Embedded Microprocessors Group Project

Dr. L. K. Verma

Email: Dr.Lalit@gmail.com

Coventry University

Assignment Submission Dates

- Assignment 1 − 19 Jan 2018
- Assignment 2 − 6 March 2018

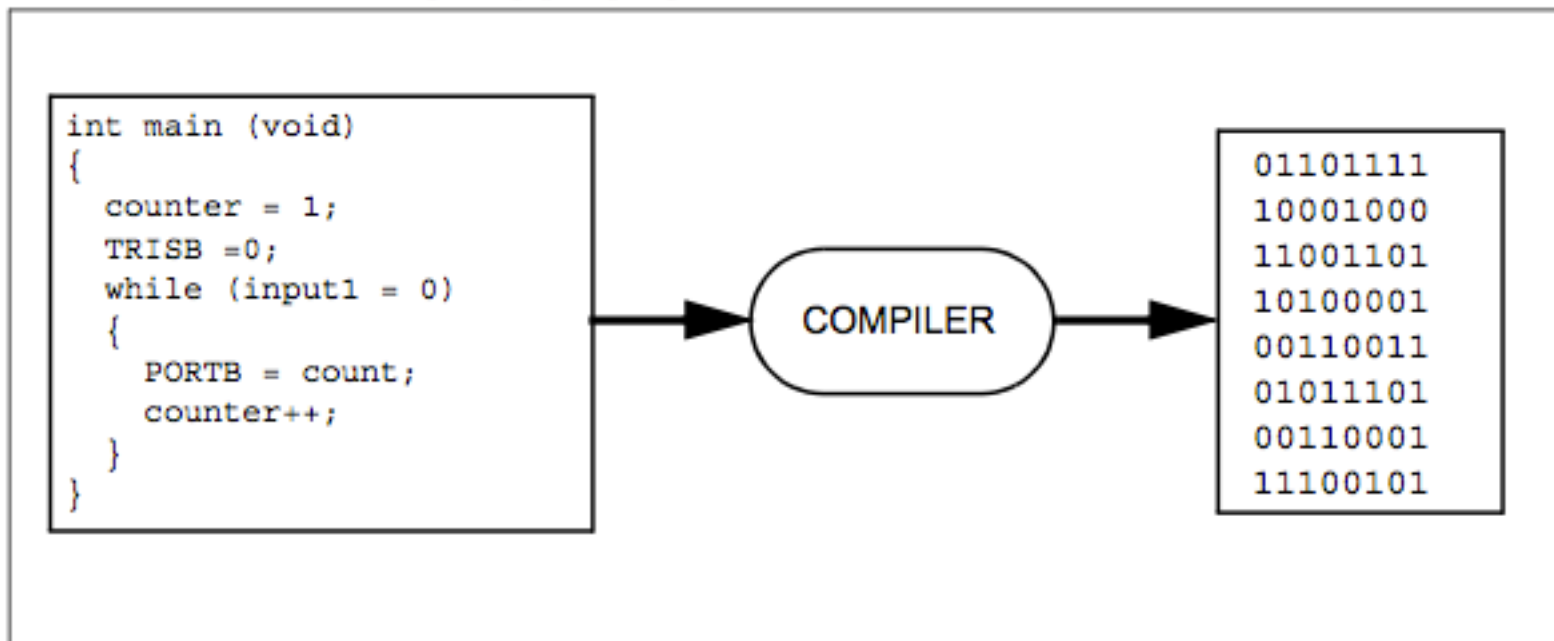# Embedded Microprocessors Group Project

- Introduction to MPLAB IDE
- Introduction to Microchip Board, Pickkit 3
- Test a program using Simulator
- Test a program using Microchip Board
- Create a delay loop
- Program to blink LED on/off
- Setting up configuration bits

Coventry University

Introduction to MPLAB IDE

- Download MPLAB-X IDE
  - http://www.microchip.com/mplab/mplab-x-ide

- Download XC8 compilers
  - http://www.microchip.com/mplab/compilers

  - http://ww1.microchip.com/downloads/en/DeviceDoc/50002053G.pdf

XC compilers

FIGURE 1-8: **A COMPILER CONVERTS SOURCE CODE INTO MACHINE INSTRUCTIONS**

```
int main (void)
{
  counter = 1;
  TRISB =0;
  while (input1 = 0)
  {
    PORTB = count;
    counter++;
  }
}
```

