range(Cells(e_row + 2, 2), Cells(e_row + 1 + no_e, 1 + no_col)).Copy
Windows(wb2_name).Activate
Sheets(sheet_equip).Select
[a2].Select
ActiveSheet.Paste
End If
Next i
Windows(wb2_name).Activate
Application.DisplayAlerts = False
ActiveWorkbook.SaveAs Filename:=new_data_file_path
ActiveWorkbook.Close

```

```

Windows(wb1_name).Activate
Sheets("Start menu").Select

```

```

Application.Workbooks.Open (software_path)
wb2_name = ActiveWorkbook.Name
Windows(wb2_name).Activate

```

```

Sheets("Initialize ").Select
Sheets("Initialize ").TextBox3.Value = directory
Sheets("Initialize ").TextBox3.Text = directory
Sheets("Initialize ").TextBox4.Enabled = True
Sheets("Initialize ").TextBox4.Value = new_data_file_name & ".xls"

```

```
Sheets("Initialize ").TextBox4.Text = new_data_file_name & ".xls"
'Sheets("Initialize ").CommandButton6 = True
```

```
'Windows(new_data_file_name).Activate
'ActiveWorkbook.Close
Windows("econ").Activate
```

```
End Sub
```

```
Sub fill_out()
  With Selection.Interior
    .Pattern = xlSolid
    .PatternColorIndex = xlAutomatic
    .ThemeColor = xlThemeColorDark1
    .TintAndShade = -4.99893185216834E-02
    .PatternTintAndShade = 0
  End With
End Sub
```

```
Sub clear_data()
  Sheets("int_data").Visible = True
  Sheets("int_data").Select
  Range("C18:S18").ClearContents
  Sheets("int_data").Visible = False
  Sheets("econ").Select
End Sub
```

```
Sub record_path()

  pj_name = ThisWorkbook.Sheets("int_data").Range("pj_name")
  case_number = ThisWorkbook.Sheets("int_data").Range("case_number")
  data_file_name = "econ_" & pj_name & case_number
  path1 = ThisWorkbook.Path
  new_data_file_path = path1 & "\ECON\ECON_data_file\" & data_file_name &
  ".xls"
  Sheets("int_data").Range("path_econ") = new_data_file_path
End Sub
```

```
Sub del_old_datafile()
```

```
'add sub for delete old data wb
```

```
-----  
End Sub
```

```
Sub import_econ_summary()  
ThisWorkbook.Activate  
software_name = Sheets("Start menu").TextBox3.Value  
Windows(software_name).Activate  
ATCI = Range("TCIR").Value  
ATPC = Range("OP23C").Value  
ANP = Range("Etotal").Offset(13, 0).Value  
AROI = Range("ROR").Value  
APBP = Range("PBP").Value  
ANR = Range("NR").Value  
ANPV = Range("ANPW").Value  
DCFRA = Range("ADCFR").Value  
ThisWorkbook.Activate  
Sheets("ECON results").Select  
[c5] = Sheets("int_data").Range("pj_name")  
[c6] = Sheets("int_data").Range("case_number")  
Cells(9, 2) = ATCI  
Cells(9, 3) = ATPC  
Cells(9, 4) = ANP  
Cells(9, 5) = AROI  
Cells(9, 6) = APBP  
Cells(9, 7) = ANR  
Cells(9, 8) = ANPV  
Cells(9, 9) = DCFRA  
Range(Cells(10, 1), Cells(3000, 3000)).Delete  
End Sub
```

```
Sub im_capital_cost()  
Application.DisplayAlerts = False  
ThisWorkbook.Activate  
software_name = Sheets("Start menu").TextBox3.Value  
Windows(software_name).Activate  
Sheets("Capital Cost").Select  
pec = Sheets("Capital Cost").Range("PEC")  
ThisWorkbook.Activate  
last_row = Sheets("ECON results").Range("b" & Rows.Count).End(xlUp).row  
Sheets("ECON results").Select  
Cells(last_row + 2, 2) = "Capital Cost"  
Range(Cells(last_row + 2, 1), Cells(last_row + 2, 18)).Select  
Call Module1.econ_cut  
Cells(last_row + 4, 2) = "Purchase Equipment Cost"  
Cells(last_row + 4, 2).Font.Bold = True  
Cells(last_row + 4, 3) = pec
```

```

Range(Cells(last_row + 4, 2), Cells(last_row + 4, 3)).Select
Call Module1.create_grid
Windows(software_name).Activate
last_store_row = Sheets("Capital Cost").Range("h" & Rows.Count).End(xlUp).row
Range(Cells(10, 1), Cells(last_store_row, 8)).Copy
Cells(10, 1).Select

Selection.PasteSpecial Paste:=xlPasteValuesAndNumberFormats, Operation:= _
    xlNone, SkipBlanks:=False, Transpose:=False
Selection.Copy
ThisWorkbook.Activate
Sheets("ECON results").Select
Cells(last_row + 6, 2).Select
ActiveSheet.Paste
add_row = last_store_row - 10 + last_row + 6
Cells(add_row + 2, 2) = "Addmore"
[j8].Select
End Sub

