Electronic Prototyping

Interrupt, Flow chart, Batteries, Fritzing

Lesson 7

Interrupt

Interrupts

- An interrupt, in microcontroller context, is a signal that temporarily stops what the CPU is currently working at.
- When a sketch is executed, the top most lines are run first. So logically the *setup()* function is run before the *loop()* function. The *loop()*function is an endless loop so there is no way to exit it.
- If we will now use interrupts, we add a third function named *isr()*. ISR is short for *Interrupt Service Routine*. This is where the program jumps to whenever there is an interrupt. An ISR cannot have any parameters, and they shouldn’t return anything.

Interrupts

- When does the program jump to isr()? For the Arduino platform, there will be an interrupt when specific pins change their state. If the interrupt pin is normally high, when it becomes low, then the interrupt is triggered and the program jumps to isr().

```
void setup(){
}

void loop(){
}

void isr(){
}
```

Interrupt pins for different Arduino boards

BOARD	DIGITAL PINS USABLE FOR INTERRUPTS
Uno, Nano, Mini, other 328-based	**2, 3**
Uno WiFi Rev.2	all digital pins
Mega, Mega2560, MegaADK	2, 3, 18, 19, 20, 21
Micro, Leonardo, other 32u4-based	0, 1, 2, 3, 7
Zero	all digital pins, except 4
MKR Family boards	0, 1, 4, 5, 6, 7, 8, 9, A1, A2
Due	**all digital pins**
101	all digital pins (Only pins 2, 5, 7, 8, 10, 11, 12, 13 work with **CHANGE**)

Syntax: attachInterrupt()

- **attachInterrupt**(digitalPinToInterrupt(pin), ISR, mode);   Arduino Uno

- **attachInterrupt**(pin, ISR, mode);   Arduino Due

Syntax: attachInterrupt()

mode: defines when the interrupt should be triggered. Four constants are predefined as valid values:

- **LOW** to trigger the interrupt whenever the pin is low,
- **CHANGE** to trigger the interrupt whenever the pin changes value
- **RISING** to trigger when the pin goes from low to high,
- **FALLING** for when the pin goes from high to low.

The Due, Zero and MKR1000 boards allows also:

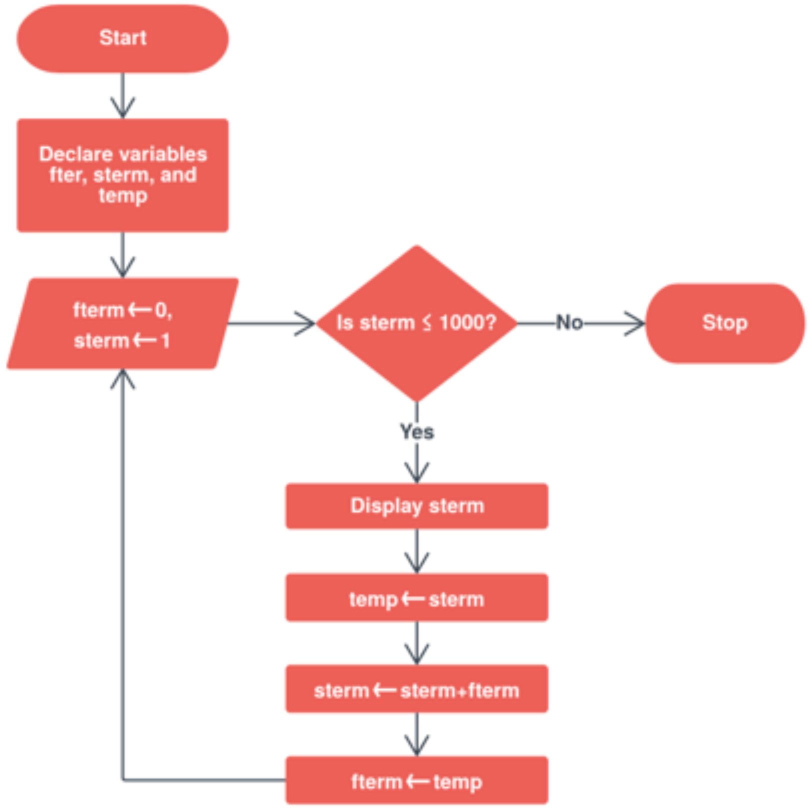
- **HIGH** to trigger the interrupt whenever the pin is high.

