Computer design

PDP-11/40E datapath and microprogramming

P.J. Drongowski
SandSoftwareSound.net

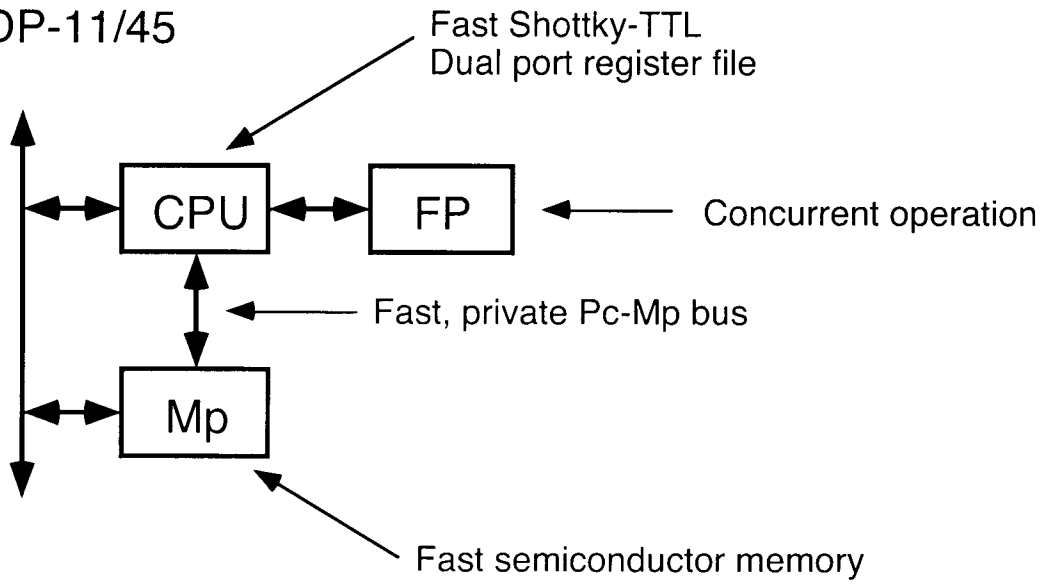
PDP-11 family

Model	Performance	Integration	Technology	Cycle time
11/05	1.4	MSI	TTL	300
11/20	1.6	SSI	TTL	280
11/40	2.8	MSI/LSI	TTL	140/200/300
11/45	6.8	MSI/LSI	TTL/S	150
11/60	3.7	MSI/LSI	TTL/S	170

Model	CS size	Boards	Packages	Types
11/05	256 x 40	2 hex	203	60
11/20	Random	6+6+2	523	27
11/40	256 x 56	4+1	417	53
11/45	256 x 64	7+1	696	78
11/60	2560 x 48	6 hex	648	74

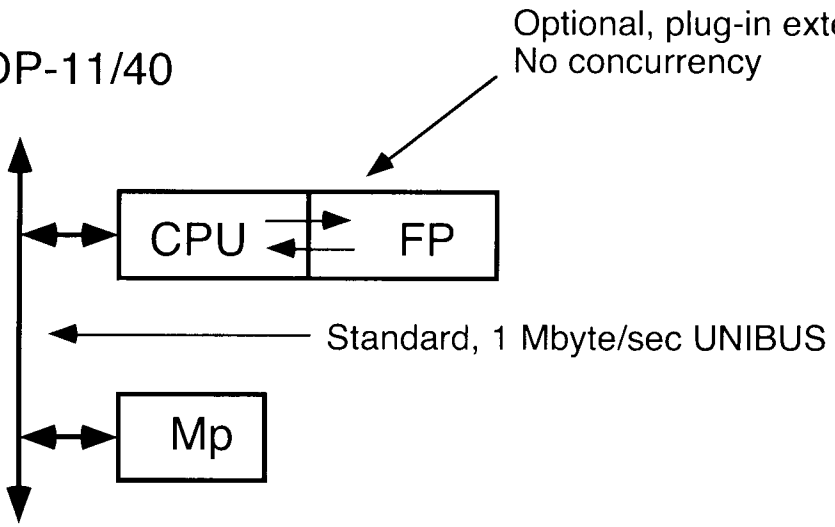
- What is the context for the PDP-11/40 design?
- Contemporary siblings in the family
  - PDP-11/45
    - ◊ High performance (scientific / laboratory)
    - ◊ Virtual memory
    - ◊ Integral floating point unit
  - PDP-11/05
    - ◊ Low cost, low level of performance
    - ◊ No floating point
    - ◊ No virtual memory
- Design target
  - Intermediate cost and performance
  - Subset of 11/45 virtual memory
  - Floating point option (subset of 11/45)

