CPE 310L
EMBEDDED SYSTEM DESIGN (CPE)

LABORATORY 11
INTER-INTEGRATED CIRCUIT

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
UNIVERSITY OF NEVADA, LAS VEGAS

GOAL
The goal of this lab is to interface the SPI chip to the AVR microcontroller. Follow the schematics and the sample code provided to construct this lab.

BACKGROUND
I2C
Inter-INTEGRATED Circuit is another serial communication protocol used to transfer data between two ICs. In AVR I2C is a two wire interface relying on the Serial Data (SDA) and Serial Clock (SCL) pins. Like the SPI protocol, I2C supports master-slave relation among devices. One of the main design advantages is that ICs can be removed and added without affecting the other ICs in the system.

AVR REGISTERS FOR TWO-WIRE INTERFACE (TWI)

TWAR (TWI Slave Address Register)
Holds the 7-bit Slave address which the TWI will respond when programmed as a Slave Transmitter or Receiver (not needed in Master Mode.)

TWBR (TWI Bit Rate Register)
Bit rate generator is a frequency divider which generates the SCL clock frequency in the Master mode

TWCR (TWI Control Register)
Used to control the various operations of the TWI. Examples include start and stop conditions, write collision, and Receiver acknowledge

TWDR (TWI Data Register)
Contains the next byte to be transmitted or last byte that was received

TWSR (TWI Status Register)
Reflects the status of the TWI logic
CODE

```c
#include <avr/io.h>

// i2c functions
void i2c_stop()
{
    //write 1 to TWINT to clear interrupt flag
    //enable TWI
    //write 1 to enable stop condition
}

void i2c_write(unsigned char data)
{
    //copy data to TWI Data register
    //write 1 to TWINT to clear interrupt flag
    //enable TWI
    //loop here if TWINT is 1
}

void i2c_start(void)
{
    //write 1 to TWINT to clear interrupt flag
    //enable TWST A
    //enable TWI
    //loop here if TWINT is 1
}

void i2c_init(void)
{
    // set prescaler bits to 0
    // SCL freq. is 50k for XTAL = 8M
    // enable TWI module
}

// Setting Date
int main(void)
{
    // i2c initialization
    i2c_init(); // initialize I2C module
    i2c_start(); // transmit START condition
    i2c_write(0b11010000); // set register pointer to 7
    i2c_write(0x07); // set value of location 7 to 0
    i2c_stop(); // transmit STOP condition

    for (int k = 0; k < 100; k++) // wait for a short time
    {
        i2c_start(); // transmit START condition
        i2c_write(0b11010000); // address DS1307 for write
        i2c_write(0x04); // set register pointer to 4
        i2c_write(0x19); // set date to 0x10 = 19 BCD
        i2c_write(0x10); // set month to 0x10 = 10 BCD
        i2c_write(0x09); // set year to 0x09 = 09 BCD
    }

    while(1) // stop here
    return 0;
}
// Setting Time
int main(void)
{
    // i2c initialization
    i2c_init(); // initialize I2C module
    i2c_start(); // transmit START condition
    i2c_write(0b11010000); // transmit START condition
    i2c_write(0x07); // set register pointer to 7
    i2c_write(0x00); // set value of location 7 to 0
    i2c_stop(); // transmit STOP condition

    for (int k = 0; k < 100; k++) { // wait for a short time
        i2c_start(); // transmit START condition
        i2c_write(0b11010000); // address DS1307 for write
        i2c_write(0); // set register pointer to 4
        i2c_write(0x55); // set date to 0x10 = 19 BCD
        i2c_write(0x56); // set month to 0x10 = 10 BCD
        i2c_write(0b00010110); // set hour = 16 in 24 hours mode
        i2c_stop(); // transmit STOP condition

        while(1); // stop here
    }
    return 0;
}

PRELAB

1. DS1307 operation
1. Describe the operation of DS1307 chip.
2. Explain the function of all the register mentioned in background section of this document.

EXPERIMENTS

1. The use of DS1307
Program the current date and time onto the DS1307 chip. Use the microcontroller to read the current date and time and display it to the screen every 30 seconds (use USART/UART interface).
POSTLAB

Include the following elements in the report document:

<table>
<thead>
<tr>
<th>Section</th>
<th>Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Theory of operation</td>
</tr>
<tr>
<td></td>
<td><em>Include a brief description of every element and phenomenon that appears during the experiments.</em></td>
</tr>
<tr>
<td>2</td>
<td>Prelab report</td>
</tr>
<tr>
<td>3</td>
<td>Results of the experiments</td>
</tr>
<tr>
<td></td>
<td><strong>Experiment</strong></td>
</tr>
<tr>
<td>1</td>
<td>a. Code with comments</td>
</tr>
<tr>
<td></td>
<td>b. Picture of the circuit</td>
</tr>
<tr>
<td>4</td>
<td>Answer the questions</td>
</tr>
<tr>
<td></td>
<td><strong>Question no.</strong></td>
</tr>
<tr>
<td>1</td>
<td>What is I2C?</td>
</tr>
<tr>
<td>2</td>
<td>What is TWI?</td>
</tr>
<tr>
<td>3</td>
<td>List the uses of SPI.</td>
</tr>
<tr>
<td>5</td>
<td>Conclusions</td>
</tr>
<tr>
<td></td>
<td><em>Write down your conclusions, things learned, problems encountered during the lab and how they were solved, etc.</em></td>
</tr>
</tbody>
</table>

![Fig. 1. DS1307 configuration](image-url)