

เอกสารอ้างอิง

1. กฤษดา วิศวธีรานนท์ และ คณะ, "การศึกษาสู่ทางการลงทุนผลิตภัณฑ์สาขาโทรศัพท์", รายงานการวิจัยเสนอต่อคณะกรรมการส่งเสริมการลงทุน, สำนักบริการวิชาการ จุฬาลงกรณ์มหาวิทยาลัย, 18 มกราคม 2530.
2. กฤษดา วิศวธีรานนท์, "คู่มือสายโทรศัพท์อิเล็กทรอนิกส์ขนาดเล็ก", รายงานการวิจัย หนังสือประดิษฐ์, จุฬาลงกรณ์มหาวิทยาลัย, 31 พฤษภาคม 2531.
3. อนุรักษ์ เกื้อนศิริ, "การสร้างขุมสายโทรศัพท์สาขาขนาดเล็กระบบอิเล็กทรอนิกส์", วิทยานิพนธ์ปริญาโทมหาบัณฑิต, ภาควิชาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย, 2528.
4. Bellamy J., DIGITAL TELEPHONY, John Willey & Sons, 1982.
5. Carbaugh D.H., Drew G.G., Ghiron H. and Hoover E.S.,
NO.1 ESS CALL PROCESSING,
The Bell System Technical Journal, Semtember 1986.
6. Hills M.T. , Kano S., PROGRAMMING ELECTRONIC SWITCHING SYSTEM
real time aspects and their language application,
Peter Peregrinus LTD, 1981.
7. เจริญชัย เจริญทุ่งเมือง, กฤษดา วิศวธีรานนท์,
"การพัฒนาโปรแกรมควบคุมสำหรับคู่สาขาโทรศัพท์อัตโนมัติโดยใช้ภาษาSTL",
การประชุมทางวิชาการวิศวกรรมไฟฟ้าครั้งที่ 10,
จุฬาลงกรณ์มหาวิทยาลัย, พฤศจิกายน 2530.
8. ไมโคร 80, "มาเขียนโปรแกรมโครงสร้างกันเถอะ", เขมิกอนคัคเตอร์ อิเล็กทรอนิกส์,
ฉบับที่ 55, หน้า 116-124, 2526.
9. Deitel H.M., An Introduction to Operating Systems,
Addison-Wesley, 1983.
10. Thom Hogam, CP/M USER GUIDE, Osborne/McGraw Hill, 1982.
11. Shaw A.C., The Logical Design of Operating Systems,
Prentice-Hall, 1974.

ภาคผนวก

รายละเอียดโปรแกรมภาษา STL

```

((((((((((((((((((((((((((((((((((((

```

```
state CALLED {0}
```

```

((((((((((((((((((((((((((((((((((((

```

```
state BUSY {1}
```

```

{ +--- X          }
{ +--- TONE (busy) }

```

```

if on_hook ($x) then
  cut ($icm,$x)
  tone ($icm,stop)
  close ($icm)
  goto IDEL {3}
end

```

```

if time_out then
  cut ($icm,$x)
  tone ($icm,stop)
  close ($icm)
  goto LOCK {2}
end

```

```

((((((((((((((((((((((((((((((((((((

```

```
state LOCK {2}
```

```
{ +--- X }
```

```

if on_hook ($x) then
  goto IDEL {3}
end

```

```

((((((((((((((((((((((((((((((((((((

```

```
state IDEL {3}
```

```
{ +--- X }
```

```

if $class < 1 then
  goto IDEL {3}
end

```

```

((((((((((((((((((((

```

```
if $FN = CALL_B then
```

```

if $FN_ex (> $x) then
  if $FN_ex.$state = IDEL {3} then
    if open ($icm) then
      ring ($x,on)
      $ex      := $FN_ex
      $ex.$ex  := $x
      $ex.$state := CALLED {3}
      goto CALL_B_RING {63}
    end
  end
end
end
end