PDP-11/45



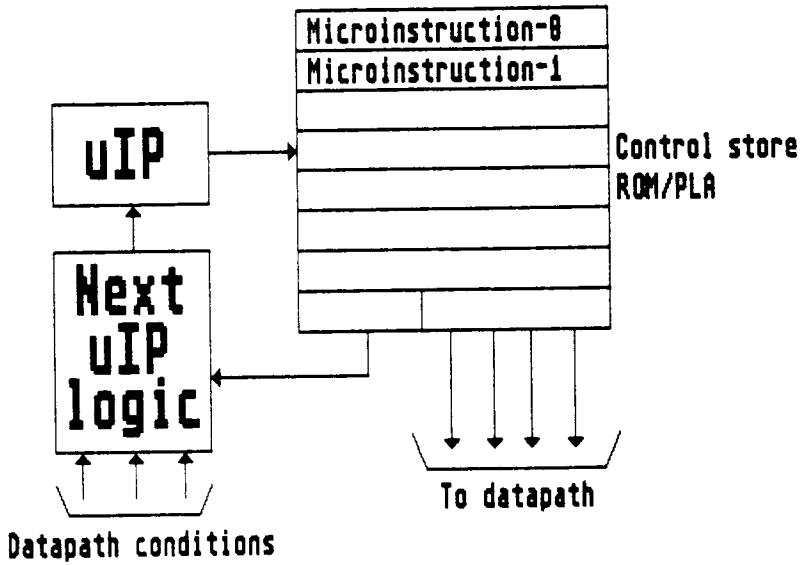
- Three processor modes
- Two sets of general registers (fast context switching)
- Separate instruction and data spaces in process
- Eight segments per address space
- Three sets of segmentation registers
- Dedicated floating point registers in FPU

