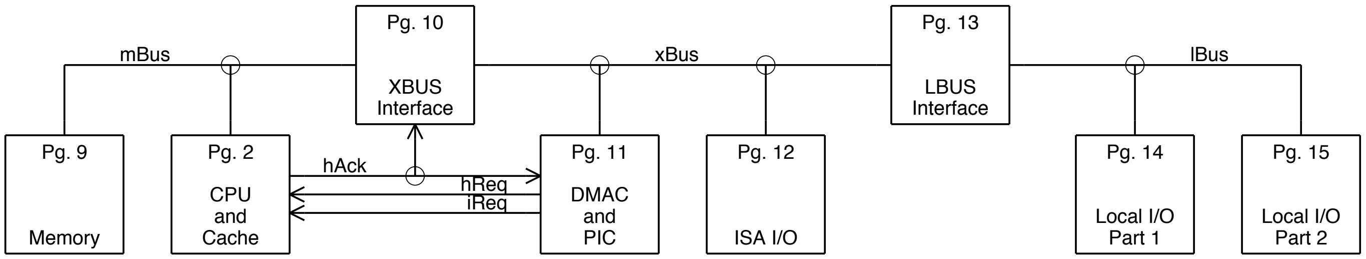
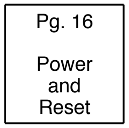
Include

sys.def

Declarations

```
VAR
  mBus: MBusType;
  xBus: XBusType;
  lBus: LBusType;
  hReq, hAck: SIGNAL;
  iReq: IReqType;
```





Title: ALPHA PC

File: sys01.imp

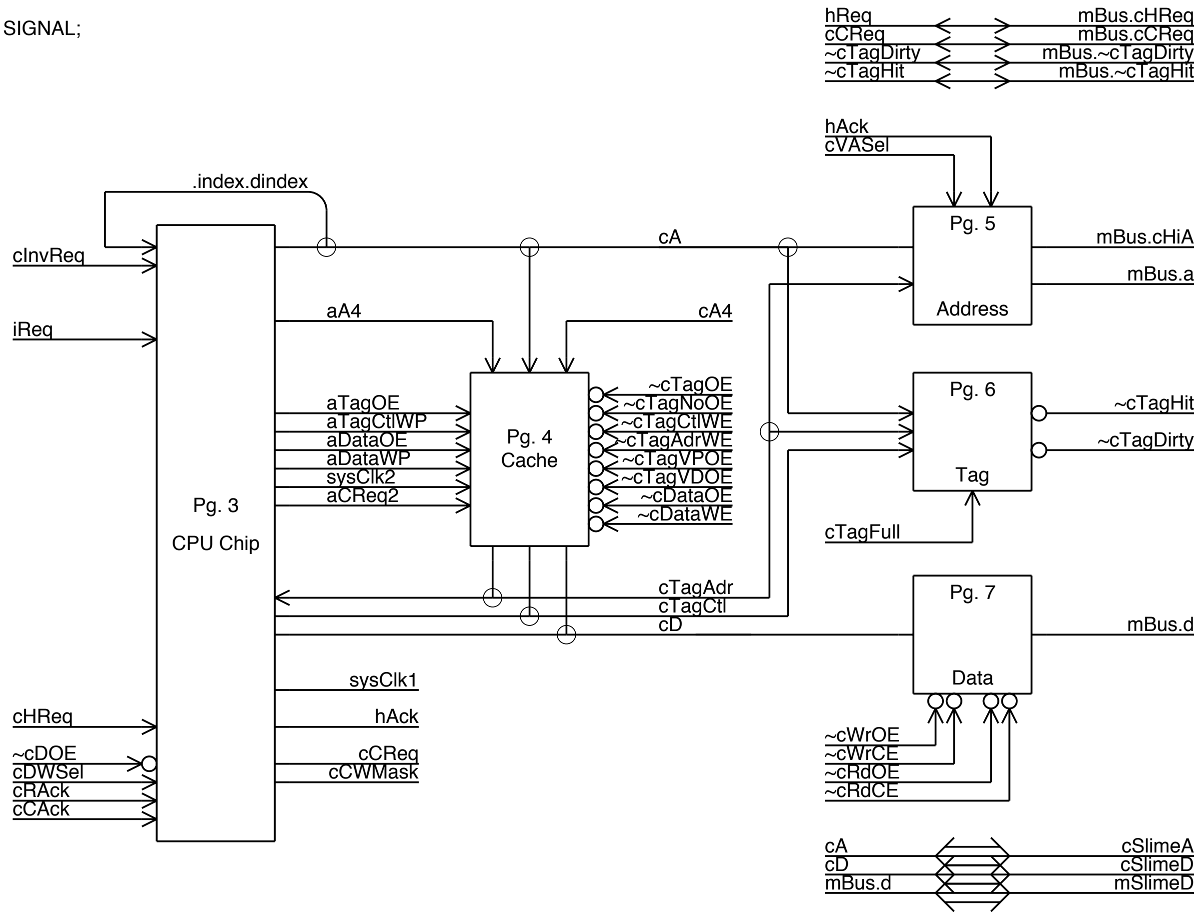
Rev: A

Page: 1

Last modified: Wed May 22 11:16:00 1991 by dgc

Declarations

```
VAR
  cA: CpuAType;
  cTagAdr: TagAdrType;
  cTagCtl: CpuTagCtlType;
  cD: CpuDataType;
  aA4, aCReq2, aTagOE, aTagCtlWP: SIGNAL;
  cA4, ~cTagNoOE, ~cTagOE, ~cDataOE: SIGNAL;
  ~cTagHit, ~cTagDirty, ~cTagCtlWE, ~cTagAdrWE: SIGNAL;
  aDataOE: CpuDataCEOEType;
  aDataWP, ~cDataWE: CpuDataWEType;
  sysClk1, sysClk2, ~cTagVPOE, ~cTagVDOE: SIGNAL;
  cInvReq, ~cDOE, cDWSel: SIGNAL;
  cTagFull, cHReq, cVASel: SIGNAL;
  ~cRdOE, ~cWrCE: SIGNAL;
  ~cRdCE, ~cWrOE: Bits2Type;
  cCReq: CpuCReqType;
  cCWMask: CpuCWMaskType;
  cRAck: CpuRAckType;
  cCAck: CpuCAckType;
  { slime access }
  cSlimeA: CpuSlimeAType;
  cSlimeD: CpuSlimeDType;
  mSlimeD: MemSlimeDType;
```