Example

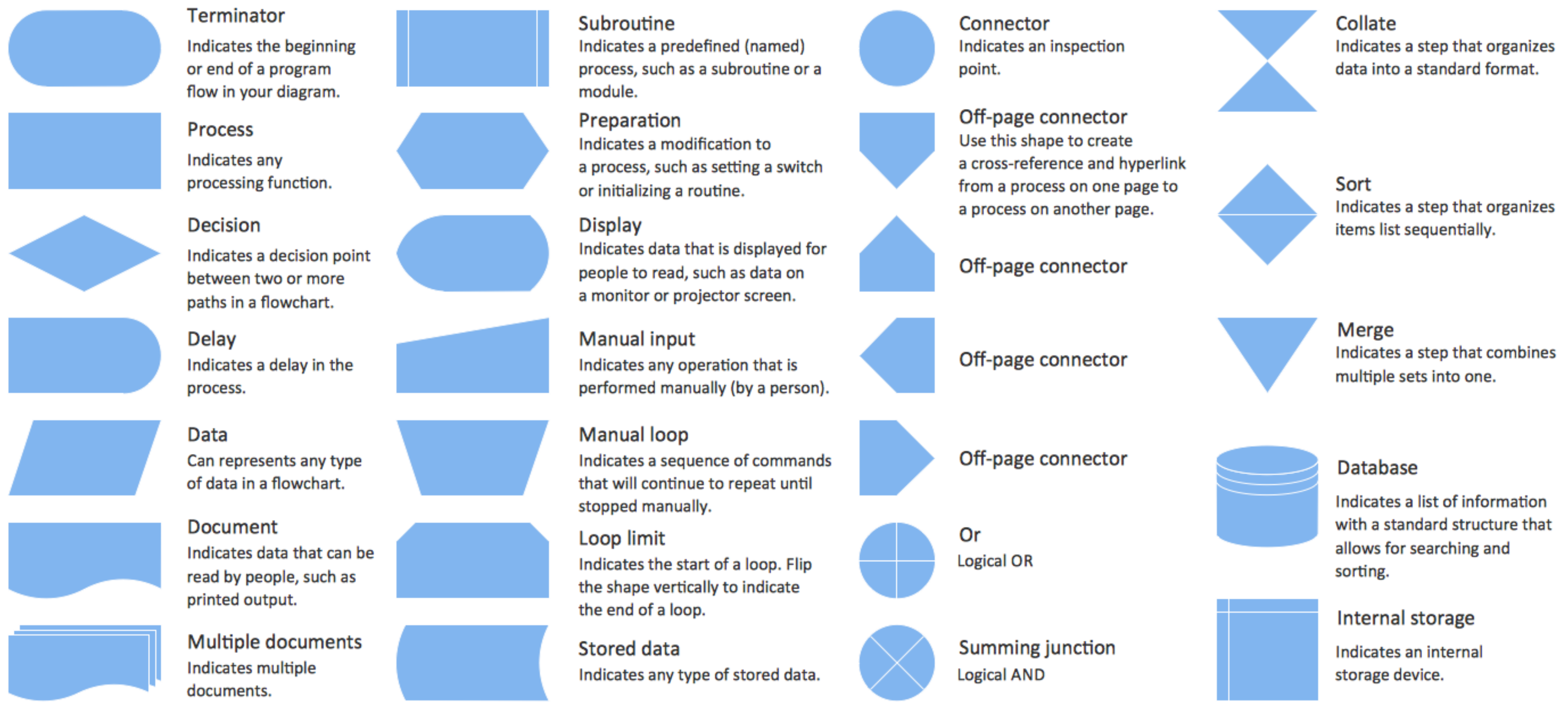
```
const byte ledPin = 13;
const byte interruptPin = 2;
volatile byte state = LOW;

void setup() {
pinMode(ledPin, OUTPUT);
pinMode(interruptPin, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(interruptPin), blink, CHANGE);   //Arduino Uno
attachInterrupt(2, blink, CHANGE);   //Arduino Due
}
void loop() {
digitalWrite(ledPin, state);
}
void blink() {
state = !state;
}
```

Flow chart

Flow chart


Start
Declare variables fter, sterm, and temp
fterm ← 0, sterm ← 1
Is sterm ≤ 1000?
No
Stop
Yes
Display sterm
temp ← sterm
sterm ← sterm+fterm
fterm ← temp

Flow chart elements



Terminator
Indicates the beginning or end of a program flow in your diagram.

Process
Indicates any processing function.

Decision
Indicates a decision point between two or more paths in a flowchart.

Delay
Indicates a delay in the process.

Data
Can represents any type of data in a flowchart.