COMPILER

```
01101111
10001000
11001101
10100001
00110011
01011101
00110001
11100101
```

Coventry
University

5

Introduction to Microchip Board

PIC18F87J11

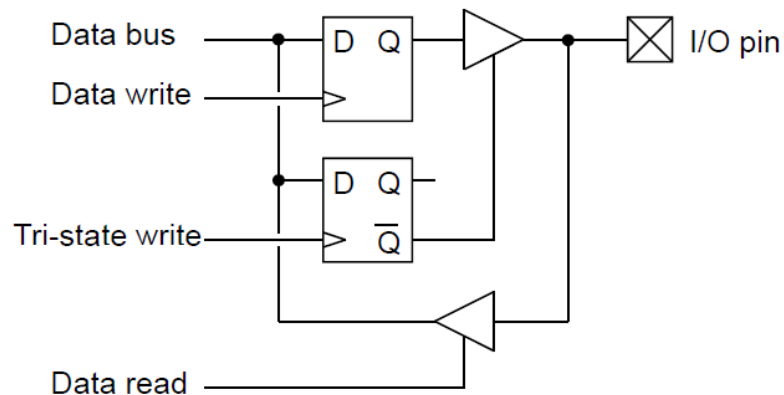http://ww1.microchip.com/downloads/en/DeviceDoc/50001721C.pdf
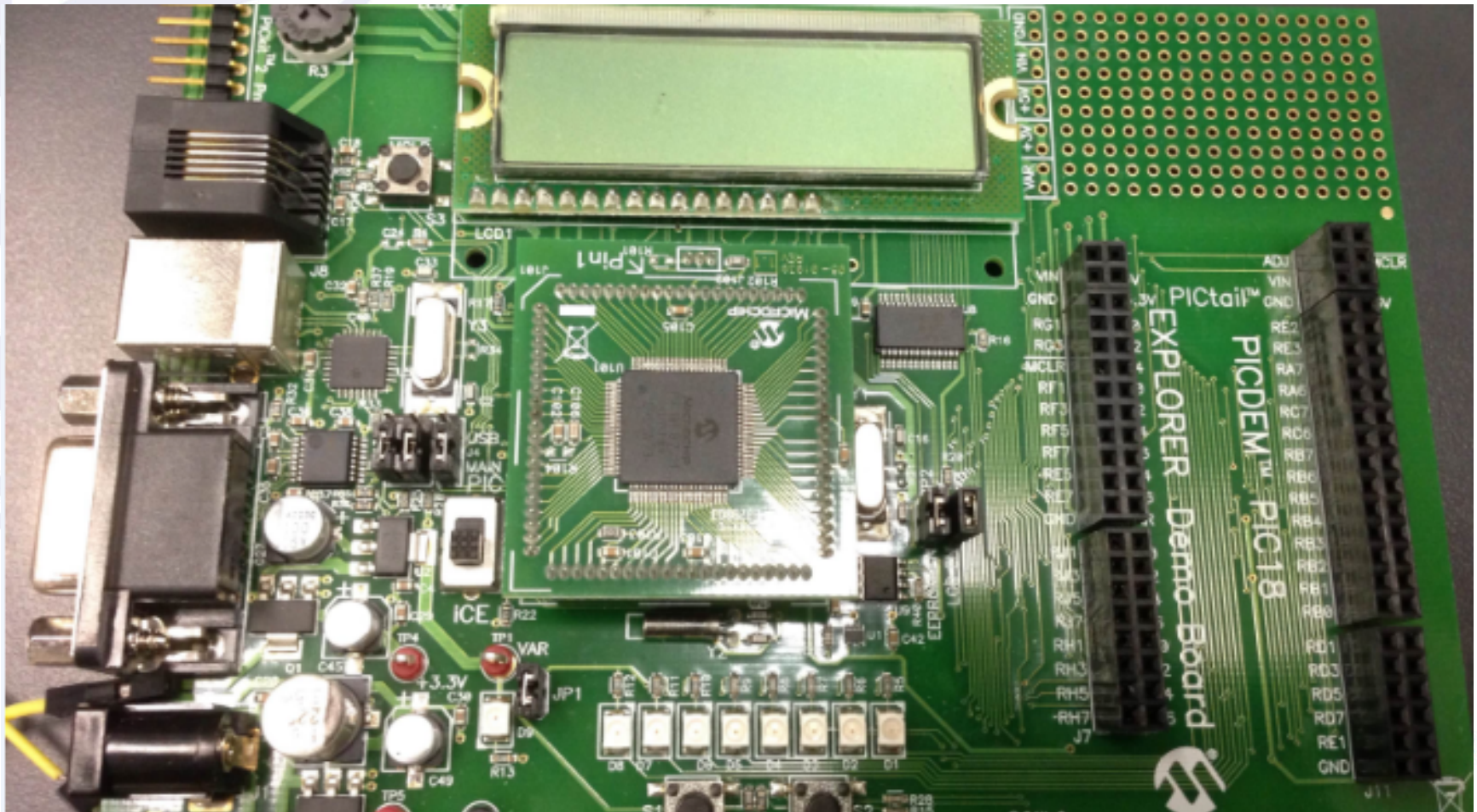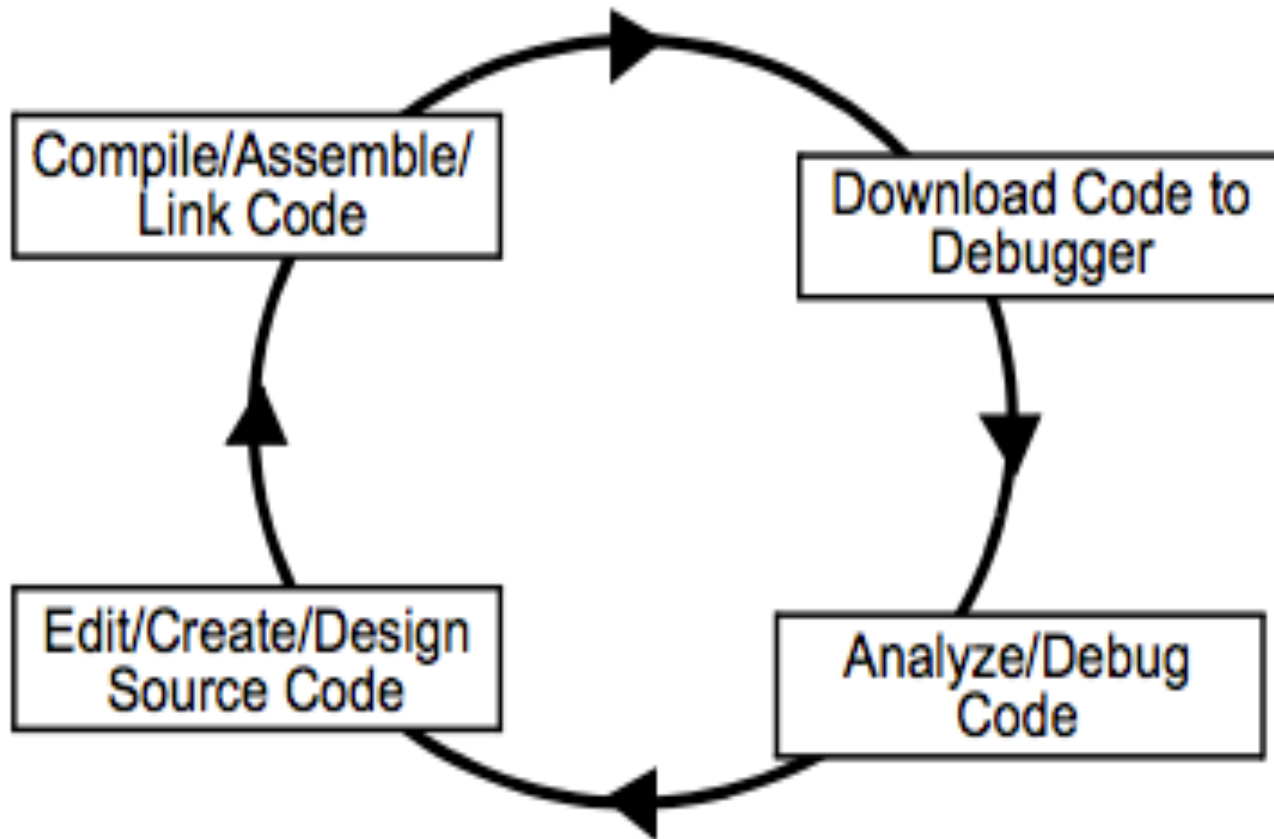
Use Port Or Latch?