Local

sys08

Pg. 8

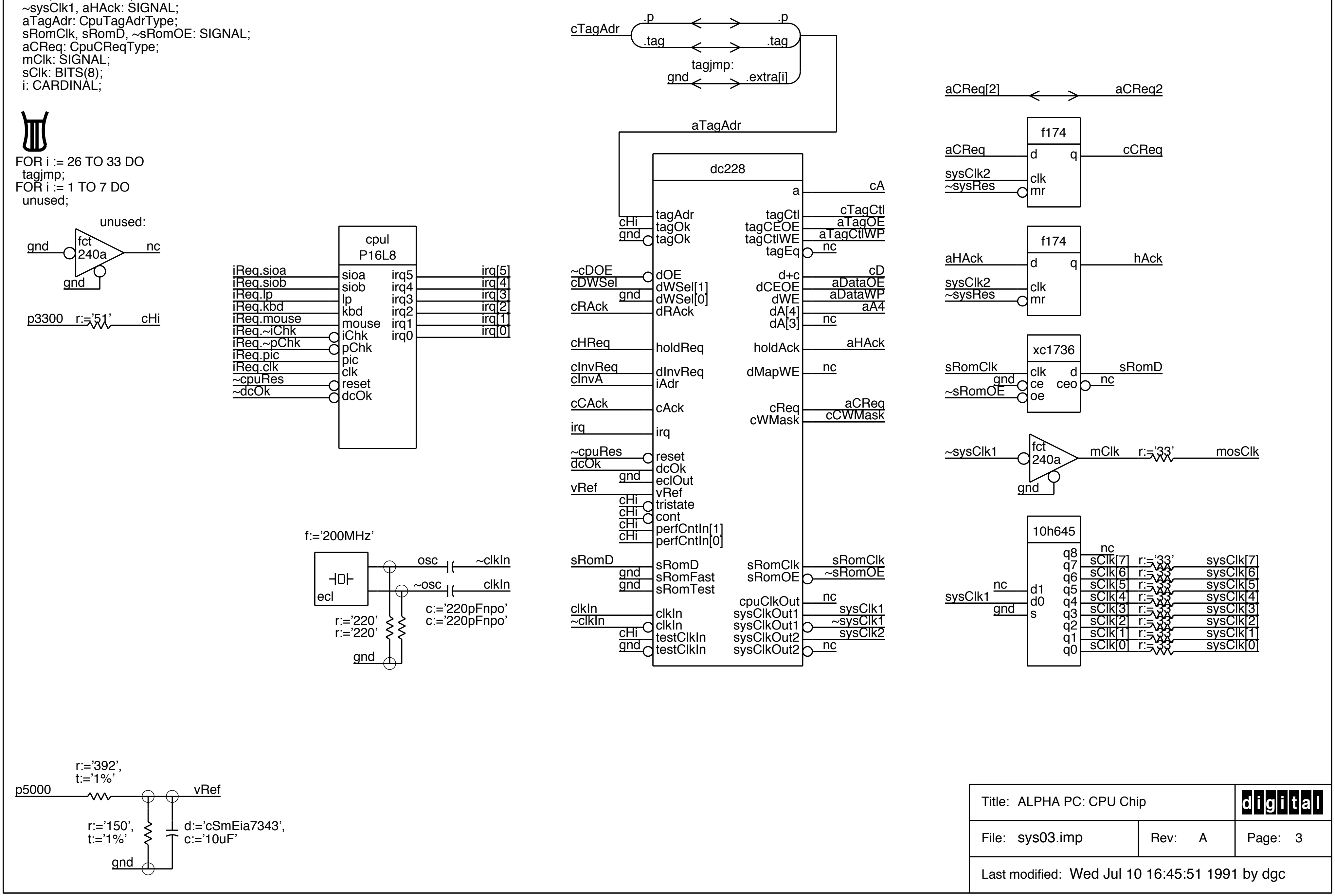
Control

Title: ALPHA PC: CPU and Cache		digital
File: sys02.imp	Rev: A	Page: 2
Last modified: Wed Jul 3 09:21:16 1991 by dgc		

Declarations

```
VAR
  irq: CpuIrqType;
  osc, ~osc, clkIn, ~clkIn: SIGNAL;
  vRef, cHi: SIGNAL;
  ~sysClk1, aHAck: SIGNAL;
  aTagAdr: CpuTagAdrType;
  sRomClk, sRomD, ~sRomOE: SIGNAL;
  aCReq: CpuCReqType;
  mClk: SIGNAL;
  sClk: BITS(8);
  i: CARDINAL;
```

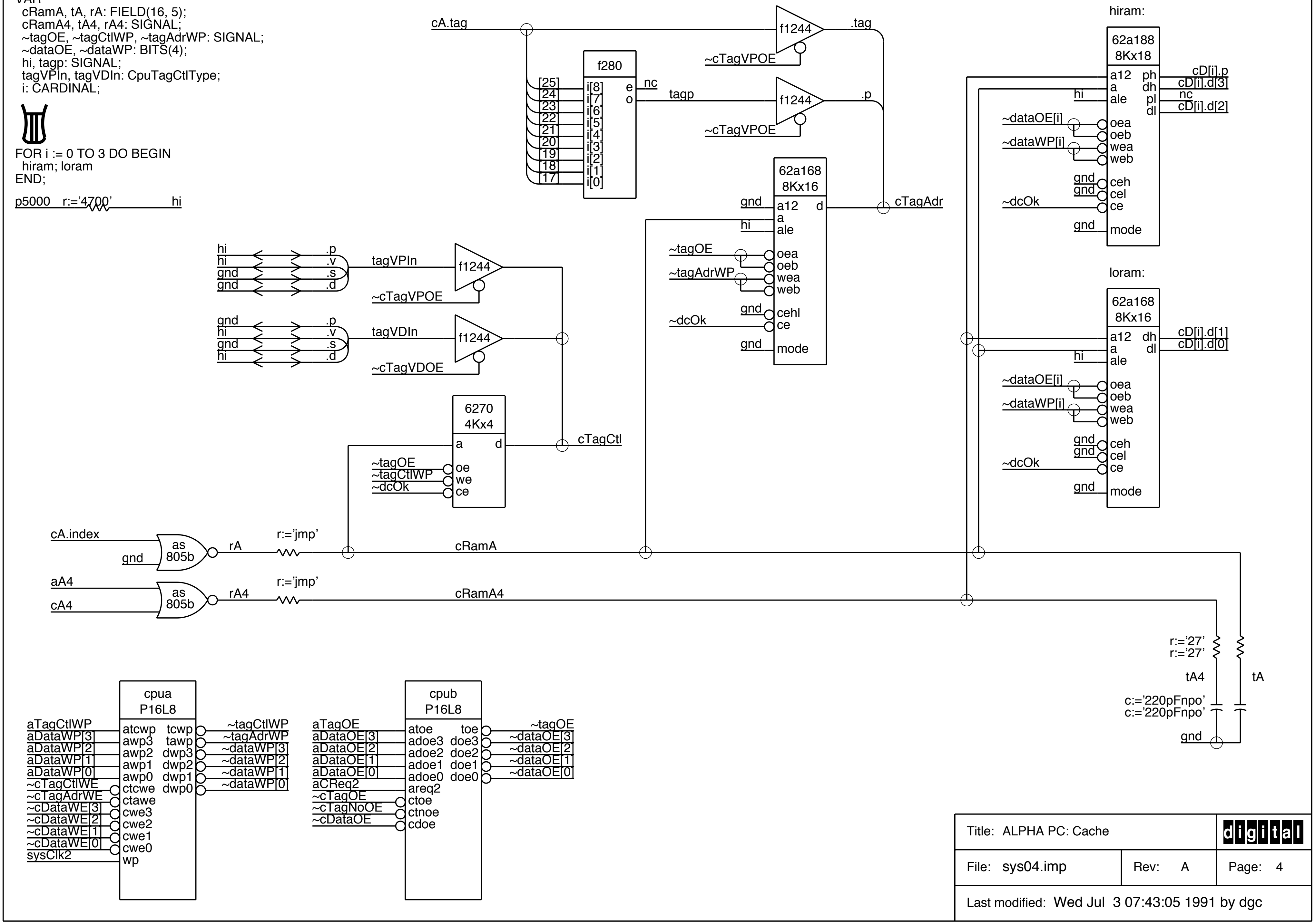
```
FOR i := 26 TO 33 DO
  tagjmp;
FOR i := 1 TO 7 DO
  unused;
```



Title: ALPHA PC: CPU Chip		digital
File: sys03.imp	Rev: A	Page: 3
Last modified: Wed Jul 10 16:45:51 1991 by dgc		

Declarations

```
VAR
  cRamA, tA, rA: FIELD(16, 5);
  cRamA4, tA4, rA4: SIGNAL;
  ~tagOE, ~tagCtlWP, ~tagAdrWP: SIGNAL;
  ~dataOE, ~dataWP: BITS(4);
  hi, tagp: SIGNAL;
  tagVPIn, tagVDIn: CpuTagCtlType;
  i: CARDINAL;

FOR i := 0 TO 3 DO BEGIN
  hiram; loram
END;
```



