

```
include(sec1.m4) dnl
include(sec2.m4) dnl
include(sec3.m4) dnl
include(sec4.m4) dnl
```

```
.globl _acc80
.globl _cflas
.globl _pc80
.globl _bc80
.globl _sp80
```

```
// for testins
```

```
.globl _a80
.globl _b80
.globl _c80
.globl _d80
.globl _e80
.globl _h80
.globl _l80
.globl _flas80
```

```
.globl _sim
.globl _ram
```

```
.data
.even
af80:
_a80:
_acc80:
    .byte 0
_flas80:
_cflas:
    .byte 0
bc80:
_bc80:
_c80:
    .byte 0
_b80:
    .byte 0
de80:
_e80:
    .byte 0
_d80:
    .byte 0
_hl80:
hl80:
_l80:
    .byte 0
_h80:
    .byte 0
_sp80:
sp80:
    .word 0
_pc80:
```

pc80:
 .word 0

.text
.even

_sim:

 call csv0 // saves registers from callins c program

 ldb rh3,_flas80
 ldb r13,_a80
 ldb rh4,_b80
 ldb r14,_c80
 ldb rh5,_d80
 ldb r15,_e80
 ldb rh6,_h80
 ldb r16,_l80
 ld r7,sp80
 ld r9,_ram
 add r7,r9
 ld r2,pc80
 add r2,r9

// first

 clr r1
 ldb r11,*r2
 add r1,r1
 ld r12,inst(r1)
 jp *r12

leave:

 ldb _flas80,rh3
 ldb _a80,r13
 ldb _b80,rh4
 ldb _c80,r14
 ldb _d80,rh5
 ldb _e80,r15
 ldb _h80,rh6
 ldb _l80,r16
 sub r7,r9
 ld sp80,r7
 sub r2,r9
 ld pc80,r2

 jp cret0

add8(b)
add8(c)
add8(d)
add8(e)
add8(h)
add8(l)
add8(m)
add8(a)
adc8(b)
adc8(c)
adc8(d)
adc8(e)
adc8(h)
adc8(l)
adcm
adc8(a)
sub8(b)
sub8(c)
sub8(d)
sub8(e)
sub8(h)
sub8(l)
sub8(m)
sub8(a)
sbb8(b)
sbb8(c)
sbb8(d)
sbb8(e)
sbb8(h)
sbb8(l)
sbbm
sbb8(a)
ana8(b)
ana8(c)
ana8(d)
ana8(e)
ana8(h)
ana8(l)
ana8(m)
ana8(a)
xra8(b)
xra8(c)
xra8(d)
xra8(e)
xra8(h)
xra8(l)
xra8(m)
xra8(a)
ora8(b)
ora8(c)
ora8(d)
ora8(e)
ora8(h)
ora8(l)
ora8(m)
ora8(a)

cmp8(b)
cmp8(c)
cmp8(d)
cmp8(e)
cmp8(h)
cmp8(l)
cmp8(m)
cmp8(a)

con(r,c)
con(r,m)
con(r,nc)
con(r,nz)
con(r,P)
con(r,Pe)
con(r,Po)
con(r,z)
ret8:

inc r7,\$1
ldb r12,*r7
inc r7,\$1
ldb rh2,*r7
add r2,r9
thread

con(j,c)
con(j,m)
con(j,nc)
con(j,nz)
con(j,P)
con(j,Pe)
con(j,Po)
con(j,z)
jmp8:

inc r2,\$1
ldb r10,*r2
inc r2,\$1
ldb rh0,*r2
ld r2,r0
add r2,r9
thread

con(c,c)
con(c,m)
con(c,nc)
con(c,nz)
con(c,P)
con(c,Pe)
con(c,Po)
con(c,z)
call8:

ea2
inc r2,\$1
sub r2,r9
ldb *r7,rh2
dec r7,\$1
ldb *r7,r12
dec r7,\$1
ld r2,r0
thread

rst(0)
rst(1)
rst(2)
rst(3)
rst(4)
rst(5)
rst(6)

```

rst(7)
POP8(b)
POP8(d)
POP8(h)
POP8(PSW)
push8(b)
push8(d)
push8(h)
push8(PSW)
im(adi,addb)
aci8:
    inc    r2,#1
    ldb    r11,*r2
    loadflas
    adcb    r13,r11
    storeflas
    inc    r2,#1
    thread
im(sui,subb)
sbi8:
    inc    r2,#1
    ldb    r11,*r2
    loadflas
    sbcb    r13,r11
    storeflas
    inc    r2,#1
    thread
im(ani,andb)
im(xri,xorb)
im(ori,orb)
im(cpi,cpb)
xthl8:
    inc    r7,#1
    exb    r16,*r7
    inc    r7,#1
    exb    r16,*r7
    dec    r7,#2
    inc    r2,#1
    thread
pchl8:
    ld     r2,r9
    add    r2,r6
    thread
xchs8:
    ex     r5,r6
    inc    r2,#1
    thread
sphl8:
    ld     r7,r9
    add    r7,r6
    inc    r2,#1
    thread
di8:
ei8:
out8:
in8:

