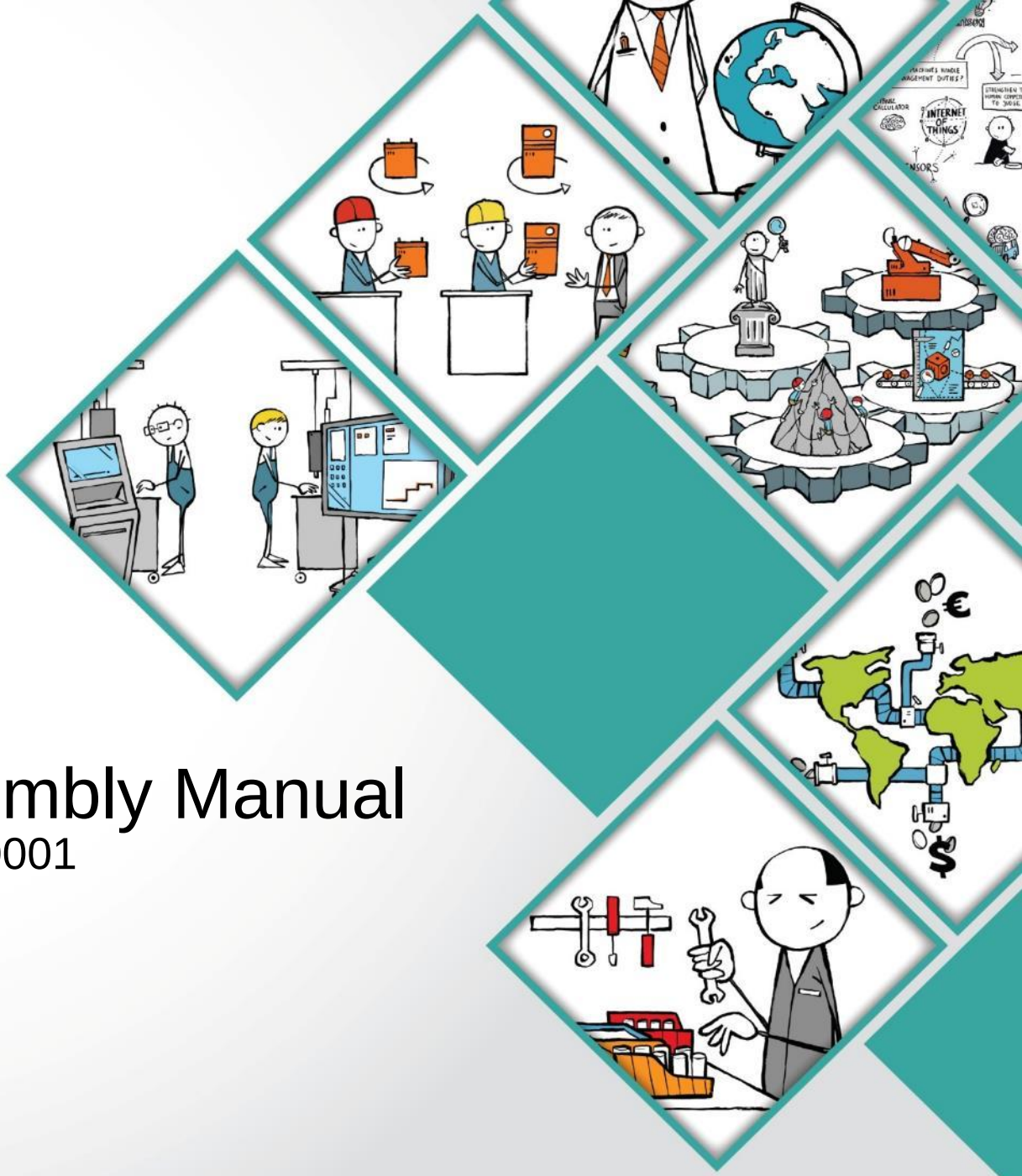
IoT FAB



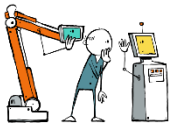
IoT Event Box Assembly Manual

IoT Testbed Motion Sensor TT0001

ROI LEAN FAB

EP 05000_v03

Additional Information



The Tip icon highlights expert shortcuts or **simple ideas** that can make life easier for you.

This icon highlights something that´s particularly technical. We´ve tried to avoid unnecessary jargon and complexity, but some **background information** can give you a better understanding of what you´re doing. Sections highlighted with this icon might be worth rereading to make sure you understand.

Although we´d like to think that reading this document is an unforgettable experience, we´ve highlighted some points that you might want to particularly **commit to memory**. They´re either important take-aways, or they are fundamental to the project you´re working on.

As you would on the road, slow down when you see a **warning sign**. It highlights an area where things could go wrong.

Simulation

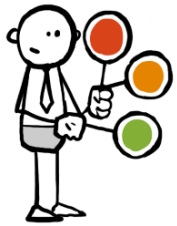
Further information

Survey

Video example

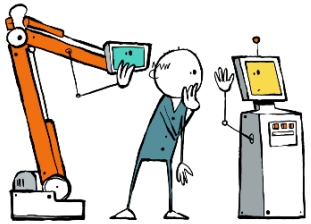
Reference slide

Introduction to Lean & IoT & Six Sigma FAB



LEAN FAB

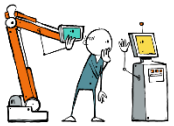
Six Sigma FAB



IoT FAB

The Training and simulation center in Jesenice - Prague

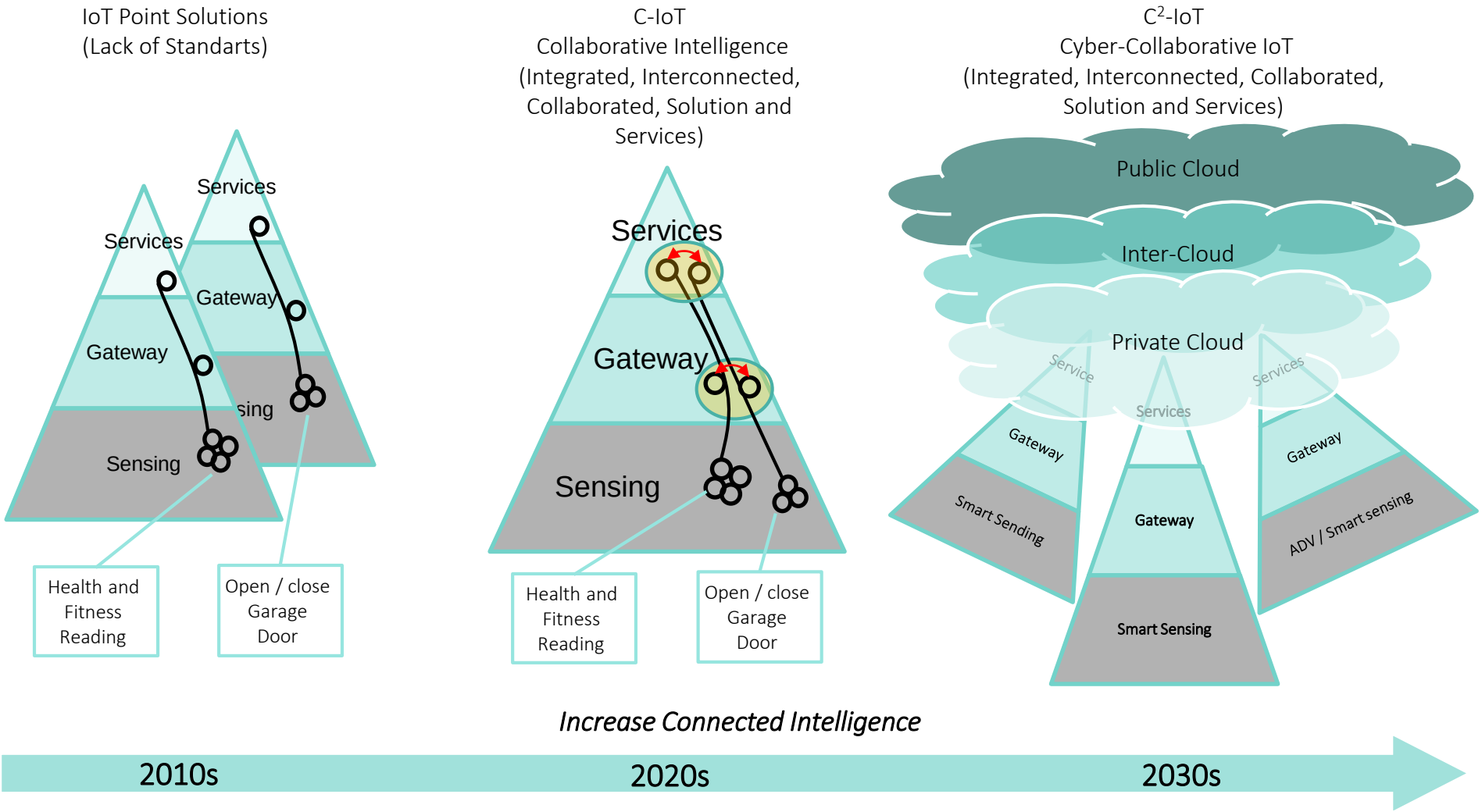
Agenda



1.	**Introduction to IoT**
2.	Prerequisites for Event Box Assembly
3.	Event Box Assembly Procedure
4.	Connection of Arduino Yun to WiFi Network
5.	Arduino IDE Download and Operation

Introduction to IoT

Vision of the IoT development



Source: Collaborative Internet of Things (C-IoT), BEHMAN, WOO

© ROI Management Consulting – EP 05000_Event Box Manual_v03

Introduction to IoT

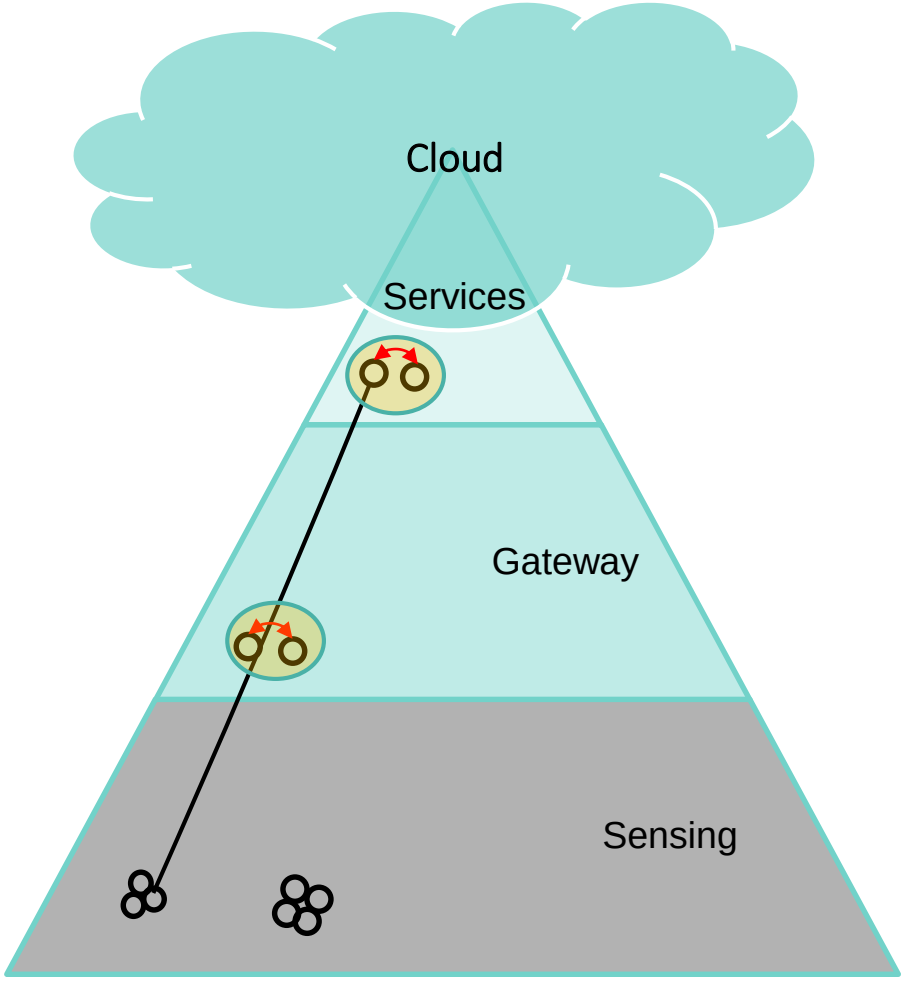
Todays focus

- Abstract model of the Motion Sensor

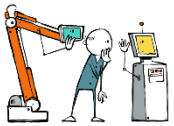
  - ThingSpeak Database
  - ThingSpeak REST Api

