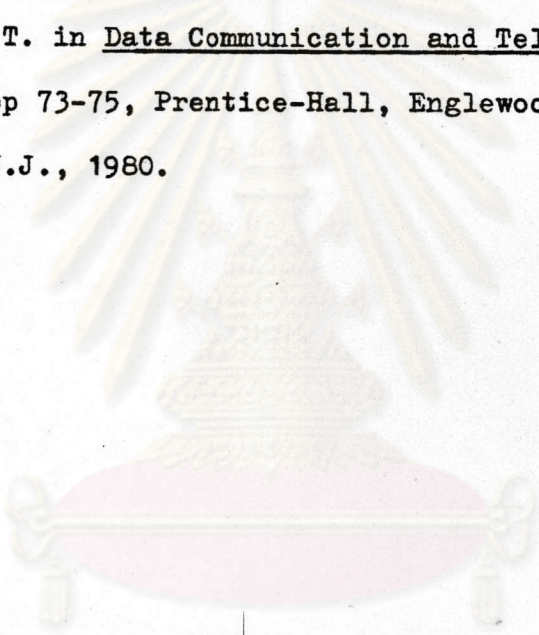


เอกสารอ้างอิง

1. Raphael, H. A. "Join micros into intelligent networks"
Electronic Design. 20 (11), (1975) : 52-57.
2. Weissberger, A. J. "Analysis of Multiple-Microprocessor
System Architecture" Computer Design. 16 (7),
(1977) : 151-163.
3. Herzog, J. H. "A Design Perspective for Real-Time Task
Control in Distributed Systems" IEEE Trans. on
Industrial Electronic. 30 (1), (1983) : 46-51.
4. Wyss, C. R. "A Conceptual Approach to Real-Time
Programming" Byte. 8 (5), (1983) : 452-462.
5. Fathi, E. T. and Krieger, M. "Multiple Microprocessor
System : What, Why and When" COMPUTER. 16 (3),
(1983) : 23-31.
6. Civeta, P. and Gonte, D. "An Experience with a Multimicro-
processor System" IEEE MICRO. 2 (2), (1982) :
38-50.
7. Fuller, S. H. "Multi-Microprocessors : An Overview and
Working Example" Proc. of the IEEE. 66 (2),
(1978) : 216-228.
8. Glover, J. R. and Tavora, Carlos. J. "A Design for the
Remote Link Unit-A Multiprocessor Remote Terminal"
IEEE Trans. on Industrial Electronics and Control
Instrumentation. 28 (2), (1981) : 82-90.

9. Libes, S. and Garetz, M. in Interfacing to S-100/IEEE 696 microcomputers. pp 255-274, OSBORNE/McGraw-Hill, Berkeley, California, 1981.
10. Woods, D. E. and Serafin, R. D. "A Multi-Microcomputer Based Distributed Front End Communications Subsystem for a Power Control Center" IEEE Trans. on Power Apparatus and Systems. 101 (1), (1982) : 180-184.
11. Silverman, G. "The Modular Multiprocessor-A model for Laboratory Instrument Design" IEEE MICRO. 2 (2), (1982) : 51-61.
12. Balph, T. and Black, J. "Multiprocessing could bring out a system's best" Electronic Design. 27 (7), (1982) : 219-224.
13. Peatman, J. B. in Digital Hardware Design. pp 319-325, McGraw-Hill Kogakusha. Tokyo, 1980.
14. Loewer, B. "The Z-80 in Parallel" Byte. 3 (6), (1978) : 60-63.
15. Polczynski, M. H. "Multiple uP Control System raises throughput without bus conflicts" Electronic Design. 27 (3), (1982) : 281-285.
16. Petriu, E. "N-Channel Asynchronous Arbiter Resolves Resources Allocation Conflicts" Computer Design. 19 (8), (1980) : 126-129.
17. Hojberg, K. S. "An Asynchronous Arbiter Resolves Resource Allocation Conflicts On a Random Priority Basis"

- Computer Design. 16 (8), (1977) : 120-123.
18. Osborne, A. and Kane, G. in 4 & 8-Bit Microprocessor Handbook. pp 7.1-7.17, OSBORNE/McGraw-Hill, Berkeley, California, 1981.
19. Rony, P. R. "Interfacing Fundamentals : Bidirectional I/O Using Two Semaphores" Computer Design. 20 (4), (1981) : 184-188.
20. Housley, T. in Data Communication and Teleprocessing. pp 73-75, Prentice-Hall, Englewood Cliffs, N.J., 1980.