```


thread
badcod:

nop8:

nop
pc1
thread

lxi(b)
lxi(d)
lxi(h)
lxi(sp)
stax(b)
stax(d)
ldax(b)
ldax(d)
inx(b)
inx(d)
inx(h)
inx(sp)
dex(b)
dex(d)
dex(h)
dex(sp)
inr(b)
inr(c)
inr(d)
inr(e)
inr(h)
inr(l)
inr(m)
inr(a)
der(b)
der(c)
der(d)
der(e)
der(h)
der(l)
der(m)
der(a)
dad(b)
dad(d)
dad(h)
dad(sp)
mvi(b)
mvi(c)
mvi(d)
mvi(e)
mvi(h)
mvi(l)
mvi(m)
mvi(a)
rlc
rrc
ral
rar
shld
lhld
daa
cma

sta
stc
lda
cmc

```
mov(b,b)
mov(b,c)
mov(b,d)
mov(b,e)
mov(b,h)
mov(b,l)
mov(b,m)
mov(b,a)
mov(c,b)
mov(c,c)
mov(c,d)
mov(c,e)
mov(c,h)
mov(c,l)
mov(c,m)
mov(c,a)
mov(d,b)
mov(d,c)
mov(d,d)
mov(d,e)
mov(d,h)
mov(d,l)
mov(d,m)
mov(d,a)
mov(e,b)
mov(e,c)
mov(e,d)
mov(e,e)
mov(e,h)
mov(e,l)
mov(e,m)
mov(e,a)
mov(h,b)
mov(h,c)
mov(h,d)
mov(h,e)
mov(h,h)
mov(h,l)
mov(h,m)
mov(h,a)
mov(l,b)
mov(l,c)
mov(l,d)
mov(l,e)
mov(l,h)
mov(l,l)
mov(l,m)
mov(l,a)
mov(m,b)
mov(m,c)
mov(m,d)
mov(m,e)
mov(m,h)
mov(m,l)
hlt
mov(m,a)
```

```
mov(a,b)
mov(a,c)
mov(a,d)
mov(a,e)
mov(a,h)
mov(a,l)
mov(a,m)
mov(a,a)
```

```

.data
.even
inst: .word      nop8
      .word 1xib8      | 1
      .word staxb8     | 2
      .word inxb8      | 3
      .word inrb8      | 4
      .word dcrb8      | 5
      .word mvib8      | 6
      .word rlc8       | 7
      .word badcod     | 8
      .word dadb8      | 9
      .word ldaxb8     | 10
      .word dcxb8      | 11
      .word inrc8      | 12
      .word dcrcl8     | 13
      .word mvic8      | 14
      .word rrc8       | 15
      .word badcod     | 16
      .word lxic8      | 17
      .word staxd8     | 18
      .word inxd8      | 19
      .word inrd8      | 20
      .word dcrd8      | 21
      .word mvid8      | 22
      .word ral8       | 23
      .word badcod     | 24
      .word dadd8      | 25
      .word ldaxd8     | 26
      .word dcxd8      | 27
      .word inre8      | 28
      .word dcre8      | 29
      .word mvie8      | 30
      .word rar8       | 31
      .word badcod     | 32
      .word lxih8      | 33
      .word shld8      | 34
      .word inxh8      | 35
      .word inrh8      | 36
      .word dcrh8      | 37
      .word mvih8      | 38
      .word daa8       | 39
      .word badcod     | 40
      .word dadh8      | 41
      .word lhld8      | 42
      .word dcxh8      | 43
      .word inrl8      | 44
      .word dcr18      | 45
      .word mvl8       | 46
      .word cma8       | 47
      .word badcod     | 48
      .word lxisp8     | 49
      .word sta8       | 50
      .word inxsp8     | 51
      .word inrm8      | 52
      .word dcrn8      | 53