  - Internet
  - Provider
  - Router

  - Sensor & Arduino
  - Edge Analytics



Agenda

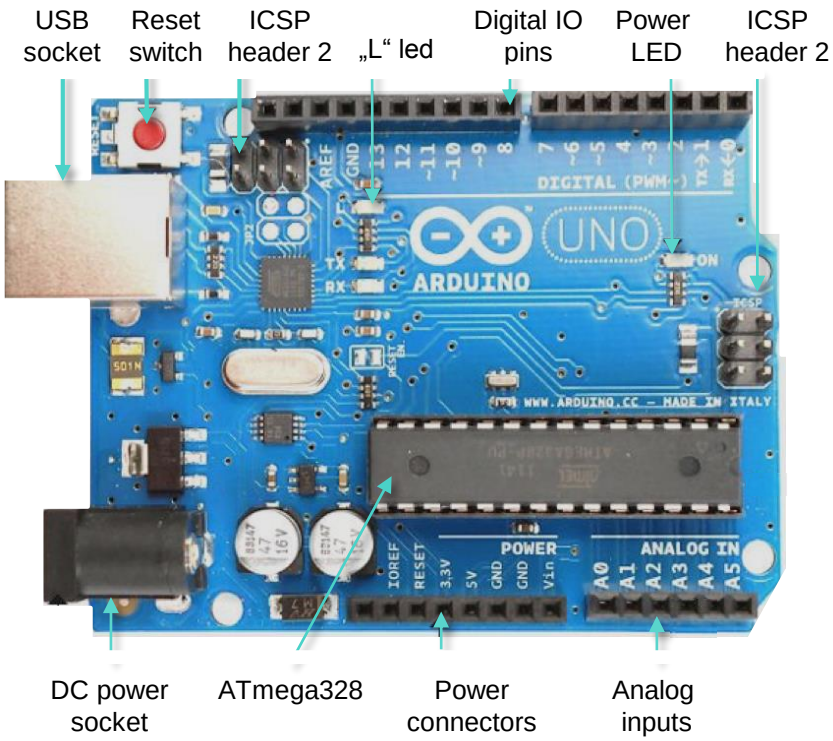


1. Introduction to IoT
2. **Prerequisites for the Event Box Assembly**
3. Event Box Assembly Procedure
4. Connection of Arduino Yun to WiFi Network
5. Arduino IDE Download and Operation

Introduction to IoT

Arduino (1)

- The project is based on a family of microcontroller board designs manufactured primarily by SmartProjects in Italy, using various 8-bit Atmel AVR microcontrollers or 32-bit Atmel ARM processors.
- These systems provide sets of digital and analog I/O pins that can be interfaced to various expansion boards ("shields") and other circuits.
- The boards feature serial communications interfaces, including USB on some models, for loading programs from personal computers.
- For programming the microcontrollers, the Arduino platform provides an integrated development environment (IDE) based on the Processing project, which includes support for C, C++ and Java programming languages.



Introduction to IoT

Arduino (2)

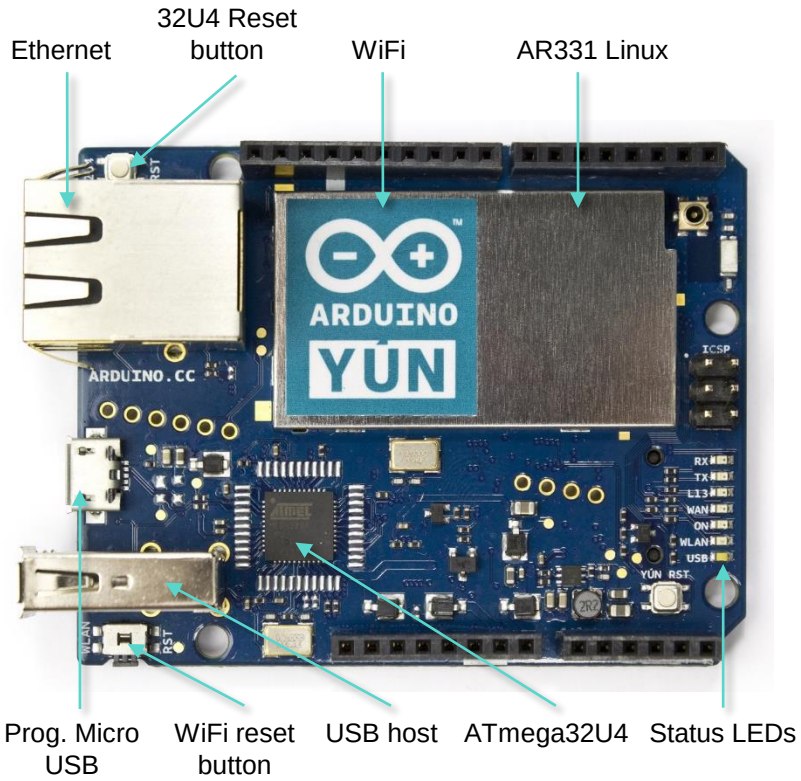
- Arduino is an open-source platform, which can be connected to all kinds of sensors and other devices.
- It was first released in 2005 as a modest tool for students of professor Massimo Banzi in Ivrea, Italy.
- Banzi co-developed Arduino while teaching at a design school in northwest Italy, simply because there weren't any good hardware options for his students. "We had to figure out something that would be simple, cheaper, USB plug and play, and you could program on Windows, Mac, and Linux," he said. Arduino was expected to be useful "in that particular tiny context," but it morphed into something much bigger. "It sort of escaped the lab“, Banzi said.
- The Arduino core team is made from four more people – David Cuartielles, Gianluca Martino, Tom Igoe and David Mellis.

Further information

Introduction to IoT

Arduino Yun

- The Arduino Yún is a microcontroller board based on the ATmega32u4 and the Atheros AR9331.
- The Atheros processor supports a Linux distribution based on OpenWrt named OpenWrt-Yun.
- Arduino´s flash memory is 32 KB and clock speed is 16 MHz.
- The board has built-in Ethernet and WiFi support, a USB-A port, micro-SD card slot, 20 digital input/output pins (of which 7 can be used as PWM outputs and 12 as analog inputs), a 16 MHz crystal oscillator, a micro USB connection, an ICSP header and a 3 reset buttons.
- The most significant differences in comparison with Arduino Uno: Yun has integrated WiFi, linux with open wrt distribution, two processors, bigger flash memory and microUSB connector.

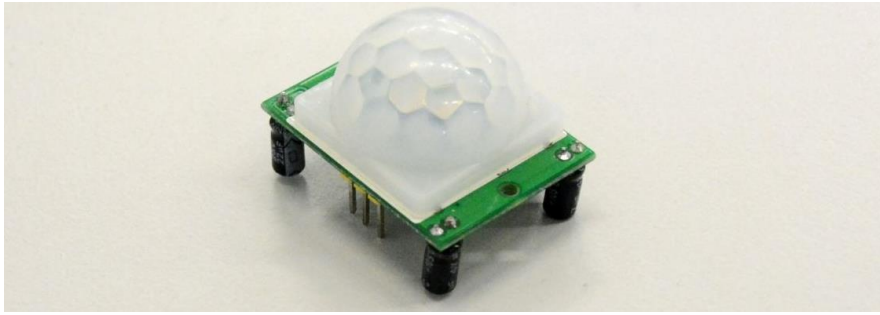


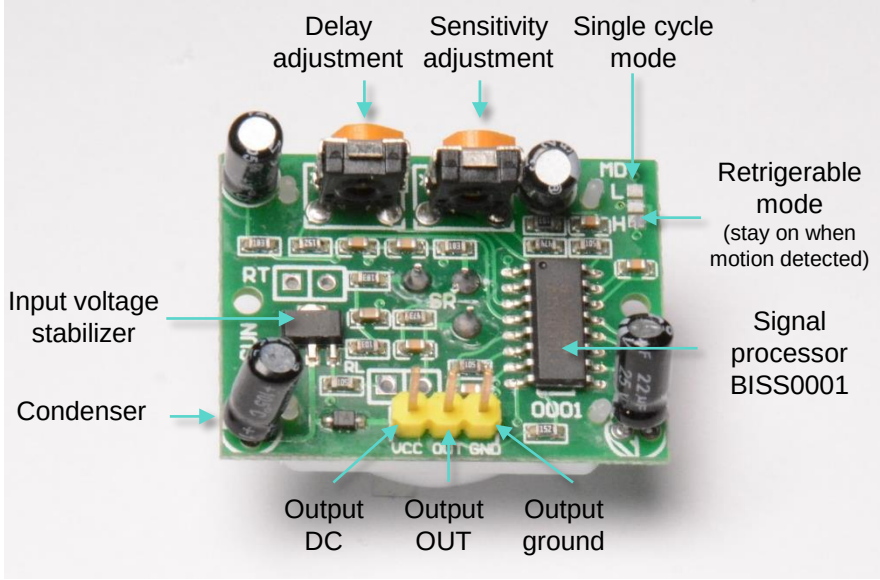
Introduction to IoT

PIR Motion Sensor

Technical specification:

- Infrared sensor for movement detection
- Circuit board dimensions are 32 mm x 24 mm
- Sensitivity and length of duration are adjustable
- Working voltage range is 4,5V – 20V DC
- Static current is less than 60uA
- Sensing distance is 7 m (adjustable)
- Sensing range is less than 104°
- Time delay is between 5 and 200 s (adjustable; default setting is 5 s ± 3 %)
- Blocking time is 2,5 s (default)
- Working temperature from -20°C to + 80°C
- Trigger mode: L unrepeatable trigger / H repeated trigger (default)



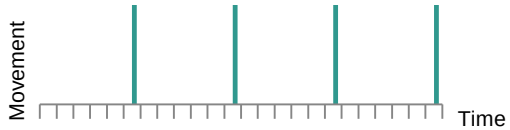


Introduction to IoT

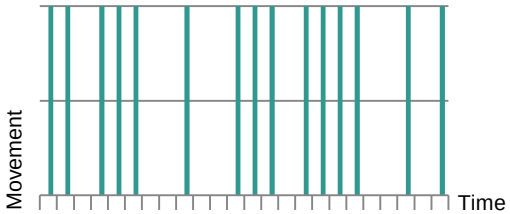
Edge Analytics

- The PIR motion sensor detects movement and when there is a motion the value is HIGH and reversely, the value is LOW when there is no motion
- If motion sensed, Arduino waits for 2 seconds
- Arduino transfers a movement count every 2 minutes

Data frequency:



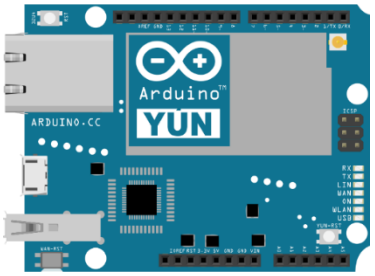
Algorithm processed movements



Random movements

PIR Motion Sensor

Value HIGH or LOW



Value of current motions

ThingSpeak

Introduction to IoT

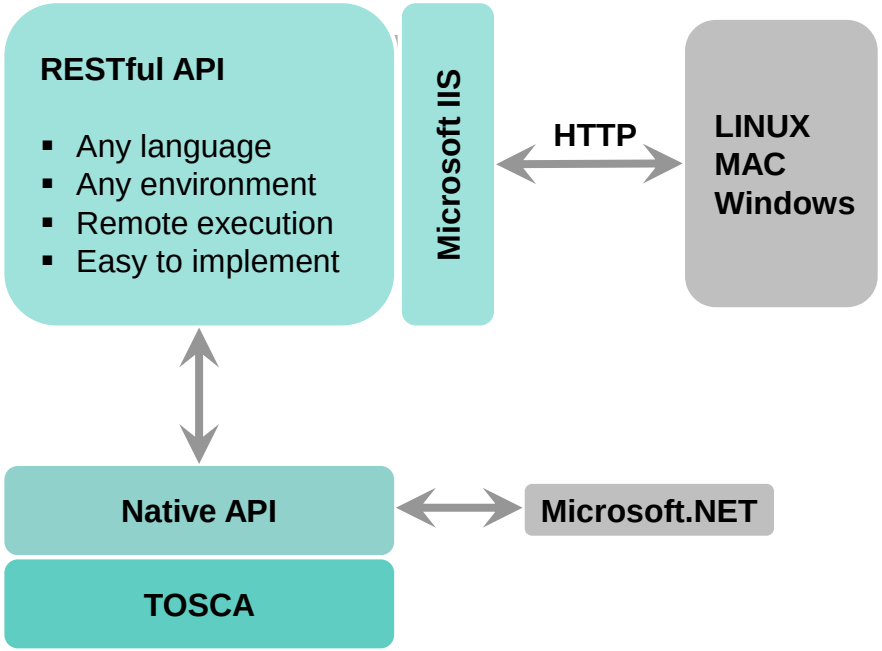
API

- An API makes a software component in terms of its operations, inputs, outputs, and underlying types available.
- An API defines functionalities that are independent of their respective implementations, which allows definitions and implementations to vary without compromising the interface.
- A good API makes it easier to develop a program by providing all the building blocks, a programmer then puts the blocks together.
- API key is used for data assing to a selected channel, where the data are assigned by their fieldname to a specific graph.
- Key word is word „update“, because it determines the API function.