Title: ALPHA PC: Cache

digital

File: sys04.imp

Rev: A

Page: 4

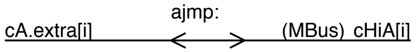
Last modified: Wed Jul 3 07:43:05 1991 by dgc

Declarations

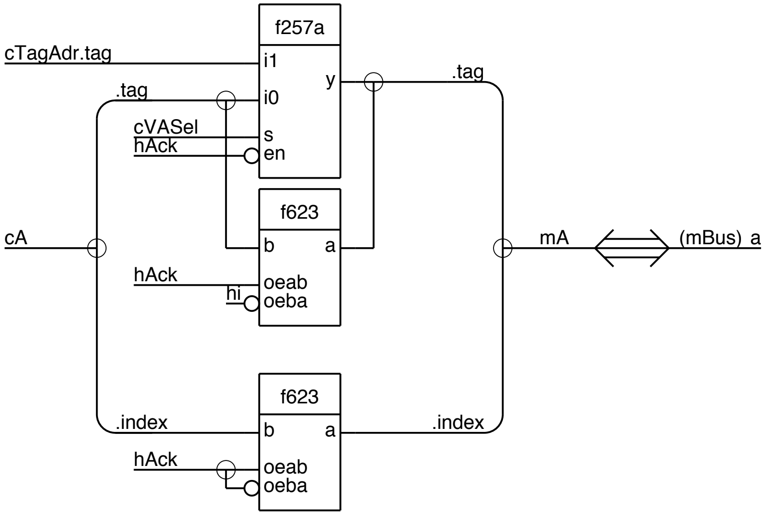
```
VAR
  mA: AType;
  hi: SIGNAL;
  i: CARDINAL;
```

```
FOR i := 26 TO 33 DO
  ajmp;
```

p5000 r:='4700' hi



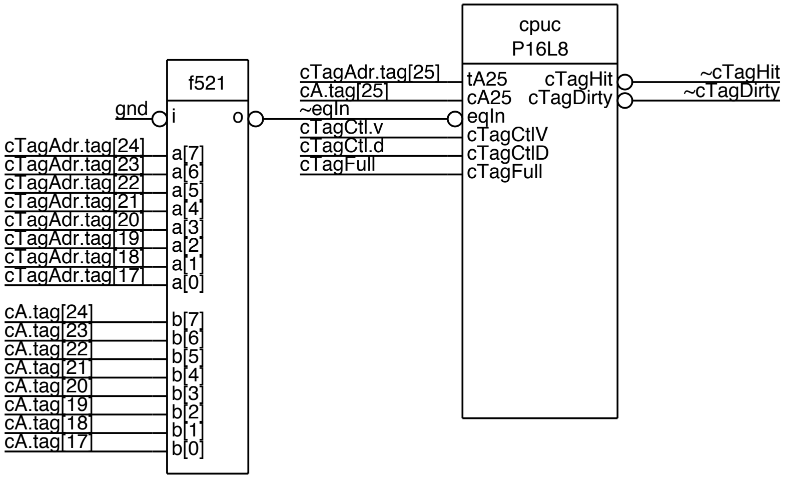
These address bits go to only a single place,
so they do not need to be buffered.



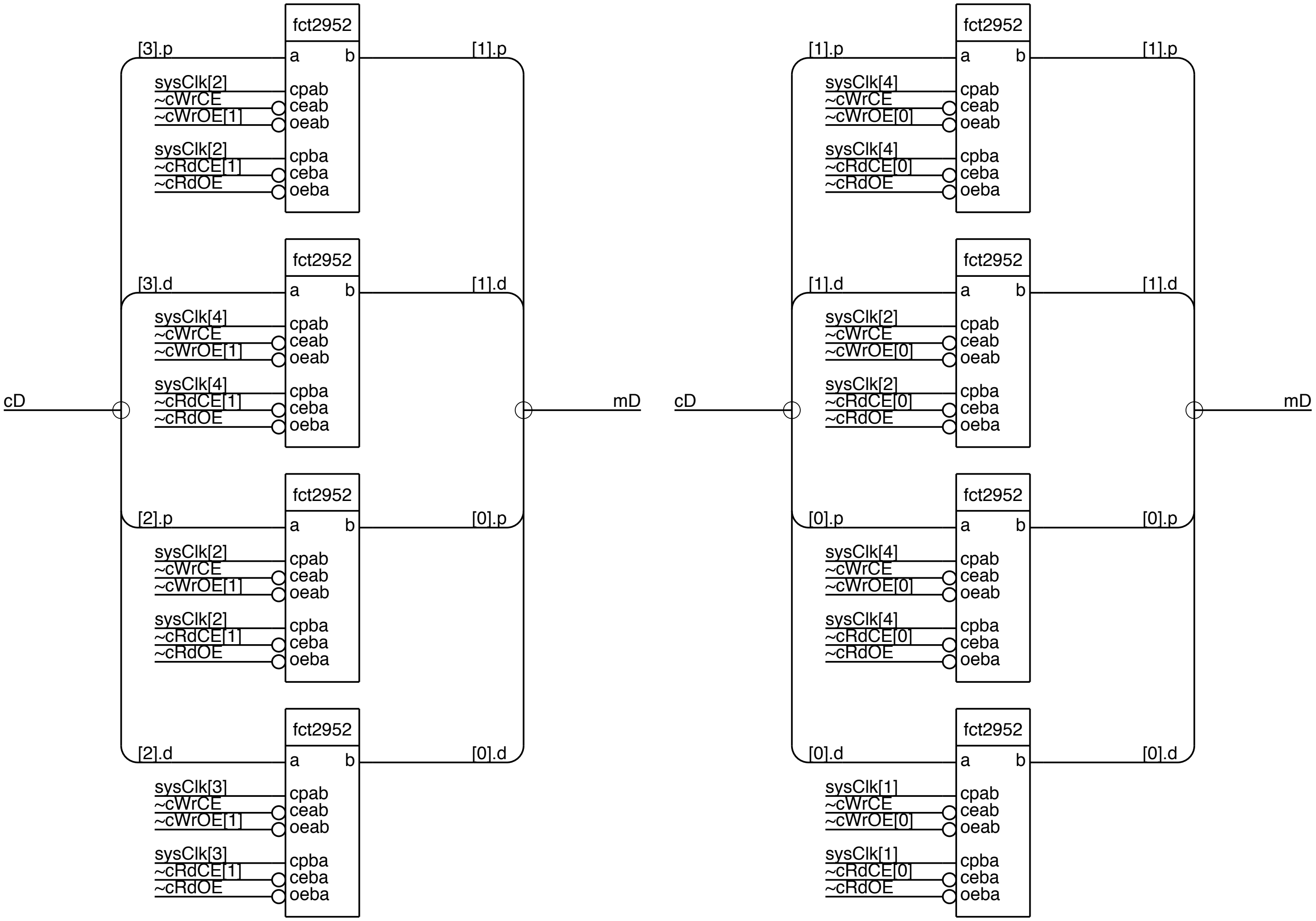
Title: ALPHA PC: Address			digital
File: sys05.imp	Rev: A		Page: 5
Last modified: Thu Jun 6 08:20:55 1991 by dgc			

Declarations

VAR
 ~eqIn: SIGNAL;