Document
Indicates data that can be read by people, such as printed output.

Multiple documents
Indicates multiple documents.

Subroutine
Indicates a predefined (named) process, such as a subroutine or a module.

Preparation
Indicates a modification to a process, such as setting a switch or initializing a routine.

Display
Indicates data that is displayed for people to read, such as data on a monitor or projector screen.

Manual input
Indicates any operation that is performed manually (by a person).

Manual loop
Indicates a sequence of commands that will continue to repeat until stopped manually.

Loop limit
Indicates the start of a loop. Flip the shape vertically to indicate the end of a loop.

Stored data
Indicates any type of stored data.

Connector
Indicates an inspection point.

Off-page connector
Use this shape to create a cross-reference and hyperlink from a process on one page to a process on another page.

Off-page connector

Off-page connector

Off-page connector

Or
Logical OR

Summing junction
Logical AND

Collate
Indicates a step that organizes data into a standard format.

Sort
Indicates a step that organizes items list sequentially.

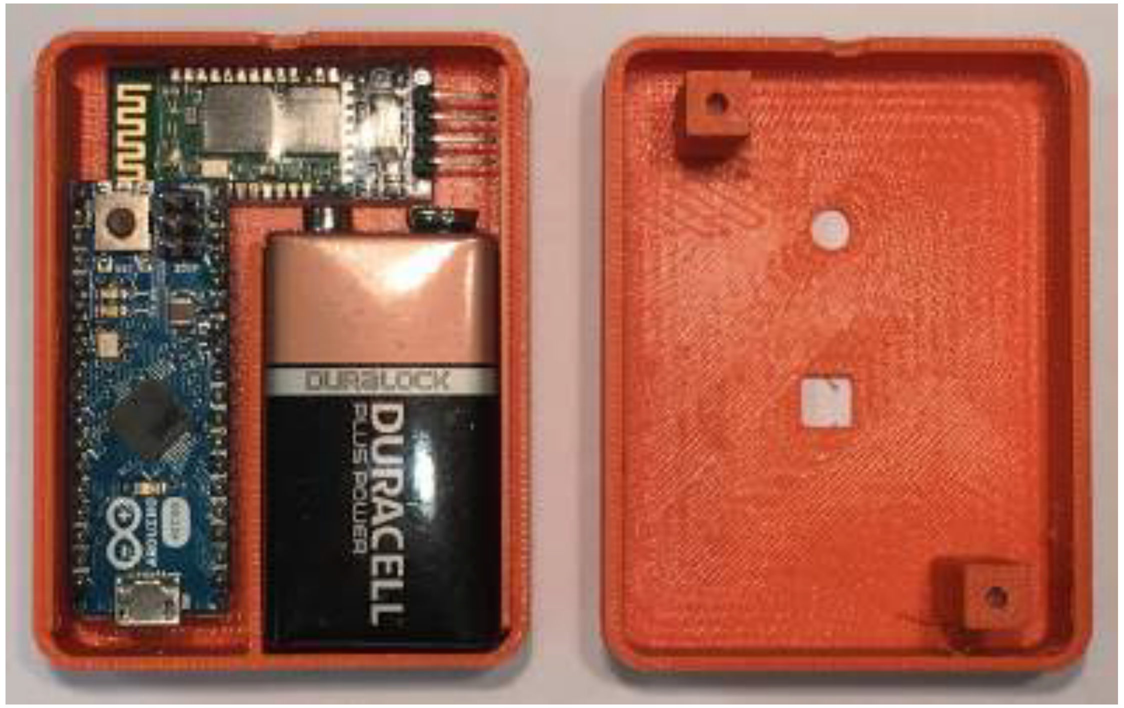
Merge
Indicates a step that combines multiple sets into one.

Database
Indicates a list of information with a standard structure that allows for searching and sorting.

Internal storage
Indicates an internal storage device.