```

.word mvim8	54
.word stc8	55
.word badcod	56
.word dadsp8	57
.word lda8	58
.word dcxsp8	59
.word inra8	60
.word dcra8	61
.word mvia8	62
.word cmc8	63
.word movbb8	64
.word movbc8	65
.word movbd8	66
.word movbe8	67
.word movbh8	68
.word movbl8	69
.word movbm8	70
.word movba8	71
.word movcb8	72
.word movcc8	73
.word movcd8	74
.word movce8	75
.word movch8	76
.word movcl8	77
.word movcm8	78
.word movca8	79
.word movdb8	80
.word movdc8	
.word movdd8	
.word movde8	
.word movdh8	
.word movdl8	
.word movdm8	
.word movda8	
.word moveb8	
.word movec8	
.word moved8	90
.word movee8	
.word moveh8	
.word movel8	
.word movem8	
.word movea8	
.word movhb8	
.word movhc8	
.word movhd8	
.word movhe8	
.word movhh8	100
.word movhl8	
.word movhm8	
.word movha8	
.word movlb8	
.word movlc8	
.word movld8	
.word movle8	
.word movlh8	
.word movll8	

.word movlm8	110
.word movla8	
.word movmb8	
.word movmc8	
.word movmd8	
.word movme8	
.word movmh8	
.word movml8	
.word hlt8	
.word movma8	
.word movab8	120
.word movac8	
.word movad8	
.word movae8	
.word movah8	
.word mova18	
.word movam8	
.word movaa8	
.word addb8	
.word addc8	
.word addd8	
.word adde8	
.word addh8	
.word addl8	
.word addm8	
.word adda8	
.word adcb8	-120
.word adcc8	
.word adcd8	
.word adce8	
.word adch8	
.word adcl8	
.word adcm8	
.word adca8	
.word subb8	
.word subc8	
.word subd8	-110
.word sube8	
.word subh8	
.word subl8	
.word subm8	
.word suba8	
.word sbbb8	
.word sbbc8	
.word sbbd8	
.word sbbe8	
.word sbbh8	-100
.word sbb18	
.word sbbm8	
.word sbb8	
.word anab8	
.word anac8	
.word anad8	
.word anae8	
.word anah8	
.word anal8	

.word anam8	1 -90
.word anaa8	
.word xrab8	
.word xrac8	
.word xrad8	
.word xrae8	
.word xrah8	
.word xral8	
.word xram8	
.word xraa8	
.word orab8	1 -80
.word orac8	
.word orad8	
.word orae8	
.word orah8	
.word oral8	
.word oram8	
.word oraa8	
.word cmpb8	
.word cmpc8	
.word cmpd8	1 -70
.word cmpe8	
.word cmp8	
.word cmpl8	
.word cmpm8	
.word cmpa8	
.word rnz8	
.word rorb8	
.word jnz8	
.word jmp8	
.word cnz8	1 -60
.word pushb8	
.word adi8	
.word rst08	
.word rz8	
.word ret8	
.word jz8	
.word badcod	
.word cz8	
.word call8	
.word aci8	1 -50
.word rst18	
.word rnc8	
.word popd8	
.word jnc8	
.word out8	
.word cnc8	
.word pushd8	
.word sui8	
.word rst28	
.word rc8	1 -40
.word badcod	
.word jc8	
.word in8	
.word cc8	
.word badcod	