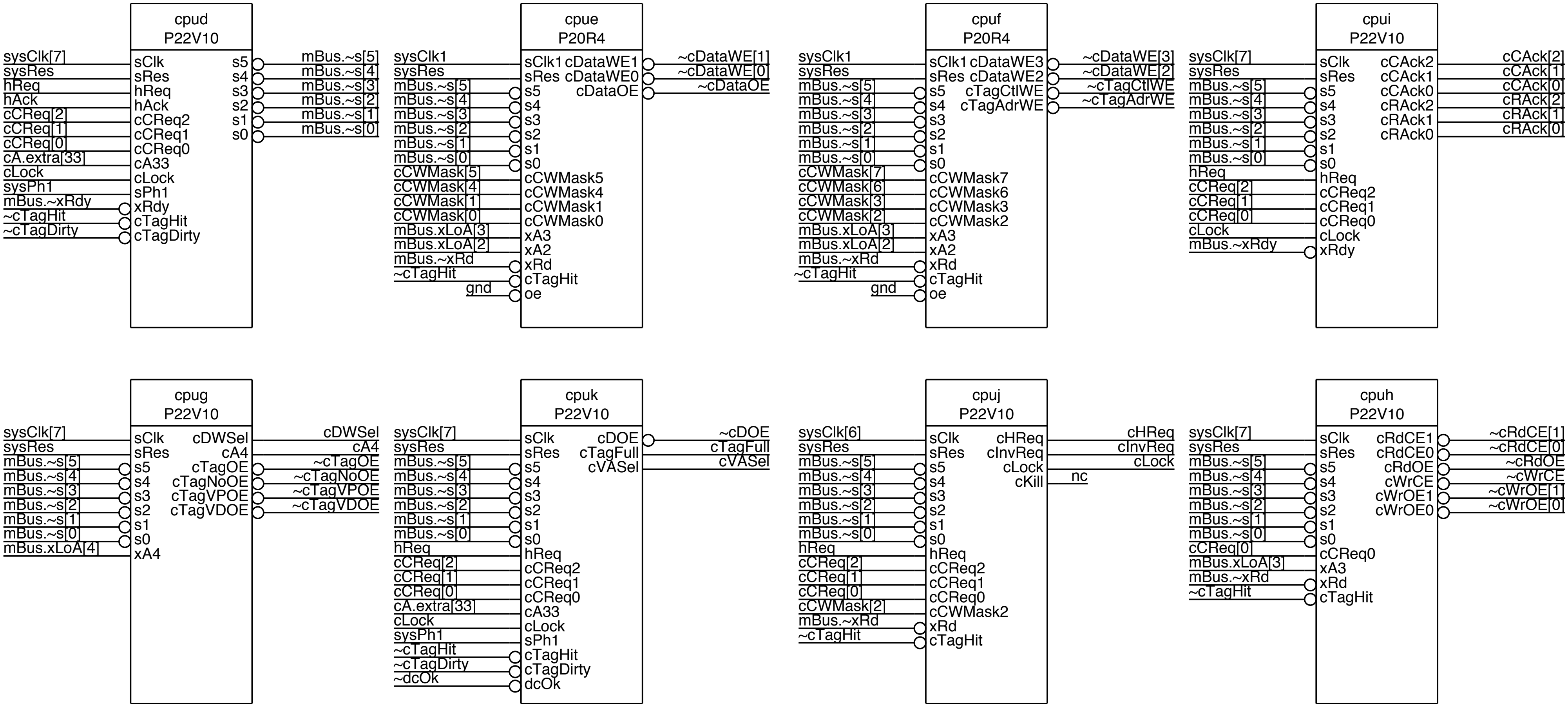
Title: ALPHA PC: Tag		digital
File: sys06.imp	Rev: A	Page: 6
Last modified: Wed Jul 3 08:05:53 1991 by dgc		



Title: ALPHA PC: Data		digital
File: sys07.imp	Rev: A	Page: 7
Last modified: Fri Jun 28 07:35:52 1991 by dgc		

Declarations

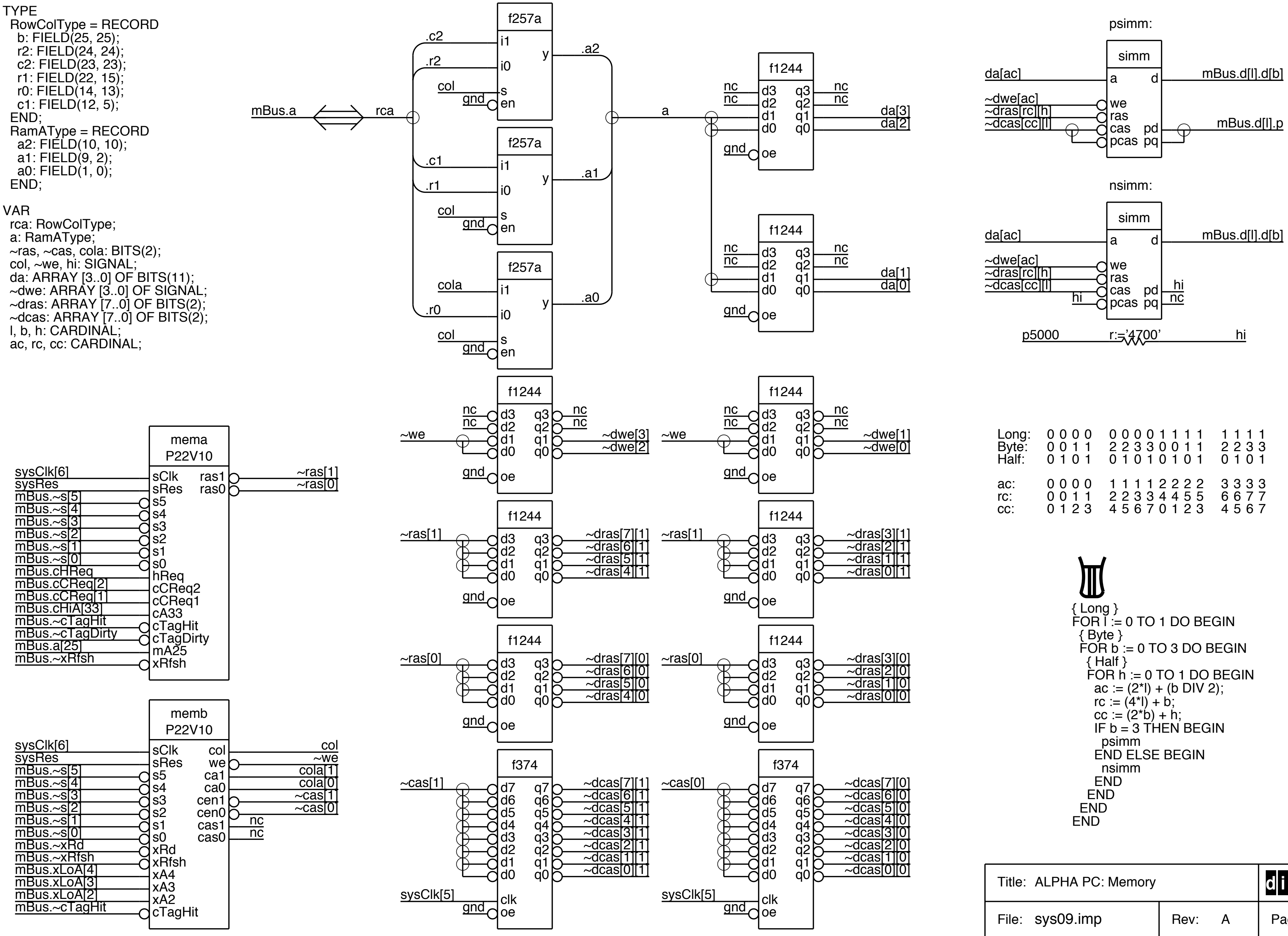
```
VAR
  cLock: SIGNAL;
```



Title: ALPHA PC: Control		digital
File: sys08.imp	Rev: A	Page: 8
Last modified: Fri Jul 12 09:19:21 1991 by dgc		