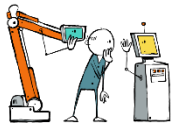
thingspeak_update_API + thingspeak_write_API_key + thingspeakfieldname + data

http://api.thingspeak.com/update?key=RWI2E3R0UAF62ZT4&field1=0

API CALLS EXAMPLES:
api.openweathermap.org/data/2.5/weather?q=London
api.openweathermap.org/data/2.5/weather?q=London,uk



{"coord":{"lon":-0.13,"lat":51.51},"weather":[{"id":520,"main":"Rain,,"description":"light intensity shower rain","icon":"09d"},{"id":721,"main":"Haze","description":"haze","icon":"50d"},{"id":311,"main":"Drizzle","description":"rain and drizzle","icon":"09d"}],"base":"cmc stations","main":{"temp":289.02,"pressure":1004,"humidity":93,"temp_min":288.15,"temp_max":290.15},"wind":{"speed":4.6,"deg":160},"rain":{"1h":0.64},"clouds":{"all":92},"dt":1444040911,"sys":{"type":1,"id":5093,"message":0.0055,"country":"GB","sunrise":1444025267,"sunset":1444066145},"id":2643743,"name":"London","cod":200}



Introduction to IoT

ThingSpeak

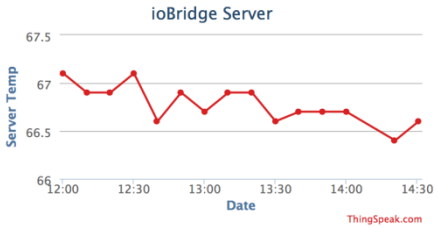
- ThingSpeak is an open source „Internet of Things“ application and API for storing and retrieval of data from devices using HTTP via Internet.
- ThingSpeak allows for example to create applications for sensor logging or location tracking.

THINGSPEAK EXAMPLE:

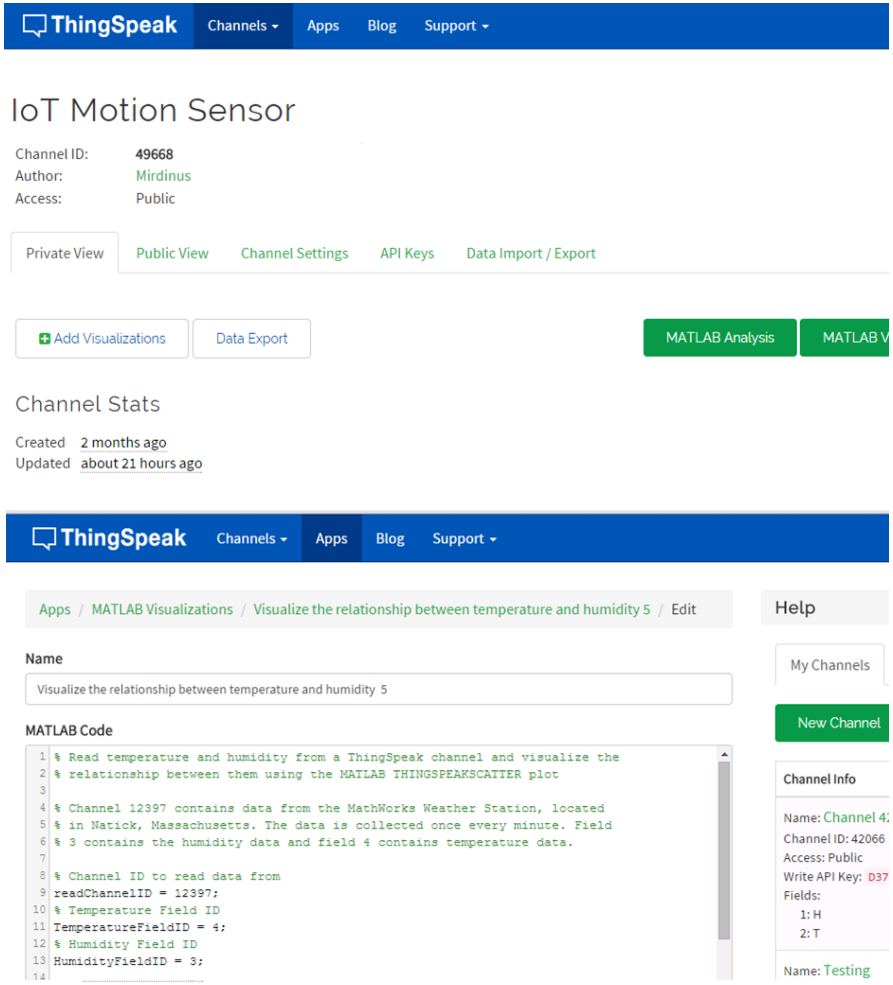
Chart Embed Code

```
<iframe width="450" height="250" style="border: 1px solid #cccccc;" src="http://
thingspeak.com/channels/CHANNEL_ID/charts/FIELD_ID"></iframe>
```

Example of Chart



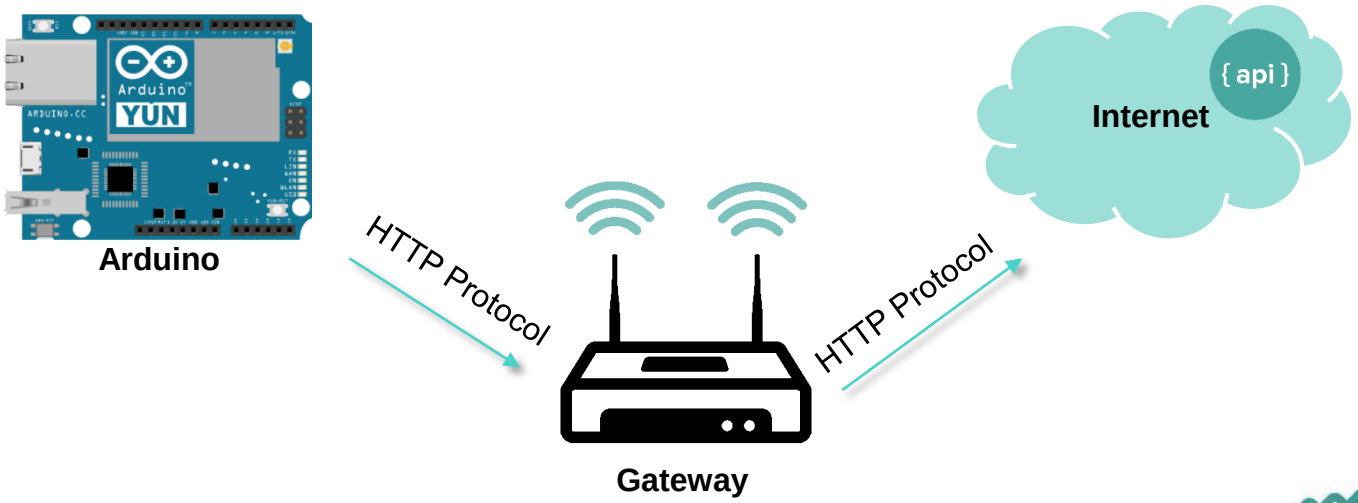
Further information



Introduction to IoT

Gateway

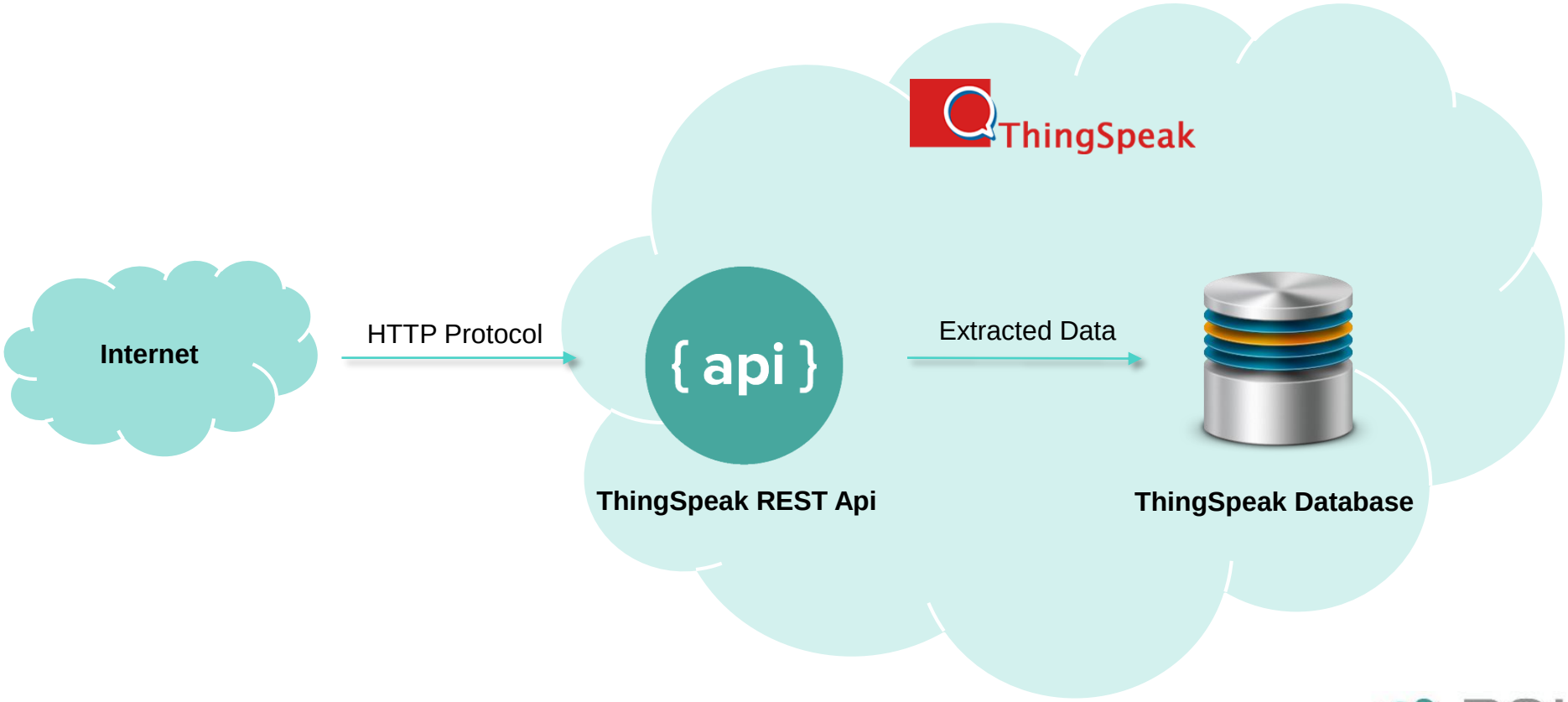
- A gateway represents a device used to connect two different networks, especially a connection to the Internet
- A gateway acts as a portal between two programs allowing them to share information by communicating between protocols on a computer or between dissimilar computers
- In our example the Arduino transfers the HTTP protocol to the Access Point, which sends them to the ThingSpeak REST Api via the internet



