DIGITAL COMMUNICATION

SIMON HOGGAN CHRISTENSEN
LABORATORIEKOORDINATOR

OVERVIEW

Wired Communication

- SPI, I^2C and U(S)ART revisited

Wireless Communication

- RF
- WiFi
  - Zigbee protocol
- Bluetooth (and BLE)
- RFID/NFC
- Component examples

DEPARTMENT OF COMPUTER SCIENCE

WIRED COMMUNICATION PROTOCOLS

DEPARTMENT OF COMPUTER SCIENCE

...MASTER/SLAVE?

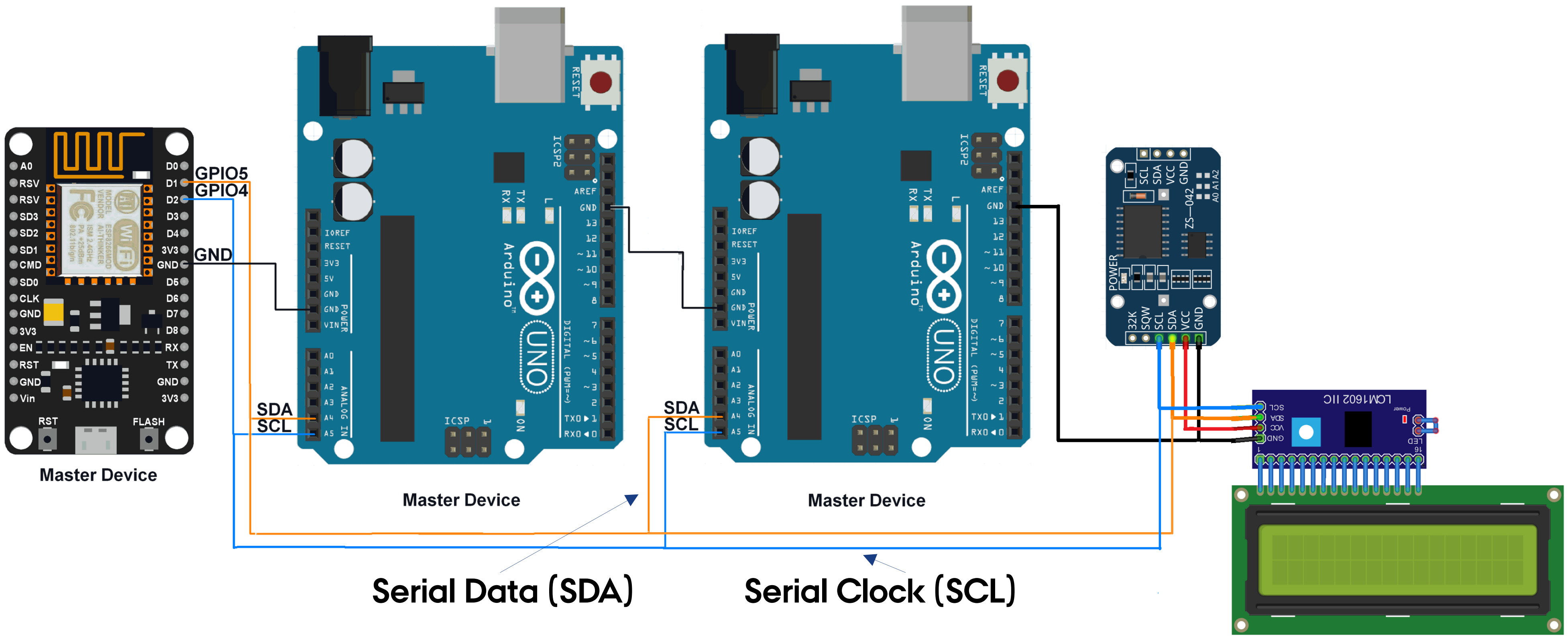
Horrible naming for a brilliant concept

Other completely acceptable naming schemes:

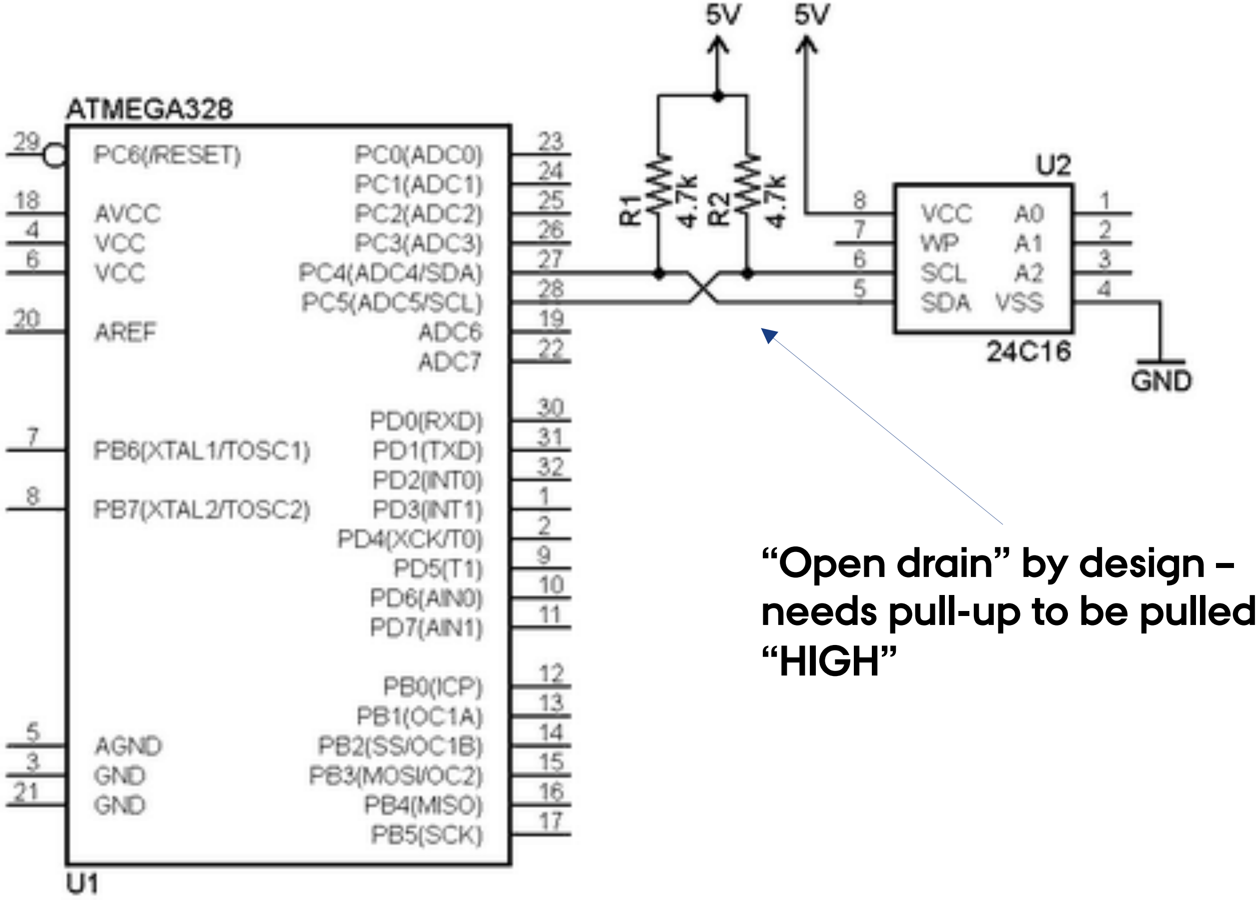
- Controller/responder
- Initiator/target
- Principal/agent

