8051 I/O and 8255

Class 6

EE4380 Fall 2001

Pari vallal Kannan

Center for Integrated Circuits and Systems

University of Texas at Dallas

Why I/O Ports

- Controllers need to get external inputs and produce external outputs
- I/O ports serve the purpose
- 8051 has 4 built-in I/O ports
- If needed additional ports can be added by interfacing it with special devices at additional cost
- Too many ports increase pin-count and device cost. Too few makes it inadequate for complex control needs
- Generally, Ports are scarce and Port usage/allotment is an engineering decision

8051 I/O Ports

- 32 pins are allotted for 4 eight bit I/O ports
  - P0, P1, P2, P3
- At power-on all are output ports by default
- To configure any port for input, write all 1's (0xFF) to the port (source of confusion)
- Ports can be read and written to like normal registers

```
mov A, #55H        ; can use A
mov P0, A          ; write A to P0
mov P1, A
mov P2, #0AAH      ; can use immediate mode
xlr P1, #0FFH      ; read-modify-write (ex-or)
mov P0, #0FFH      ; configure P0 for input
```

- Ports can be bit manipulated (single bit addressable) using cpl and setb instructions

8051 I/O Ports – Pin Muxing

- Port pins are muxed with other signals
  - P0 : Also carry A0:A7 and D0:D7
  - P1 : dedicated
  - P2 : Also carry A8:A15
  - P3 : Also carry serial I/O (TxD, RxD), Timer inputs (T0, T1), external interrupts (INT0, INT1) and read write signals (RD, WR)
- For 8051 or DS5000, with no external memory, P0, P1 and P2 are available. For 8031, only P2 is available
- To increase the number of ports, use a parallel port interface chip like 8255

8051 I/O Programming

- Simple read and write 8bits at a time

```
mov A, #0FFH        ; configure P1 for input
mov P1, A           ;
mov A, P1           ; read from P1
mov P0, A           ; write that to P0
```

- Bit manipulation

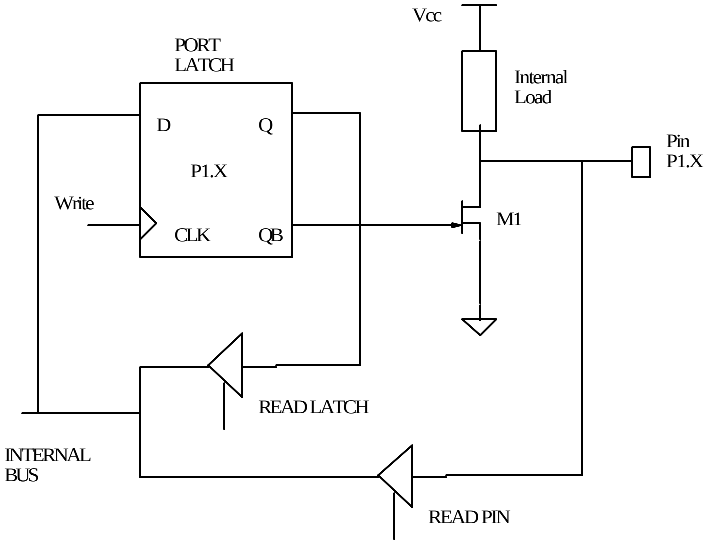
```
cpl P1.2            ; complement bit 2  of Port1
setb P1.3           ; set bit 3 for Port1 to 1
clr P0.0            ; clear bit 0 of Port0
```

8051 I/O Ports : Hardware Specs

- P0 is open drain.
  - Has to be pulled high by external 10K resistors.
  - Not needed if P0 is used for address lines
- P1, P2, P3 have internal pull-ups
- Some 8051 clones are available in 20 pin packages. They use a different muxing scheme
- Port fan-out (number of devices it can drive) is limited. Use buffers (74LS244, 74LS245, etc)
- P1, P2, P3 can drive up to 4 LS-TTL inputs
- P0 fan-out is dependant on the pull-up resistor value, limited by the max current it can sink on the output stage.

8051 I/O Ports : Input Quirks

- Port read instructions either
  - Read from the 8051 pins (“voltage” levels on the pins)
  - Read from an internal latch on the ports
- Writing 1 to the latch
  - Q=1, QB=0
  - M1 off
  - P1.x is available at tristate buffer
- Writing 0 to the latch
  - Q=0, QB=1
  - M1 ON
  - Input always gets 0
  - Can **damage** the port (M1) if P1.x is Vcc
  - Use 10K resistance between switch on P1.x and Vcc
  - Or use a SPST switch connected to GND



Input Quirks (contd.)