Introduction to IoT

Cloud Services

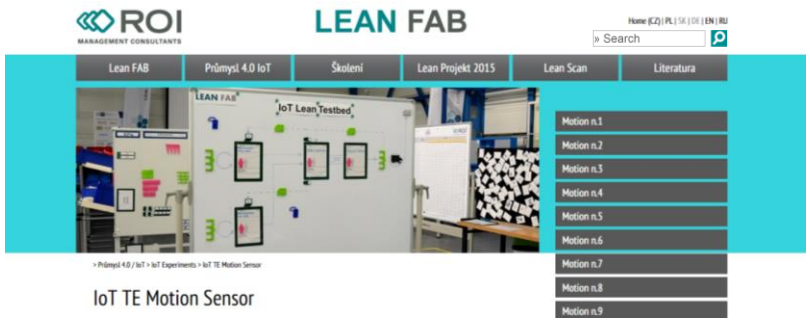
- The ThingSpeak REST Api receives the compresed data in HTTP protocol from the internet, extracts them and then sends the data to the ThingSpeak Database



Prerequisites for Event Box Assembly

Prerequisites for the Workshop

- Notebook (with operating system Windows or Mac)
- WiFi connection to internet
- Power supply 220 V
- Event Box – please see following slide „List of Components for Event Box”
- Assembly Event Box Manual (on USB stick)
- Web site (portal)

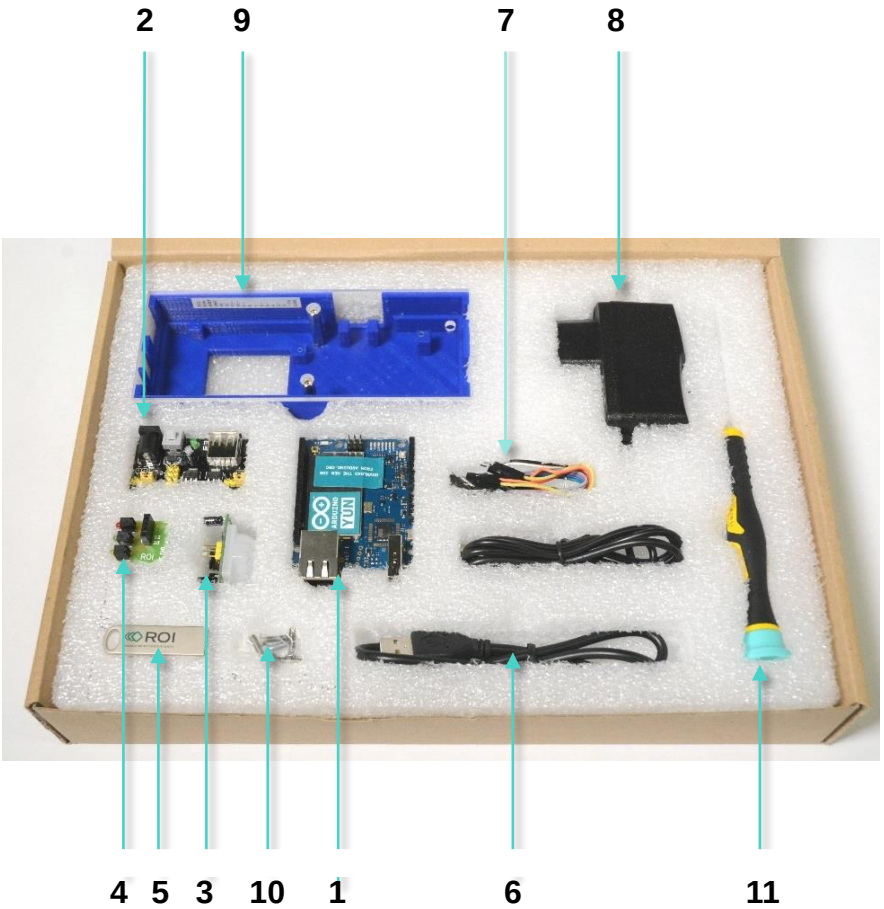


Prerequisites for Event Box Assembly

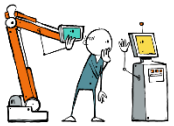
List of Components for Event Box

Box content:

1. Arduino Yun
2. Power supply 5V
3. Motion sensor PIR
4. First ROI PCB
5. Flash disk
6. Cable microUSB
7. Jump wires
8. Power supply 9V DC
9. Box with cover
10. Screws
11. Screwdriver



Agenda

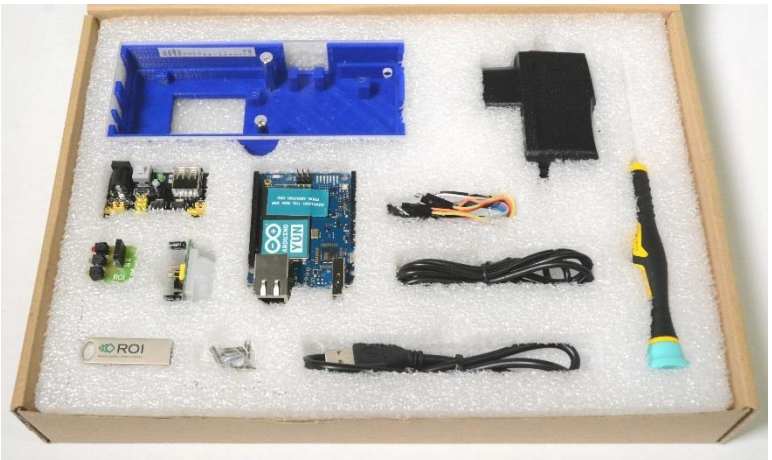


1. Introduction to IoT
2. Prerequisites for Event Box Assembly
3. **Event Box Assembly Procedure**
4. Connection of Arduino Yun to WiFi Network
5. Arduino IDE Download and Operation

Event Box Assembly Procedure

Overview

- All required componets are provided in the Event Box

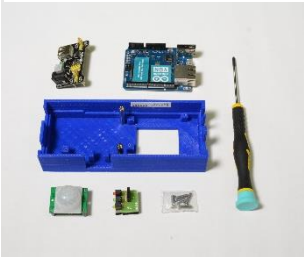


Event Box Assembly Procedure

Assembly Overview

- The assembly process is divided into four major steps:

A) Mechanical Assembly



Time required: 5 min

B) Wiring

Time required: 10 min

C) Software Download

Time required: 45 min

D) Operation

Time required: 10 min

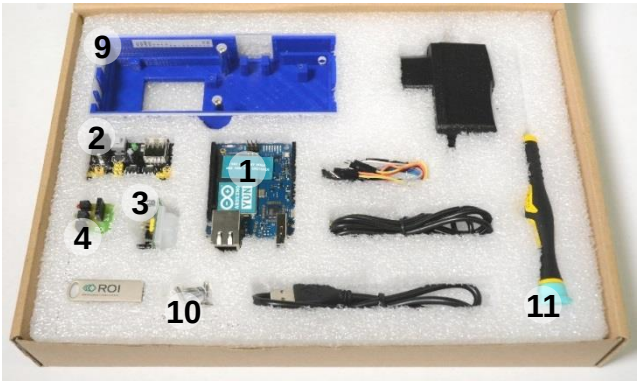
Event Box Assembly Procedure

A) Mechanical Assembly (1)

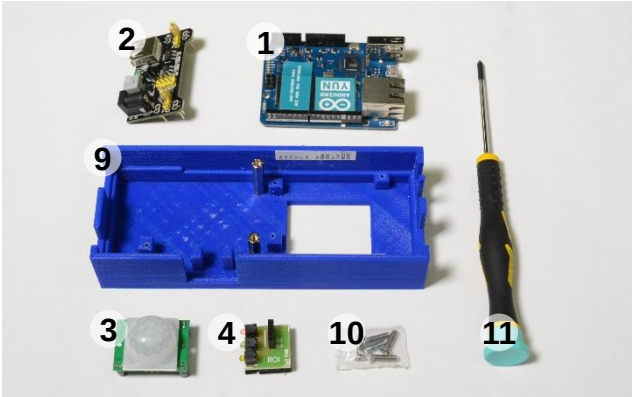
Components:

- Take out the **components 1, 2, 3, 4, 9**, **10** and the **screwdriver 11**
- In the **package (10)** are two sizes of screws – four small ones (for power supply and Arduino installation) and two big ones (for plexiglass cover fixation)

BEFORE



AFTER



Time required: 1 minute

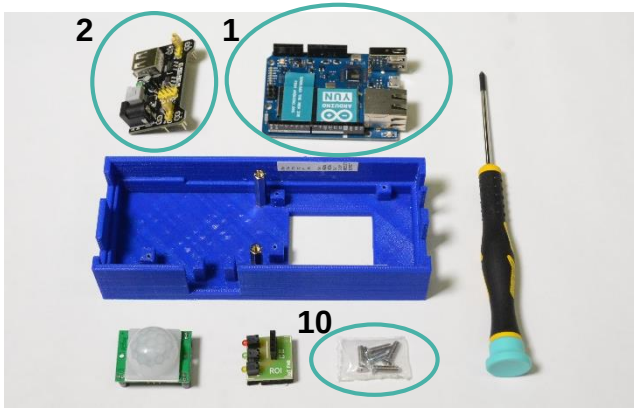
Event Box Assembly Procedure

A) Mechanical Assembly (2)

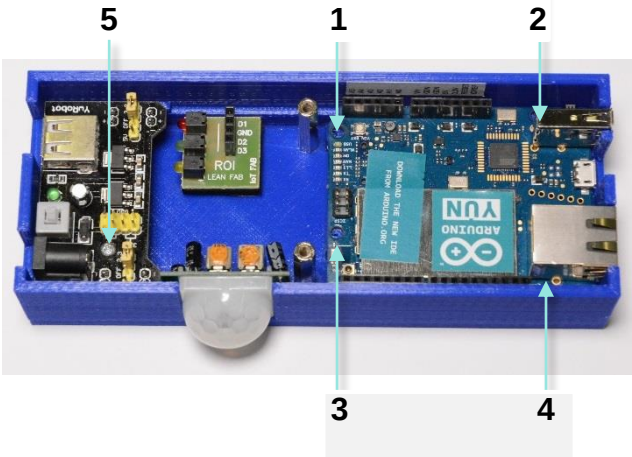
Components assembly:

- Open the **packet (10)** with screws
- Place the **Arduino Yun (1)** to its location (see photo) and fixate it with four small screws (see arrows 1,2,3,4)
- Install the **power supply (2)** and tighten it with one screw (see arrow 5)

BEFORE



AFTER



Time required: 2 minutes

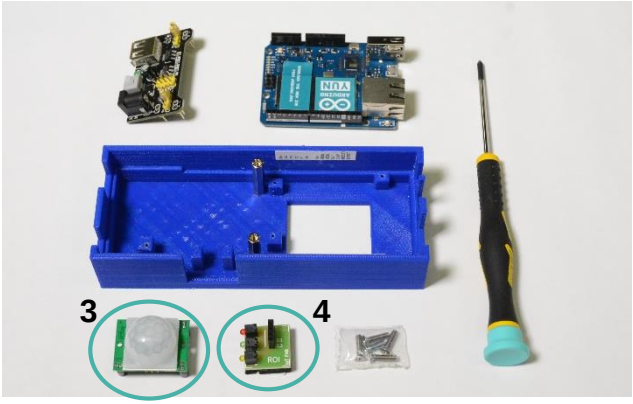
Event Box Assembly Procedure

A) Mechanical Assembly (3)