DEPARTMENT OF COMPUTER SCIENCE

I²C (INTER-INTEGRATED CIRCUIT)

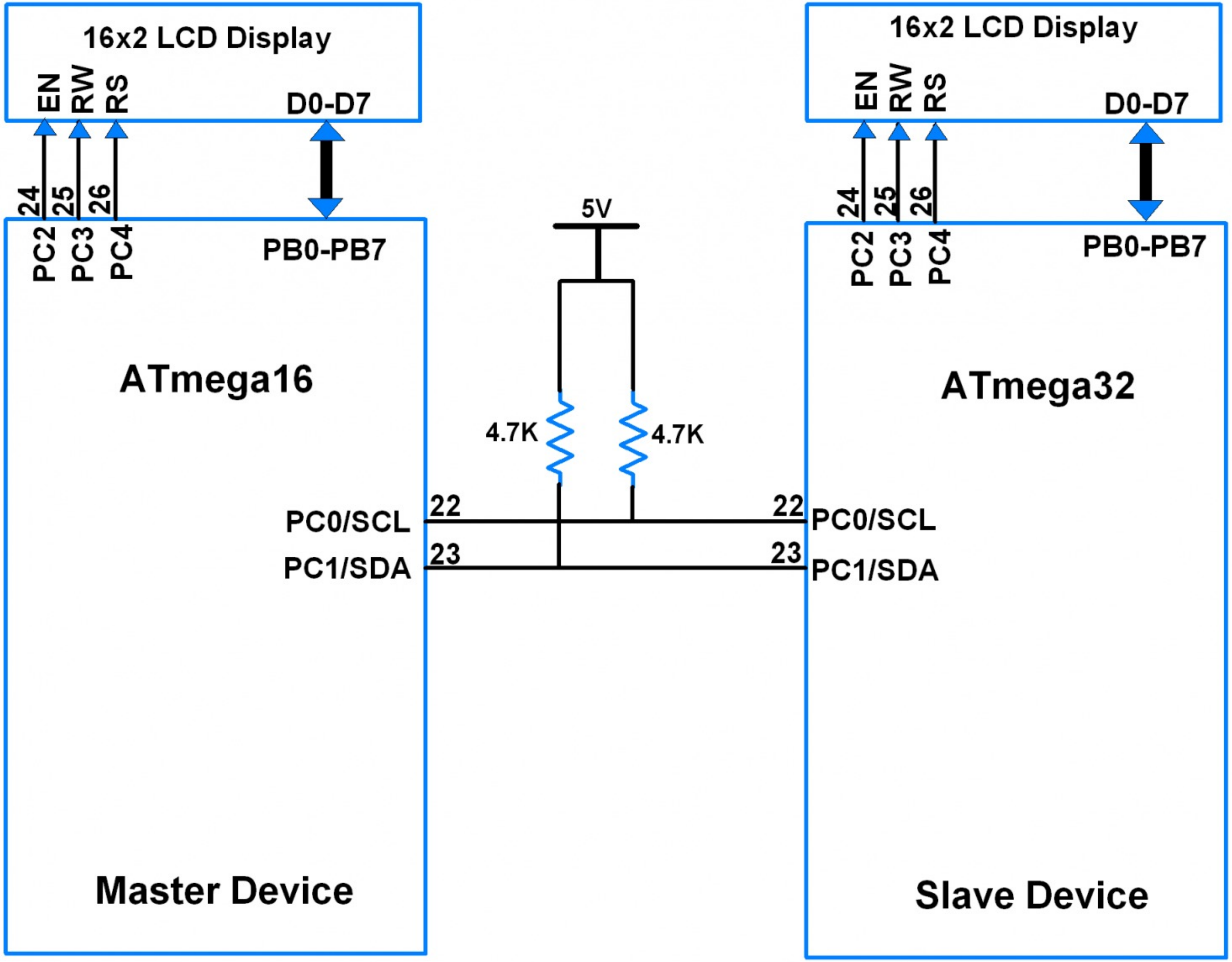


I²C (INTER-INTEGRATED CIRCUIT)