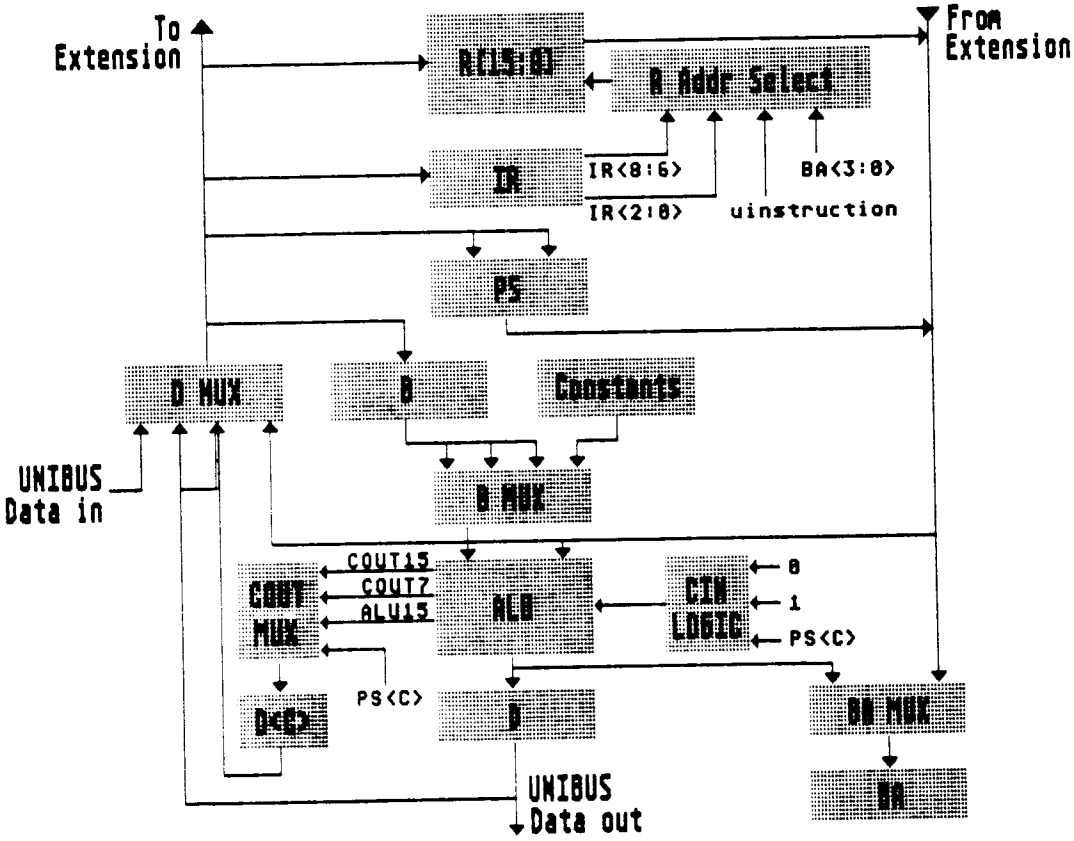
PDP-11/40



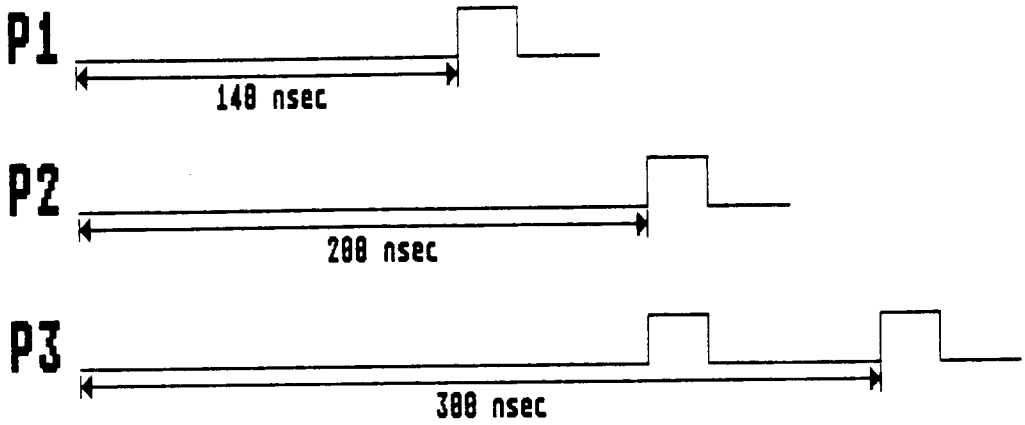
- One general register set (slow context switch)
- No instruction and data space separation
- 64 Kbyte maximum program size
- Two sets of segmentation registers
- No separate floating point registers (operands on stack)