.word sbi8	
.word rst38	
.word rpo8	
.word poph8	
.word jpo8	1 -30
.word xth18	
.word cpo8	
.word pushh8	
.word ani8	
.word rst48	
.word rpe8	
.word pch18	
.word jpe8	
.word xchs8	
.word cpe8	1 -20
.word badcod	
.word xri8	
.word rst58	
.word rpe8	
.word poppsw8	
.word jpe8	
.word di8	
.word cpe8	
.word pushpsw8	
.word ori8	1 -10
.word rst68	
.word rm8	
.word sph18	
.word jm8	
.word ei8	
.word cm8	
.word badcod	
.word cpi8	
.word rst78	1 -1

```

changequote dnl
changequote([,]) dnl
define(pc1,[inc r2,[\$]1]) dnl
define(pc2,[inc r2,[\$]2]) dnl
define(pc3,[inc r2,[\$]3]) dnl
define(ea1,[ld r1[,]r6
      add      r1[,]r9]) dnl
define(mth,[\$1 r13,*r1]) dnl
define(thread,[/* THREAD */
//JP      leave
      clr      r1
      ldb      r11,*r2
      add      r1,r1
      ld       r12,inst(r1)
      JP       *r12
/* end */]) dnl
define(ea2,[inc r2,[\$]1
      ldb      r11,*r2
      inc      r2,[\$]1
      ldb      rh0,*r2
      add      r1,r9]) dnl
define(add8,[slm(add,[\$]1)]) dnl
define(adc8,[slm(adc,[\$]1)]) dnl
define(sub8,[slm(sub,[\$]1)]) dnl
define(sbb8,[slm(sbb,[\$]1)]) dnl
define(ana8,[slm(ana,[\$]1)]) dnl
define(xra8,[slm(xra,[\$]1)]) dnl
define(ora8,[slm(ora,[\$]1)]) dnl
define(cmp8,[slm(cmp,[\$]1)]) dnl
define(add_op,[addb]) dnl
define(adc_op,[adcb]) dnl
define(sub_op,[subb]) dnl
define(sbb_op,[sbb]) dnl
define(ana_op,[andb]) dnl
define(xra_op,[xorb]) dnl
define(ora_op,[orb]) dnl
define(cmp_op,[cpb]) dnl
define(slm,[ifelse(\$2,m,mm,m)ac1(\$1,[\$]2)]) dnl
define(mac1,[\$1\$28:
      loadflag
      \$1_op  a_res,$2_res
      \$1_flagstore
      pc1
      thread]) dnl
define(mmac1,[\$1\$28:
      ea1
      loadflag
      mth(\$1_op)
      \$1_flagstore
      pc1
      thread]) dnl
define(loadflag,[ldctlb flag,rh3]) dnl
define(storeflag,[ldctlb      rh3,flag]) dnl
define(sbbm,[sbbm8:
      ea1
      ldb      r10,*r1

```

```

loadflag
sbc b    r13,r10
storeflag
pc1
thread)) dnl
define(adcm,[adcm8:
    eal
    ldb    r10,*r1
    loadflag
    adcb    r13,r10
    storeflag
    pc1
    thread)) dnl
define(add_flagstore,[storeflag]) dnl
define(adc_flagstore,[storeflag]) dnl
define(sub_flagstore,[storeflag]) dnl
define(sbb_flagstore,[storeflag]) dnl
define(ana_flagstore,[ldctlb    rh3,flag
    andb    rh3,[10x7f]) dnl
define(xra_flagstore,[ldctlb    rh3,flag
    andb    rh3,[10x3f]) dnl
define(ora_flagstore,[xra_flagstore]) dnl
define(cmp_flagstore,[storeflag]) dnl
define(inr_flagstore,[rlcb    rh3,[11
    storeflag]) dnl
define(dcr_flagstore,[inr_flagstore]) dnl
define(carryin,[rlcb    rh3,[11]) dnl
define(carryout,[rrcb    rh3,[11]) dnl
define(adi_flagstore,[add_flagstore]) dnl
define(aci_flagstore,[adc_flagstore]) dnl
define(sui_flagstore,[sub_flagstore]) dnl
define(sbi_flagstore,[sbb_flagstore]) dnl
define(ani_flagstore,[ana_flagstore]) dnl
define(xri_flagstore,[xra_flagstore]) dnl
define(ori_flagstore,[ora_flagstore]) dnl
define(cpi_flagstore,[cmp_flagstore]) dnl

```

```

chansequote dnl
chansequote([,]) dnl
define(con,[#1$28:
    ldctlb    flagg,rh3
    JP        ifelse($2,m,mi,$2,P,P1,$2), dnl
ifelse($1,r,ret8,$1,J,Jmp8,$1,c,call8)
    inc      r2,[#]ifelse($1,r,1,3)
    thread]) dnl
define(b_pair,[r4]) dnl
define(b_high,[rh4]) dnl
define(b_low,[rl4]) dnl
define(d_pair,[r5]) dnl
define(d_high,[rh5]) dnl
define(d_low,[rl5]) dnl
define(h_pair,[r6]) dnl
define(h_high,[rh6]) dnl
define(h_low,[rl6]) dnl
define(psw_pair,[r3]) dnl
define(psw_high,[rh0]) dnl
define(psw_low,[rl0]) dnl
define(sp_pair,[r7]) dnl
define(sp_high,[rh7]) dnl
define(sp_low,[rl7]) dnl
define(m_res,[*r1]) dnl
define(b_res,[rh4]) dnl
define(c_res,[rl4]) dnl
define(d_res,[rh5]) dnl
define(e_res,[rl5]) dnl
define(h_res,[rh6]) dnl
define(l_res,[rl6]) dnl
define(a_res,[rl3]) dnl
define(rst,[rst$18:
    inc      r2,[#]1
    ldb      *r7,r12
    dec      r7,[#]1
    ldb      *r7,r12
    dec      r7,[#]1
    ld       r2,r9
    add      r2,$$1
    thread]) dnl
define(pop8,[pop$18:
    inc      r7,[#]1
    ldb      $1_low,*r7
    inc      r7,[#]1
    ldb      $1_high,*r7
    PCl
    thread]) dnl
define(push8,[push$18:
    ldb      *r7,$1_high
    dec      r7,[#]1
    ldb      *r7,$1_low
    dec      r7,[#]1
    inc      r2,[#]1
    thread]) dnl
define(im,[#18:
    inc      r2,[#]1