```

```

((((((((((((((((((((

```

```

if off_hook ($x) then
  if open ($icm) then
    if open ($reg) then
      connect ($icm,$x)
      connect ($icm,$reg)
      tone ($icm,dial)
      goto DIAL {4}
    else
      close ($icm)
    end {reg}
  end {icm}
end {hook}

```

```

((((((((((((((((((((((((((((((((((((

```

```
state DIAL {4}
```

```

{ +--- X          }
{ +--- DTMF       }
{ +--- TONE (dial) }

```

```

if on_hook ($x) then
  cut ($icm,$x)
  cut ($icm,$reg)
  tone ($icm,stop)
  close ($reg)
  close ($icm)
  goto IDEL {3}
end

```

```

if dial ($x,$number) then
  tone ($icm,stop)
  goto COMMAND {5}

```

```

close ($reg)
close ($icm)
goto IDEL {3}
end

((((((((((((((((((((((((((((((((((((((((

state CALL_EX {10}

{ +--- X }

if $ex.$FN = FOLLOW then
    $ex := $ex.$FN_EX
end

((((((((((((((((((((

if $ex.$st = IDEL {3} then
    tone ($icm,ring_back)
    ring ($ex,on)
    $ex.$ex := $x
    $ex.$state := CALLED {0}
    goto RING_BACK {11}
end

tone ($icm,busy_tone)
flash ($x,clear)
goto SET_CALL_B {60}

((((((((((((((((((((((((((((((((((((((((

state RING_BACK {11}

{ +--- X }
{ +--- TONE (ring_back) }
{ }
{ +--- EX }

if off_hook ($ex) then
    ring ($ex,stop)
    tone ($icm,stop)
    connect ($icm,$ex)
    goto SPEAK {12}
end

if on_hook ($x) then
    ring ($ex,stop)
    cut ($icm,$x)

tone ($icm,stop)
close ($icm)
$ex.$state := IDEL {3}
goto IDEL {3}
end

if time_out then
    ring ($ex,stop)
    tone ($icm,busy)
    $ex.$state := 3 { idel }
    goto BUSY {1}
end

((((((((((((((((((((((((((((((((((((((((

state SAY {12}

{ +--- X }
{ +--- EX }

if on_hook ($x) then
    cut ($icm,$x)
    tone ($icm,busy)
    $ex.$icm := $icm
    $ex.$state := BUSY {1}
    goto IDEL {3}
end

if on_hook ($ex) then
    cut ($icm,$ex)
    tone ($icm,busy)
    $ex.$state := IDEL {3}
    goto BUSY {1}
end

((((((((((((((((((((((((((((((((((((((((

state CALL_CO {20}

{ +--- X }
{ +--- DTMF }

if $class < 2 then
    tone ($icm,busy)
    cut ($icm,$reg)
    close ($reg)
    goto BUSY {1}

```

```

end

if time_out then
    tone ($icm,busy)
    cut ($icm,$reg)
    close ($reg)
    goto BUSY {1}
end

((((((((((((((((((((((((((((((((((((((((

state COMMAND {5}

{ +--- X }
{ +--- DTMF }

if $number is_extension_name then goto CALL_EX {10} end
if $number is_command then goto ??? {each case} end
if $number too_less then goto NEXT {6} end
if $number error then
    tone ($icm,busy)
    cut ($icm,$reg)
    close ($reg)
    goto BUSY {1}
end

((((((((((((((((((((((((((((((((((((((((

state NEXT {6}

{ +--- X }
{ +--- DTMF }

if dial ($x,$temp) then
    concat ($number,$temp)
    goto COMMAND {5}
end

if time_out then
    tone ($icm,busy)
    cut ($icm,$reg)
    close ($reg)
    goto BUSY {1}
end

if on_hook ($x) then
    cut ($icm,$x)
    cut ($icm,$reg)