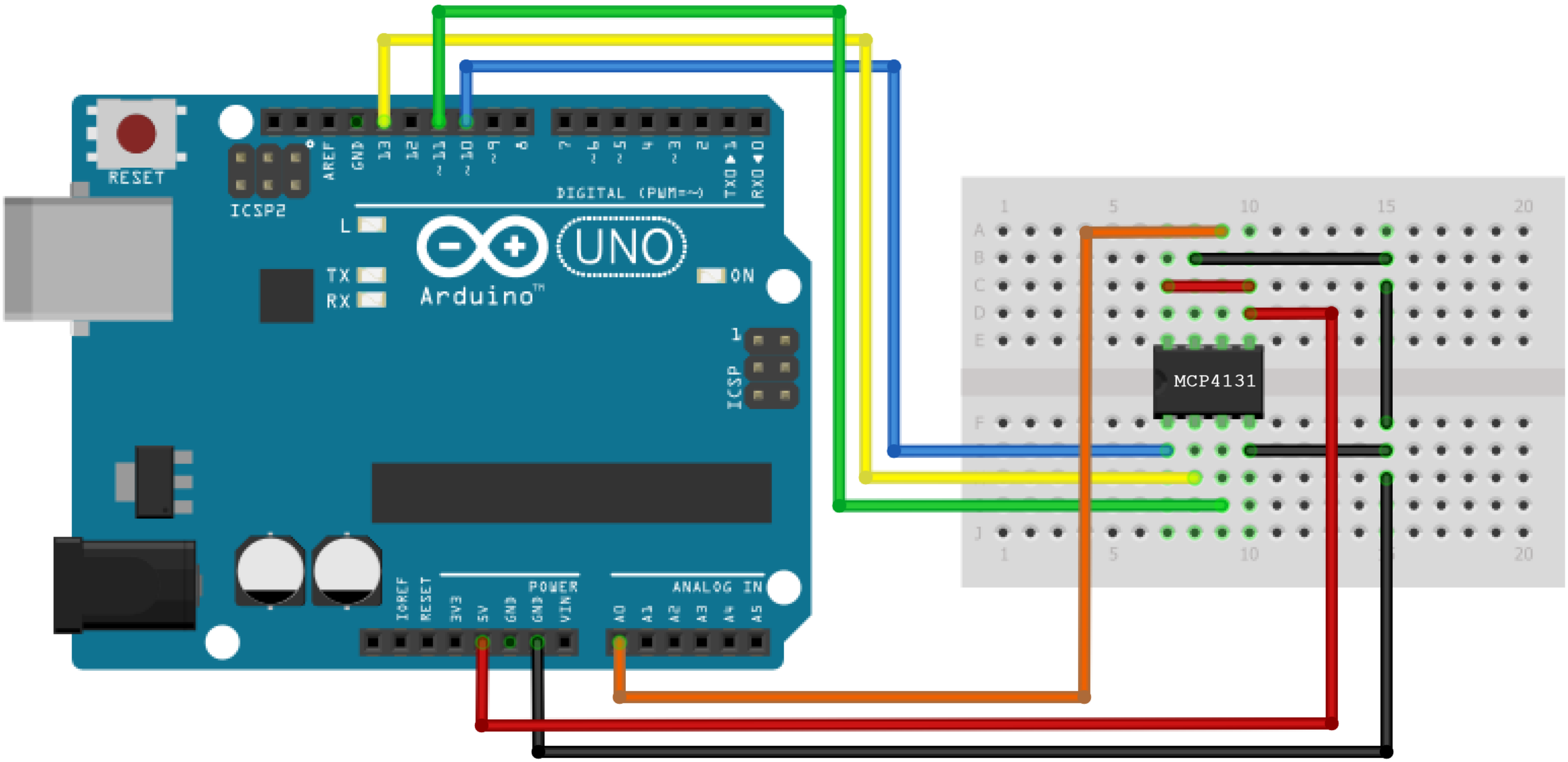


"Open drain" by design – needs pull-up to be pulled "HIGH"

DEPARTMENT OF COMPUTER SCIENCE


16x2 LCD Display
EN
RW
RS
D0-D7
24
25
26
PC2
PC3
PC4
PB0-PB7
ATmega16
PC0/SCL
PC1/SDA
22
23
Master Device
5V
4.7K
4.7K
16x2 LCD Display
EN
RW
RS
D0-D7
24
25
26
PC2
PC3
PC4
PB0-PB7
ATmega32
22
23
PC0/SCL
PC1/SDA
Slave Device

SPI (SERIAL PERIPHERAL INTERFACE)



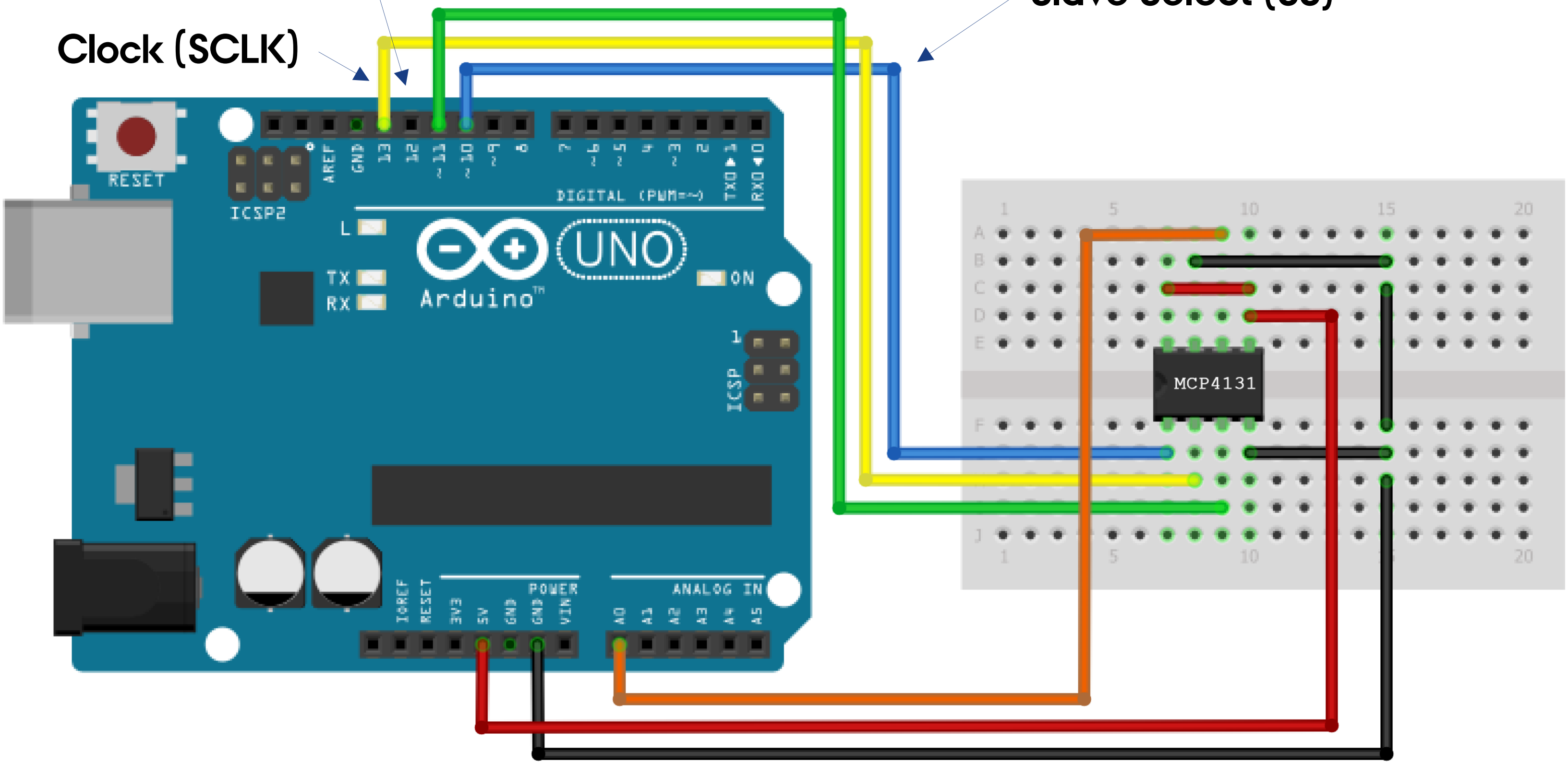
SPI (SERIAL PERIPHERAL INTERFACE)