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



ภาคผนวก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ભાગખુલાક બ.

```

1  REM  HOST COMPUTER SIMULATION PROGRAM
2  REM  FOR DEMONSTRATION; HOST ONLY DISPLAYS DATA FROM REMOTE
      TERMINAL ON CRT DISPLAY; THE DISPLAY IS UPDATED EVERY
      3 SECONDS, FIVE TIMES PER COMMAND.
3  REM  DATA RECEIVED (FROM FRONT-END COMMUNICATION SYSTEM)
      WILL BE STORED IN MEMORY LOCATION $C000-$C020 AND
      REPRESENTED BY VARIABLE NAMED "HE(1)-HE(32)" IN APPLE
      SOFT.
4  REM  HOST COMPUTER IS APPLE II plus WITH SERIAL INTERFACE
      CARD.

10  DIM  BEG (4), HE (37)
20  BEG(1) = 213 : BEG(2) = 82 : INPUT "REMOTE TERMINAL# ?" ;
      BEG(3) : INPUT "COMMAND BYTE?" ; BEG(4).
30  FOR I = 1 TO 4 : POKE 3109 + I, BEG(I) : NEXT I
40  FOR K = 1 TO 5
50  POKE 60, 38 : POKE 61, 12 : POKE 62, 41 : POKE 63, 12 :
      CALL-16128 : CALL-14015 : PR# 0 : REM REQUEST COMMAND OUTPUT
60  POKE 60, 00 : POKE 61, 12 : POKE 62, 36 : POKE 63, 12 :
      CALL-16128 : CALL-14019 : IN# 0 : REM DATA INPUT
70  FOR I = 1 TO 37 : HE(I) = PEEK(3071 + I) : NEXT I
80  HOME : PRINT TAB(12) "REMOTE TERMINAL " ; HE(36)
90  PRINT : PRINT TAB(3) "SENSOR# " ; TAB(13) ; "VALUE" ;
      TAB(23) ; "SENSOR# " ; TAB(33) ; "VALUE" : PRINT
100 FOR I = 1 TO 16 : PRINT TAB(5) ; I ; TAB(15) ; HE(I) ; TAB(25) ;
      I + 16 ; TAB(35) ; HE(I + 16) : NEXT I
110 PRINT TAB(5), "COMMAND BYTE =" ; HE(37)

```

```
120 FOR I = 1 TO 2250 : NEXT I : REM DISPLAY FOR 3 SECONDS  
130 NEXT K  
140 GOTO 20
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข.

```

;      HOST INTERFACE UNIT (HIU)
;
;      SIMULATION PROGRAM
;
;      FOR DEMONSTRATION OF
;
;      DISTRIBUTED MICROPROCESSOR SYSTEM
;
;      APPLICATION
;
;      JAN 84
;
;      *****

```

```

(OOF1)  USTAT: EQU F1 H
(OOFO)  USDA : EQU FO H
(1A10)  RTNA : EQU 1A10 H
(1A11)  RTNO : EQU 1A11 H
(1A12)  COMD : EQU 1A12 H
(1A00)  TIMB : EQU 1A00 H
(500A)  RT1DA: EQU 500A H
(5002)  RT1CD: EQU 5002 H
(1E00)  BYTE : EQU 1E00 H

```