```



```
loadflas  
$2      r13,*r2  
$1_flasstore  
inc     r2,[$!1  
thread]) dn1
```



```

define(inx,[inx$18:
    inc    $1_pair,[ $1]
    pcli
    thread]) dnl
define(dcx,[dcx$18:
    dec    $1_pair,[ $1]
    pcli
    thread]) dnl
define(inr,[inr$18:
    ifelse($1,m,eal)
    loadflas
    incb    $1_res,[ $1]
    $0_flasstore
    pcli
    thread]) dnl
define(dcr,[dcr$18:
    ifelse($1,m,eal)
    loadflas
    decb    $1_res,[ $1]
    $0_flasstore
    pcli
    thread]) dnl
define(lxi,[lxi$18:
    inc     r2,[ $1]
    ldb     $1_low,*r2
    inc     r2,[ $1]
    ldb     $1_high,*r2
    ifelse($1,sp,[add]    r7[, ]r9)
    pcli
    thread]) dnl
define(stax,[stax$18:
    ld      r1,r9
    add     r1,$1_pair
    ldb     *r1,r13
    pcli
    thread]) dnl
define(ldax,[ldax$18:
    ld      r1,r9
    add     r1,$1_pair
    ldb     r13,*r1
    pcli
    thread]) dnl
define(dad,[dad$18:
    loadflas
    add     r6,$1_pair
    storeflas
    ifelse($1,sp,[sub]    r6[, ]r9)
    pcli
    thread]) dnl
define(mvi,[mvi$18:
    inc     r2,[ $1]
    ifelse($1,m,eal)
    ldb     r10,*r2
    ldb     $1_res,r10
    pcli
    thread]) dnl

```

```

define(ral,[ral8:
    carryin
    rlcb    r13
    carryout
    pc1
    thread]) dn1
define(rar,[rar8:
    carryin
    rrcb    r13
    carryout
    pc1
    thread]) dn1
define(rlc,[rlc8:
    carryin
    rlb     r13
    carryout
    pc1
    thread]) dn1
define(rrc,[rrc8:
    carryin
    rrb     r13
    carryout
    pc1
    thread]) dn1
define(lda,[lda8:
    ea2
    ldb     r13,*r1
    pc1
    thread]) dn1
define(sta,[sta8:
    ea2
    ldb     *r1,r13
    pc1
    thread]) dn1
define(lhld,[lhld8:
    ea2
    ldb     r16,*r1
    inc     r1,[#]1
    ldb     rh6,*r1
    pc1
    thread]) dn1
define(shld,[shld8:
    ea2
    ldb     *r1,r16
    inc     r1,[#]1
    ldb     *r1,rh6
    pc1
    thread]) dn1
define(daa,[daa8:
    loadfla
    dab     r13
    storefla
    pc1
    thread]) dn1
define(stc,[stc8:
    setb     rh3,[#]7

```

```
    pci
    thread]) dnl
define(cmc,[cmc8:
    xorb    rh3,[4]0x80
    pci
    thread]) dnl
define(cma,[cma8:
    comb    r13
    pci
    thread]) dnl
```

```
define(ea4,[
    ld      r1[,r6
    add     r1[,r9]) dnl
define(mov,[mov$1$28: ifelse($1,m,ea4,$2,m,ea4)
    ldb     $1_res,$2_res
    pc1
    thread]) dnl
define(hlt,[hlt8:
    inc     r2,[$]1
    clrb    rh0
    ldb     r10,*r2
    jf      leave]) dnl
```