Master Input, Slave Output (MISO)
- not connected

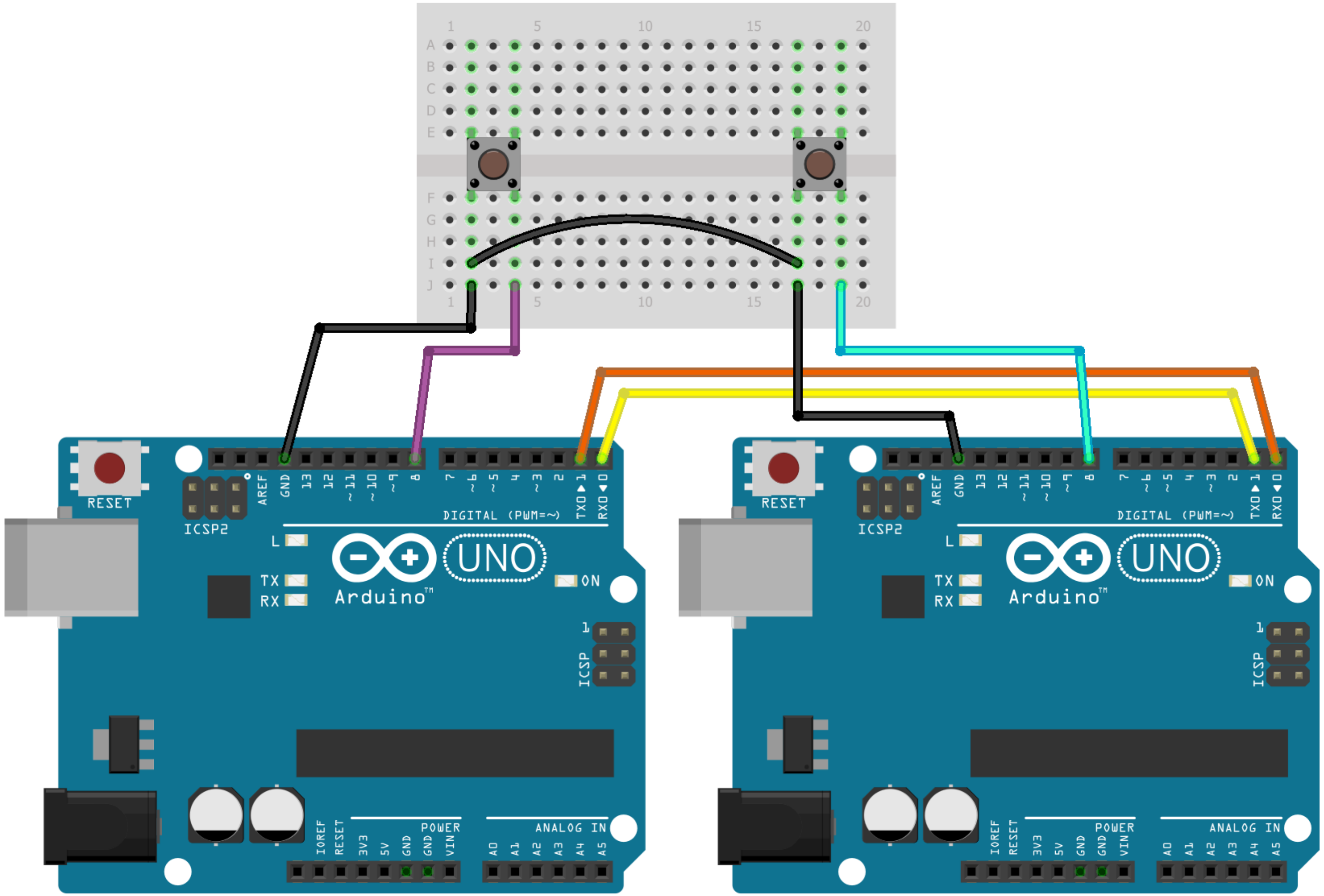
Master Output, Slave Input (MOSI)

Slave Select (SS)

Clock (SCLK)



UART/USART



DEPARTMENT OF COMPUTER SCIENCE

COMMUNICATION PROTOCOLS

I2C (Inter-integrated Circuit) communication

- Single bus with address specific calls using only two pins (SDA/SCL).
- Half-duplex (no concurrent reading and writing communication)
- Best for: Controlling lots of I2C components and/or simplifying your circuit communication + scalability

SPI (Serial Peripheral Interface)

- MISO/MOSI/SCK/SS pins often used for fastest slave/master communication – used to bootload your chips.
- Needs allocated SS-pin for each slave
- Full-duplex
- Best for: Fast/complex master/slave architecture, or connecting flash/eeprom

UART/USART (Universal Asynchronous Receiver/Transmitter) communication

- R(ead)X/T(ransmit)X pins used for serial communication – used when you program the chip and when using Serial Monitor/Plotter. USART uses a separate clock pin.
- Full-duplex
- "Open broadcast" – multiple devices attached to TX, will receive same messages
- Best for: Long distance point-to-point communication

WIRELESS COMMUNICATION

DEPARTMENT OF COMPUTER SCIENCE

OPTIONS

Wired communication

- I²C (Inter-Integrated Circuit)
- SPI (Serial Peripheral Interface)
- UART (Universal Asynchronous Receiver-Transmitter)

Wireless communication

- RF
- Wi-Fi
- Zigbee
- Bluetooth + BLE (Bluetooth Low Energy)
- RFID / NFC
- (Cellular, such as GSM)

DEPARTMENT OF COMPUTER SCIENCE

RADIO FREQUENCY COMMUNICATION

A **radio frequency** (RF) signal is a wireless electromagnetic signal used as a form of communication

Often categorised by **frequency band** or protocol

<1GHz - the 900 MHz band is used for walkie-talkies, amateur radio

- RFID active ≈ 400-900 MHz, passive ≈ 13.56-134 MHz
    - NFC operates at 13.56 MHz.