Batteries

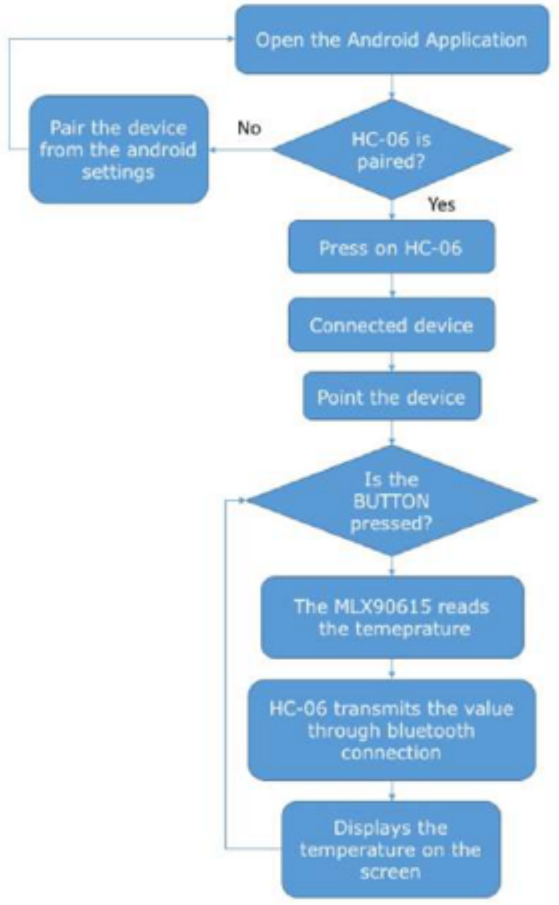
Example: contactless thermometer

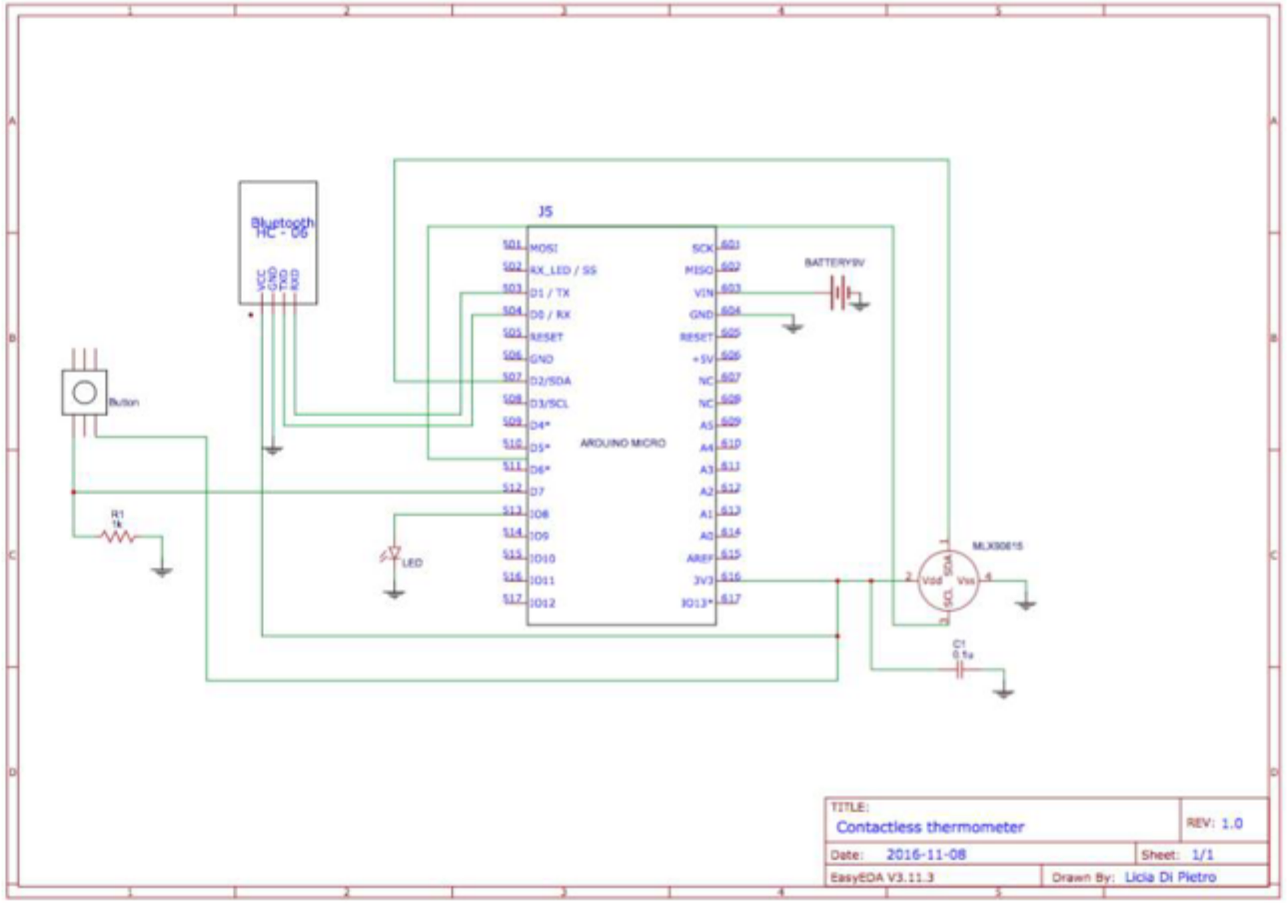


MLX90615 by Melexis

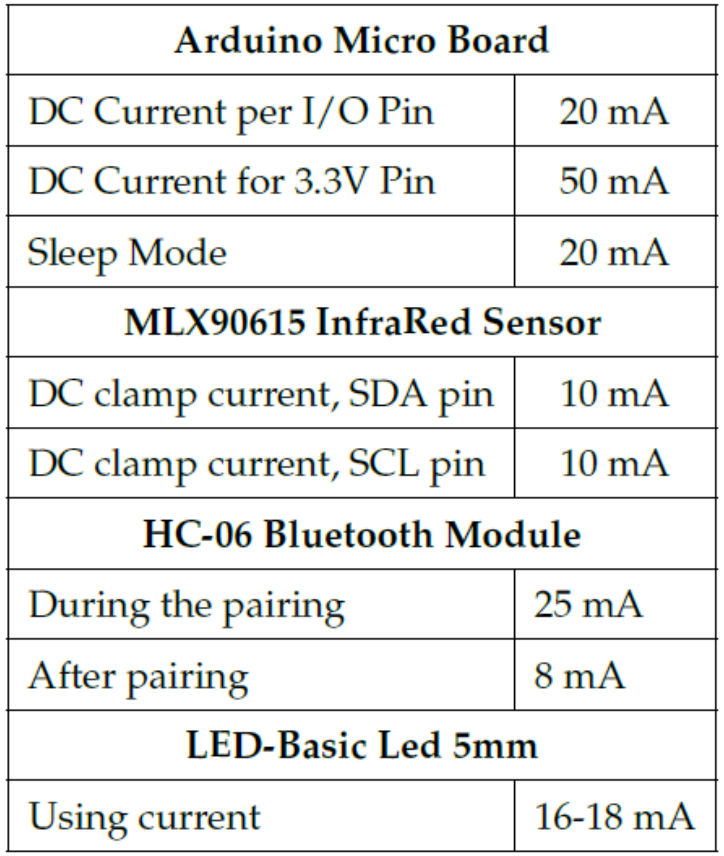
Battery capacity 550 mAh @9 V

Contactless thermometer


Open the Android Application
HC-06 is paired?
No
Pair the device from the android settings
Yes
Press on HC-06
Connected device
Point the device
Is the BUTTON pressed?
The MLX90615 reads the temeprature
HC-06 transmits the value through bluetooth connection
Displays the temperature on the screen

