

Lab 6: Serial Communication (UART, SPI, and I2C)

Micah Galos - SID: 862082339

Joseph Ayala - SID: 862080244

Due: November 30th, 2020 - 11:59PM

Abstract

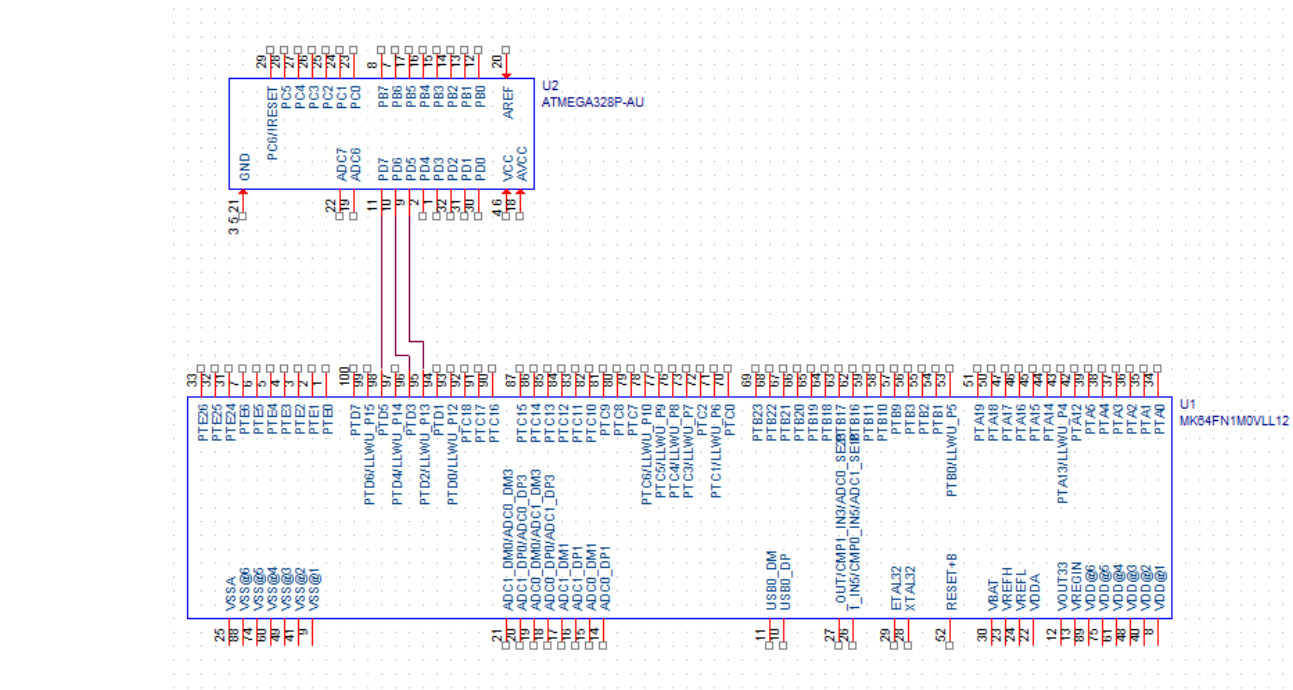
Within the course of this lab we were introduced to further implementation of the K64F microcontroller in applying what we have learned with Inter-Integrated Circuit communication systems and the onboard capabilities of the Freedom board. Within the Freedom board there is an embedded accelerometer and magnetometer we can access through the Kinetis software. From this we can get familiar with the sensors onboard the microcontroller and extend it outward to other peripherals that can be added to this microcontroller or other types for instance.

Experiment System Specification

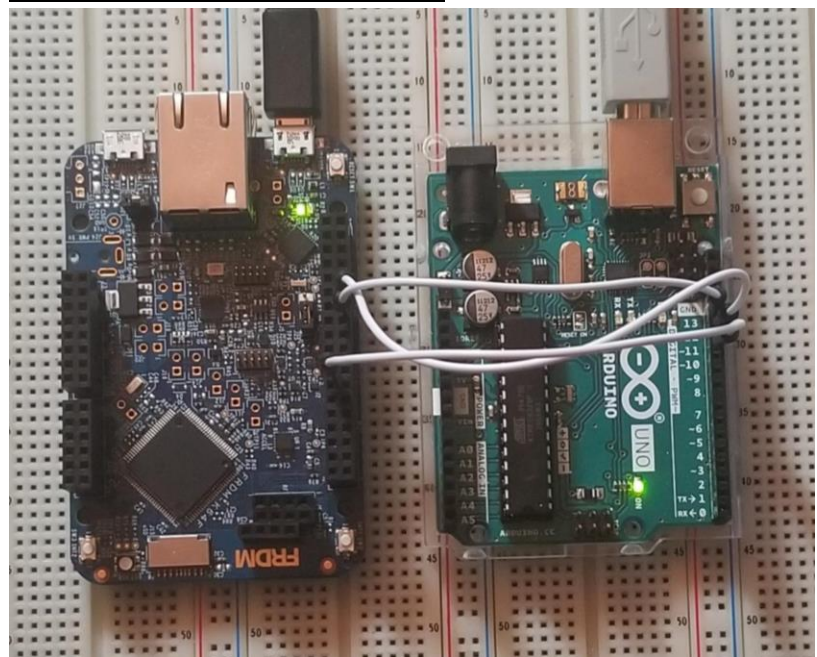
Moving forward in this lab, we adhere closely to the lab manual in downloading a new tool to access the sensors on board the Freedom microcontroller and view the serial terminal. With this tool we can observe the individual channels that are the clock and data lines for each of the onboard sensors. From these tools we are able to set up and observe the individual baud rates from the sensors to the microcontroller and the respective communication terminal.