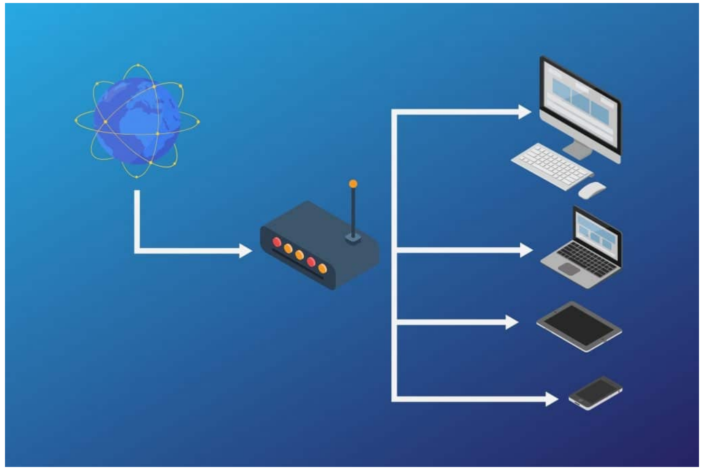
1 to 5GHz – the 2.4 GHz ISM band is very common.

- ZigBee
- Bluetooth
- Wi-Fi

>5GHz - extremely high frequency comms are either severely limited in range or require line of sight.

- Radar (F.x Air surveillance K-band = 18-27 GHz)

WIFI



WiFi ≠ Internet

WiFi is trademarked, and does not stand for Wireless Fidelity.

Just playing on the word Hi-Fi.

Computer –> WiFi -> Router (local routing) –> Modem -> ISP-> Lots of cables, servers, redirecting, addressing, and interconnectivity.

WiFi = Local radio frequency communication (2.4GHz or 5GHz)

DEPARTMENT OF COMPUTER SCIENCE

WIFI

Advantages of WiFi:

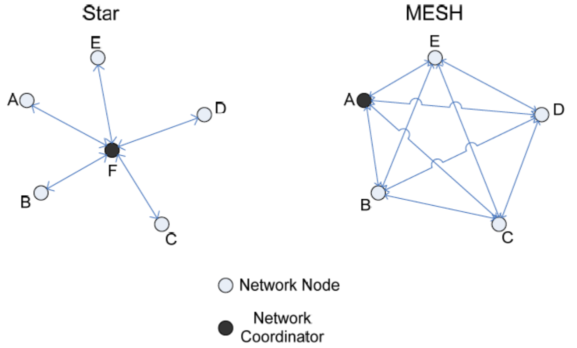
- WiFi has good range coverage
- Can (to a certain degree) penetrate walls and other obstacles
- Connection is simple and the protocol is widely used

Disadvantages of WiFi:

- (Sometimes) Security
- Connection delays
- Coverage on higher frequency band
- Speed on lower frequency band

DEPARTMENT OF COMPUTER SCIENCE

THE ZIGBEE PROTOCOL



Zigbee Coordinator (the brain) - commissions devices to the network, stores the security keys and bridges to other networks. There can be only one.

Zigbee Router - Zigbee networks may have several routers to serve as intermediate routers or to transmit data within the network.

Zigbee End Device - can only talk to the parent node (router or coordinator).

These devices spend most of their time in sleep mode and only wake up to transmit data to the parent.

Zigbee uses a mesh network layout but also supports other topologies like Star.

ZIGBEE

Advantages:

- Mesh-based network topology.
- Low-power consumption
- Secure (ish?)

Disadvantages:

- Short range (10 to 100m)
- Complicated to set up
- Single point of failure

DEPARTMENT OF COMPUTER SCIENCE

BLUETOOTH

Bluetooth uses the 2.4GHz spectrum in the ISM band.

It has a range from 10m up to 100m (longer distance = higher power consumption).

Bluetooth is an ad-hoc type of network and provides point to point (P2P) connections.

Bluetooth pairing required for exchange of data.

DEPARTMENT OF COMPUTER SCIENCE

CLASSIC VERSUS LE

Bluetooth 5 Classic – The traditional Bluetooth has a high throughput, mostly used for wireless audio and file transmission.

- Up to 7 devices can be connected to a single master
- Range reliably up to 100m (if designed correctly) – max 240m
- Mbit/s = 50
- Battery life - days

Bluetooth Smart – Bluetooth Low Energy (LE) (sometimes branded as Bluetooth Smart) transmits just state information.

- Up to 128 devices can be connected to a single master
- Range reliably up to 30m (if designed correctly) – can be extended a ton
- Mbit/s = 2 (burst)
- Battery life – months (but probably not with over-the-counter SoCs)

DEPARTMENT OF COMPUTER SCIENCE

RADIO-FREQUENCY IDENTIFICATION (RFID)

One of the simplest communication methods.

RFID technology includes tags and readers.

Passive or active – range from a few centimeters to many meters.

RFID Readers have 3 main components

- Antenna - to receive and transmit signals to/from the tags.
- Power source – to ”wake up” passive tags
- Reader controller - manages the information read by the reader.

DEPARTMENT OF COMPUTER SCIENCE

RFID

Tags - the end points in an RFID system. They store identity information along with other information as required

- Active Tags: have a power source, so they have more range.
- Passive Tags: no power source, only get activated in the vicinity of a reader (they use the radio energy transmitted by the reader).

RFID tags can have 3 main components:

- IC - for storing identity information, processing it and modulating/demodulating the RF signals
- Antenna - to receive and send the radio signals
- Power source- if an active tag

DEPARTMENT OF COMPUTER SCIENCE

NEAR-FIELD COMMUNICATION (NFC)

A subset of the RFID protocol

Specifically designed for very short-range communication (less than a few centimeters)

Very power-efficient communication protocols.

NFC has 2 types of devices (device can act as a tag or a reader):

- Initiator: initiates the communication. It actively generates an RF field, that can also power the passive target.
- Target: receives the information from the initiator. The target can either be passive (e.g. NFC tags) or active for peer to peer communication, such as in phones

DEPARTMENT OF COMPUTER SCIENCE

COMPONENT EXAMPLES

Many, many options – this is a few examples, and a non-exhaustive list

DEPARTMENT OF COMPUTER SCIENCE

XBEE PRO

Zigbee protocol component (can be used for fx. Philips Hue)

SPI connection to SoC

Can be bought with onboard microcontroller

DEPARTMENT OF COMPUTER SCIENCE

RC522 RFID READER/WRITER

Inexpensive SPI RFID reader/writer

I2C alternative = PN532

Mainly suitable for NFC – result will vary based on tag

In most Arduino kits

DEPARTMENT OF COMPUTER SCIENCE

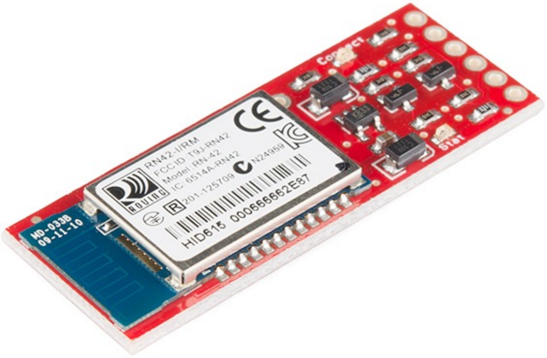
BLUETOOTH MATE

Bluetooth UART module

Retired – and varying documentation

Alternatives:

