# 16.317: Microprocessor Systems Design I

Summer 2016

Homework 5

Due **1:00 PM, Monday, 6/20/16**

**Notes:**

- While typed submissions are preferred, handwritten submissions are acceptable.
- All solutions must be legible and contained in one file. Archive files are not acceptable.
- Electronic submissions should be e-mailed to Dr. Geiger at Michael_Geiger@uml.edu. Please include your name as part of your filename (for example, mgeiger_hw6.pdf).
- This assignment is worth a total of 100 points.

For each of the following complex operations, write a sequence of PIC 16F1829 instructions that performs an equivalent operation. Assume that X, Y, and Z are 16-bit values split into individual bytes as shown in the following cblock directive, which defines two additional variables you can use:

```
cblock 0x70
     XH, XL     ; High and low bytes of X
     YH, YL     ; High and low bytes of Y
     ZH, ZL     ; High and low bytes of Z
     TEMP       ; Temporary byte, if needed
     COUNT      ; Loop counter, if needed
endc
```

Each question on this assignment is worth 20 points.

1. Perform the 16-bit addition: X = Y + Z. Do not change Y or Z when performing this operation.

2. Perform the 16-bit subtraction: X = Y – Z. Do not change Y or Z when performing this operation.

3. Perform a 16-bit arithmetic right shift: X = Y >> ZL. (Note that, because the shift amount is no greater than 15, a single byte is sufficient to hold that value.) Do not change Y or ZL when performing this operation.

4. Given an 8-bit variable, YL, perform the multiplication:

   YL = YL * 10

Hint: Note that multiplication by a constant amount can be broken into a series of shift and add operations. For example, in general:

- X * 2 can be implemented by shifting X to the left by 1 (X << 1)
- X * 5 can be implemented as (X * 4) + X = (X << 2) + X

5. Given two 8-bit variables stored in XL and YL, copy the value of bit position YL within variable XL into the carry flag. For example:

- If XL = 0x03 and YL = 0x00, set C to the value of bit 0 within XL.
  - Since XL = 0x03 = $0000\ 001\underline{1}_2$, C = 1
- If XL = 0xC2 and YL = 0x04, set C to the value of bit 4 within XL.
  - Since XL = 0xC2 = $110\underline{0}\ 0011_2$, C = 0

Note that:

- This operation is very similar to the bit test (BT) instruction in the x86 architecture.
- Since YL is not a constant, you cannot use the value of YL directly in any of the PIC bit test instructions (for example, `btfsc XL, YL` is not a valid instruction).
- Your code should not modify either XL or YL.