```

ORG 2000 H

```

```

2000 319F1F      LD SP, 1F9F H ; Set Stack Pointer.
2003 3E0B      USARIN: LD A, 0B H ; Put USART in command.
2005 D3F1      OUT(USTAT), A ; mode
2007 3E40      LD A, 40 H ; reset it
2009 D3F1      OUT(USTAT), A
200B 3ECE      LD A, CE H ; set for 8 bit/2 stop bit/
200D D3F1      OUT(USTAT), A no parity/16X
200F 3E37      LD A, 37 H ; enable transmit and receive
2011 D3F1      OUT(USTAT), A

```

```

;

```

```

;

```

```

2013 0603    MAIN : LD B, 03 H    ; Set number of command
                                     ; byte from host
2015 11101A    LD DE, RTNA    ; point to data storing
                                     ; location
2018 CD3C20    CALL BCIN    ; waiting for command
201B 210250    LD HL, RT1CD    ; send command to RLU and
                                     ; get data for host
                                     ;
201E CD8C20    HOSED: CALL COMLOAD ; send command to RLU
2021 CDA220    CALL VACHK    ; look for data location
                                     ; of RT n
2024 0620     LD B, 20 H    ; and send to host
2026 CD5620    CALL BCOUT
2029 21001A    LD HL, TIMB    ; send dummy
202C 0603     LD B, 03 H
202E CD5620    CALL BCOUT
2031 21111A    LD HL, RTNO    ; return to host, RT
2034 0602     LD B, 02 H    ; and command
2036 CD5620    CALL BCOUT
2039 C31318    JP MAIN      ; waiting for new command
                                     ;
203C CD4D20    BCIN : CALL DATAIN ; waiting for data
203F FED5     CP D5 H      ; If it is "U" then
2041 20F9     JR NZ, BCIN    ; get command
2043 CD4D20    BCIN1: CALL DATAIN ; get 3 following byte
2046 E67F     AND 7F H      ; as host command and
2048 12      LD (DE), A      ; keep them at RTNA,

```



```

2049 13          INC DE          ; RTNO and COMD
204A 10F7        DJNZ BCIN1
204C C9          RET
204D DBF1  DATAIN: IN A, (USTAT)
204F CB4F        BIT 1, A
2051 28FA        JR Z? BCIN
2053 DBFO        IN A, (USDA)
2055 C9          RET

;

2056 CD6920 BCOUT : CALL DUMMY    ; Waiting for host to ready
2059 DBF1  COUT  : IN A, (USTAT)  ; check USART status
205B CB47        BIT 0, A        ;
205D 28FA        JR Z, COUT      ; loop until ready
205F 7E          LD A, (HL)      ;
2060 D3FO        OUT (USDA), A    ; then send character
2062 110100      LD DE, 01 H
2065 19          ADD HL, DE      ; send next data
2066 10F1        DJNZ COUT      ; any more? if no,
2068 C9          RET            ; return
2069 F5  DUMMY : PUSH AF        ; Save status
206A C5          PUSH BC
206B D5          PUSH DE
206C E5          PUSH HL
206D 11121A      LD DE, COMD    ; get host command
2070 21001E      LD HL, BYTE
2073 0603        LD B, 3
2075 1A  LOOP  : LD A, (DE)

```

```

2076 CD7806 CALL HEX7SG
2079 1B DEC DE
207A 10F9 DJNZ LOOP
207C DD21001E LD IX, BYTE
2080 0650 LD B, 50
2082 CD2406 SCAN: CALL SCAN 1 ; display it for 50 times
2085 10FB DJ NZ SCAN
2087 E1 POP HL
2088 D1 POP DE
2089 C1 POP BC
208A F1 POP AF
208B C9 RET

;

208C 210250 COMLD: LD HL, RT1 CD ; get address of command
208F 3A111A LD A, (RTNO) ; slot for RT 1
2092 47 LD B, A
2093 1002 ANO: DJNZ UPPER ; If it is RT 1 then
2095 1806 JR LOAD ; push command
2097 112800 UPPER: LD DE, 28 H ; If not, point to next
209A 19 ADD HL, DE ; RT
209B 18F6 JR ANO
209D 3A121A LOAD: LD A, (COMD) ; get command from host
20A0 77 LD (HL), A ; send to selected RT
20A1 C9 RET