- HC-06 (UART, Bluetooth 2.0)
- HC-10 (UART, Bluetooth 4.0)
- JDY-33 (UART, Bluetooth 4.2)



DEPARTMENT OF COMPUTER SCIENCE

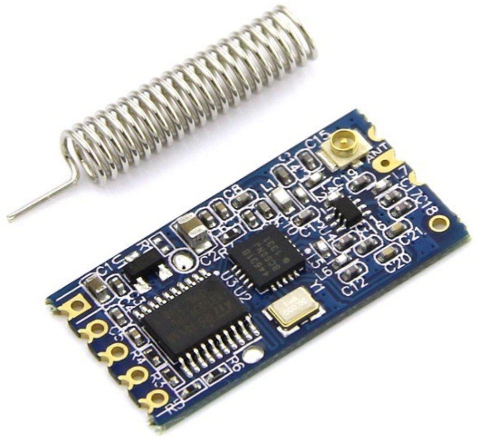
HC-12 RADIO MODULES

Long range wireless communication between multiple Atmegas

- with distances up to 1.8km!

Uses AT commands, which can be sent from an Arduino, a PC, or any other microcontroller using the serial port (UART)

Can support: one to one, one to many, or many to many.



DEPARTMENT OF COMPUTER SCIENCE

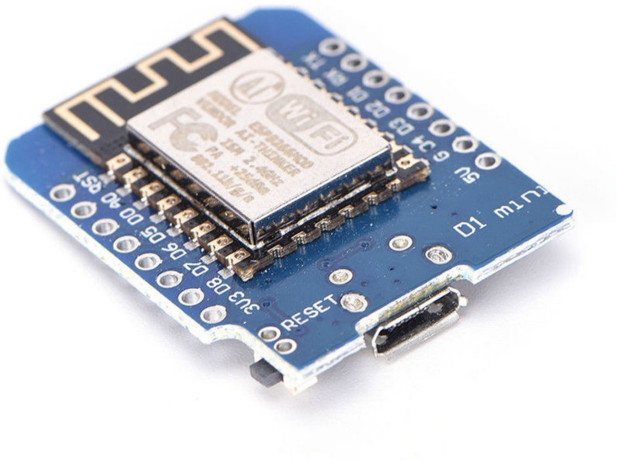
ESP32/ESP8266

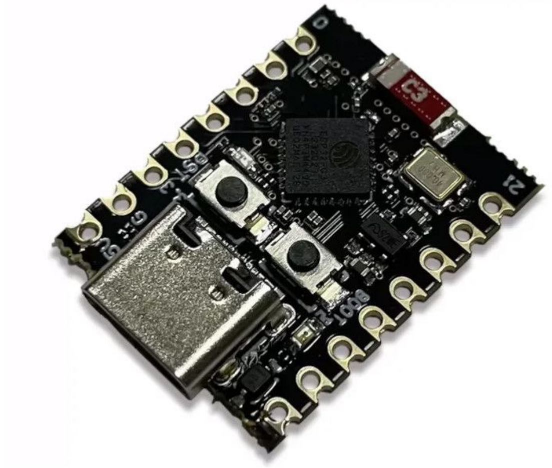
Many protocols on board

SoC + communication

D1 Mini = ESP8266

ESP32c3 Super Mini





DEPARTMENT OF COMPUTER SCIENCE

PRACTICAL EXERCISE 5

I^{2}C MASTER AND SLAVE

SIMON HOGGAN CHRISTENSEN
LABORATORIEKOORDINATOR

CHALLENGE

Set up a master with two slaves via I^2C

You will now use the I2C protocol to allow a master-ATmega328p to send information to both a slave-ATmega328 and a slave-ESP32

This exercise and getting the communication flow from master to slave open gives new possibilities:
- Actual concurrency
- More GPIO via two (or more) Atmegas
- Wireless communication via ESP32
- Allocating a microcontroller to noisy circuit elements (motors etc.)
- Understanding I^2C addressing for future I^2C components

DEPARTMENT OF COMPUTER SCIENCE

COMPONENTS

Your prototype must use the following components:

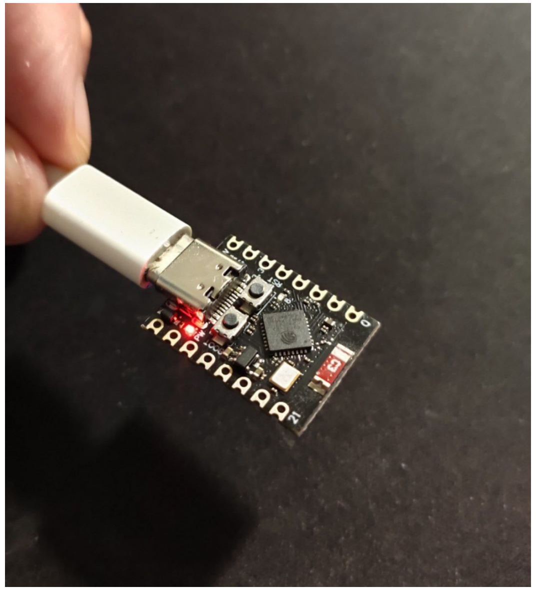
- 2 x ATmega328p with 16MHz external crystal (logic)

- 1 x ESP32c3 Super Mini (at component table, needs soldered)

- 2 x Light Emitting Diode (LED) with appropriate protection

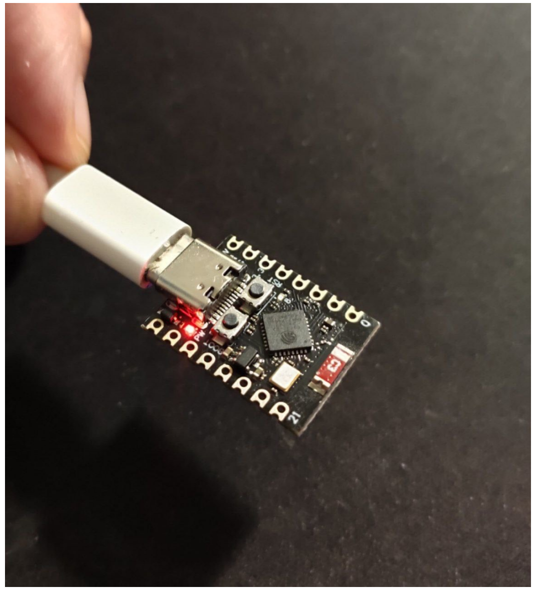
- Power from battery (breadboard PSU acceptable)