```

```

end
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
if search $co , $co.$state = CO_IDEL {30} then
    connect ($icm,$co)
    do_off_hook ($co)
    $last := ''
    $co.$state := CALLED {0}
    goto SEND {21}
end

tone ($icm,busy)
cut ($icm,$reg)
close ($reg)
goto BUSY {1}

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
state SEND {21}

{ +--- X }
{ +--- DTMF }
{ +--- CO }

if class < 3 then
    if $last = '0????????' then
        tone ($icm,busy)
        cut ($icm,$reg)
        cut ($icm,$co)
        do_on_hook ($co)
        close ($reg)
        $co.$state := CO_IDEL{30}
        goto BUSY {1}
    end
end

if class < 4 then
    if $last = '100????????' then
        tone ($icm,busy)
        cut ($icm,$reg)
        cut ($icm,$co)
        do_on_hook ($co)
        close ($reg)
        $co.$state := CO_IDEL{30}
        goto BUSY {1}
    end
end
end

```

```

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
if on_hook ($x) then
    cut ($icm,$x)
    cut ($icm,$co)
    cut ($icm,$reg)
    do_on_hook ($co)
    close ($reg)
    close ($icm)
    $co.$state := CO_IDEL{30}
    goto IDEL {3}
end

if time_out then
    cut ($icm,$reg)
    close ($reg)
    flash ($x,clear)
    goto CO_SAY {22}
end

if dial ($x,$temp) then
    do_DIAL ($co,$temp)
    concat ($last,$temp)
    goto SEND {21}
end

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
state CO_SAY {22}

{ +--- X }
{ +--- CO }

if flash ($x) then
    if open ($reg) then
        cut ($icm,$co)
        tone ($icm,dial2)
        connect ($icm,$reg)
        music ($co,on)
        goto HOLD_DIAL {94}
    end
end

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
if on_hook ($x) then
    cut ($icm,$x)
    cut ($icm,$co)

```


end

```

if class < 4 then
  if $number = '100???' then
    tone ($icm,busy)
    cut ($icm,$reg)
    cut ($icm,$co)
    do_on_hook ($co)
    close ($reg)
    $co.$state := CO_IDLE{30}
    goto BUSY {1}
  end
end
end

```

```

<<<<<<<<<<<<<<<<

```

```

cut ($icm,$reg)
close ($reg)
do_dial ($co,$last)
goto CO_SAY {22}

```

```

<<<<<<<<<<<<<<<<

```

```

state CALL_B {60}

```

```

{ +--- X          }
{ +--- TONE (busy) }

```

```

if flash ($x) then
  $FN := CALL_B
  $FN_EX := $ex
  tone ($icm,ack)
  goto ACK {1}
end

```

```

if on_hook ($x) then
  cut ($icm,$x)
  tone ($icm,stop)
  close ($icm)
  goto IDEL {3}
end

```

```

if time_out then
  cut ($icm,$x)
  tone ($icm,stop)
  close ($icm)
  goto LOCK {2}
end

```

```

<<<<<<<<<<<<<<<<

```

```

state CANCEL {61}

```

```

{ +--- X          }
{ +--- DTMF       }

```

```

$FN := 0
tone ($icm,ack)
cut ($icm,$reg)
close ($reg)
goto ACK {1}

```

```

<<<<<<<<<<<<<<<<

```

```

state CALL_B_RING {63}

```

```

{ +--- X          }
{                }
{ +--- EX         }

```

```

if off_hook ($x) then
  $FN := 0
  ring ($x,stop)
  ring ($ex,on)
  connect ($icm,$ring_back)
  connect ($icm,$x)
  goto RING_BACK {11}
end

```

```

if off_hook ($ex) then
  ring ($x,stop)
  close ($icm)
  $ex.$state := ?IDEL
  goto ?IDEL
end

```

```

if time_out then
  $FN := 0
  ring ($x,stop)
  close ($icm)
  $ex.$state := ?IDEL
  goto ?IDEL
end