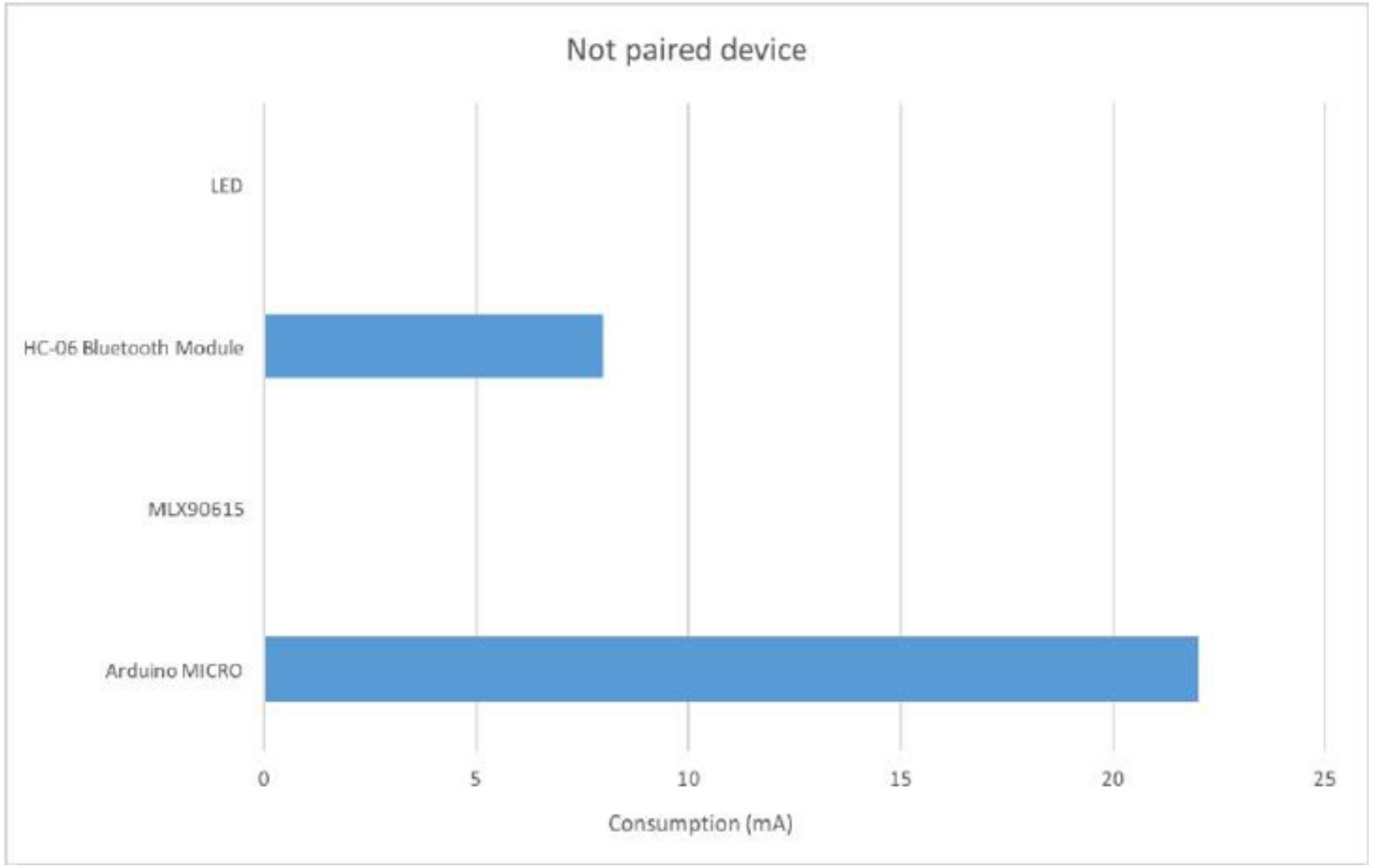


Current consumption

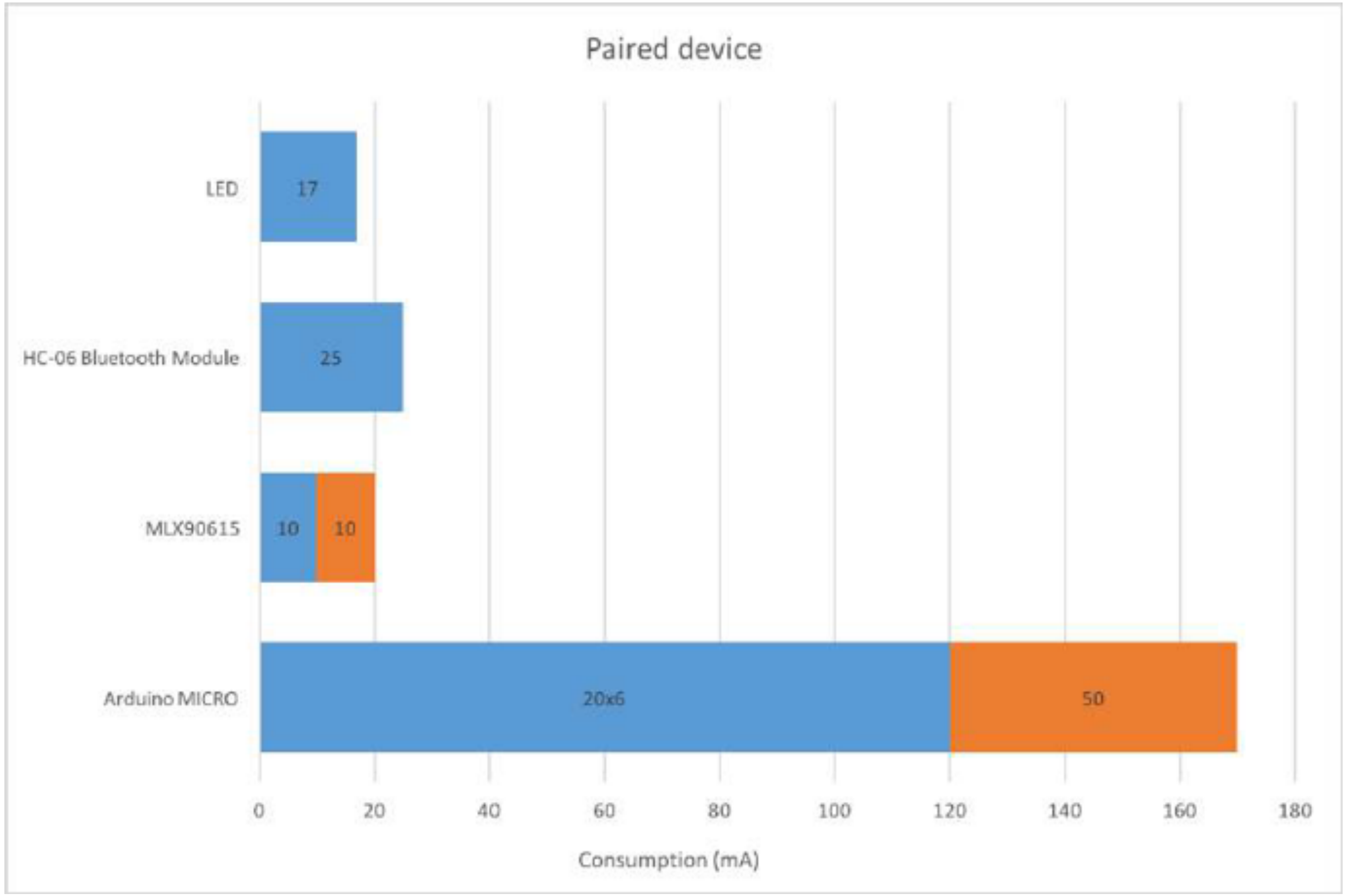


Arduino Micro Board	
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Sleep Mode	20 mA
MLX90615 InfraRed Sensor	
DC clamp current, SDA pin	10 mA
DC clamp current, SCL pin	10 mA
HC-06 Bluetooth Module	
During the pairing	25 mA
After pairing	8 mA