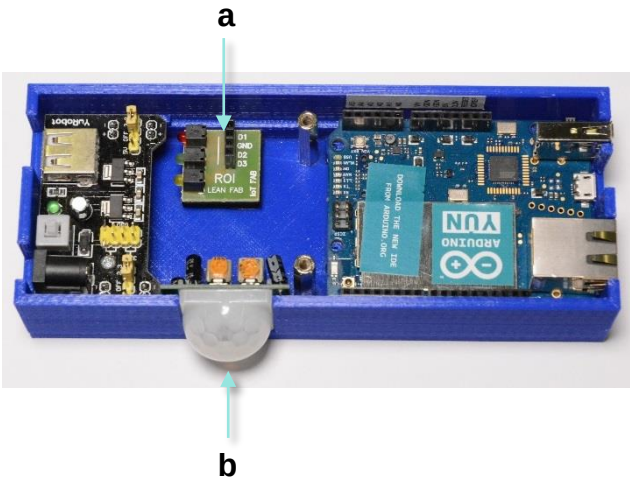
Components assembly:

- Peel off the protective foil from the **PCB (4)** and place the PCB to a location between the Arduino Yun and the Power supply (see arrow a)
- Fit the **motion sensor PIR (3)** into the cut-out section in the Box and orient the connectors down (see arrow b)

BEFORE



AFTER



Time required: 2 minutes

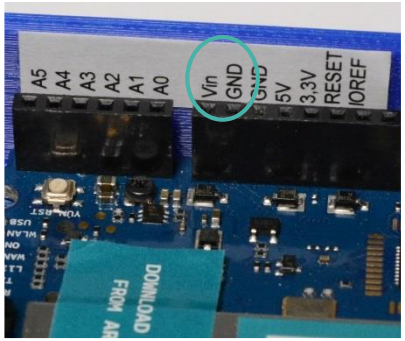
Event Box Assembly Procedure

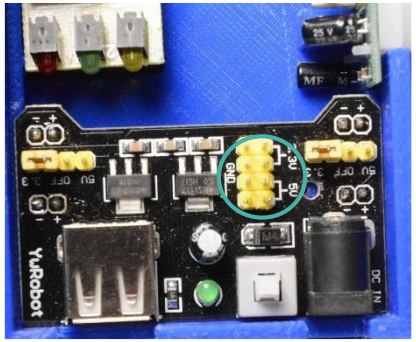
B) Wiring (1)

Connecting the components :

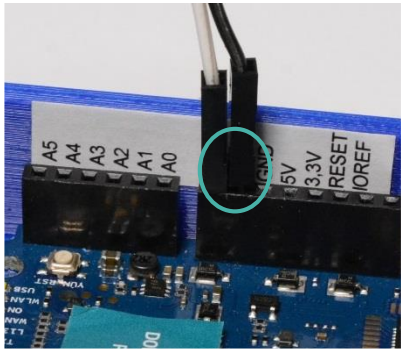
1. Connect the **black cable** to the output pin „GND” on the power supply (2) and to the input pin „GND” on the Arduino Yun (1) (see photos)
2. Connect the **white cable** to the output pin „5V” on the power supply (2) and to the input pin „Vin” on the Arduino Yun (1) (see photos)

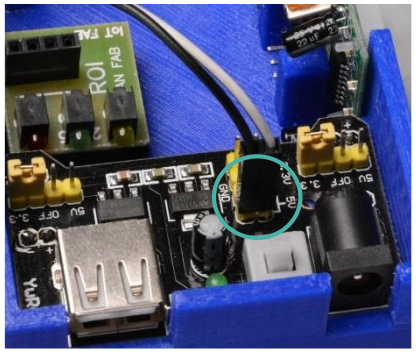
BEFORE



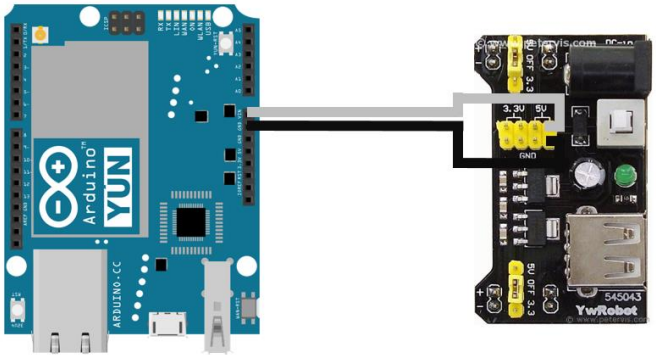


AFTER





CONNECTION SCHEME



Time required: 2 minutes

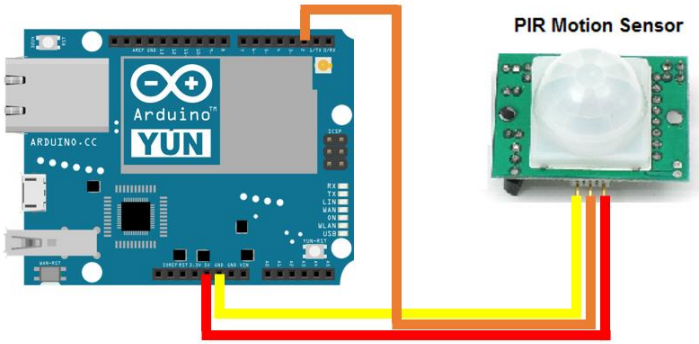
Event Box Assembly Procedure

B) Wiring (2)

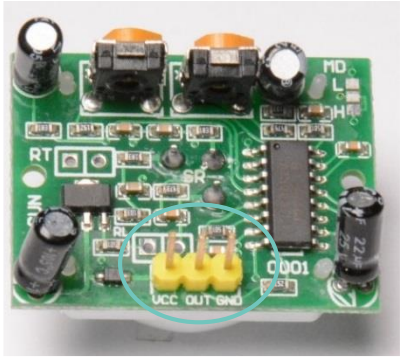
Connecting the components :

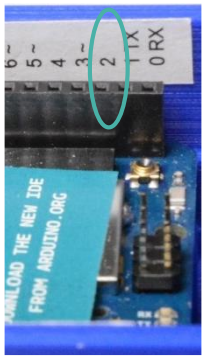
3. Connect the **red cable** to the input pin „5V” on the Arduino Yun (1) and to the output pin „VCC” on the motion sensor PIR* (3) (see photos)
4. Connect the **yellow cable** to the input pin „GND” on the Arduino Yun (1) and to the output pin „GND” on the motion sensor PIR (3) (see photos)
5. Connect the **orange cable** to the input pin „2” on the Arduino Yun (1) and to the output pin „OUT” on the motion sensor PIR (3) (see photos)

CONNECTION SCHEME

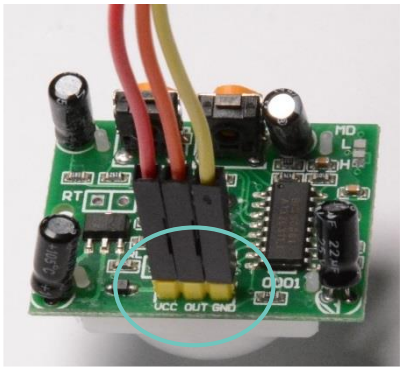


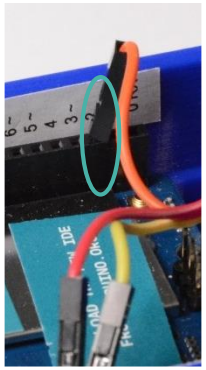
BEFORE

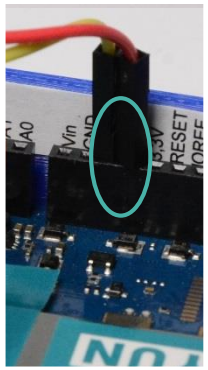




AFTER







* You may take out the motion sensor PIR for better accessibility of its outputs.

Time required: 4 minutes

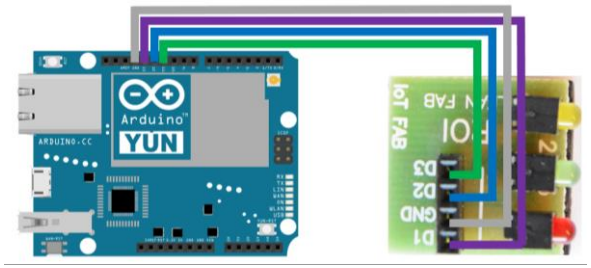
Event Box Assembly Procedure

B) Wiring (3)

Connecting the components :

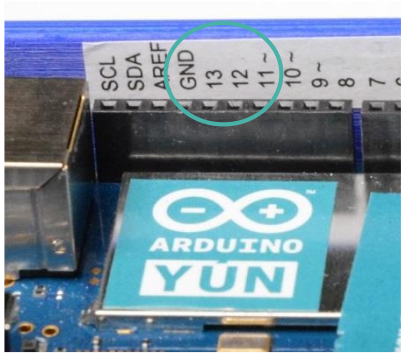
6. Connect the **grey cable** to the input pin „GND” on the Arduino Yun (1) and to the input pin „GND” on the PCB (4) (see photos)
7. Connect the **purple cable** to the input pin „13” on the Arduino Yun (1) and to the input pin „D1” on the PCB (4) (see photos)
8. Connect the **blue cable** to the input pin „12” on the Arduino Yun (1) and to the input pin „D2” on the PCB (4) (see photos)
9. Connect the **green cable** to the input pin „11” on the Arduino Yun (1) and to the input pin „D3” on the PCB (4) (see photos)

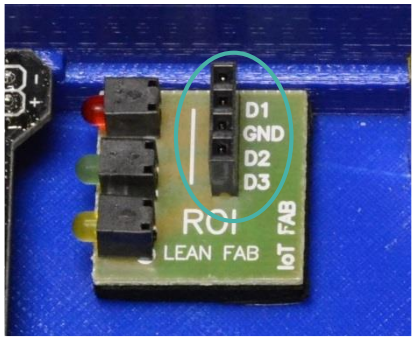
CONNECTION SCHEME



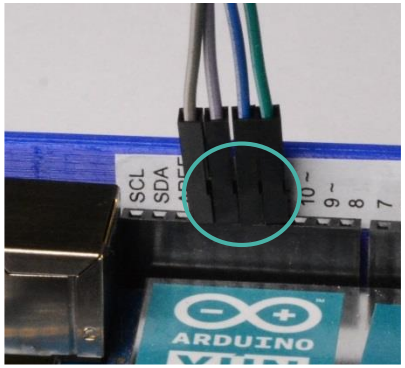
Time required: 4 minutes

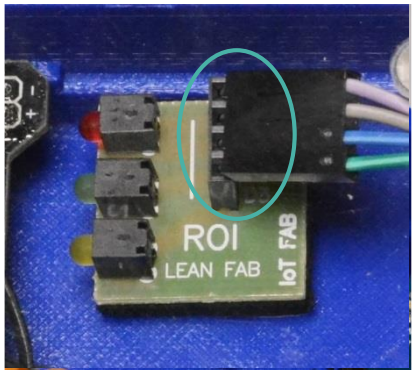
BEFORE



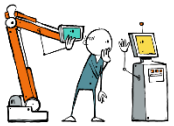


AFTER





Agenda



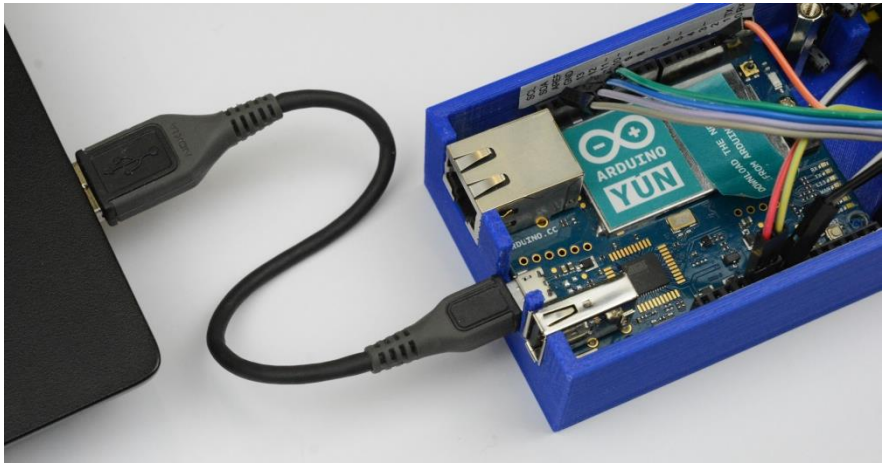
1. Introduction to IoT
2. Prerequisites for Event Box Assembly
3. Event Box Assembly Procedure
4. **Connection of Arduino Yun to WiFi Network**
5. Arduino IDE Download and Operation

