

Electronics Serial Communication-I2C

Terry Sturtevant

Wilfrid Laurier University

June 28, 2017

Serial Communication -I²C

Serial Communication -I²C

- Inter-Integrated Circuit Interface

Serial Communication -I²C

- Inter-Integrated Circuit Interface
- Master/slave communication

Serial Communication -I²C

- Inter-Integrated Circuit Interface
- Master/slave communication
- Uses 2 signals (and Ground),
SDA and SCL

Serial Communication -I²C

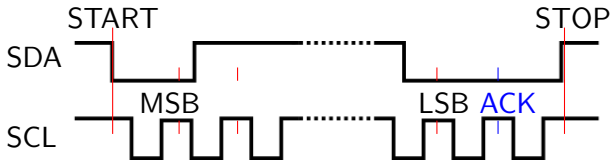
- Inter-Integrated Circuit Interface
- Master/slave communication
- Uses 2 signals (and Ground),
SDA and SCL
- Many slaves can be on the same bus since each has an address

Serial Communication -I²C

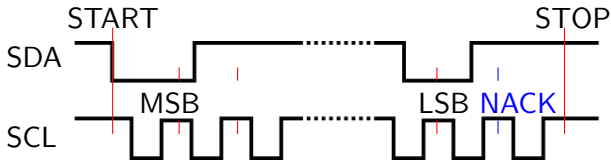
- Inter-Integrated Circuit Interface
- Master/slave communication
- Uses 2 signals (and Ground),
SDA and SCL
- Many slaves can be on the same bus since each has an address
Device addresses are pre-programmed, but can usually be changed

Serial Communication -I²C

- Inter-Integrated Circuit Interface
- Master/slave communication
- Uses 2 signals (and Ground),
SDA and SCL
- Many slaves can be on the same bus since each has an address
Device addresses are pre-programmed, but can usually be changed
- Synchronous, so master controls clock rate



- I²C ; bits are read when SCL is HIGH
- ACK is sent by receiver if OK
sender must release SDA after LSB



- I²C ; bits are read when SCL is HIGH
- NACK is sent by master-receiver if OK
sender must release SDA after LSB



- I²C write to slave register



- I²C write to slave register



- I²C write to slave register



- I²C write to slave register



- I²C write to slave register



- I²C write to slave register



- I²C write to slave register



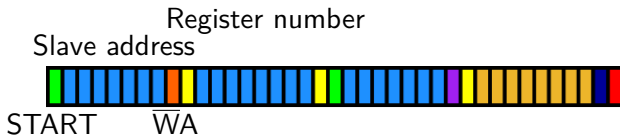
- I²C read from slave register



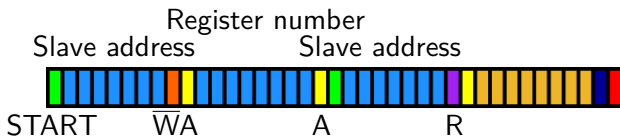
- I²C read from slave register



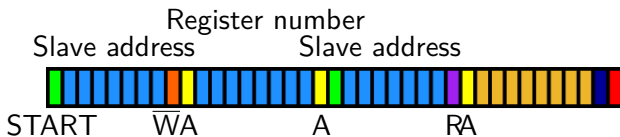
- I²C read from slave register



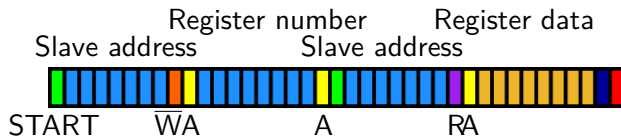
- I²C read from slave register



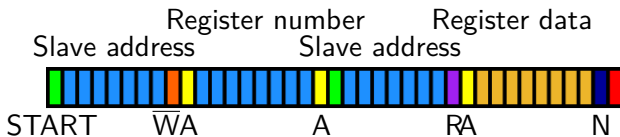
- I²C read from slave register



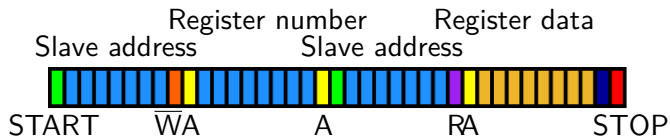
- I²C read from slave register



- I²C read from slave register



- I²C read from slave register



- I²C read from slave register

Arduino Wire

Arduino Wire

- **Wire.begin()**
initiallize I²C bus

Arduino Wire

- **Wire.begin()**
initiallize I²C bus
- **Wire.beginTransmission(44)**
transmit to address d'44

Arduino Wire

- **Wire.begin()**
initiallize I²C bus
- **Wire.beginTransmission(44)**
transmit to address d'44
- **Wire.write(byte(0x00))**
transmit byte

Arduino Wire

- **Wire.begin()**
initiallize I²C bus
- **Wire.beginTransmission(44)**
transmit to address d'44
- **Wire.write(byte(0x00))**
transmit byte
- **Wire.write(val)**
transmit value

Arduino Wire

- **Wire.begin()**
initiallize I²C bus
- **Wire.beginTransmission(44)**
transmit to address d'44
- **Wire.write(byte(0x00))**
transmit byte
- **Wire.write(val)**
transmit value
- **Wire.endTransmission()**
end tranmission

Arduino Wire (write example)

Arduino Wire (write example)

```
#include <Wire.h>
void setup() {
  Wire.begin(); // join i2c bus
}
byte val = 0;
void loop() {
  Wire.beginTransmission(44);
  Wire.write(byte(0x00));
  Wire.write(val);
  Wire.endTransmission();
}
delay(500);
}
```

Arduino Wire (read example)

Arduino Wire (read example)

- **Wire.requestFrom(44,2,stop)**
request 2 bytes from address d'44
stop is a boolean; true to stop

Arduino Wire (read example)

- **Wire.requestFrom(44,2,stop)**
request 2 bytes from address d'44
stop is a boolean; true to stop
- **Wire.read()**
read bytes

Arduino Wire (read example)

Arduino Wire (read example)

```
#include <Wire.h>
void setup()
{
  Wire.begin();           // join i2c bus
}
void loop()
{
  Wire.requestFrom(2, 6);
  while(Wire.available())
  {
    char c = Wire.read(); // receive a byte
  }
  delay(500);
}
```