As for the second portion of this lab we will be communicating between the Freedom microcontroller and the Arduino Microcontroller to practice implementing a different type of communication protocol. This protocol in particular is Serial Peripheral Interface (SPI) which sets up a Parent and Child relationship between the sensor/component with the microcontroller.

Through this implementation of SPI communication we will be adhering again to the lab manual to ensure that the SPI communication from the Arduino microcontroller is received on the Freedom microcontroller.



Board and Circuit

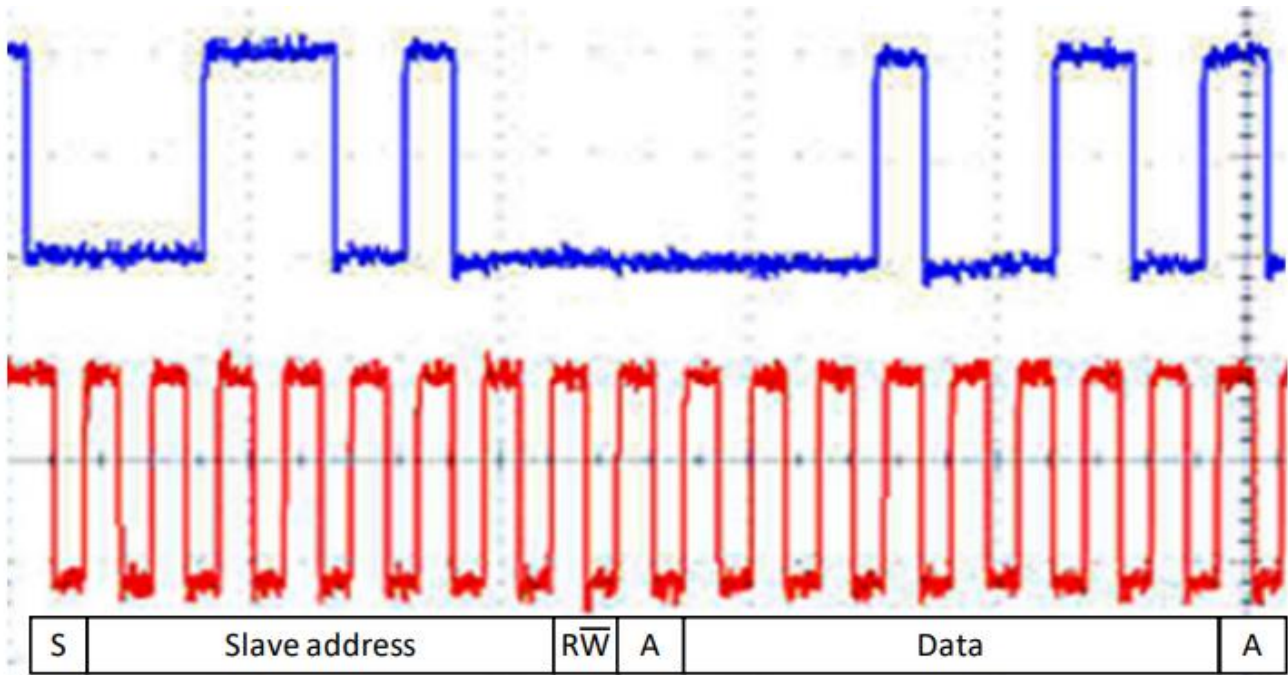


Technical Problems

No technical problems occurred during the process of finishing the lab.

Questions

2. An I2C communication has been captured by a mixed signal oscilloscope. SDA is the blue waveform and SCL is the red one. The address and data portions are indicated in the picture



a) What is the address of the slave?

From Blue Waveform

In binary	As Hex
00110100	0x36

b) What is the data value being transferred? Is the master reading or writing the data?

In binary	As Hex
00010101	0x27

The master is writing the data

Conclusion

Overall, the lab process was a success. Setting up the components per each part in their respective projects was quite tedious. After uploading starter code and loading up the debugger, I was amazed by the output results. The position of the accelerometer changes when shifting the K64F board in both parts was shown to be successful.