Declarations

```
TYPE
  RowColType = RECORD
    b: FIELD(25, 25);
    r2: FIELD(24, 24);
    c2: FIELD(23, 23);
    r1: FIELD(22, 15);
    r0: FIELD(14, 13);
    c1: FIELD(12, 5);
  END;
  RamAType = RECORD
    a2: FIELD(10, 10);
    a1: FIELD(9, 2);
    a0: FIELD(1, 0);
  END;

VAR
  rca: RowColType;
  a: RamAType;
  ~ras, ~cas, cola: BITS(2);
  col, ~we, hi: SIGNAL;
  da: ARRAY [3..0] OF BITS(11);
  ~dwe: ARRAY [3..0] OF SIGNAL;
  ~dras: ARRAY [7..0] OF BITS(2);
  ~dcas: ARRAY [7..0] OF BITS(2);
  l, b, h: CARDINAL;
  ac, rc, cc: CARDINAL;
```



Long:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
Byte:	0	0	1	1	2	2	3	3	0	0	1	1	2	2	3	3
Half:	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
ac:	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
rc:	0	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7
cc:	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7

```
{ Long }
FOR l := 0 TO 1 DO BEGIN
  { Byte }
  FOR b := 0 TO 3 DO BEGIN
    { Half }
    FOR h := 0 TO 1 DO BEGIN
      ac := (2*l) + (b DIV 2);
      rc := (4*l) + b;
      cc := (2*b) + h;
      IF b = 3 THEN BEGIN
        psimm
      END ELSE BEGIN
        nsimm
      END
    END
  END
END
```

Title: ALPHA PC: Memory

digital

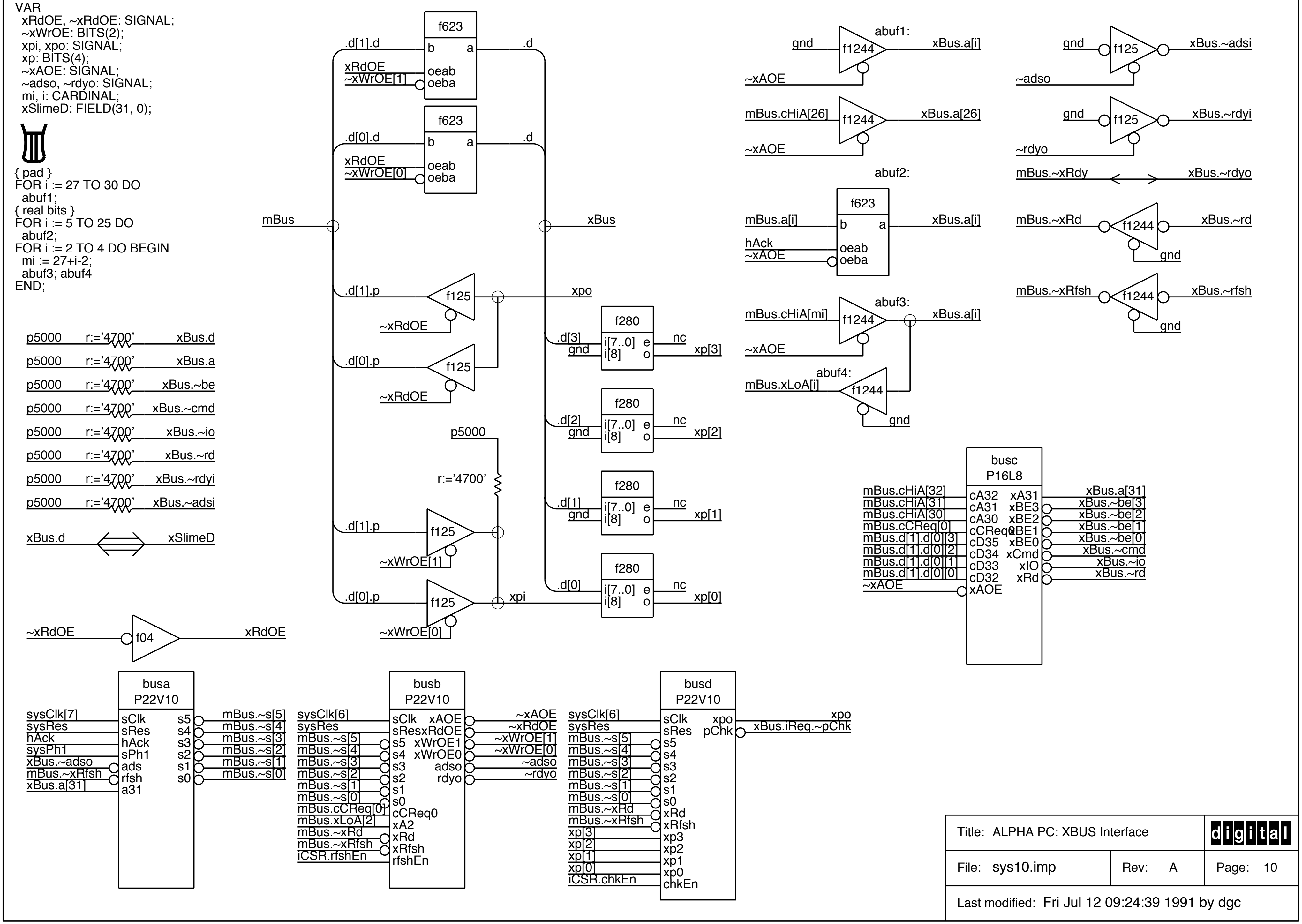
File: sys09.imp | Rev: A | Page: 9

Last modified: Mon Jul 15 08:35:38 1991 by dgc

Declarations

```
VAR
  xRdOE, ~xRdOE: SIGNAL;
  ~xWrOE: BITS(2);
  xpi, xpo: SIGNAL;
  xp: BITS(4);
  ~xAOE: SIGNAL;
  ~adso, ~rdyo: SIGNAL;
  mi, i: CARDINAL;
  xSlimeD: FIELD(31, 0);
```

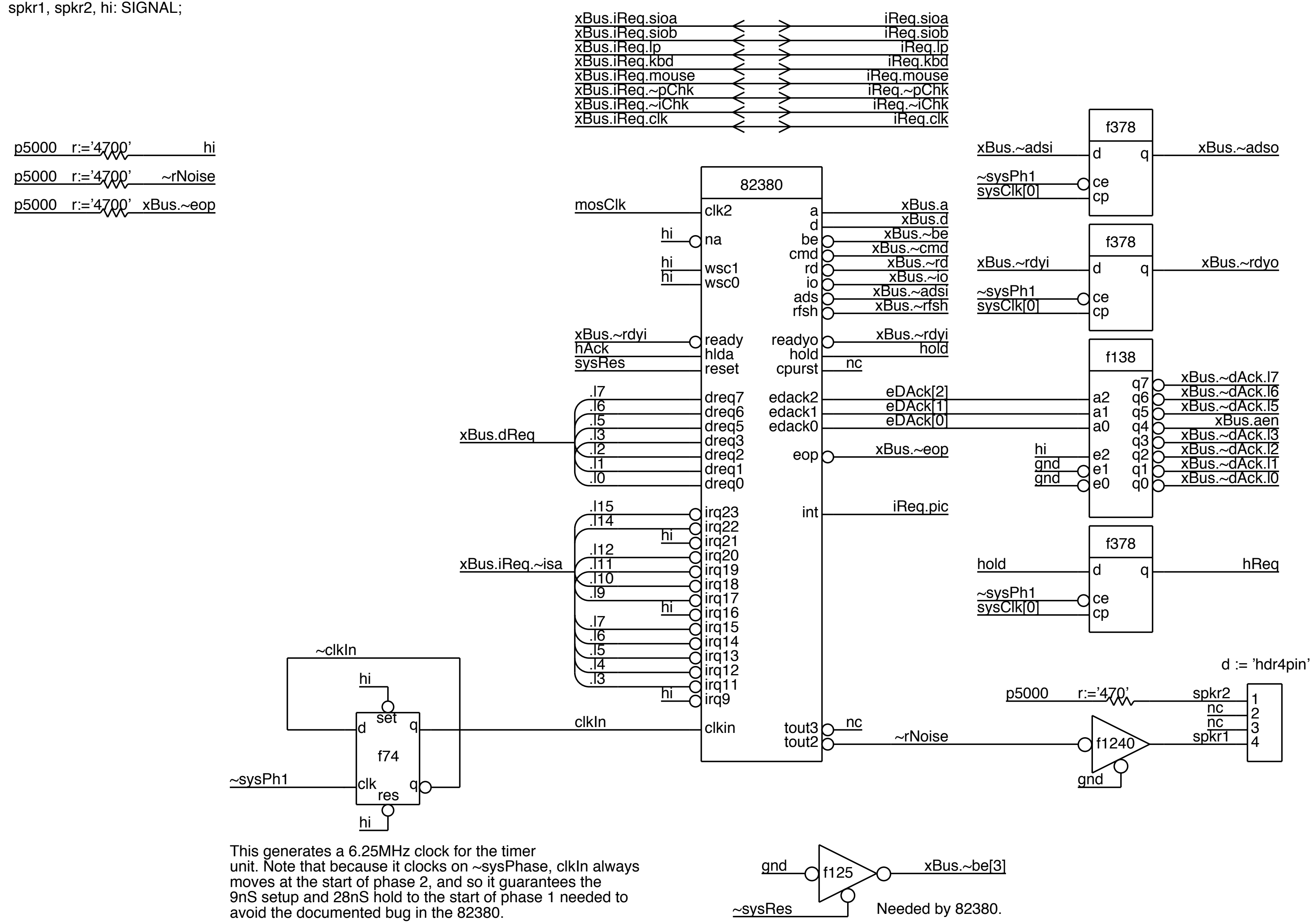
```
{ pad }
FOR i := 27 TO 30 DO
  abuf1;
{ real bits }
FOR i := 5 TO 25 DO
  abuf2;
FOR i := 2 TO 4 DO BEGIN
  mi := 27+i-2;
  abuf3; abuf4
END;
```



