

# **COMP140 Individual Creative Computing Project**

-  
w\

COMP140: Individual Creative Computing Project

# **1: Hardware Interfaces**

# Register Attendance



Figure 1: Attendance monitoring is in place. It is your responsibility to ensure that you have signed yourself in.

# Learning Outcomes

After this session you will be able to:

- ▶ **Select** the appropriate method for communication
- ▶ **Outline** the difference between I2C and SPI
- ▶ **Implement** a hardware interface between the Arduino and a sensor

# What is Serial Communication?

- ▶ Arduino to Multimedia Computer (retro term alert)
- ▶ Arduino to Arduino
- ▶ Arduino to sensors

Some kind of picture?

# Types of Serial Communications

- ▶ **I2C**
- ▶ RS232
- ▶ **USB**
- ▶ R2422
- ▶ CAN
- ▶ Microwire
- ▶ RS485
- ▶ **SPI**
- ▶ **TTL**

# Protocols

A protocol is the language that governs communication between systems or devices.

- ▶ Midi
- ▶ DMX-512
- ▶ X10
- ▶ USB
- ▶ RS485
- ▶ UDP/IP
- ▶ TCP/IP

# Physical Agreement



# Timing Agreement

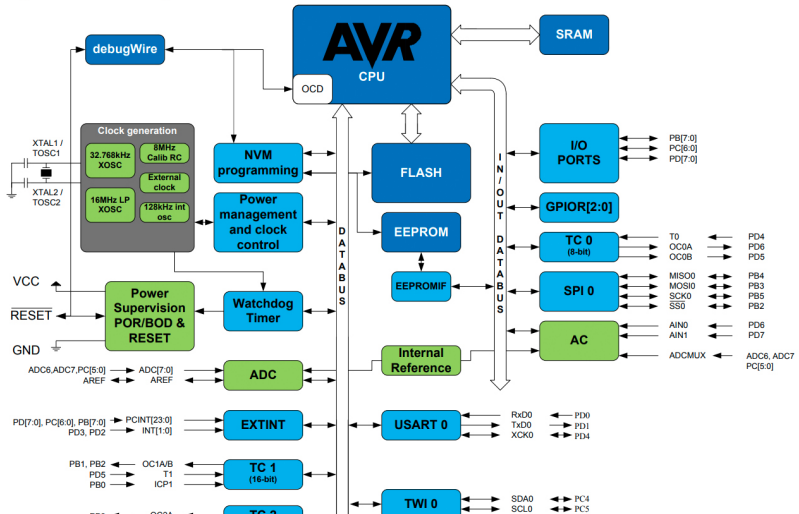
Interfaces require

# Electrical Agreement

# Package Size

# Hexadecimal

# Serial vs. Parallel



# Serial/UART

- ▶ This is how an Arduino talks to a computer
- ▶ Arduino has an onboard UART to Seral Converter
- ▶ Requires an agreed baud rate: `Serial.begin(9600)`
- ▶ Two communication lines - RX to receive and TX to transmit
- ▶ TX connects RX, RX to TX
- ▶ Used by some peripherals such as Bluetooth modules

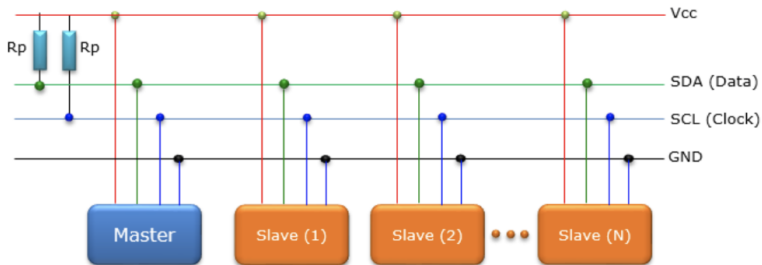




# Inter-Integrated Circuit (I<sup>2</sup>C)

- ▶ Pronounced I Square C
- ▶ Uses a clock Signal
- ▶ Two communication lines - SDA (data) and SCL (clock)
- ▶ Multiple I<sup>2</sup>C devices can communication on the same data lines (bussed)
- ▶ Slave devices have an address

# I<sup>2</sup>C Bus Example

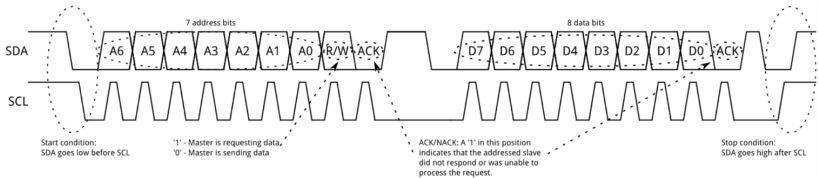


Examples

# Wire Library

- ▶ I<sup>2</sup>C Library for Arduino
- ▶ Arduino can be either host or slave
- ▶ Particularly useful for wired Arduino-to-Arduino communication

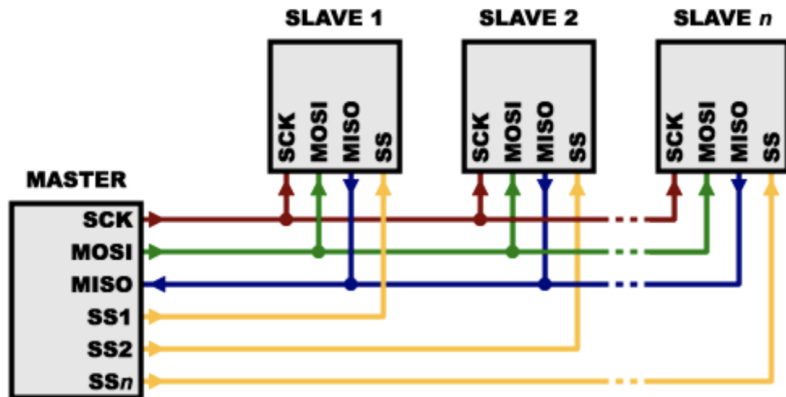
# I<sup>2</sup>C Communication Example



# Serial Peripheral Interface (SPI)

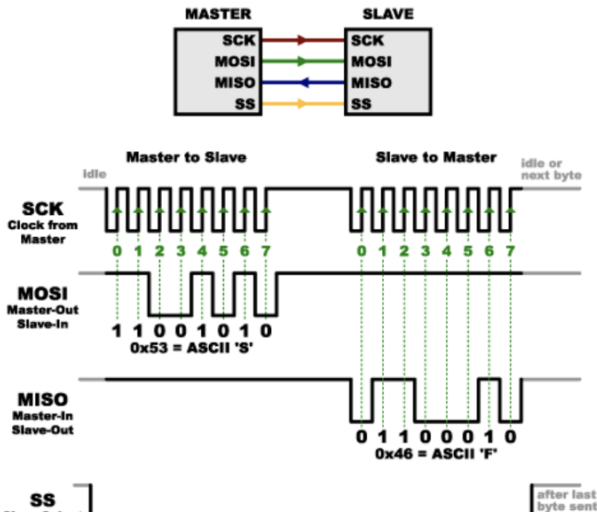
- ▶ Uses a clock signal
- ▶ Requires 4 lines
- ▶ MISO - master in slave out
- ▶ MOSI - master out slave in
- ▶ SCK - serial clock
- ▶ CS - chip select
- ▶ Not addressed - each slave device requires a discrete chip select line.

# SPI Bus Example



# SPI Communication Example

## SPI COMMUNICATION EXAMPLE

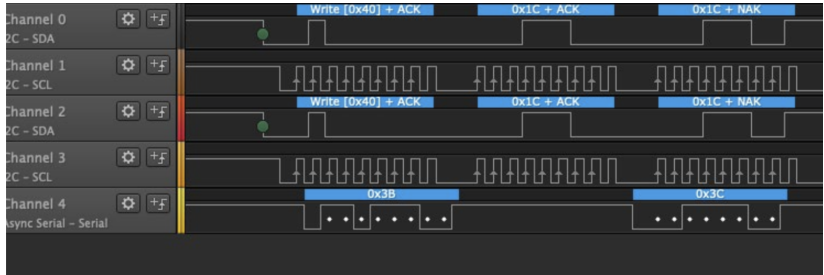


# Logic Analysers

- ▶ Not doing what you expect? Spy on them!
- ▶ Logic analysers are the digital equivalent of an oscilloscope
- ▶ Can have up to 8 signal capture lines
- ▶ Capture digital logic (HIGH or LOW)
- ▶ Easy to connect to existing circuits
- ▶ USB connection for data capture
- ▶ We have a couple in the Games Academy



# Logic Analyser Software



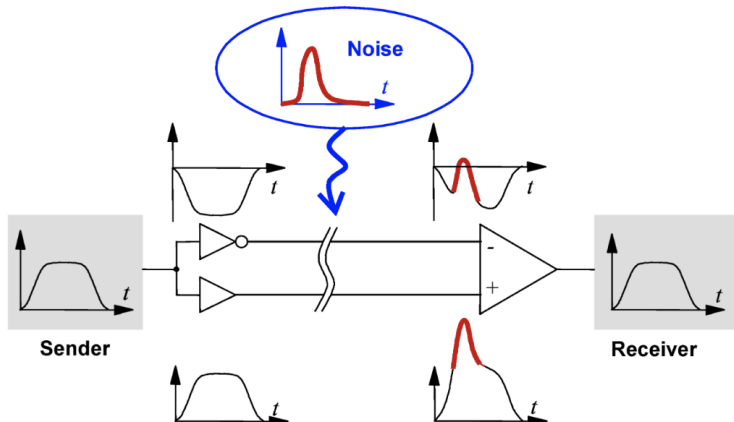
# Differential Signal

- ▶ Low voltage signals are susceptible to noise over long distance
- ▶ Noise can turn a digital 1 into a 0
- ▶ Differential Signal is one solution
- ▶ Used by USB and in professional audio
- ▶ The hardwork is done for you: PCA9615 chip converts I<sup>2</sup>C to diff



# How does Differential Signal Work?

- ▶ Two lines carry the same signal, but one is inverted
- ▶ The two lines cancel each other out
- ▶ All that is left is the noise
- ▶ The noise is then cancelled out



# Activity - MPU-6050

