I2C Subsystem

1. A master begins communication by transmitting a single start bit. How can the slave(s) devices differentiate between a start bit and a data bit?

   ANSWER: A start bit is generated by changing the state of the data line (SDA) from high to low while the clock line (SCL) is high. In contrast, transitions between two data states, for example from 1 (high) to 0 (low), will only occur when the clock line (SCL) is low.

2. How does the I2C interface protocol define packet transmission stop?

   ANSWER: A stop bit is generated by changing the state of the data line (SDA) from low to high, again only while the clock line (SCL) is high.

3. How does the I2C interface protocol define a bit of data (1 or 0)?

   ANSWER: A bit of data is defined by a clock pulse. The data line must be stable during this period of time.

4. How many bits comprise a data packet?

   ANSWER: 9

5. How many slaves can be addressed by a master?

   ANSWER: The slave address (SLA) is 7-bit wide, suggesting up to 2^7 slaves. However, address is reserved, so in practice only 127 slaves may be addressed.

6. What is the bit ordering for transmitting both an address and data?

   ANSWER: Most significant bit (MSB) first

7. What is the difference between a packet and a session?

   ANSWER: A single session is defined by the master sending an address (SLA-R/W) packet and reading or writing one or more data packets.

8. Once a slave decodes the 7 address bits and determines the master is talking to it, how does the slave know if master is going to Read data from the slave or Write data to the slave?

   ANSWER: After the 7 address bits, the master will transmit a read bit (SLA+R) or a write bit (SLA+W).

9. Once a slave decodes the 7 address bits and learns the direction of communications (SLA+R/W), it acknowledges (ACK) that it is ready by setting to zero the data line (SDA). This is the 9th bit in the address packet. How can the slave bring this line low without

generating contention on the SDA line? In other words how can both the master and a slave use the same line to both send and receive data?

ANSWER: Each of the two wires (SDA and SCL), include a pull-up resistor. A slave device writing to SDA only needs to pull the line low for each ‘0’written; otherwise, it leaves the line alone to write a ‘1’, which occurs due to the lines being pulled high externally by the pull-up resistor. This is commonly known as a wired-AND configuration. Here is a link where you can learn more about wired-AND and wired-OR configurations. http://en.wikipedia.org/wiki/Open_collector

10. When the slave sends data to the master (SLA-R), the slave controls the data line (SDA), who generates the clock pulse?

ANSWER: The master always generates the clock pulse.

11. If the master always generates the clock pulse, then why does the clock line (SCL) use an open-collector output configuration allowing multiple devices to share this bus line?

ANSWER: The I2C interface protocol allow multiple masters.

Use **Figure 21-9 “**Overview of the TWI Module” found in the ATmega328P datasheet and in the lecture material, to help answer the next few questions.

12. Assuming that the ATmega328P is the master, which block or blocks comprising the I2C module, can be ignored?

ANSWER: The Address Match Unit is used in conjunction with the 7-bit address portion of the SLA-R/W packet to determine if the master is addressing the slave. If the ATmega328P is the master, then this feature is not required.

13. Which registers comprising the I2C module can be accessed by the AVR microprocessor?

ANSWER: Address/Data Shift Register (TWDR), Bit Rate Register (TWBR), Address Register (TWAR), Status Register (TWSR), and the Control Register (TWCR).

14. At what address can you find these registers?

ANSWER:

0xBC	Control Register (TWCR) x
0xBB	Address/Data Shift Register (TWDR)
0xBA	Address Register (TWAR)
0xB9	Status Register (TWSR)
0xB8	Bit Rate Register (TWBR)

15. Is this an I/O Address or an Extended I/O Address? In other words, is the address in the I/O address space, the data memory address space, or both?

ANSWER: The I2C module registers are outside the I/O address space of the AVR processor and are in the Extended I/O Address space.

16. Before you can use a microcontroller subsystem, it needs to be configured. Which registers would you write to in order to configure the I2C subsystem?

    ANSWER: TWBR and TWAR (if ATmega is acting as a slave device)

In the next series of questions we are going to look at both the Arduino code to work with the digital compass and the more generic C++ code to write a single packet of data (START, SLA+W, DATA, STOP).

17. Given the following three Arduino “Wire” instructions. What part of the C++ code example (see I2C Lecture C++ Write example) corresponds to the first Arduino instruction (`Wire.beginTransmission`).

```
// step 1: instruct sensor to read echoes
   Wire.beginTransmission(compassAddress); // transmit to device
   Wire.send('A'); // command sensor to measure angle
   Wire.endTransmission(); // stop transmitting
```

ANSWER:

```
TWCR = (1<<TWINT)|(1<<TWSTA)|(1<<TWEN);
while (!(TWCR & (1<<TWINT)))
;
if ((TWSR & 0xF8) != START) ERROR();
TWDR = SLA_W;
TWCR = (1<<TWINT) | (1<<TWEN);
while (!(TWCR & (1<<TWINT)))
;
if ((TWSR & 0xF8) != MT_SLA_ACK) ERROR();
```

18. The C++ instruction `TWDR = SLA_W;` loads the Address/Data Shift Register with the address of the slave plus clear the R/W bit. For the compass example, what value would be loaded into the TWDR register?

    ANSWER: The 8-bit compass address is 66 (0x42). We use only the high 7-bits to form the SLA part of the SLA+W packet or 33 (0x21). To this address we concatenate the Write bit (logic 0). Let’s look at this in binary

    SLA_W = 010 0001 _0 = 0100 0010 = 0x42.

19. What part of the C++ code corresponds to the following Arduino code?

```
Wire.send('A');          // command sensor to measure angle
```

ANSWER:

```
TWDR = DATA;
TWCR = (1<<TWINT) | (1<<TWEN);
```

```
while (!(TWCR & (1<<TWINT)))
;
if ((TWSR & 0xF8) != MT_DATA_ACK) ERROR();
```

20. The C++ instruction `TWDR = DATA;` loads the Address/Data Shift Register with the data to be written to the slave. For the compass example, what value would be loaded into the TWDR register?

    ANSWER: The argument sent to the `send` method is the ASCII Character 'A" which equals 0x41.

21. What part of the C++ code corresponds to the following Arduino code?

    ```
    Wire.endTransmission(); // stop transmitting
    ```

    ANSWER:

    ```
    TWCR = (1<<TWINT)|(1<<TWEN)|(1<<TWSTO);
    ```

22. What would the SCL clock frequency be set to, on an Arduino operating at a frequency of 16MHz, and assuming TWBR = 0x40 and TWSR bits TWPS1 = 1 and TWPS0 = 0.

    ANSWER: 16 MHz / (16 + 2(64)(16)) = 7.7519 KHz