Connection of Arduino Yun to WiFi Network

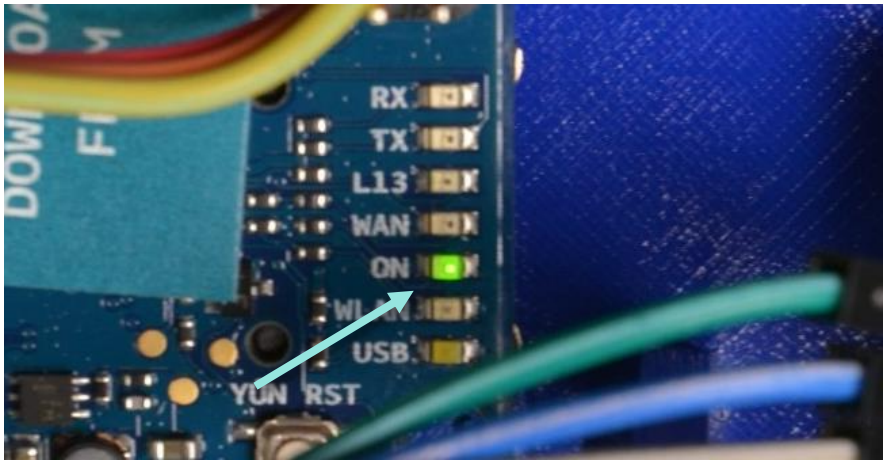
C) Required Steps (1)

- Connect the **cable miniUSB (6)** to the Arduino Yun and to your PC (photo I.),
- **Green diode** lights up as a proof of correct connection (photo II., see the arrow)

I.



II.



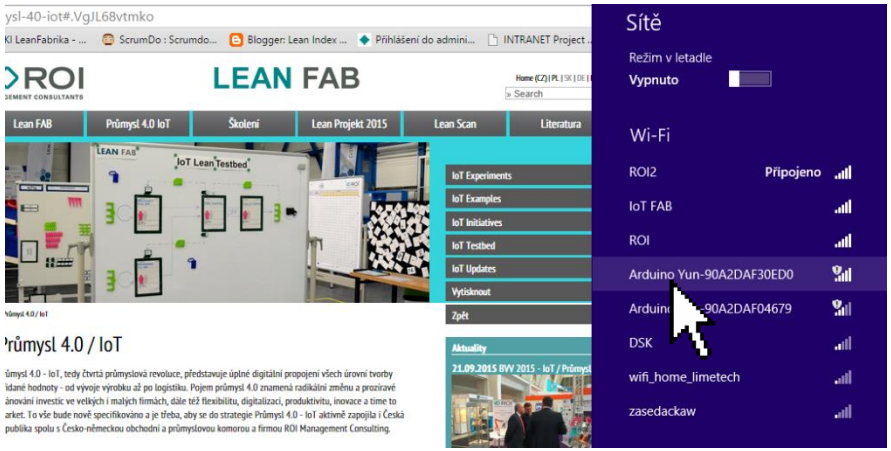
Time required: 1 minute

Connection of Arduino Yun to WiFi Network

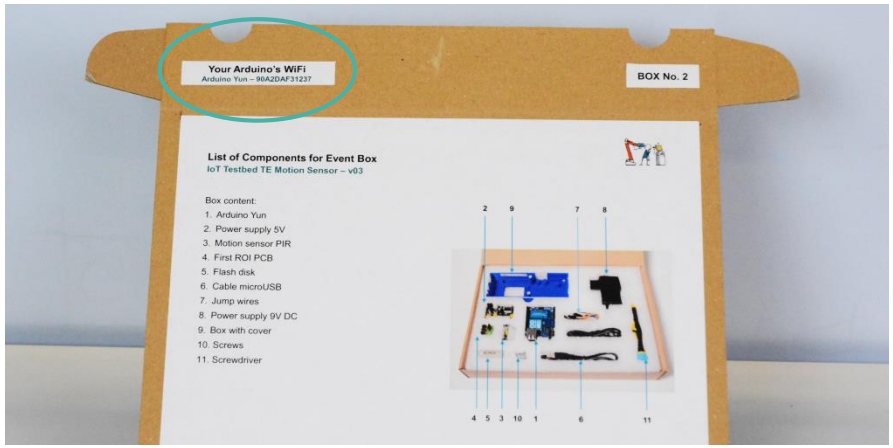
C) Required Steps (2)

- For the change of the WiFi settings it is needed to connect to the Arduino´s WiFi. On your PC select *WiFi Settings* and then choose the network of your Arduino (see screenshot I.; for the WiFi name see the Box cover – foto II.)
- There might be a minor time lapse between connecting the Arduino to the PC and the visibility of the Arduino´s WiFi, to which we connect

I.



II.



Time required: 3 minutes

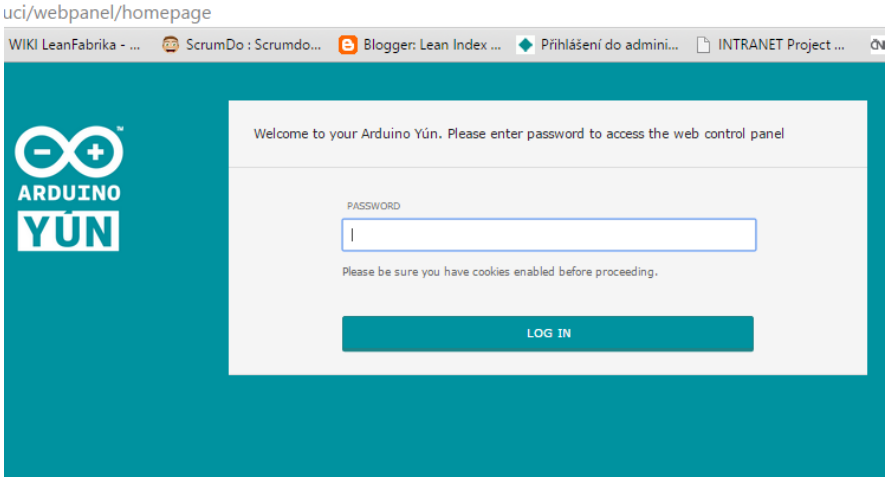
Connection of Arduino Yun to WiFi Network

C) Required Steps (3)

- After this step, type in the Arduino´s address into the internet browser **http://192.168.240.1** and log in (screenshot I.)
- **Password: arduino** (screenshot II.)

I.

II.



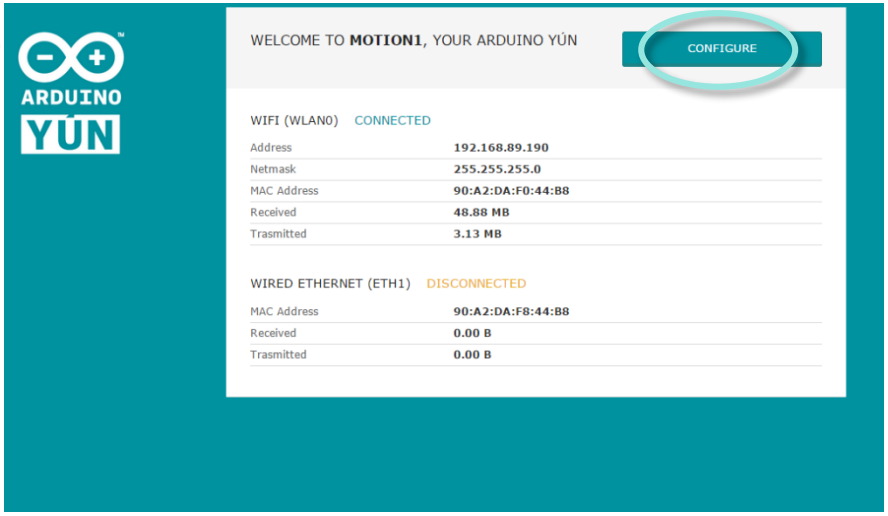
Time required: 2 minutes

Connection of Arduino Yun to WiFi Network

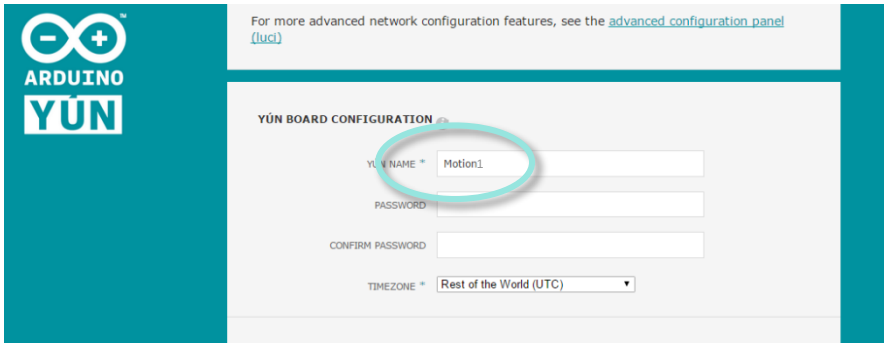
C) Required Steps (4)

- After successful log-in click on button *Configure*, which would enable you to configure the Arduino (see screenshot I.)
- Change the name of the Arduino to **Motion**Number (number of your Box, which is written on the Box cover – for example **Motion1**) – see screenshot II.
- Leave empty rest of the boxes in the formular

I.



II.

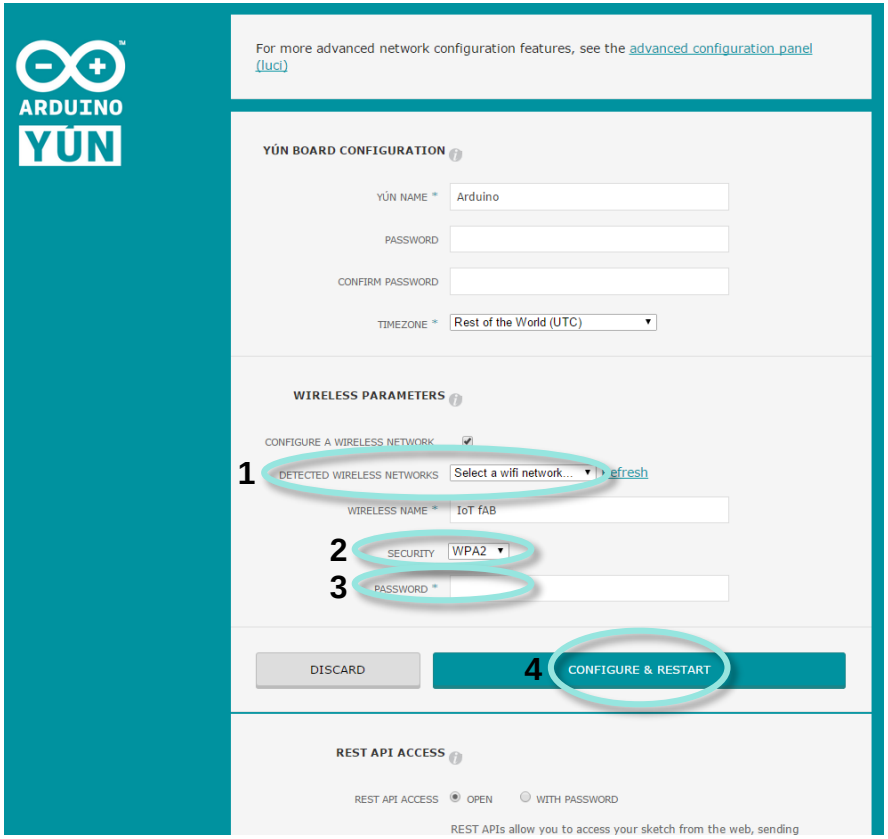


Time required: 2 minutes

Connection of Arduino Yun to WiFi Network

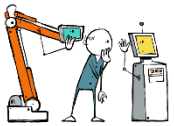
C) Required Steps (5)