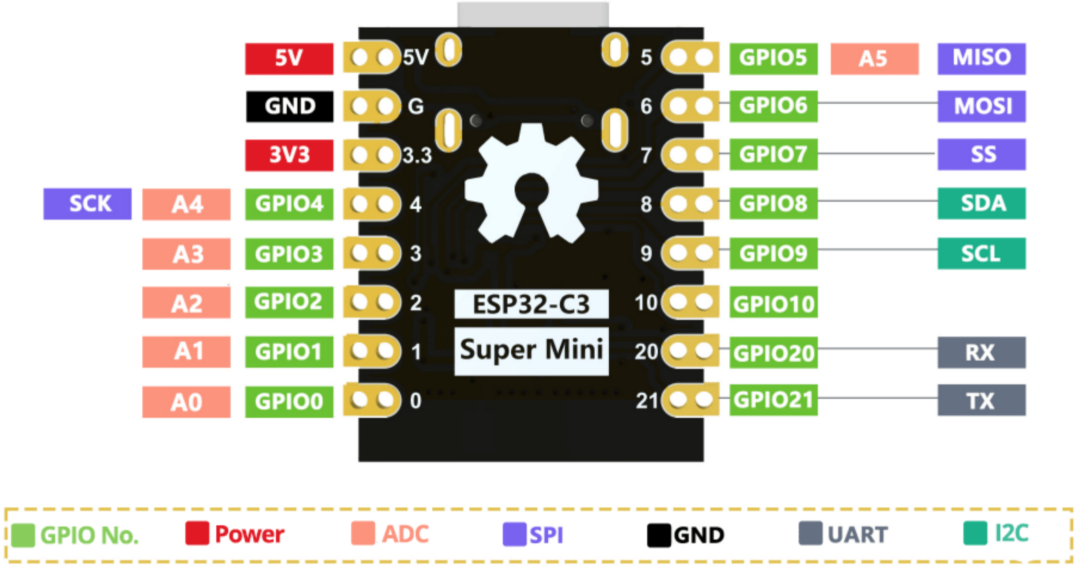
- Lots of jumper wires ☺



ESP32C3 SUPERMINI

- WiFi (802.11b/g/n protocol, 2.4GhHz)
- BLE 5.0
- Deep sleep mode (consumption of about 43uA)
- Can be wired to work directly with 3.7 LiPo.
- ADC (A0-A4)
- PWM (all GPIO)
- UART, SPI, I^{2}C
- Costs 2-3 times as much as ATmega328p (<u>used to be 10-12 times as much</u>)





GOAL

ATMega328 Master handles counting.

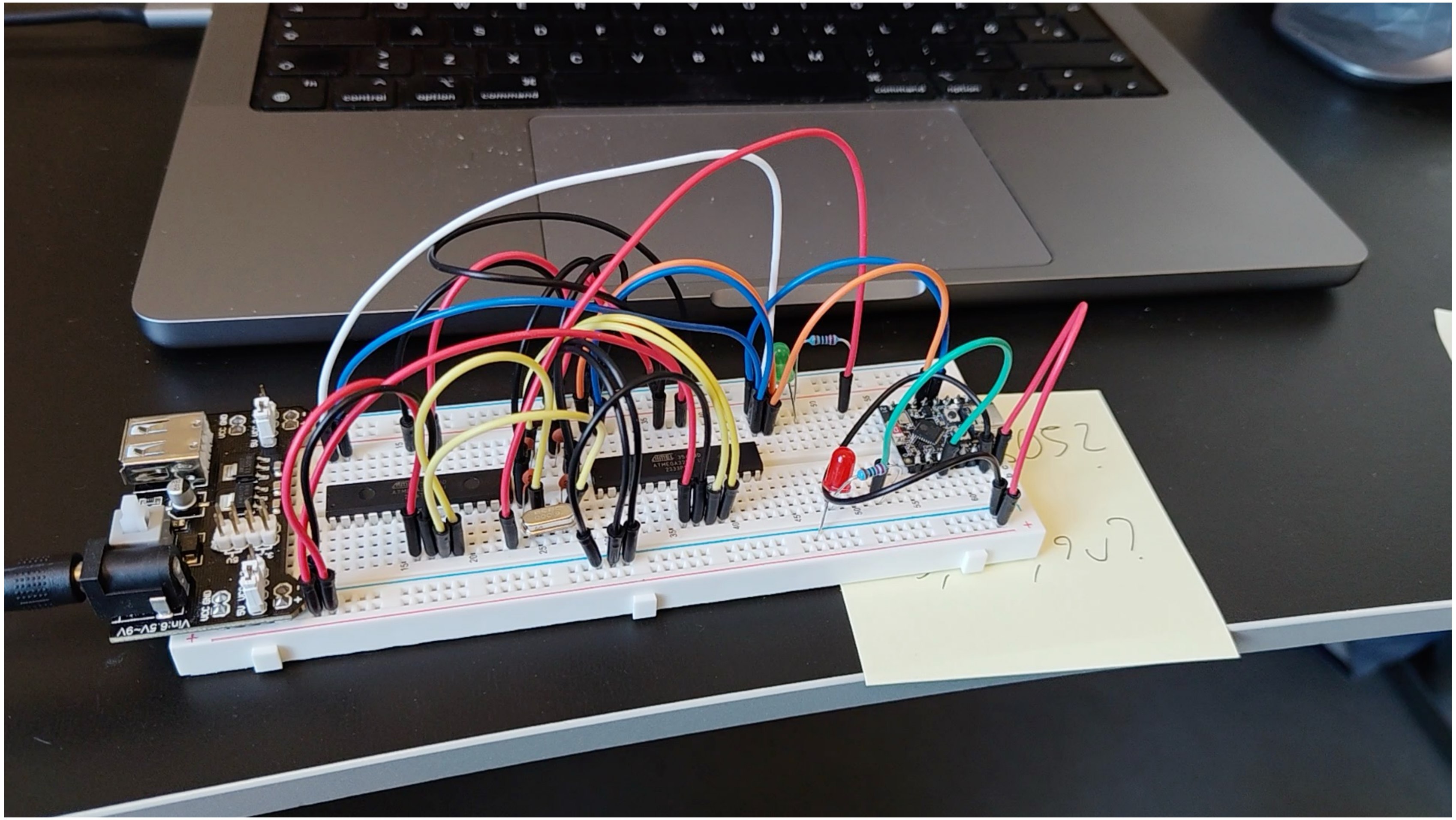
ATMega328 Slave blinks an LED when x=1&3

ESP32c3 Slave won't start loop until connected to phone hotspot
ESP32c3 Slave blanks LED when x=2&5, once connected to WiFi