Title: ALPHA PC: XBUS Interface		digital
File: sys10.imp	Rev: A	Page: 10
Last modified: Fri Jul 12 09:24:39 1991 by dgc		

Declarations

```
VAR
  clkIn, ~clkIn: SIGNAL;
  eDAck: BITS(3);
  ~rNoise, hold: SIGNAL;
  spkr1, spkr2, hi: SIGNAL;
```



This generates a 6.25MHz clock for the timer unit. Note that because it clocks on ~sysPhase, clkIn always moves at the start of phase 2, and so it guarantees the 9nS setup and 28nS hold to the start of phase 1 needed to avoid the documented bug in the 82380.

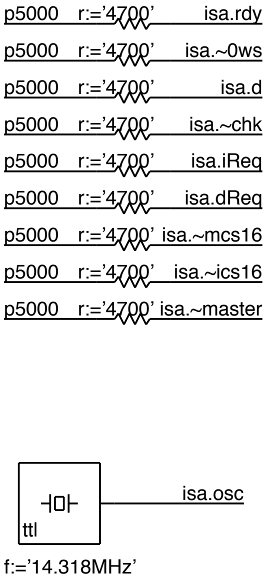
Needed by 82380.

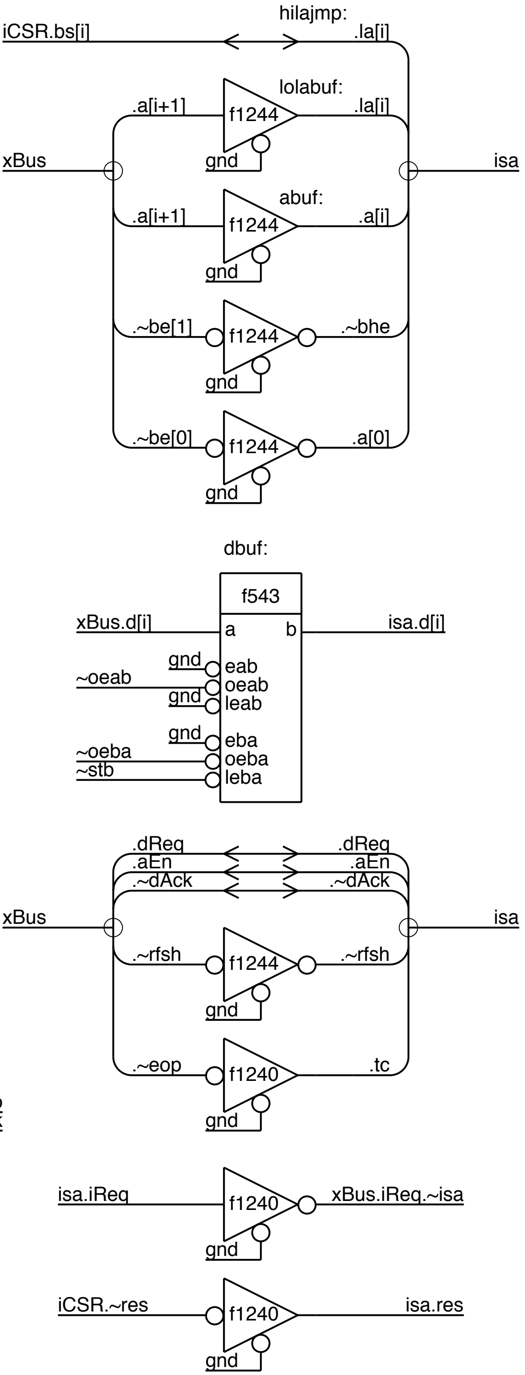
Title: ALPHA PC: DMAC and PIC		digital
File: sys11.imp	Rev: A	Page: 11
Last modified: Wed Jul 3 11:32:46 1991 by dgc		