;

20A2 210A50 VARCK: LD HL, RT1DA ; get address of data
20A5 31111A LD A, (RTNO) ; slot of RT 1

```

```
20A8 47          LD B, A
20A9 1001      TES:  DJNZ UP          ; if it is RT 1 then
20AB C9          RET                ; return
20AC 112800    UP:   LD DE, 28 H      ; if not, point to next
20AF 19        ADD HL, DE          ; RT
20B0 18F7      JR TES
```

;*****



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

; REMOTE LINK UNIT (RLU)
; SIMULATION PROGRAM
; FOR DEMONSTRATION OF
; DISTRIBUTED MICROPROCESSOR SYSTEM
; APPLICATION
; JAN 84
; *****

```

```

(5002) RT 1 CD: EQU 5002 H
(500A) RT 1 DA: EQU 500A H
(5282) RT17 CD: EQU 5282 H
(528A) RT17 DA: EQU 5284 H
(8150) LOBYE : EQU 8150 H
(8151) HIBYE : EQU 8151 H
(8100) BUFFER : EQU 8100 H
      BUFFER : DFS 32

```

ORG 1000 H

```

1000 310083      LD SP, 8300 H ; Set stack pointer.
1003 0E01  INIA  : LD C, 01 H   ; begin with RT 1.
1005 3E02      LD A, 02 H   ; set command code to 02.
1007 C30F10     JP POLL      ; poll data.
100A 0E01      LD C, 01 H   ; from RT 1.
100C CD3010  MAIN  : CALL CMRD   ; get command from shared
                        ; memory.
100F CD5010  POLL  : CALL POLSIM ; get data from RT and
                        ; push data in data buffer.
1012 C5        PUSH BC
1013 CD9010     CALL DALD   ; transfer 32 data bytes
                        ; from buffer to data slot.

```

```

1016  C1          POP BC
1017  3E10        LD A, 10H      ; max number of RT is 16.
1019  B9          CP C          ; Is last RT polled?
101A  28EE        JR NZ, MAIN   ; no, poll next, yes, go
                                   ; back to first RT.

101C  0C          INC C
101D  18ED        JR MAIN

;

;      CMRD ROUTINE:  read command from
;      command slot in shared memory
;      REMARK:  replace RT 1 CD
;      with RT 17 CD for RLU 2,
;      This program is for RLU 1.

      ORG 1030 H

1030  41          CMRD : LD B, C      ; get RT number.
1031  210250      LD HL, RT1CD     ; get starting address of
1034  112800      LD DE, 28 H     ; command slot.  For RT 1.
1037  1002      TEST1: DJNZ CONT1  ; If it is RT 1 then Return.
1039  77          LD (HL), A
103A  C9          RET
103B  19          CON1 : ADD HL,DE  ; If it is not, point to
                                   ; RT n.

103C  C33710     JP TEST 1
;      POLSIM:  simulate polling by generating 32
;      data bytes randomly, then load them in
;      BUFFER.

;
;      ORG 1050

```

```

1050 AF POLSIM: XOR A ; clear carry flag.
1051 210500 LD HL, 05 H ; do this routine 5 times.
1054 CD6010 POL 1 : CALL RANDOM ; generate 32 data.
1057 110100 LD DE, 01 H ; bytes randomly
105A ED52 SBC HL, DE ; repeat until time out.
105C 20F6 JR NZ, POL 1
105E C9 RET
ORG 1060 H
1060 0620 RANDOM: LD B, 20 H ; get number of byte to
; generate.
1062 110081 LD DE, BUFFER ; keep them at BUFFER
1065 C5 RAN 1 : PUSH BC
1066 CD7510 CALL BYRADM ; one byte of data is
1069 C1 POP BC ; generated.
106A 3A5081 LD A, (LOBYE) ; then
106D 12 LD (DE), A ; store it at BUFFER.
106E 13 INC DE
106F 10F4 DJNZ RAN 1 ; repeat until 32 bytes.
1071 C9 RET
;
ORG 1075
1075 3A5181 BYRADM: LD A, (HIBYE) ; B contain HI Byte.
1078 47 LD B, A ; A contain LO Byte.
1079 3A5081 LD A, (LOBYE) ; store new HI Byte from
107C 325181 LD (HIBYE), A ; LO Byte.
107F CB1F RRA ; carry = B8.
1081 CB18 RRB ; B = B8 B7...B1