Sub im_operating_cost()
Application.DisplayAlerts = False
ThisWorkbook.Activate
last_row = Sheets("ECON results").Range("b" & Rows.Count).End(xlUp).row
Sheets("ECON results").Select
Cells(last_row + 2, 2) = "Operating Cost"
Range(Cells(last_row + 2, 1), Cells(last_row + 2, 18)).Select
Call Module1.econ_cut
software_name = Sheets("Start menu").TextBox3.Value
Windows(software_name).Activate
Sheets("Operating Cost").Select
|b6:i10].Copy
ThisWorkbook.Activate
Sheets("ECON results").Select
Cells(last_row + 4, 2).Select
ActiveSheet.Paste
Windows(software_name).Activate
Sheets("Operating Cost").Select
last_store_row = Range("m" & Rows.Count).End(xlUp).row
Range(Cells(12, 2), Cells(last_store_row, 15)).Copy
Cells(12, 2).Select
Selection.PasteSpecial Paste:=xlPasteValuesAndNumberFormats, Operation:= _
    xlNone, SkipBlanks:=False, Transpose:=False
Selection.Copy
ThisWorkbook.Activate
Sheets("ECON results").Select
Cells(last_row + 10, 2).Select
ActiveSheet.Paste
add_row = last_store_row - 12 + last_row + 10

```

```

Cells(add_row + 2, 2) = "Addmore"
[j8].Select
End Sub
Sub im_economic_evaluation()
Application.DisplayAlerts = False
ThisWorkbook.Activate
software_name = Sheets("Start menu").TextBox3.Value
Sheets("ECON results").Select
last_row = Sheets("ECON results").Range("b" & Rows.Count).End(xlUp).row
Cells(last_row + 2, 2) = "Economic Evaluation"
Range(Cells(last_row + 2, 1), Cells(last_row + 2, 18)).Select
Call Module1.econ_cut
Windows(software_name).Activate
Sheets("Economic Evaluation").Select
Range("B7:L18").Copy
ThisWorkbook.Activate
Sheets("ECON results").Select
Cells(last_row + 4, 2).Select
ActiveSheet.Paste
Windows(software_name).Activate
Sheets("Economic Evaluation").Select
ActiveSheet.ChartObjects("CumulativeCashFlow").Activate
Application.CutCopyMode = False
ActiveChart.ChartArea.Copy
ThisWorkbook.Activate
Sheets("ECON results").Select
Cells(last_row + 17, 2).Select
ActiveSheet.Paste
[j8].Select
End Sub

```

```

Sub im_pie_chart()
Application.DisplayAlerts = False
ThisWorkbook.Activate
software_name = Sheets("Start menu").TextBox3.Value
Sheets("ECON results").Select
Range(Cells(11, 19), Cells(160, 19)).Select
    With Selection.Interior
        .Pattern = xlSolid
        .PatternColorIndex = xlAutomatic
        .ThemeColor = xlThemeColorLight1
        .TintAndShade = 0
        .PatternTintAndShade = 0
    End With
Cells(11, 20) = "PIE Chart"
Range(Cells(11, 20), Cells(11, 39)).Select
Call Module1.econ_cut

```

```
Windows(software_name).Activate  
Sheets("PIE Chart").Select  
[b13:r58].Copy  
ThisWorkbook.Activate  
Sheets("ECON results").Select  
Cells(13, 21).Select  
ActiveSheet.Paste  
[j8].Select  
End Sub
```

```
Sub im_sensitivity_analysis()  
Application.DisplayAlerts = False  
ThisWorkbook.Activate  
software_name = Sheets("Start menu").TextBox3.Value  
Sheets("ECON results").Select  
Cells(60, 20) = "Sensitivity Analysis"  
Range(Cells(60, 20), Cells(60, 39)).Select  
Call Module1.econ_cut  
Windows(software_name).Activate  
Sheets("Sensitivity Analysis").Select  
ActiveSheet.ChartObjects("Sensitivity").Activate  
Application.CutCopyMode = False  
ActiveChart.ChartArea.Copy  
ThisWorkbook.Activate  
Sheets("ECON results").Select  
Cells(62, 21).Select  
ActiveSheet.Paste  
[j8].Select  
End Sub
```


CURRICULUM VITAE

Name: Ms. Sawitree Kalakul

Date of Birth: May 12, 1988

Nationality: Thai

University Education:

2007-2011 Bachelor Degree of Engineering (Petrochemical and Polymeric materials), Faculty of Engineering, Silpakorn University, Thailand

Work Experience:

2010 Position: Student Internship
Company name: PTT Chemical PCL (Laboratory center)

Presentations:

1. Sawitree, K., Pomthong, M., Kitipat, S. and Rafiqul, G. (2013, April 23) Software Integration of Life Cycle Assessment and Economic Analysis for Process Evaluation. Paper presented at the 4rd Research Symposium on Petrochemical and Materials Technology and the 19th PPC Symposium on Petroleum, Petrochemicals, and Polymers, Bangkok, Thailand.