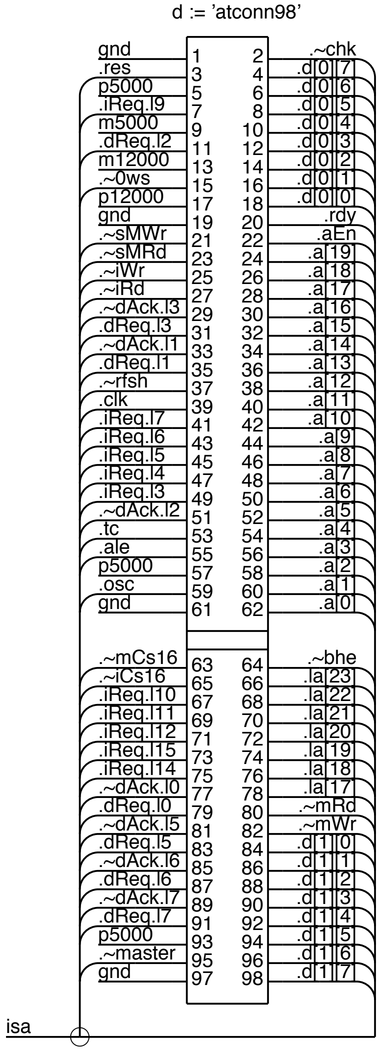
Declarations

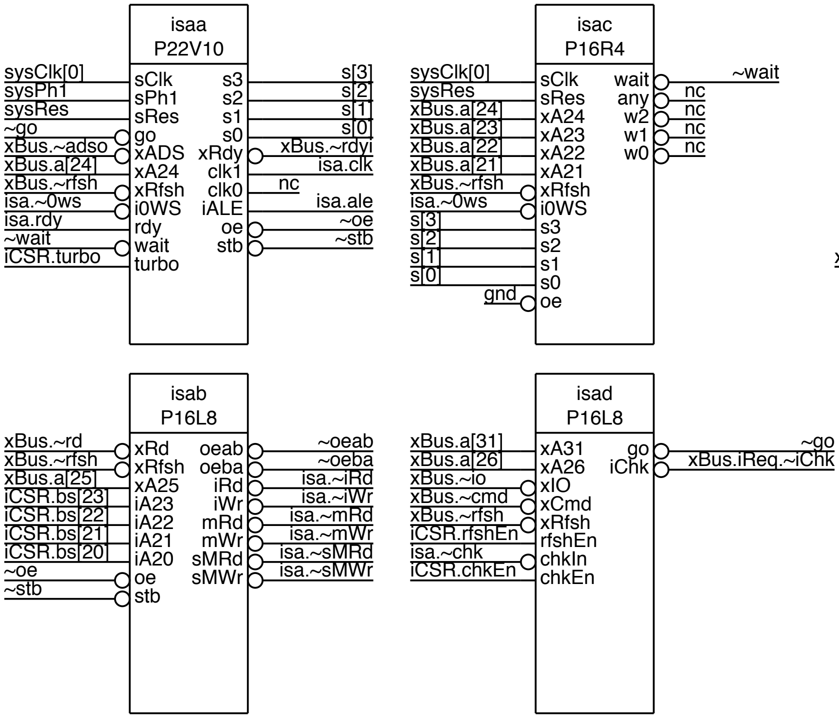
```
VAR
  ~oe, ~oeab, ~oeba, ~stb, ~go, ~wait: SIGNAL;
  s: BITS(4);
  isa: ISABusType;
  i: CARDINAL;
```

```
{ a[19..1] <- xA[20..2] }
FOR i := 1 TO 19 DO
  abuf;
{ la[19..17] <- xA[20..18] }
FOR i := 17 TO 19 DO
  lolabuf;
{ la[23..20] <- bank[23..20]  }
FOR i := 20 TO 23 DO
  hilajmp;
{ 2 data bytes }
FOR i := 0 TO 1 DO
  dbuf;
```









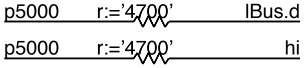
Title: ALPHA PC: ISA I/O

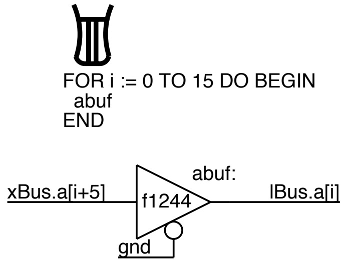
File: sys12.imp | Rev: A | Page: 12

Last modified: Mon Jul 15 13:02:38 1991 by dgc

Declarations

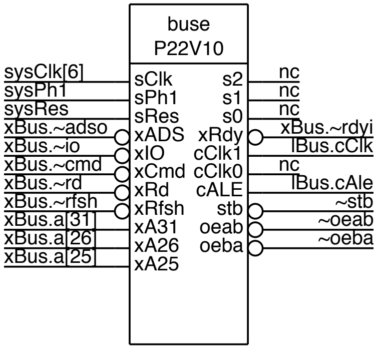
```
VAR
  ~oeab, ~oeba, ~stb, hi: SIGNAL;
  i: CARDINAL;
```

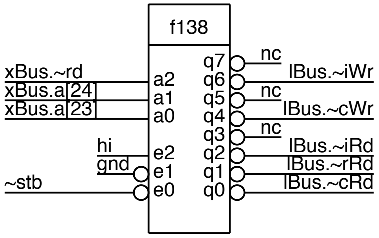


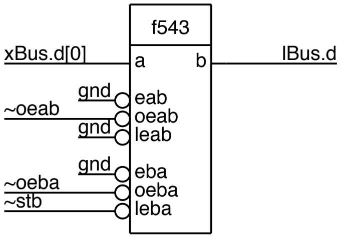


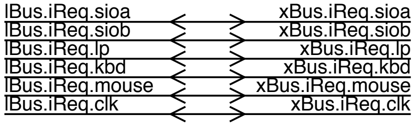
```
FOR i := 0 TO 15 DO BEGIN
  abuf
END
```

xA[31]	xA[26..25]	xA[24..23]	Device
1	00	00	Combo chip
1	00	01	ROM (RO)
1	00	10	ISA CSR
1	00	11	Unused (stb + ack)
1	01	xx	Unused (ack)





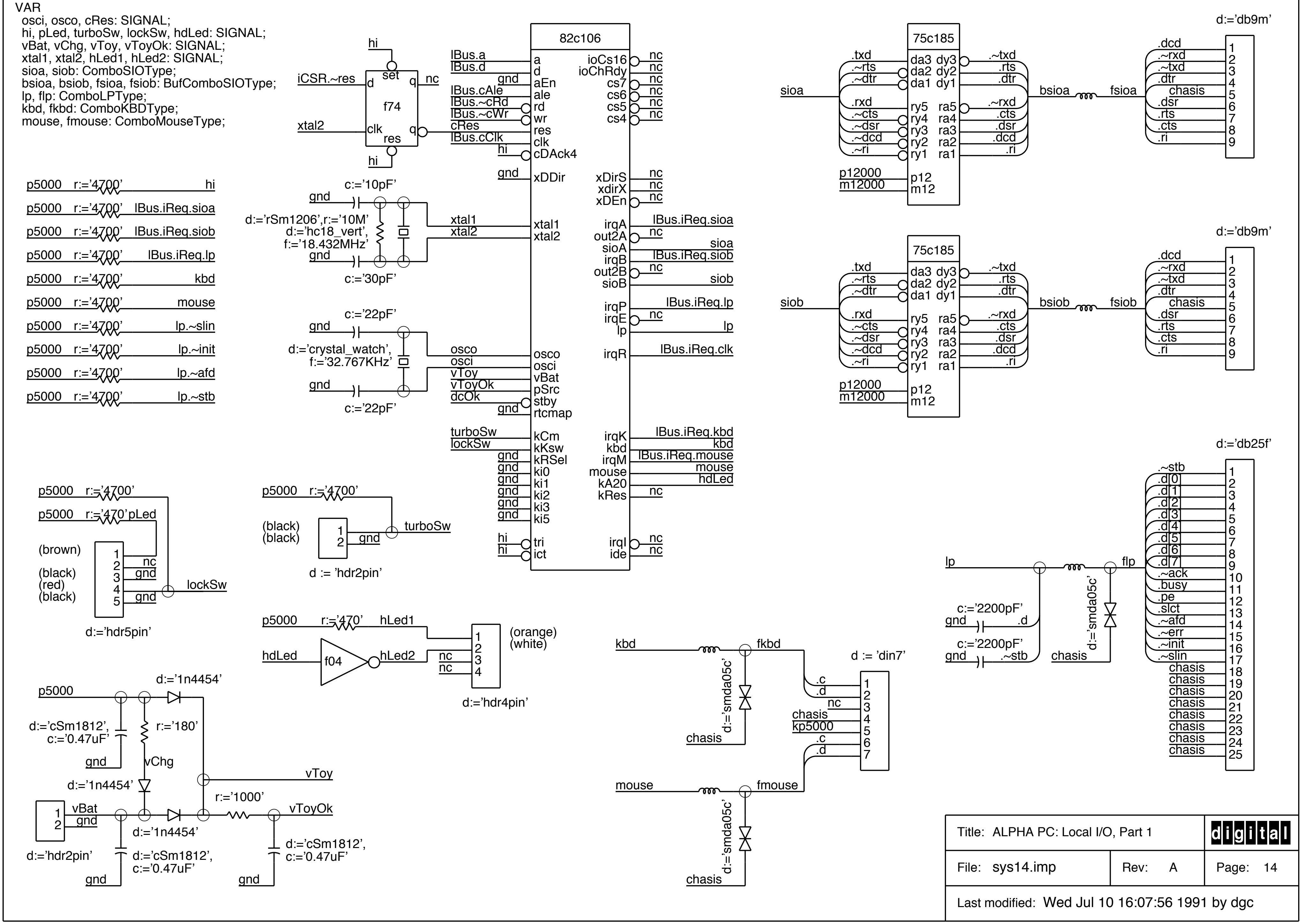




Title: ALPHA PC: LBUS Interface		digital
File: sys13.imp	Rev: A	Page: 13
Last modified: Fri Jul 12 09:20:10 1991 by dgc		