```

```

1083  A8          XOR B          ; XOR with LO Byte.
1084  325081     LD (LOBYE), A ; store random byte in
1087  C9          RET              ; LO Byte.

;

;          DALD:  move data from buffer to
;          data slot in shared memory
;          REMARK:  replace RT1 DA with RT17DA
;          for RLU 2

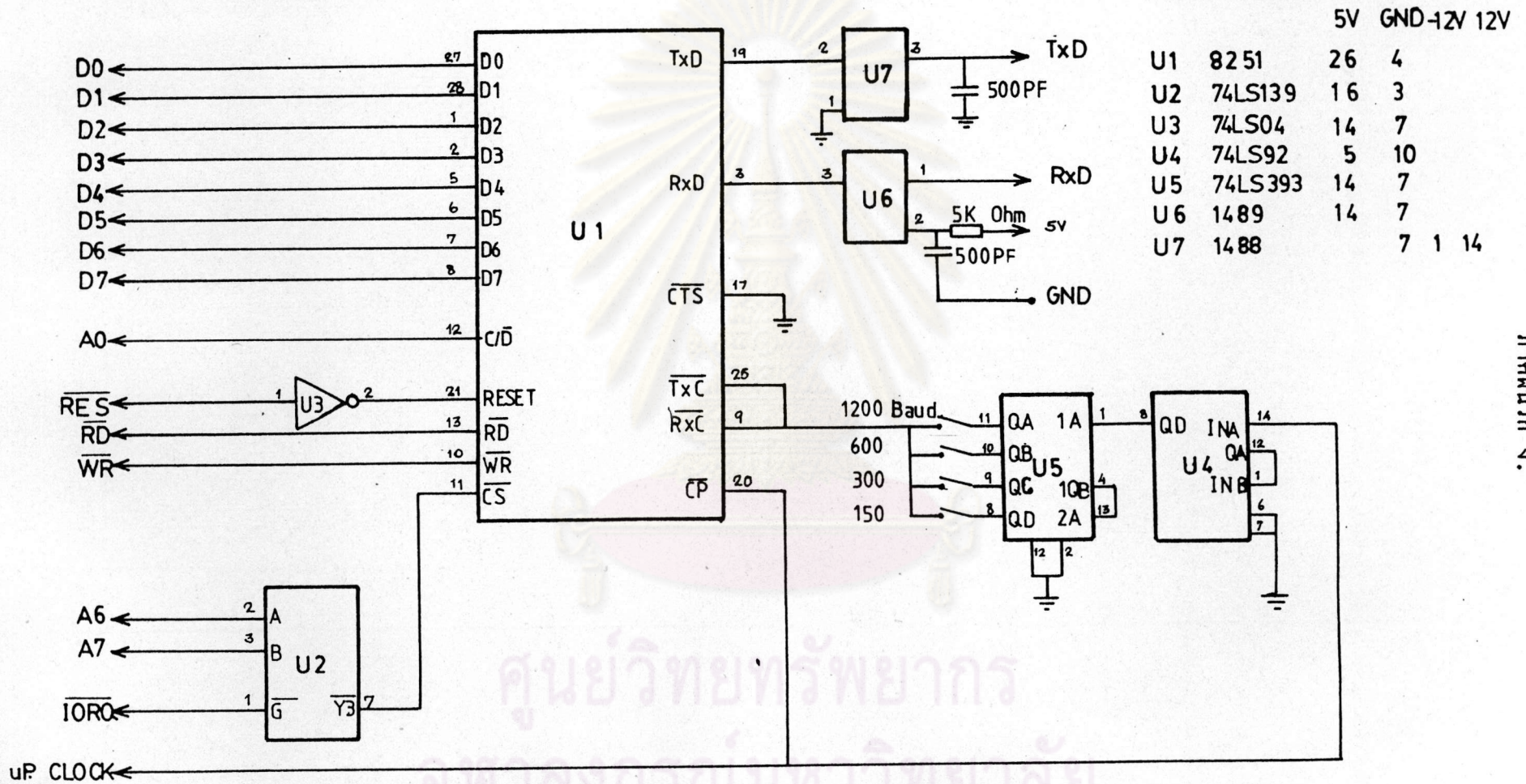
          ORG 1090
1090  41          DALD : LD B, C          ; get RT.
1091  210A50     LD HL, RT1DA ; point to data slot.
1094  112800     LD DE, 28H ; for RT n.
1097  100C      TEST2: DJNZ CONT 2 ;
1099  110081     LD DE, BUFFER ; move data from BUFFER.
109C  0620      LD B, 20 H ; to RT n data slot.
109E  1A          NEX : LD A, (DE)
109F  77          LD (HL), A
10A0  23          INC DE
10A1  13          INC HL
10A2  10FA      DJNZ NEX
10A4  C9          RET
10A5  19          CONT2: ADD HL, DE
10A6  C39710    JP TEST 2