LED-Basic Led 5mm	
Using current	16-18 mA

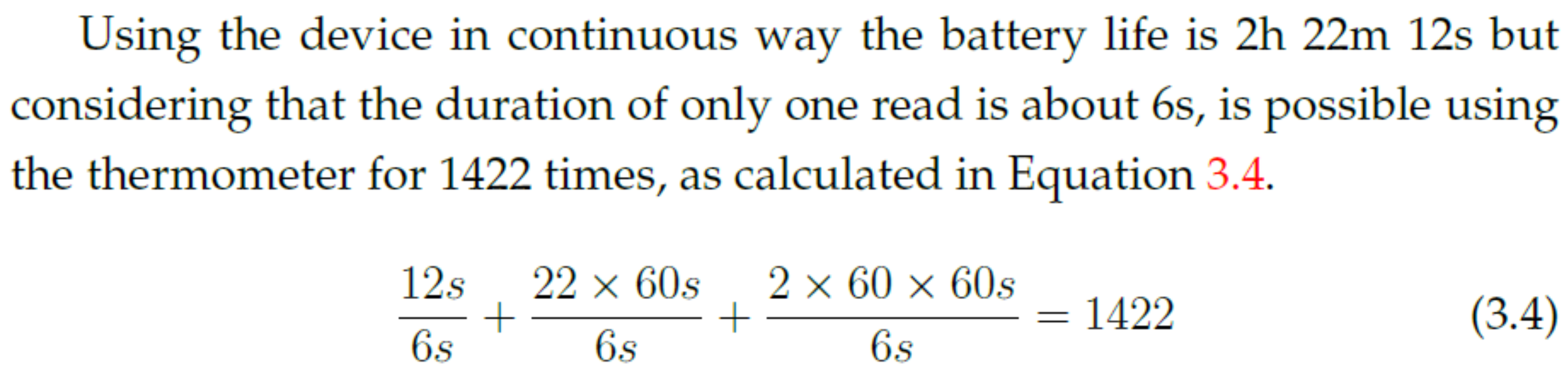
Current consumption



Current consumption



Current consumption



Using the device in continuous way the battery life is 2h 22m 12s but considering that the duration of only one read is about 6s, is possible using the thermometer for 1422 times, as calculated in Equation 3.4.