```

```

<<<<<<<<<<<<<<<<

```



```
state SET_FOLLOW {70}
```

```
{ +--- X      }
{ +--- DTMF  }
```

```
if dial ($x,$temp) then
  concat ($number,$temp)
```

```
  if $number is_extension_name then
    tone ($icm,ack)
    cut  ($icm,$reg)
    close ($reg)
    $FN := FOLLOW_ME
    $FN_EX := $ex
    goto ACK {1}
  end
```

```
  if $number too_less then
    goto SET_FOLLOW
```

```
  if $number error then
    tone ($icm,busy)
    cut  ($icm,$reg)
    close ($reg)
    goto BUSY {1}
  end
```

```
end
```

```
if time_out then
  tone ($icm,busy)
  cut  ($icm,$reg)
  close ($reg)
  goto BUSY {1}
end
```

```
if on_hook ($x) then
  cut ($icm,$x)
  cut ($icm,$reg)
  close ($reg)
  close ($icm)
  goto IDEL {3}
end
```

```
<<<<<<<<<<<
```

```
state HOLD_BUSY {91}
```

```
{ +--- X      }
{ +--- TONE (busy2) }
{          }
{ +--- CO     }
{ +--- music  }
```

```
if flash ($x) then
  music ($co,stop)
  tone ($icm,stop)
  connect ($icm,$co)
  goto CO_SAY {22}
end
```

```
if on_hook ($x) then
  music ($co,stop)
  tone ($icm,stop)
  cut  ($icm,$x)
  close ($icm)
  do_on_hook ($co)
  $co.$state := CO_IDLE {30}
  goto IDEL {3}
end
```

```
if time_out then
  music ($co,stop)
  tone ($icm,stop)
  cut  ($icm,$x)
  close ($icm)
  do_on_hook ($co)
  $co.$state := CO_IDLE {30}
  goto LOCK {1}
end
```

```
<<<<<<<<<<<
```

```
state HOLD_IDLE {93}
```

```
93n0:                  ret
```

```
<<<<<<<<<<<
```

```
state HOLD_DIAL {94}
```

```
{ +--- X      }
{ +--- DTMF  }
```



```

close ($reg)
goto HOLD_BUSY {91}
end

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

state HOLD_NEXT {96}

{ +--- X      }
{ +--- DTMF   }
{             }
{ +--- CO     }
{ +--- music  }

if dial ($x,$temp) then
  concat ($number,$temp)
  goto HOLD_COMMAND {95}
end

if time_out then
  tone ($icm,busy2)
  cut ($icm,$reg)
  close ($reg)
  goto HOLD_BUSY {91}
end

if on_hook ($x) then
  music ($co,stop)
  cut ($icm,$x)
  cut ($icm,$reg)
  close ($reg)
  close ($icm)
  do_on_hook ($co)
  $co.$state := CO_IDEL {30}
  goto IDEL {3}
end

if flash ($x) then
  music ($co,stop)
  cut ($icm,$reg)
  close ($reg)
  connect ($icm,$co)
  goto CO_SAY {22}
end

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

```

```

state HOLD_CALL_EX {100}

{ +--- X      }
{ +--- DTMF   }
{             }
{ +--- CO     }
{ +--- music  }

if $ex.$FN = FOLLOW then
  $ex := $ex.$FN_EX
end

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

cut ($icm,$reg)
close ($reg)

if $ex.$state = IDEL {3} then
  tone ($icm,ring_back2)
  ring ($ex,on)
  $ex.$ex := $x
  $ex.$state := CALLED {0}
  goto HOLD_RING_BACK {101}
end

tone ($icm,busy2)
goto HOLD_BUSY {91}

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

state HOLD_RING_BACK {101}

{ +--- X      }
{ +--- TONE (ring_back2) }
{             }
{ +--- EX     }
{             }
{ +--- CO     }
{ +--- music  }

if off_hook ($ex) then
  ring ($ex,stop)
  tone ($icm,stop)
  connect ($icm,$ex)
  goto HOLD_SAY {102}
end

