**CprE 288 – Homework Question Set 3**

## Question 1: GPIO Programming

Assume that any GPIO ports used have been initialized properly.

1. Write code that copies the GPIODATA value from Port A to Port B and clears bit 0 of Port B (PB0). Don’t change the value of Port A.

GPIO\_PORTB\_DATA\_R = GPIO\_PORTA\_DATA\_R & 0xFE;

1. The DATA register for Port C is at address 0x400063FC. Write code for the port\_ptr pointer variable to access the Port C DATA register and read the value of the DATA register into my\_data. **Do not use any register macros from the system header file for part b.**

unsigned long \*port\_ptr;

uint8\_t my\_data;

//your code here

port\_ptr = 0x400063FC;

my\_data = \*port\_ptr;

1. Write a function that copies the value of *any* port’s DATA register into any other port’s DATA register. It should use two call-by-reference parameters, and the function would be called by passing in addresses of (or pointers to) the DATA registers for the two ports.

Void copyPortData(unsigned long \*port1, unsigned long \*port2) {

\*port2 = \*port1;

}

## MYQuestion 1: GPIO Programming

Assume that any GPIO ports used have been initialized properly.

1. Write code that copies the GPIODATA value from Port A to Port B and clears bit 0, 2, and 7 of Port B (PB0, PB2, PB7). Don’t change the value of Port A when putting the value in Port B. After clear all bits in port A.

GPIO\_PORTB\_DATA\_R = GPIO\_PORTA\_DATA\_R & 0x7A;

GPIO\_PORTA\_DATA\_R &= 0x0;

1. The DATA register for Port E is at address 0x40024000. Write code for the port\_ptr pointer variable to access the Port E DATA register and read the value of the DATA register into my\_data but clearing bits 0 and 1 for my\_data. **Do not use any register macros from the system header file for part b.**

unsigned long \*port\_ptr;

uint8\_t my\_data;

//your code here

port\_ptr = 0x40024000;

my\_data = \*port\_ptr & 0x2;

1. Write a function that swaps the value of *any* two given port’s DATA register. It should use two call-by-reference parameters, and the function would be called by passing in addresses of (or pointers to) the DATA registers for the two ports.

Void swapPortData(unsigned long \*port1, unsigned long \*port2) {

Unsigned long temp = \*port2;

\*port2 = \*port1;

\*port1 = temp;

}

B) I believe this question deals with serial communication which entails inputs, outputs, interfaces, and ports and data registers for ports. This concept appears in this question through working with different GPIO data ports and setting their data to different variables and also using masks. This covers the ports part of the serial communication. It also can be seen very clearly that the Data portion of the GPIO register is used in Part A and Part B.

C) I used the Tivia data sheet to let me know which bits were reserved and which ones were allocated for Data in the GPIO register. Upon finding the section in the GPIODATA section on page 665 it let me know that it uses the last 8 bits, 0-7. This let me know the correct number of bytes to use when I was masking the data. I used table 10-2 to verify that I was calling the right pins that lined up with the GPIO port. Finally I took the base address for GPIO PORTE from the data sheet on page 659.