Microcontroller Interface to Memory  Lab 3

— Internal memory
  • Access sequential data locations.
  • Use address *indirection*.
  • Perform memory test.

— External memory
  • Develop interface to external SRAM.
  • Use handshaking techniques from lab 2.
  • Perform memory test.
— Goal: access a number of sequential locations.
— Problem: locations are fixed in instructions.
— Instructions are programmed into EEPROM.

See Figure 2-7 in Peatman.
— Move value 0x1 to consecutive locations.
— Sequence of instructions.
— Approach is clumsy, long.

TEMP0 EQU 0x20
TEMP1 EQU 0x21
....
....
TEMP95 EQU 0x7F

movecode movlw 0x1
    movwf TEMP0
    movwf TEMP1
    ....
    ....
    movwf TEMP95
— Indirection register \textit{FSR} holds address.
— Instructions use \textit{INDF} as target “register”.

\textbf{See Figure 2-8 in Peatman.}
— Move value 0x1 to consecutive locations.
— Create a loop.
— Memory test involves write followed by read.

```
movlw 0x20          ; initialize first address
movwf FSR           ; set indirection register
movecode movlw 0x1   ; move test value to W register
movwf INDF          ; move value to indirect register
incf FSR,F          ; point to next data location
....                ; test if 0x80 reached
btfss STATUS, Z     ; if yes, then leave loop
goto movecode       ; restart loop
....                ; rest of code here
```
— 2 8Kx8 SRAM devices
— 1 74F373 tristate buffer
— 1 7032 PLD for control
— 1 16F877 PIC
Signal Interface

— Address A0-A12 - specifies data location.
— Data I/O1-I/O8 - data value.
— CS - chip select (enable).
— WE - write enable.
— OE - output enable.

See Page 2 in HM6264 data sheet.
Read Transaction

1. OE* asserted during transaction
2. address asserted
3. CS* asserted
4. data available (transfer to PIC)
5. CS*, OE* deasserted.
6. address deasserted.

See Fig 4.17 in Clements.
— Ordering of operations is very important.
— PLD generates OE*, CS*, WE*.
— PIC generates address.

See Fig 4.17 in Clements.
Write Transaction

SRAM device

1. OE* stays deasserted during transaction.
2. address asserted.
3. CS* asserted.
4. WE* asserted.
5. data driven by PIC (any time during 1-4).
6. CS*, WE* deasserted.
7. data and address deasserted.

See Fig 4.24 in Clements.
Write Transaction

— Ordering of operations is important.
— PLD generates CS, WE, OE signals.
— PIC generates address/data signals.

See Fig 4.24 in Clements.
— Use different address spaces to access data values.
— In our system: three different address spaces.
  1. 16F877 instruction space
  2. 16F877 data space
  3. external address space.
— External address space ranges from 0x0 to 0x1F
— 16 data values stored in each SRAM
— Top four data bits for each SRAM tied to ground through 4.7K resistors.
— Top nine address pins on each SRAM driven by ground.
Address Decoding Strategies

— Full address decoding

⋆ all addresses used in decoding.
⋆ detects out-of-range accesses.
⋆ requires extra hardware (especially pins).
⋆ more logic may lead to slower evaluation.

— Partial address decoding

1. only decode addresses that select between devices.
2. out-of-range accesses not detected.
3. careful design needed.
4. fewer pins/gates.
— Only one OE signal should be active at a time.
— Unused address/data signals tied off.
— PLD generates CS, WE, OE signals.
— CS1* pin on SRAMs tied to ground.
— PICSTROBE indicates address/data transfer.
— PLDSTROBE indicates data transfer response.
— Note that SRAMs are self-timed
— PICSTROBE and PLDSTROBE allow for handshaking
— Example - read transaction
  * PICSTROBE assert - ”Valid address on the bus”
  * PLDSTROBE assert - ”Valid data on the bus”
  * PICSTROBE deassert - ”Got the data”
  * PLDSTROBE deassert - ”All control signals deasserted”
PLD Operation

- PLD operation can be modeled as a state machine
- PLD receives PICSTROBE to start operation.
- PLD generates SRAM control signals in proper order
- Data transfer occurs.
- PLD send acknowledgement in form of PLDSTROBE
- Each group can choose their own protocol.
Goals of the Assignment

— Test a block of internal or external memory
  * 0x20-0x7F internal
  * 0x0-0x1F external
— Terminal prompt asks for two pieces of information
  1. internal/external access
  2. value to be written
— PIC code writes value to chosen memory locations.
— PIC code reads values from locations and compares against original.
— If test is successful, positive message displayed on terminal.
— If failure, message indicates which memory, address, expected value, and actual value.
— Implementation should be robust.
Steps for Assignment Completion

— Complete internal memory test using FSR
— Develop PLD code for write transaction
  * Observe transaction on logic analyzer.
— Develop PLD code for read transaction
  * Observe transaction on logic analyzer.
— Integrate code together.
Things to Keep In Mind

— Only one chip should drive the external bus at a time.

— SRAM interface signals should be asserted and deasserted in the correct order.

— The logic analyzer is your friend.

— Use simple tests to check hardware functionality.

— Get started early.
Reading

— Clements handout.
— Peatman book - pages 18-21, 131
— HM6264B data sheet
— 16F877 data sheet - page 27
— Assignment focus on memory interfacing.
— Internal memory tested via indirect addressing.
— External interface constructed via buffers and PLDs.
— Data value written to a sequence of locations.
— Read and written results tested by PIC
— Test status sent to terminal.