```

```
if on_hook ($x) then  
  music ($co,stop)  
  ring ($ex,stop)  
  tone ($icm,stop)  
  cut ($icm,$x)  
  close ($icm)  
  do_on_hook ($co)  
  $co.$state := CO_IDEL {30}  
  $ex.$state := IDEL {3}  
  goto IDEL {3}  
end
```

```
if time_out then  
  ring ($ex,stop)  
  tone ($icm,busy2)  
  $ex.$state := IDEL {3}  
  goto HOLD_BUSY {90}  
end
```

```
if flash ($x) then  
  music ($co,stop)  
  ring ($ex,stop)  
  tone ($icm,stop)  
  connect ($icm,$co)  
  $ex.$state := IDEL {3}  
  goto CO_SAY {22}  
end
```

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

st_102 { hold_speak }

```
{ +--- X }  
{ +--- EX }  
{ }  
{ +--- CO }  
{ +--- music }
```

```
if on_hook ($x) then  
  music ($co,stop)  
  cut ($icm,$x)  
  connect ($icm,$co)  
  flash ($ex,clear)  
  $ex.$icm := $icm  
  $ex.$co := $co  
  $ex.$state := CO_SAY {22}  
  goto IDEL {3}  
end
```

```
if on_hook ($ex) then  
  music ($co,stop)  
  cut ($icm,$ex)  
  connect ($icm,$co)  
  $ex.$state := IDEL {3}  
  goto CO_SAY {22}  
end
```

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

```
state CALL_PU {120}  
  
{ +--- X }  
{ +--- DTMF }
```

```
if dial ($x,$temp) then  
  concat ($number,$temp)  
  if $number is_extension_name then  
    cut ($icm,$reg)  
    close ($reg)  
    goto PU {121}  
  end
```

```
if $number too_less then  
  goto CALL_PU  
end
```

```
if $number error then  
  tone ($icm,busy)  
  cut ($icm,$reg)  
  close ($reg)  
  goto BUSY {1}  
end  
end
```

```
if time_out then  
  tone ($icm,busy)  
  cut ($icm,$reg)  
  close ($reg)  
  goto BUSY {1}  
end
```

```
if on_hook ($x) then  
  cut ($icm,$x)  
  cut ($icm,$reg)  
  close ($reg)  
  close ($icm)
```

```
goto IDEL {3}  
end
```

```
((((((((((((((((((((((((((((((((((((((((
```

```
state PU {121}
```

```
{ +---- X }
```

```
if $ex.$state = CO_INPUT {31} then  
ring ($ex,stop)  
$icm := $ex.$icm  
$co := $ex.$co  
$ex.$state := IDEL {3}  
goto CO_INPUT {31}  
end
```

```
if $ex.$state = CALLED {0} then  
if $ex.$ex.$state = RING_BACK {11} then  
ring ($ex,stop)  
$ex.$state := IDEL {3}  
$ex.$ex := $x  
$ex := $ex.$ex  
goto CALLED {0}  
end  
end
```

```
if $ex.$state = CALLED {0} then  
if $ex.$state = HOLD_RB {101} then  
ring ($ex,stop)  
$ex.$state := IDEL {3}  
$ex.$ex := $x  
$ex := $ex.$ex  
goto CALLED {0}  
end  
end
```

```
((((((((((((((((((((((((((((((((((((((((
```



ประวัติผู้เขียน

นายเจริญชัย เจริญทั้งเมือง เกิดเมื่อวันที่ 8 กันยายน 2508 ที่กรุงเทพมหานคร
ได้รับปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมไฟฟ้า จากจุฬาลงกรณ์มหาวิทยาลัย เมื่อปี
2528 และเข้าศึกษาต่อปริญญาโทบัณฑิต ที่ภาควิชาวิศวกรรมไฟฟ้า ในปีเดียวกัน