- For WiFi setting scroll to *Wireless Parametres* and then choose *Detected Wireless Networks* and select your network (see screenshot)
- *Wireless Name* is filled in automatically
- In field *Security* choose *WPA2* (if needed) and in field *Password* fill-in the password of your WiFi
- Finally, click on the button *Configure & Restart*
- After this step will be the Arduino connected to the desired WiFi and the communication via the Arduino IDE will be enabled (please see chapter 5 of the agenda)



Time required: 5 minutes

Agenda

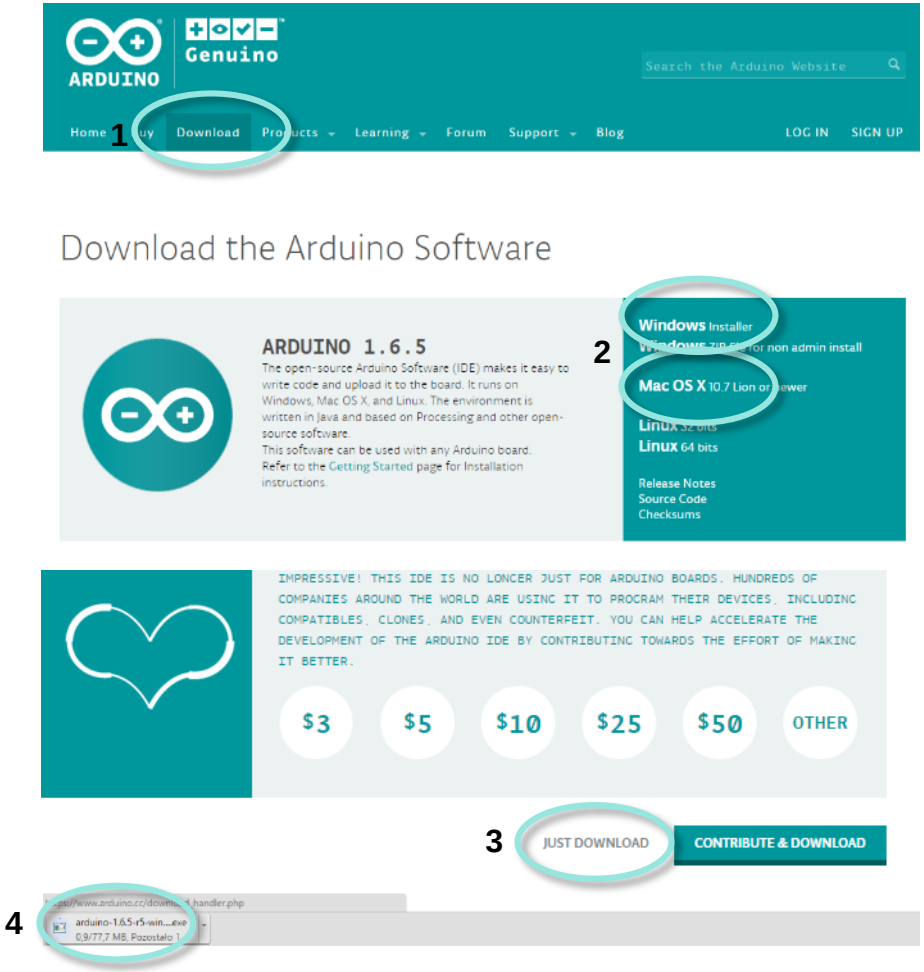


1. Introduction to IoT
2. Prerequisites for Event Box Assembly
3. Event Box Assembly Procedure
4. Connection of Arduino Yun to WiFi Network
5. **Arduino IDE Download and Operation**

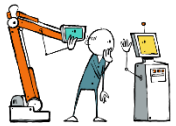
Arduino IDE Download and Operation

C) Download of Official Arduino IDE for Arduino Boards

- From the webpage www.Arduino.cc, the tab *Download,* download latest Arduino IDE for the operating system Windows (**Windows Installer**) or for operating system Mac (**Mac OS X**) – see screenshot
- Click *Just download* button
- Download progress is shown on the bottom of the page



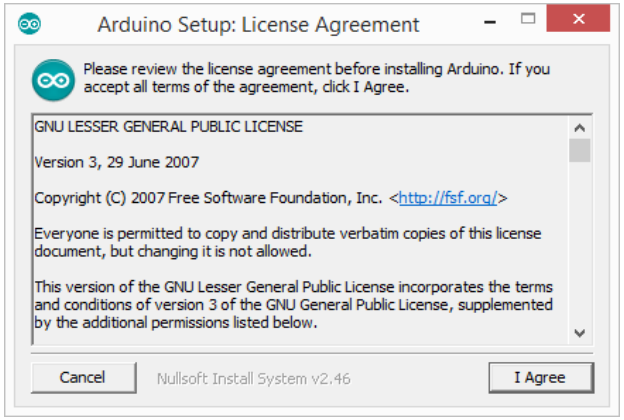
Time required: 10 minutes

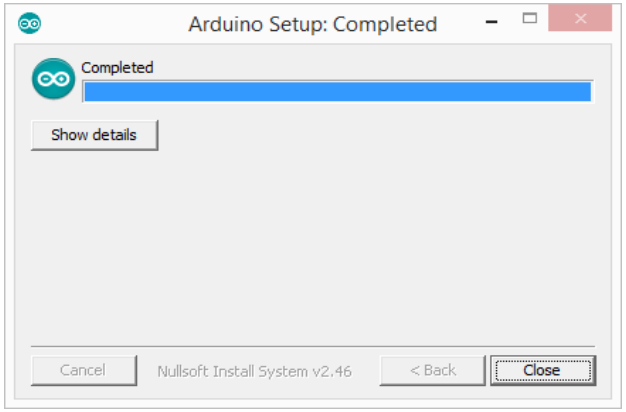


Arduino IDE Download and Operation

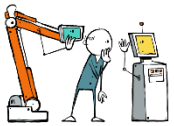
C) Installing Official Arduino IDE for Arduino Boards

- **Open** the Arduino IDE after its installation and follow the instructions
- After completion of the installation close the installation program
- From now on, the Arduino IDE is being **successfully installed** on your PC





Time required: 4 minutes



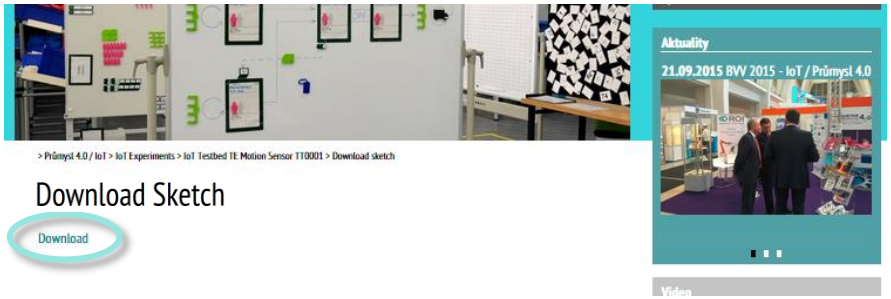
Arduino IDE Download and Operation

C) Download Lean FAB Sketch for your own Arduino Yun

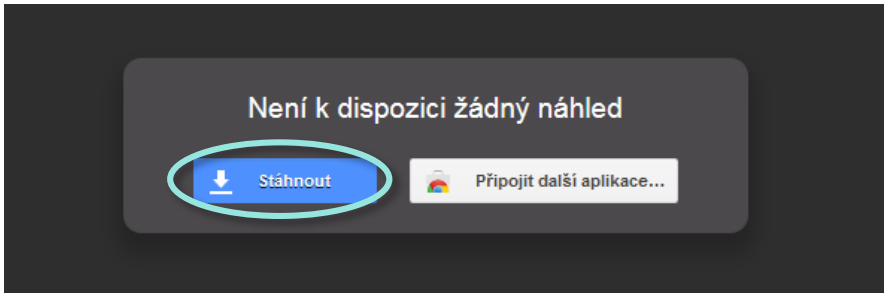
- Go to the website www.lean-fabrika.cz
- Choose tab: *Průmysl 4.0 IoT>IoT Experiments>IoT Testbed TE Motion Sensor TT001>Download Sketch* (see screenshot I.)
- Press the *Download* button (see screenshot II.) and then again on Google drive (see screenshot III.)

I.

II.



III.

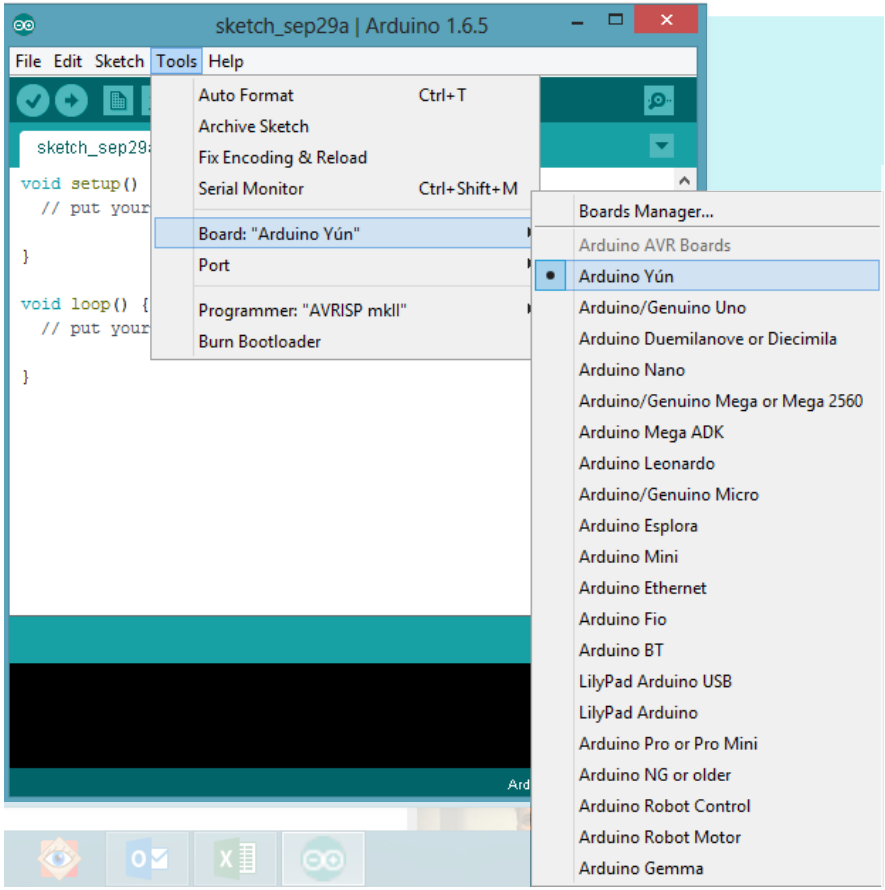


Time required: 2 minutes

Arduino IDE Download and Operation

C) Edit parameters for your own Arduino Yun

- Open the Arduino IDE
- In tab *Tools>Board* choose *Arduíno Yún*

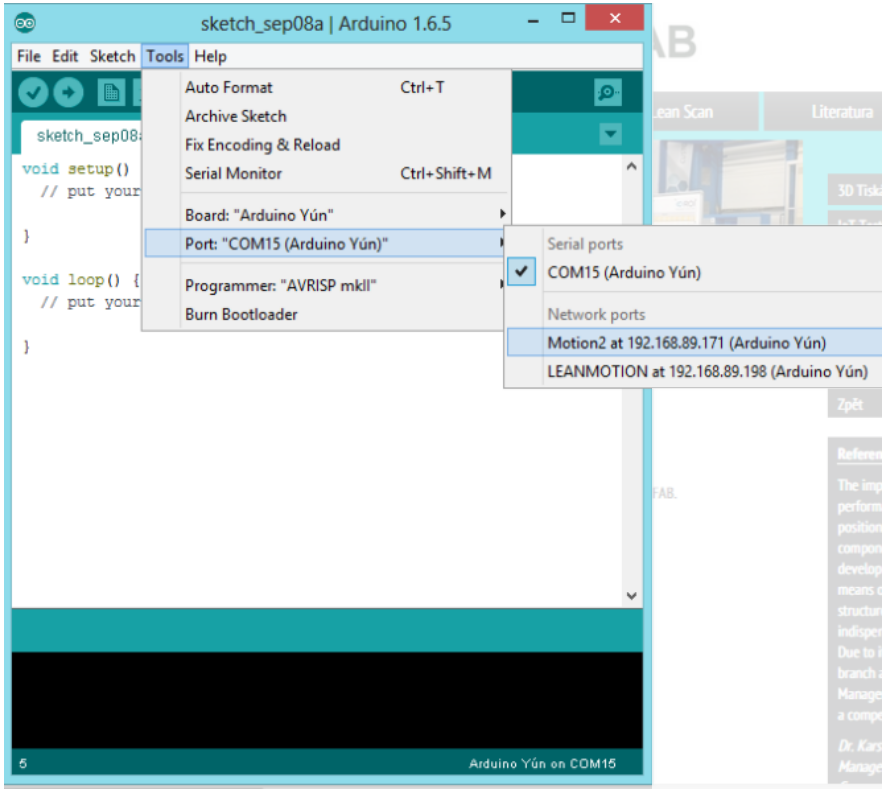


Time required: 3 minutes

Arduino IDE Download and Operation

C) Edit parameters for your own Arduino Yun (1)