Declarations

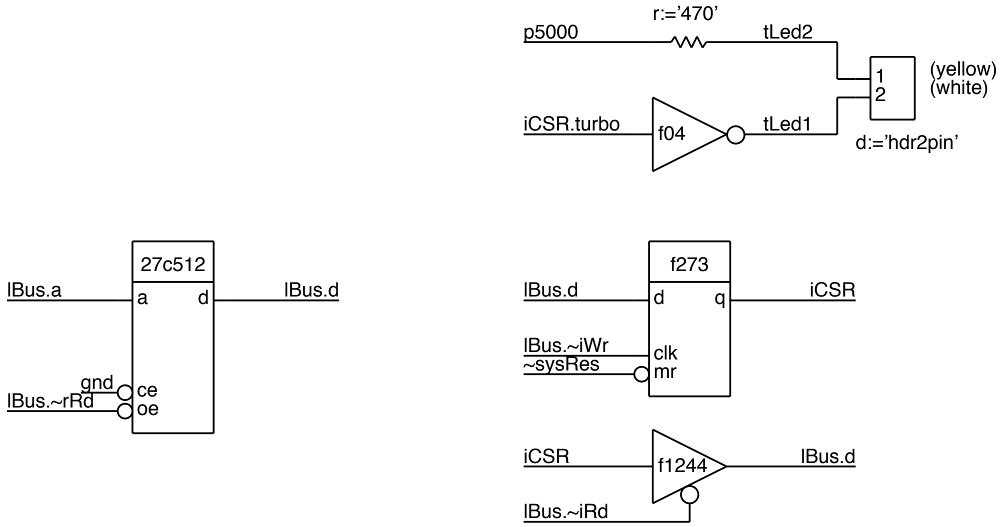
```
VAR
  osci, osco, cRes: SIGNAL;
  hi, pLed, turboSw, lockSw, hdLed: SIGNAL;
  vBat, vChg, vToy, vToyOk: SIGNAL;
  xtal1, xtal2, hLed1, hLed2: SIGNAL;
  sioa, siob: ComboSIOType;
  bsioa, bsiob, fsioa, fsiob: BufComboSIOType;
  lp, flp: ComboLPType;
  kbd, fkbd: ComboKBDType;
  mouse, fmouse: ComboMouseType;
```



Title: ALPHA PC: Local I/O, Part 1		digital
File: sys14.imp	Rev: A	Page: 14
Last modified: Wed Jul 10 16:07:56 1991 by dgc		

Declarations

```
VAR
  tLed1, tLed2: SIGNAL;
```

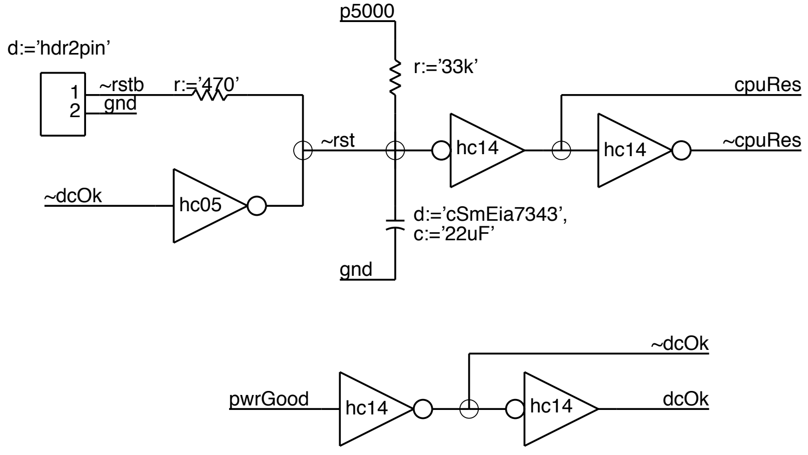


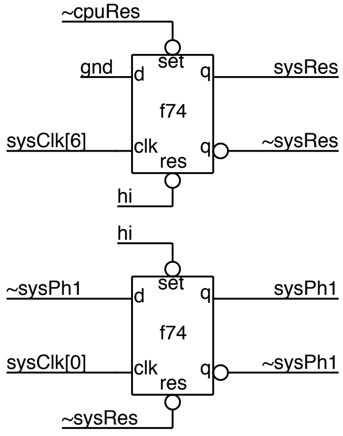
Title: ALPHA PC: Local I/O, Part 2		
File: sys15.imp	Rev: A	Page: 15
Last modified: Tue May 28 10:30:21 1991 by dgc		

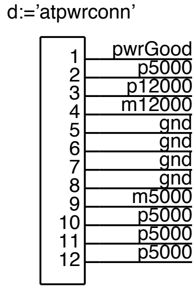
Declarations

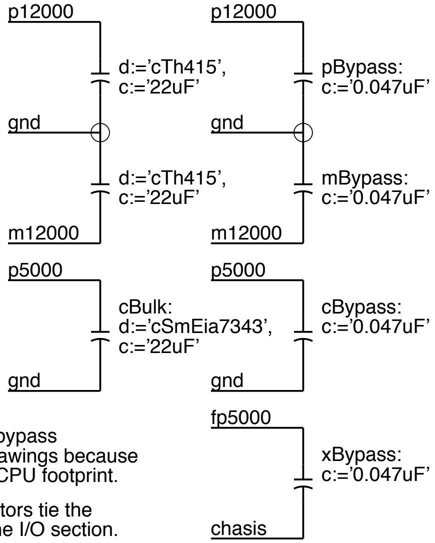
```
CONST
  NCBypass = 161;
  NXBypass = 10;
  NCBulk = 5;
  NPBypass = 2;
  NMBypass = 2;

VAR
  adj: SIGNAL;
  pwrGood, hi: SIGNAL;
  ~rstb, ~rst: SIGNAL;
  i: CARDINAL;
```

```
FOR i := 1 TO NCBypass DO
  cBypass;
FOR i := 1 TO NXBypass DO
  xBypass;
FOR i := 1 TO NCBulk DO
  cBulk;
FOR i := 1 TO NPBypass DO
  pBypass;
FOR i := 1 TO NMBypass DO
  mBypass;
FOR i := 1 TO 5 DO
  unused05;
FOR i := 1 TO 2 DO
  unused14;
```



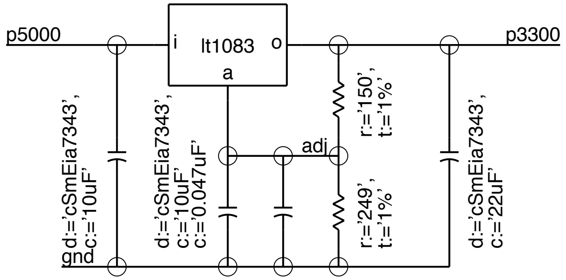


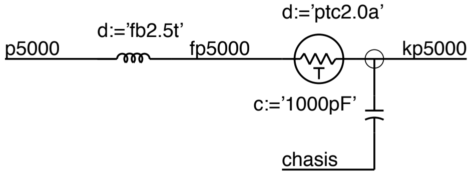


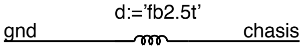


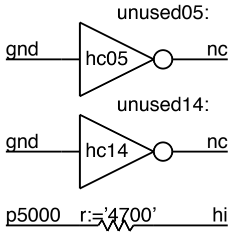
There are no -3.3V bypass capacitors on the drawings because they are part of the CPU footprint.

The xBypass capacitors tie the planes together in the I/O section.









Title: ALPHA PC: Power		digital
File: sys16.imp	Rev: A	Page: 16
Last modified: Mon Jul 15 13:03:24 1991 by dgc		