

Lecture #1

Introduction to IoT

Android Things 2021

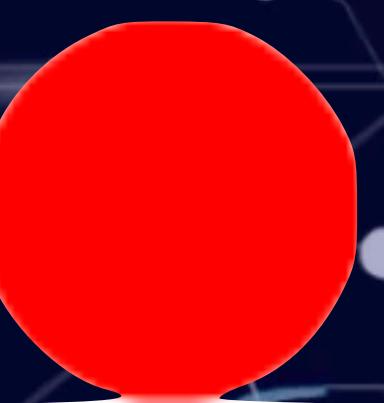
Prerequisites

- Modern programming language
- Object oriented
- Statically types
- IDE - IntelliJ/Android Studio



What you should know...

- Basics:
 - Object-oriented programming
 - Classes, methods
 - Exception handling
 - Android



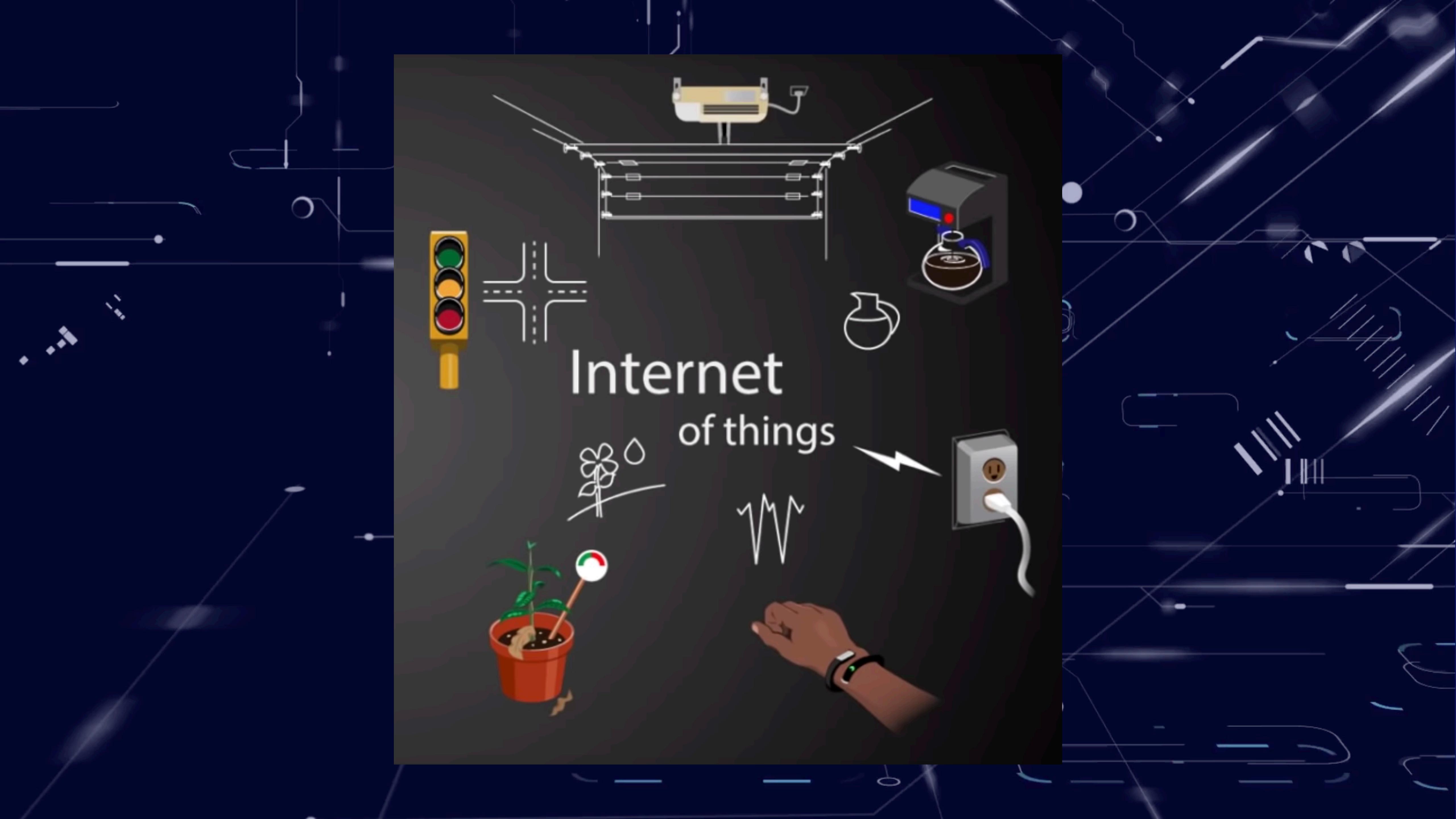
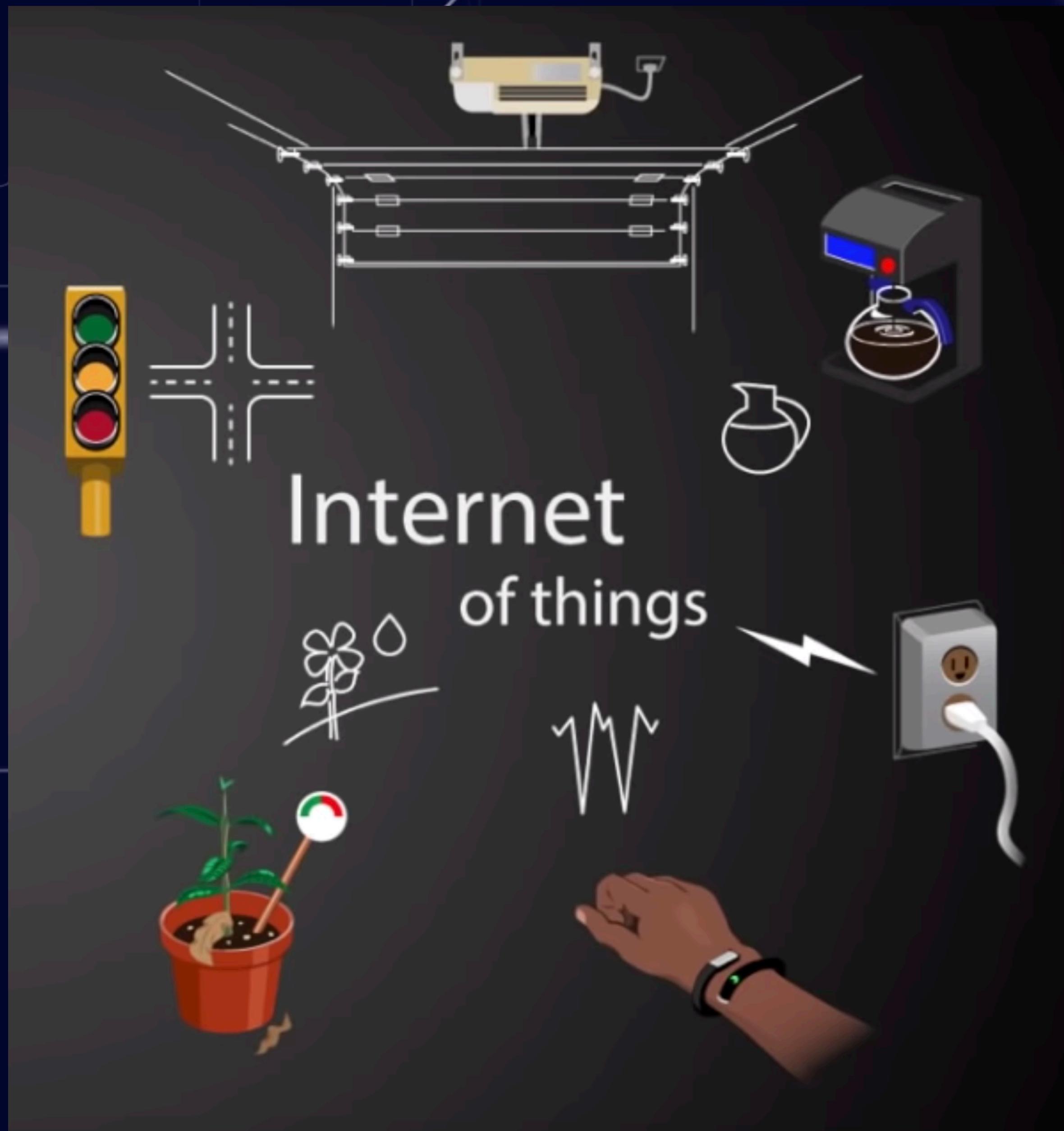
Bonus