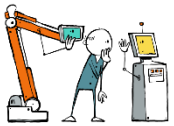
- In tab *Tools> Port* select the name of your Arduino – **Network ports: your number of motion sensor** or if you don´t see it, use option **Serial ports: COM with number**



Time required: 1 minute

Arduino IDE Download and Operation

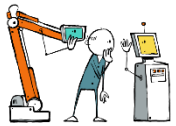
C) Edit parameters for your own Arduino Yun (2)



Sketch installation:

- Tab *File>Open…* and select previously downloaded sketch named „**ROI_Motion_sensor_v03**" from webpage www.lean-fabrika.cz
- In following pop-up window click on the *OK* button

Time required: 2 minutes



Arduino IDE Download and Operation

C) Sketch Modification (1)

- According to the **number of your Box** (printed on the Box cover):

Edit API key:

- For your API key please see next page - *Sketch Modification (2)*, table column *API key*

from "key=RWI2E3R0UAF62ZT4"

to "key=*YourAPIkey*"

(see screenshot)

Edit fieldname:

- For your fieldname please see next page - *Sketch Modification (2)*, table column *Field number*

from "&field1="

to "*YourFieldNumber*"

(see screenshot)

ROI_Motion_sensor_TT0001 | Arduino 1.6.5

File Edit Sketch Tools Help

ROI_Motion_sensor_TT0001

```
#include <Bridge.h>
#include <HttpClient.h>

String thingspeak_update_API     = "http://api.thingspeak.com/update?";
String thingspeak_write_API_key = "key=RWI2E3R0UAF62ZT4";                //EDIT YOUT API KEY
String thingspeakfieldname       = "&field1="; //                         //EDIT YOUR FIELD
int motion_1 = 2;
int tricet = 0;
int pohyb = 0;

void setup() {
  pinMode (motion_1, INPUT);
  pinMode(13, OUTPUT);
  pinMode(12, OUTPUT);
  delay(4000);
  Bridge.begin();
  Serial.begin(9600);
  digitalWrite(12, HIGH);
}

void loop () {
  delay(100);
  int sensor_1 = digitalRead(motion_1);
  if (sensor_1 == HIGH) {
    pohyb++;
    digitalWrite(13, HIGH);
    delay(1900);
    digitalWrite(13, LOW);
    tricet += 19;
  }
  tricet++;
  if(tricet >= 6000)
  {
    Serial.println(pohyb);
    postToThinspeak(pohyb);
    pohyb = 0;
```

Time required: 5 minutes

Arduino IDE Download and Operation

C) Sketch Modification (2)

- Sketch modification table:

Number of Box	Api Key	Field Number
2	IYK0HH1RTQQ3E9CO	&field2=
3	IYK0HH1RTQQ3E9CO	&field3=
4	IYK0HH1RTQQ3E9CO	&field4=
5	IYK0HH1RTQQ3E9CO	&field5=
6	IYK0HH1RTQQ3E9CO	&field6=
7	IYK0HH1RTQQ3E9CO	&field7=
8	IYK0HH1RTQQ3E9CO	&field8=
9	SPPG1JPKZGVCWYXE	&field1=
10	SPPG1JPKZGVCWYXE	&field2=
11	SPPG1JPKZGVCWYXE	&field3=
12	SPPG1JPKZGVCWYXE	&field4=
13	SPPG1JPKZGVCWYXE	&field5=
14	SPPG1JPKZGVCWYXE	&field6=
15	SPPG1JPKZGVCWYXE	&field7=
16	SPPG1JPKZGVCWYXE	&field8=
17	3AO2E6LS7OMU3U1X	&field1=
18	3AO2E6LS7OMU3U1X	&field2=
19	3AO2E6LS7OMU3U1X	&field3=
20	3AO2E6LS7OMU3U1X	&field4=
21	3AO2E6LS7OMU3U1X	&field5=
22	3AO2E6LS7OMU3U1X	&field6=
23	3AO2E6LS7OMU3U1X	&field7=
24	3AO2E6LS7OMU3U1X	&field8=
25	5CQAW8WCIENX7649	&field1=
26	5CQAW8WCIENX7649	&field2=
27	5CQAW8WCIENX7649	&field3=
28	5CQAW8WCIENX7649	&field4=
29	5CQAW8WCIENX7649	&field5=
30	5CQAW8WCIENX7649	&field6=
31	5CQAW8WCIENX7649	&field7=
32	5CQAW8WCIENX7649	&field8=

Time required: 2 minutes

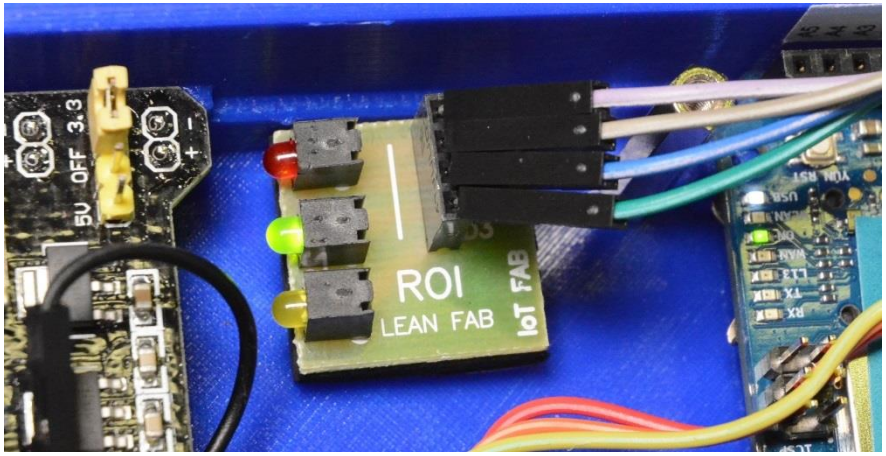
Arduino IDE Download and Operation

C) Sketch Modification (3)

- For uploading the modified sketch on the Arduino Yun click on *Upload* (see screenshot I., arrow symbol) in the left upper corner
- **Password: arduino** (if required)
- Successful Arduino IDE upload is being announced by sign *Done uploading* in the left bottom corner – **green diode on the PCB lights up** (see photo II.)
- Yellow diode on the PCB signalizes Arduino´s connection to the internet
- Red diode on the PCB lights up, when the Arduino detects motion

I.

II.

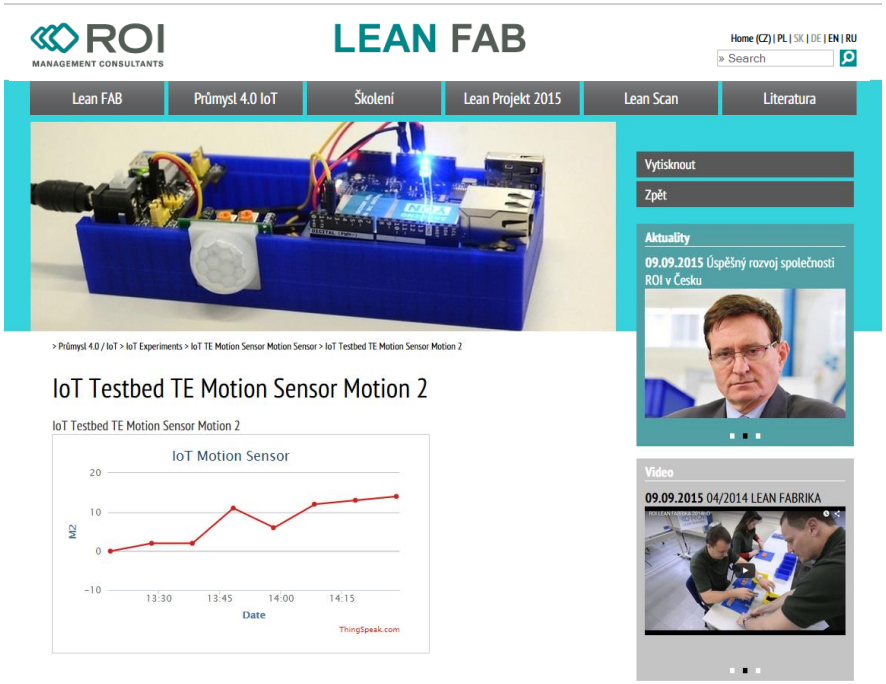


Time required: 2 minutes

Arduino IDE Download and Operation

D) Report

- After configuration and upload of the Arduino IDE, the information about movement monitored by the motion sensor will be vizualized in a graph on the Lean FAB webpage – www.lean-fabrika.cz
- Go to: *Průmysl 4.0 IoT>IoT Experiments>Motion sensors*
- And there select the **number of your motion sensor**
- Data actualization is every 2 minutes



Time required: 6 minutes

Arduino IDE Download and Operation

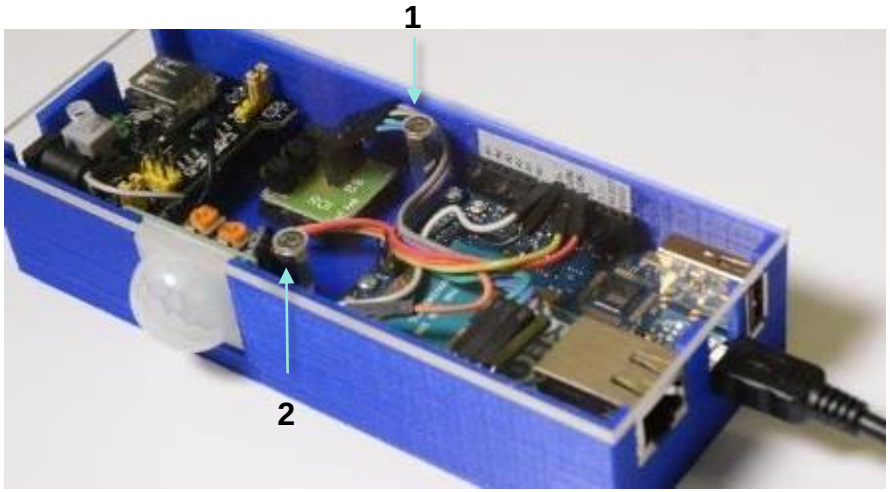
D) Cover montage

Cover montage:

- Softly **bend cables into the Box** and adjust their position (see photo I.)
- Put the **cover from plexiglass** on the Box and fix it with two big screws in the proper places (see arrow 1 and 2, photo II.)

I.

II.



Time required: 4 minutes