The latch is the *output* latch onto which values are written. The port is the voltage at the actual pin

- If it's an Input pin you must use PORT
- If its an output and you want to know what you have written to -> use LAT
- If its an output and you want to know the actual state at the pin -> use PORT

Simplified diagram representing a single data I/O pin of one of the ports A-E:

Data bus

Data write

Tri-state write

Data read

D Q

D Q

$\overline{Q}$

I/O pin

Design Cycle

Compile/Assemble/Link Code → Download Code to Debugger → Analyze/Debug Code → Edit/Create/Design Source Code → (cycle)

PIC Family to use

- Soldered Prototype − PIC18F4550
- PIC Development board − PIC18F87J11

Coventry
University

Sample test – Turn ON PortD LEDs

```c
void main()
    {
        PORTA = 0x00; //set RA0-RA5 low
        TRISA = 0x00; //set PORTA to output
        //superloop
        while(1)
        {
            PORTA = 0xFF;
        }
    }
```

Implement a Delay

```
#define _XTAL_FREQ 8000000
void delay1s (void);


void delay1s (){
    for (int i=0; i<50; i++){
        _delay_ms(20);
    }
}
```

Test Input Port, If loop

```
void main(void) {
    TRISB=1;  // port B is input
    TRISD=0;  // port D is output
    LATD=0;

    Motor = 0;  // turn the motor off

    while(1){
        if (Button == 0){  // it means button is pressed
        Motor = 1;
        }
    }
    return;
}
```