Microprogrammable controller

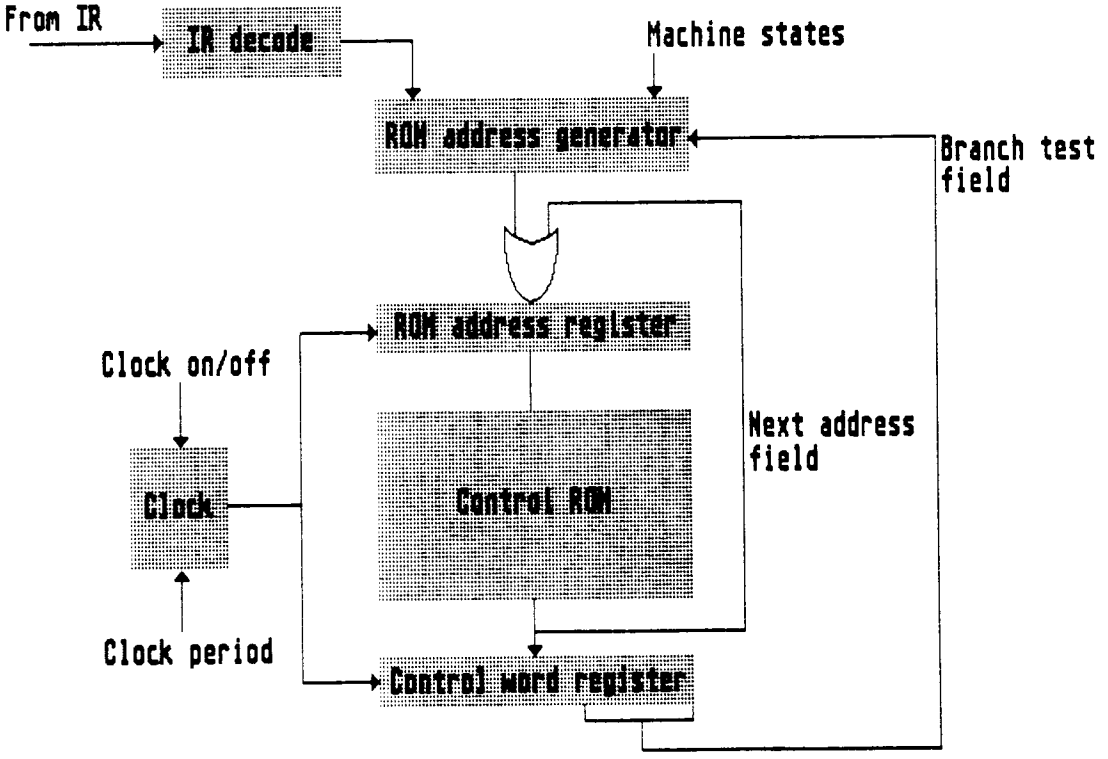




Clocking



Clocking



11/40 controller

PDP-11/40 uinstruction fields

Register enables

CLKB P1 & P3
- 0 No op
- 1 B ⇳ DMUX

CLKD P2
- 0 No op
- 1 D ⇳ ALU result

CLKBA P1 & P2
- 0 No op
- 1 BA ⇳ BA MUX

CLKIR P1 & P3
- 0 No op
- 1 IR ⇳ D MUX

WR H
- 0 No op
- 1 R⇳D MUX<15:8>

WR L P1 & P3
- 0 No op
- 1 R⇳D MUX<7:0>

MUX selection

SBAM
- 0 ALU
- 1 RD bus

SDM
- 0 RD bus
- 1 Unibus data
- 2 D
- 3 D<C>⊡D<14:0>

SBMH
- 0 B<15:8>
- 1 B<7>
- 2 B<7:0>
- 3 Constant<15:8>

SBML
- 0 B<7:0>
- 1 B<7:0>
- 2 B<15:8>
- 3 Constant<7:0>

Register indexing

RIF<3:0> Register immediate field

SR

SRBA	BA<3:0>	Bus address reg
SRD	IR<2:0>	Destination
SRI	RIF<3:0>	Immediate
SRS	IR<8:6>	Source