No Arduinos anywhere near final version ☺

Pass = concurrently running, asynchronous LEDs + explanation of your code

DEMO



THE MASTER

The ATmega328p master

Initialized as master (no address in paratheses)

Wire.h library for I^2C-communication

Transmitting commands to specific adresses

Simple counter

Serial.println not necessary – just debugging

I2CMaster.ino

```
#include<Wire.h>
int  x = 0;
void setup() {
  // Start the I2C Bus as master (no adress-number)
  Wire.begin();
  Serial.begin(9600);
}
void loop() {
  Wire.beginTransmission(9);
  Wire.write(x);
  Wire.endTransmission();
  Wire.beginTransmission(10);
  Wire.write(x);
  Wire.endTransmission();
  x++;
  if (x > 5) x = 0;
  delay(500);
  Serial.println(x);
}

```

THE ESP32 SLAVE

The ESP32 Slave

Initialized as slave (address in Wire.begin())

Connecting to WiFi in setup, pausing loop

onReceive() handling reading from I^2C bus

Blinks LED when x = 2 and 5

I2CSlaveESP32.ino

```
#include <Wire.h>
#include <WiFi.h>

const char* ssid = "ZenfoneSimon";
const char* password =  "test1234";
int x = 0;
int   LED = 2;
void setup() {
  Serial.begin(115200);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.println("Connecting to WiFi...");
   pinMode (LED, OUTPUT);
  // Join the I2C Bus as Slave on address 10
  Wire.begin(10);
  // Attach a function to trigger when something is received.
  Wire.onReceive(receiveEvent);
  }
}
void   receiveEvent(int bytes) {
  x = Wire.read();    // read ONE character from the I2C
}
void loop() {
  Serial.println(x);
   if (x == 2) {
    digitalWrite(LED, HIGH);
    delay(200);
    digitalWrite(LED,   LOW);
    delay(200);
  }
  //If value received is 3 blink LED for 400   ms
  if (x == 5) {
    digitalWrite(LED, HIGH);
    delay(400);
    digitalWrite(LED,   LOW);
    delay(400);
  }
}

```

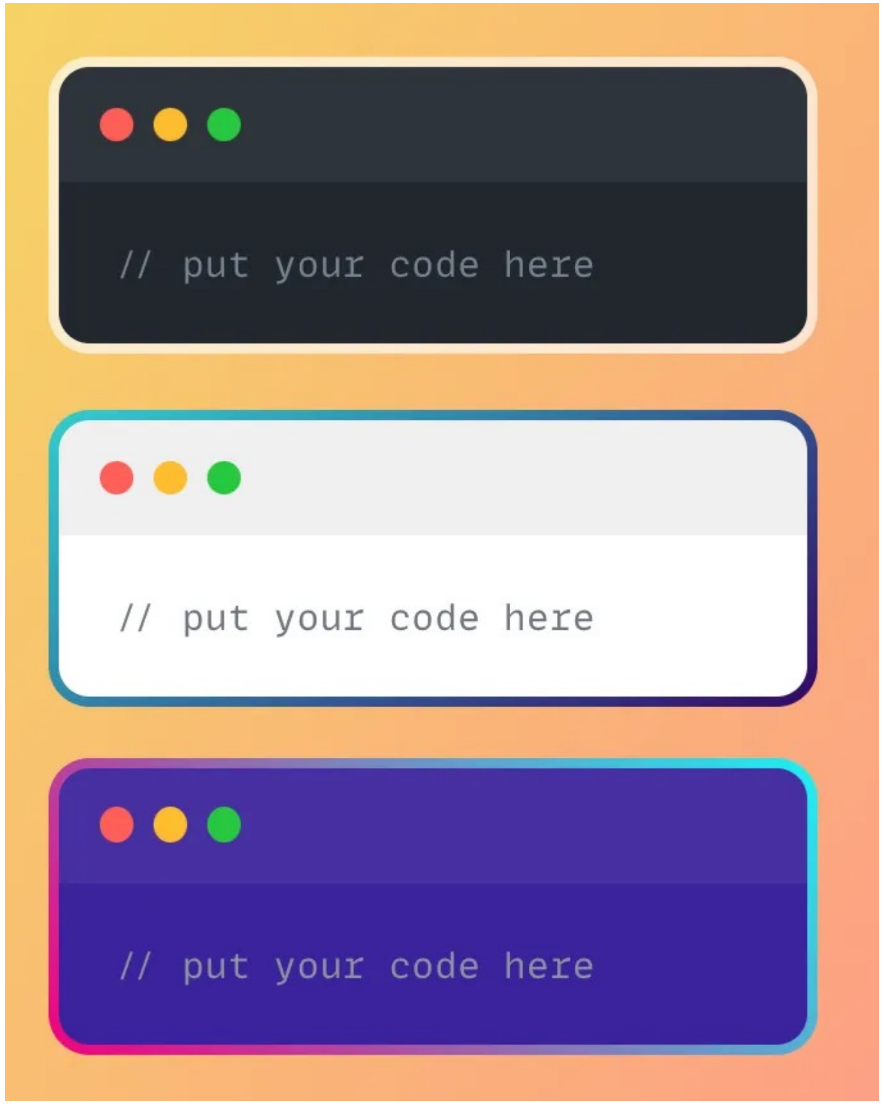
THE ATMEGA SLAVE

The ATmega328p Slave

Initialized as slave (address in Wire.begin())

onReceive() handling reading from I^2C bus

Blinks LED when x = 1 and 3 (NOT 0!!!)



TIPS

Working with the ESP32c3 Super Mini

- Needs to be set in boot mode for first time upload (hold boot switch, press and release rst switch, release boot)
- Needs the ESP32 board library installed
- Runs on a different baud rate for Serial when debugging.

https://dl.artronshop.co.th/ESP32-C3%20SuperMini%20datasheet.pdf

ATMegas:

- Separate ATmega setups – remember external crystal for both

Event handling

- Take inspiration from receiveEvent() in the ESP32 code. You will need similar handling for your ATmega slave.

SOURCES AND USEFUL LINKS

Working with the ESP32c3:

https://randomnerdtutorials.com/getting-started-esp32-c3-super-mini/

I²C slave example for Arduino UNO:

https://projecthub.arduino.cc/PIYUSH_K_SINGH/master-slave-i2c-connection-31a095

HOW TO "PASS" THE PRACTICAL EXERCISES

A full demo of your prototype against the test cases to TA or Simon/Eve.

Preferably during Study Café or lab session

Deadline: 30/09/2025

If you're unable to attend at some point, so you can't demonstrate, a video demo can be sent to TAs or Simon/Eve – include code in mail.

DEPARTMENT OF COMPUTER SCIENCE