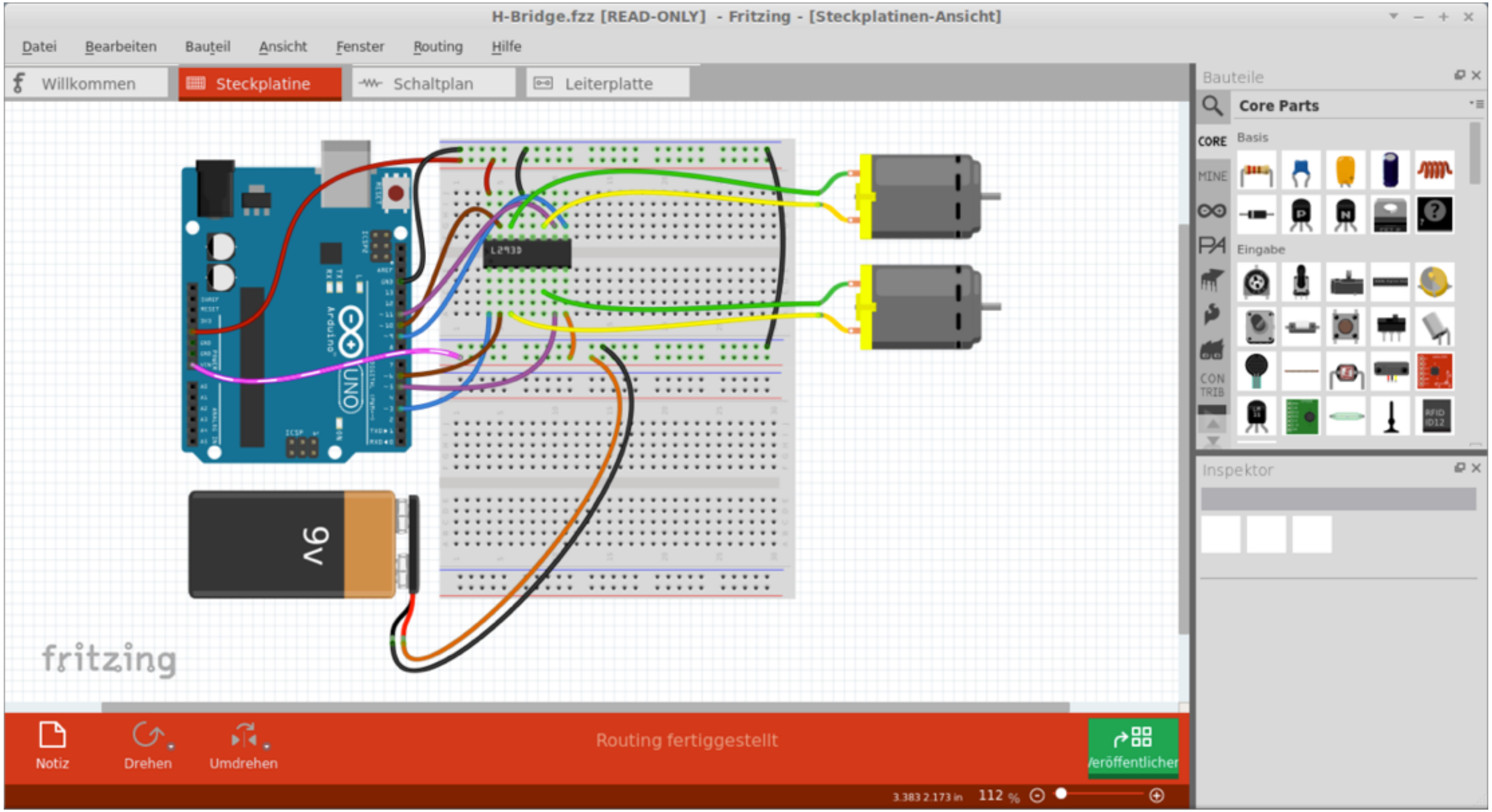
$$\frac{12s}{6s} + \frac{22 \times 60s}{6s} + \frac{2 \times 60 \times 60s}{6s} = 1422 \qquad (3.4)$$

Fritzing & Tinkercad

How to simulate your prototype?

https://fritzing.org/home/

https://www.tinkercad.com/


H-Bridge.fzz [READ-ONLY] - Fritzing - [Steckplatinen-Ansicht]
Datei
Bearbeiten
Bauteil
Ansicht
Fenster
Routing
Hilfe
Willkommen
Steckplatine
Schaltplan
Leiterplatte
Bauteile
Core Parts
CORE
MINE
CON TRIB
Basis
Eingabe
Inspektor
L293D
9V
fritzing
Notiz
Drehen
Umdrehen
Routing fertiggestellt
Veröffentlichen
3.383 2.173 in
112 %