Processor status word

SPS P1 & P3

0	D MUX
1	C
2	NZV
3	NZVC
6	PS to RD bus
7	PS ⇳ D MUX

ALU

SALUM ALU mode

0 Arithmetic

1 Logic

SALU ALU function

See 74181 spec sheet

DAD Discrete alteration of data

UNIBUS control

BUS

0 -

1 DATI (data in)

2 Await bus busy

3 DATIP (data in pause)

4 -

5 DATO (data out)

6 Restart on bus release

7 DATOB (data out byte)

Clock and control

CLKOFF

- 0 No op
- 1 Turn clock off

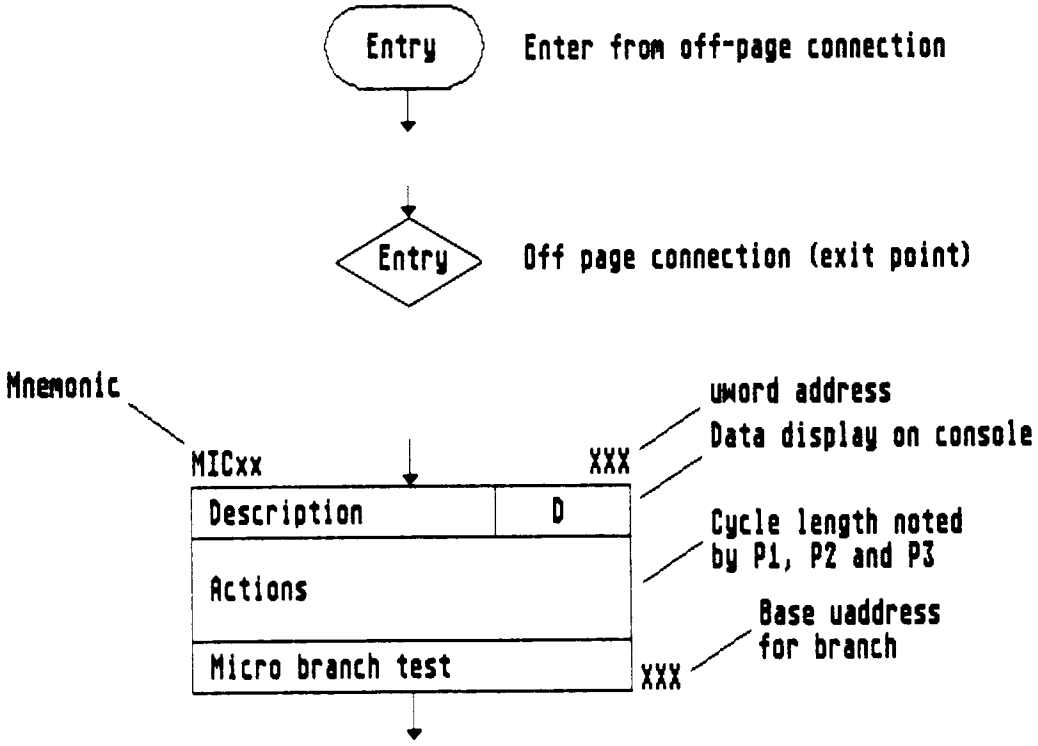
CLK

- 0 P1 140 nsec
- 2 P2 200 nsec
- 3 P3 200 + 300 nsec

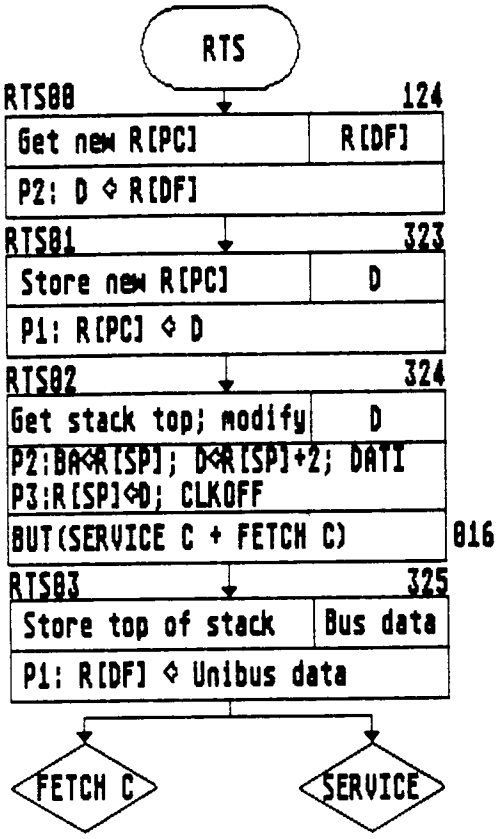
UBF<4:0> Micro branch field

UPF<7:0> Next micro address field

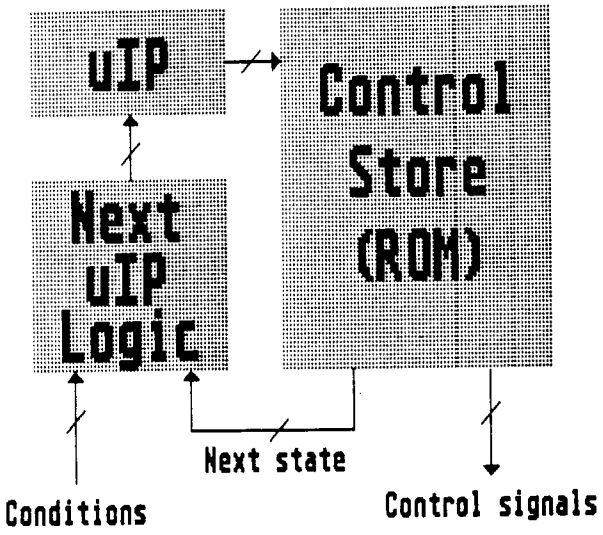
Fields # 7



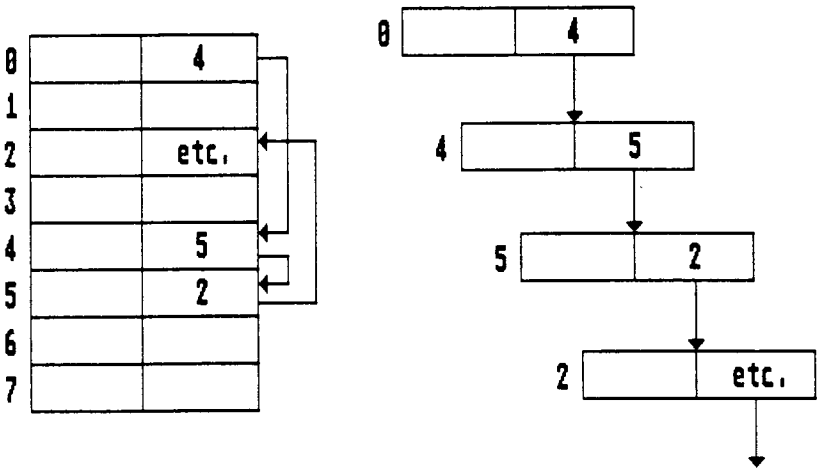
Flow symbol

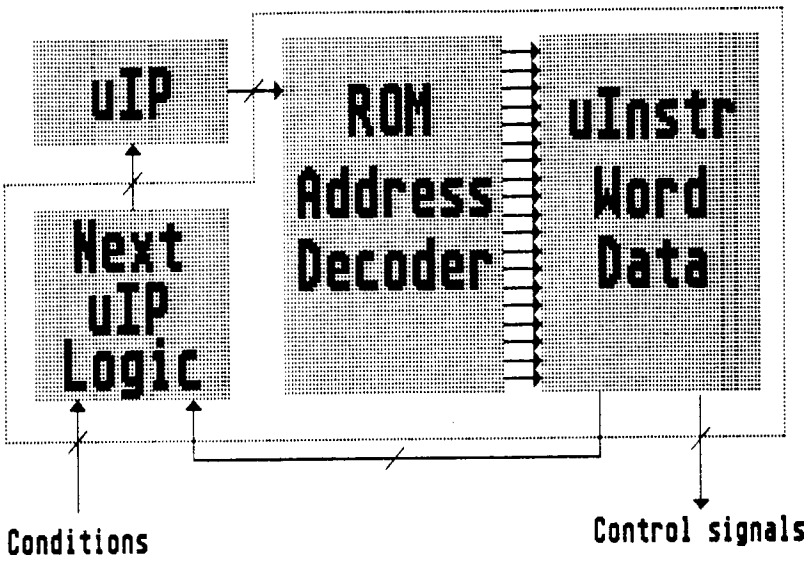

RTS
RTS00
124
Get new R[PC]
R[DF]
P2: D ◇ R[DF]
RTS01
323
Store new R[PC]
D
P1: R[PC] ◇ D
RTS02
324
Get stack top; modify
D
P2:BA◇R[SP]; D◇R[SP]+2; DATI
P3:R[SP]◇D; CLKOFF
BUT(SERVICE C + FETCH C)
816
RTS03
325
Store top of stack
Bus data
P1: R[DF] ◇ Unibus data
FETCH C
SERVICE