- Instructions that read the pins (READ_PIN is asserted)
  - mov A, Px
  - jnb Px.y …
  - jb Px.y …
  - mov C, Px.y
- Instructions that read the latch (READ_LATCH is asserted)
  - They read the last output value and not the value on the pins
  - [anl, orl, xrl] Px
  - [jbc, djnz] Px.y, target
  - [cpl, clr, setb] Px.y
  - [inc, dec] Px
  - mov Px.y, C

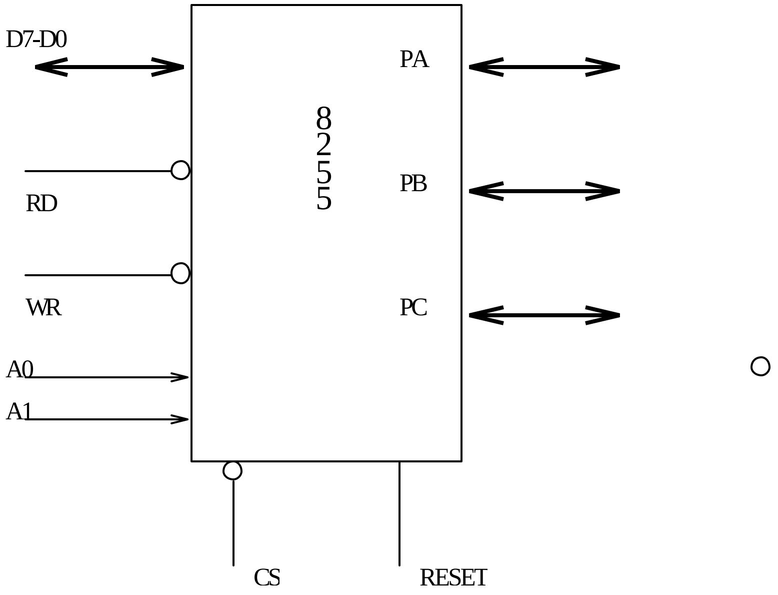
More ports on 8051 : The 8255

- Widely used I/O chip
  - 40 pins
  - Provides 3 eight bit ports PA, PB and PC
  - Port PC can be used as two 4 bit ports PCL and PCU
  - Ports have handshaking ability
  - Two address lines A0 and A1 and a Chip select CS
    - Address space of 4 bytes
    - 00b selects Port A
    - 01b selects Port B
    - 10b selects Port C
    - 11b selects a control register

8255 Functional Diagram

- CS is used to interface with 8051
- If CS is generated from lets say Address lines A15:A12 as follows,

  A15:A12 = 1000
- Base address of 8255 is
  - 1000 xxxx xxxx xx00b
  - 8000H
- Address of the registers
  - PA = 8000H
  - PB = 8001H
  - PC = 8002H
  - CR = 8003H



8255 Operating Modes

- Mode 0 : Simple I/O
  - Any of A, B, CL and CU can be programmed as input or output
- Mode 1: I/O with Handshake
  - A and B can be used for I/O
  - C provides the handshake signals
- Mode 2: Bi-directional with handshake
  - A is bi-directional with C providing handshake signals
  - B is simple I/O (mode-0) or handshake I/O (mode-1)
- BSR (Bit Set Reset) Mode
  - C alone is available for bit mode access

8255 Configuration

- Configured by writing a control-word in the CR register
- CR definition
  - D7 : 1→I/O mode, 0→BSR
  - D6,D5 : Mode selection for A and CU
    - 00→Mode0, 01→Mode1, 1x→Mode2
  - D4 : Port A control
    - 1→A input, 0→A output
  - D3 : Port CU control
    - 1→CU input, 0→CU output
  - D2 : Port B Mode selection
    - 0→B is in mode 0, 1→B is in mode 1
  - D1 : Port B control
    - 1→B input, 0→B output
  - D0 : Port CL control
    - 1→CL input, 0→CL output
- Refer to 8255 datasheet for additional options

8255 Usage: Simple Example

- 8255 memory mapped to 8051 at address 8000H base
  - PA = 8000H, PB = 8001H, PC = 8002H, CR = 8003H
- Control word for all ports as outputs in mode0
  - CR : 1000 0000b = 80H
- Code snippet

```
test:       mov A, #80H          ; control word
            mov DPTR, #8003H     ; address of CR
            movx @DPTR, A        ; write control word
            mov A, #55h          ; will try to write 55 and AA alternatively
repeat:     mov DPTR, #8000H     ; address of PA
            movx @DPTR, A        ; write 55H to PA
            inc DPTR             ; now DPTR points to PB
            movx @DPTR, A        ; write 55H to PB
            inc DPTR             ; now DPTR points to PC
            movx @DPTR, A        ; write 55H to PC
            cpl A                ; toggle A (55→AA, AA→55)
            acall MY_DELAY       ; small delay subroutine
            sjmp repeat          ; for (1)
```