# Answers to questions:

## Explain the SPI communication protocol with a timing diagram

Serial Peripheral Communication (SPI) is used to send data between a microcontroller and peripherals. For this lab the Raspberry Pi used SPI to communicate with an ADC. The SPI protocol is implanted with a master and a slave, where the master controls the clock and decides when to communicate.

In the diagram below is shown a generic timing diagram for SPI communication.

|  |
| --- |
|  |

<<< reference for diagram: author -Mikegrusin,

Website - Sparkfun

Name - Serial Peripheral Interface (SPI)

Date accessed – 1/09/2018

URL - https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi>>>

As seen in the image, the master and slave are connected with four lines, SCK, MOSI (Master Out Slave In), MISO (Master In Slave Out), and SS. SCK is the clock sent from the master so that both master and slave know when to sample the data sent to them respectively. MOSI is the serial data channel from master to slave, while MISO is the serial data channel from slave to master. SS is the channel used by the master to select or enable the slave so that it is ready to receive and send data.

Below is the timing diagram taken from the MCP3008 ADC’s datasheet for how it receives the commands from the microcontroller.

|  |
| --- |
|  |

<<reference for diagram: author -Microchip,

Name - MCP3004/3008 datasheet,

URL - <https://pdf1.alldatasheet.com/datasheet-pdf/view/194715/MICROCHIP/MCP3008.html>>>>

As seen above, for our purposes the master will send 4 bits stating the mode of operation (single or differential inputs) in the first bit and the last three select the channel to be sampled.

## Define interrupt and threaded call-back in the context of an embedded system.

<<< reference for information found on how to use interrupts

Website – raspi.tv

Name – How to use interrupts with python on the raspberry pi and rpi GPIO part 2

Date accessed – 1/09/2018

URL - https://raspi.tv/tag/threaded-callback-interrupts-on-raspberry-pi>>>

## Write a function that converts a 10-bit ADC reading from the potentiometer to a 3V3 limited voltage output.

## Write a function that converts a 10-bit ADC reading from the temperature sensor to a reading in degree Celsius (Have a look at the datasheet).

## Write a function that converts a 10-bit ADC reading from the LDR to a percentage representing the amount of light received by the LDR. (2) *The flashlight from a smartphone could be used as the maximum amount of light received by the LDR.*

## Draw a flowchart of the system.