RTS

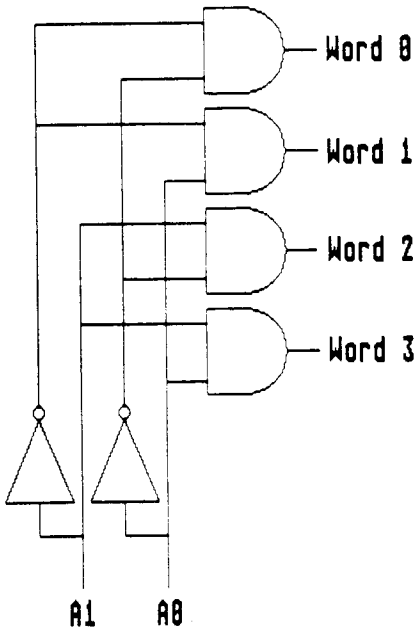


Next uaddress/linked list





2 to 4 decoder



- Expandable
- High true

ROM features

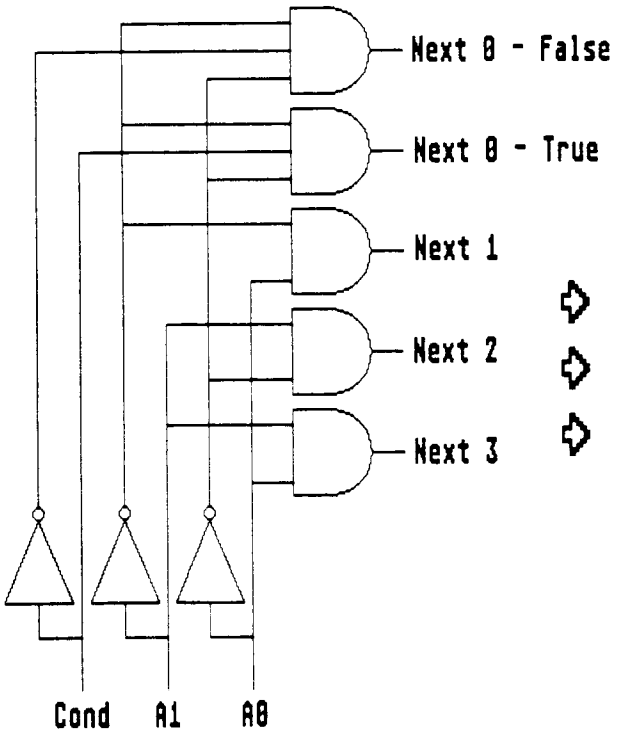
- One word line asserted at a time.
- Very regular decoder design.
- uInstr word data is programmable.

ROM disadvantages.

- Cannot ignore conditions without external logic.
- Redundant uinstruction values.