```

```

;*****

```



ภาคผนวก 1.

วงจรสื่อสารข้อมูลแบบอนุกรมของ HIU.

รายการอุปกรณ์วงจรสื่อสารข้อมูลนุกรม

U1	8251	USART
U2	74LS139	Dual 2-To-4 Line Decoder/Demultiplexers
U3	74LS04	Hex Inverters
U4	74LS92	Devide-by-Twelve Counter
U5	74LS393	Dual 4-Bit Binary Counters
U6	1489	Quad Line Receiver
U7	1488	Quad Line Driver

ศูนย์วิทยพัทยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก จ.

รายการอุปกรณ์ วงจร ARBITER และวงจรอินเทอร์เฟสหน่วยความจำร่วม

U1	74LS139	Dual 2-to-4 Line Decoder/Demultiplexer
U2	74LS74	Dual D Positive-Edge-Triggered Flip-Flop
U3	74LS123	Retriggerable Monostable Multivibrator with Clear
U4	74LS123	Retriggerable Monostable Multivibrator with Clear
U5	74LS00	Quad 2-Input NAND Gate
U6-U13	74LS245	Octal Bus Transceivers
U9	6116	2K X 8 Bit Static Random Access Memory
U21	74LS30	8-Input NAND Gate
U22	7495	4-Bit Parallel Access Shift Register
U23	7425	Dual 4-Input Positive-NOR-Gates with strobe

2 X Resistor	composition	4.7K OHMS	1/4 W.
4 X Resistor	composition	10K OHMS	1/4 W.
2 X Capacitor	ceramic	1 PF	10 V.
1 X Capacitor	ceramic	33 PF	10 V.
1 X Capacitor	ceramic	50 PF	10 V.

ประวัติผู้เขียน

นายบวร ปกัศจรรย์ เกิดเมื่อวันที่ 28 กรกฎาคม 2500 ที่กรุงเทพมหานคร สำเร็จปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมไฟฟ้า จากสถาบันเทคโนโลยีพระจอมเกล้า วิทยาเขตธนบุรี เมื่อ พ.ศ. 2523 ปัจจุบันทำงานในตำแหน่งวิศวกรไฟฟ้า กองช่างสื่อสารและเครื่องช่วยการเดินอากาศ กรมการบินพาณิชย์



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย