

PC/CP 364

Serial Communication-SPI

Terry Sturtevant

Wilfrid Laurier University

January 18, 2017

Serial Communication -SPI

Serial Communication -SPI

- Serial Peripheral Interface

Serial Communication -SPI

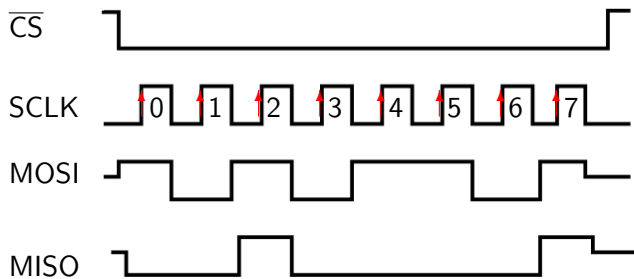
- Serial Peripheral Interface
- Master/slave communication

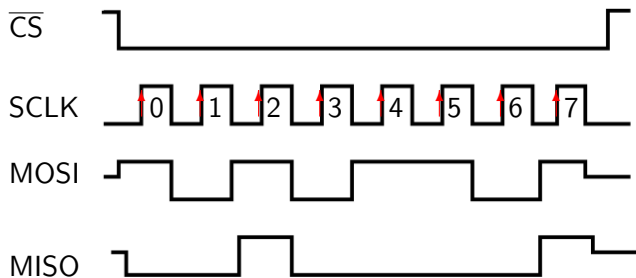
Serial Communication -SPI

- Serial Peripheral Interface
- Master/slave communication
- Uses 3 signals (and Ground),
MISO, MOSI, SCLK
and chip selects for each slave device

Serial Communication -SPI

- Serial Peripheral Interface
- Master/slave communication
- Uses 3 signals (and Ground),
MISO, MOSI, SCLK
and chip selects for each slave device
- Synchronous, so master controls clock rate





SPI transfers can happen in both directions simultaneously.

Arduino SPI

Arduino SPI

- **SPI.begin()**
initiallize SPI bus

Arduino SPI

- **SPI.begin()**
initialize SPI bus
- **response = SPI.transfer(val)**
transfer byte

Arduino SPI

- **SPI.begin()**
initialize SPI bus
- **response = SPI.transfer(val)**
transfer byte
- **response16 = SPI.transfer16(val16)**
transfer two byte value

Arduino SPI

- **SPI.begin()**
initialize SPI bus
- **response = SPI.transfer(val)**
transfer byte
- **response16 = SPI.transfer16(val16)**
transfer two byte value

Any pin can be slave select and is manipulated manually.

SPI sample code

SPI sample code

```
#include <SPI.h>
const int slaveSelectPin = 10;
void setup() {
    pinMode(slaveSelectPin , OUTPUT);
    SPI.begin();
}

void loop() {
    for (int level = 0; level < 255; level++) {
        digitalPotWrite(channel , level);
        delay(10);
    }
}
```

SPI sample code (continued)

SPI sample code (continued)

```
void digitalPotWrite(int address, int value) {  
    digitalWrite(slaveSelectPin, LOW);  
    SPI.transfer(address);  
    SPI.transfer(value);  
    digitalWrite(slaveSelectPin, HIGH);  
}
```