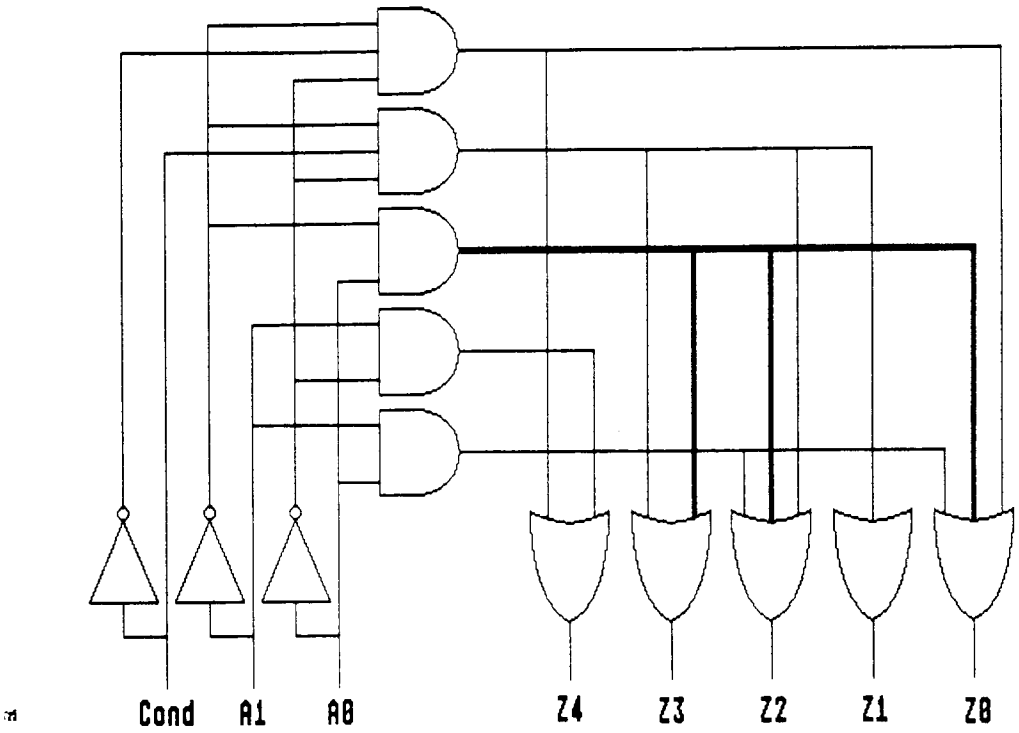
Solution ⇨ Make selection logic more general without losing good features.

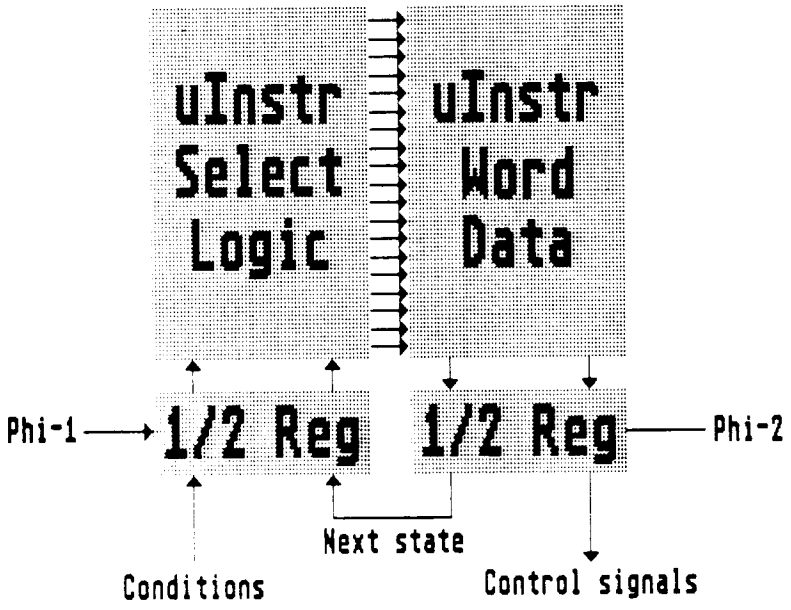
Selection logic



- Expandable
- Programmable
- One line active

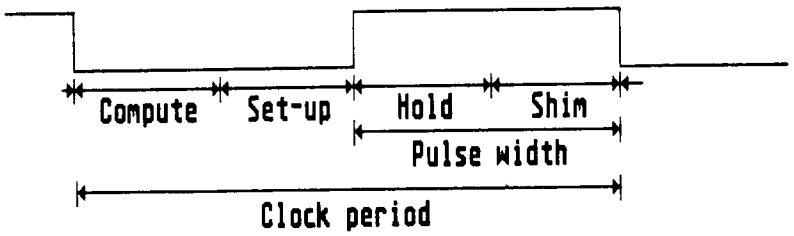
Select + uinstruction word data





Timing disciplines.

Edge driven (D-type flip/flops.)



Timing disciplines.

Two phase non-overlapping.