- Functional Programming
- Lambdas
- Higher Order Functions
- Reactive Programming
- Kotlin

BONUS

Internet of people





Why

- Collect and aggregate data.
- Remote control devices.
- Automate certain tasks.



Image source: <https://newsignature.com/articles/iot-now-eric-stein-new-signature-solutions-architect/>

What

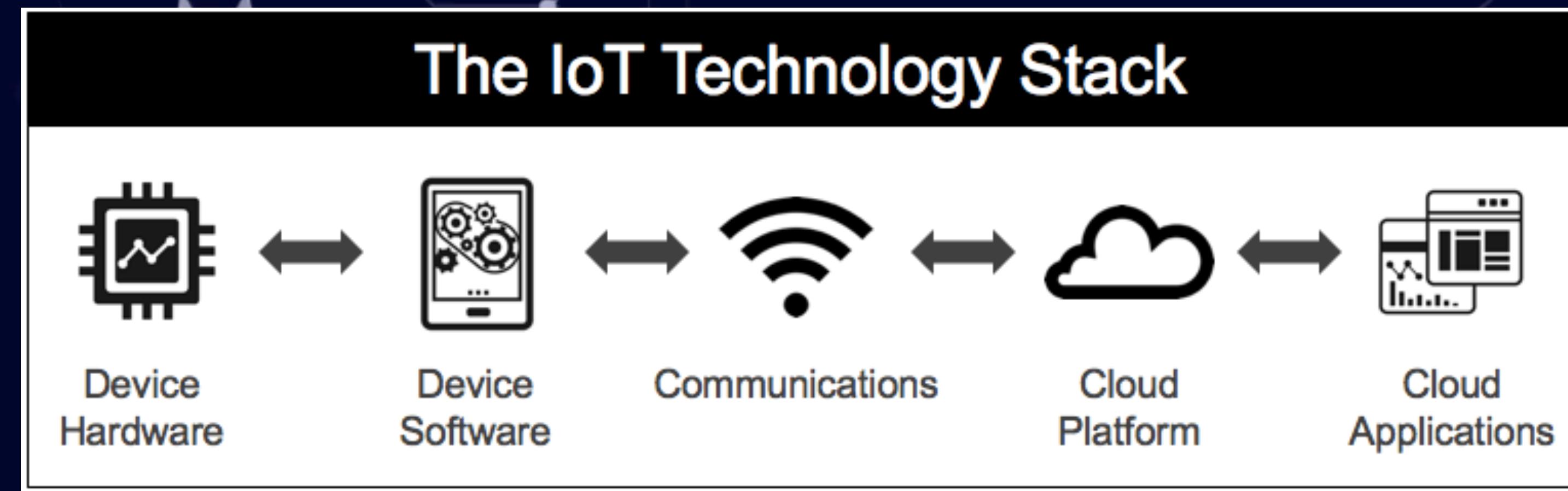
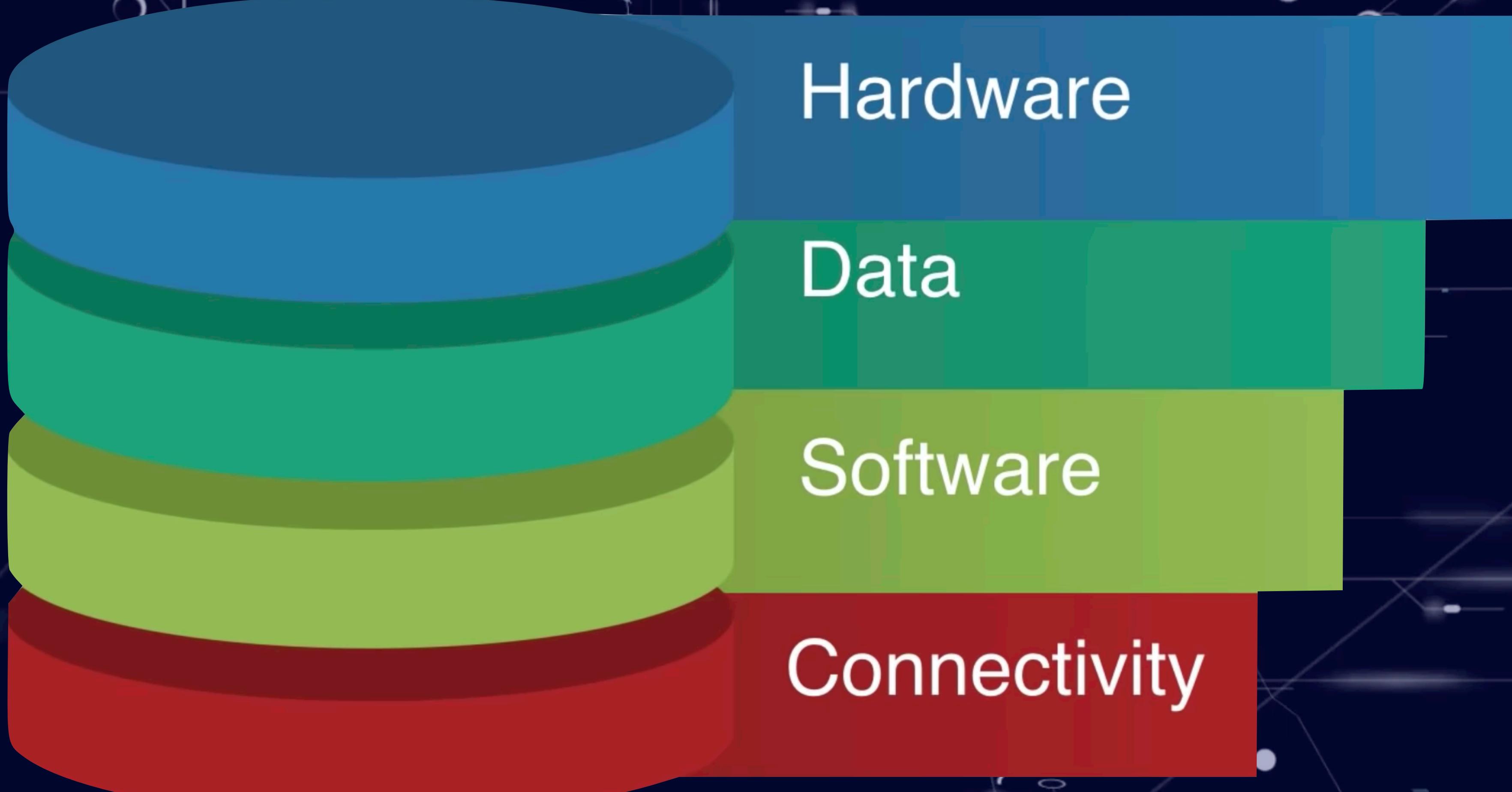
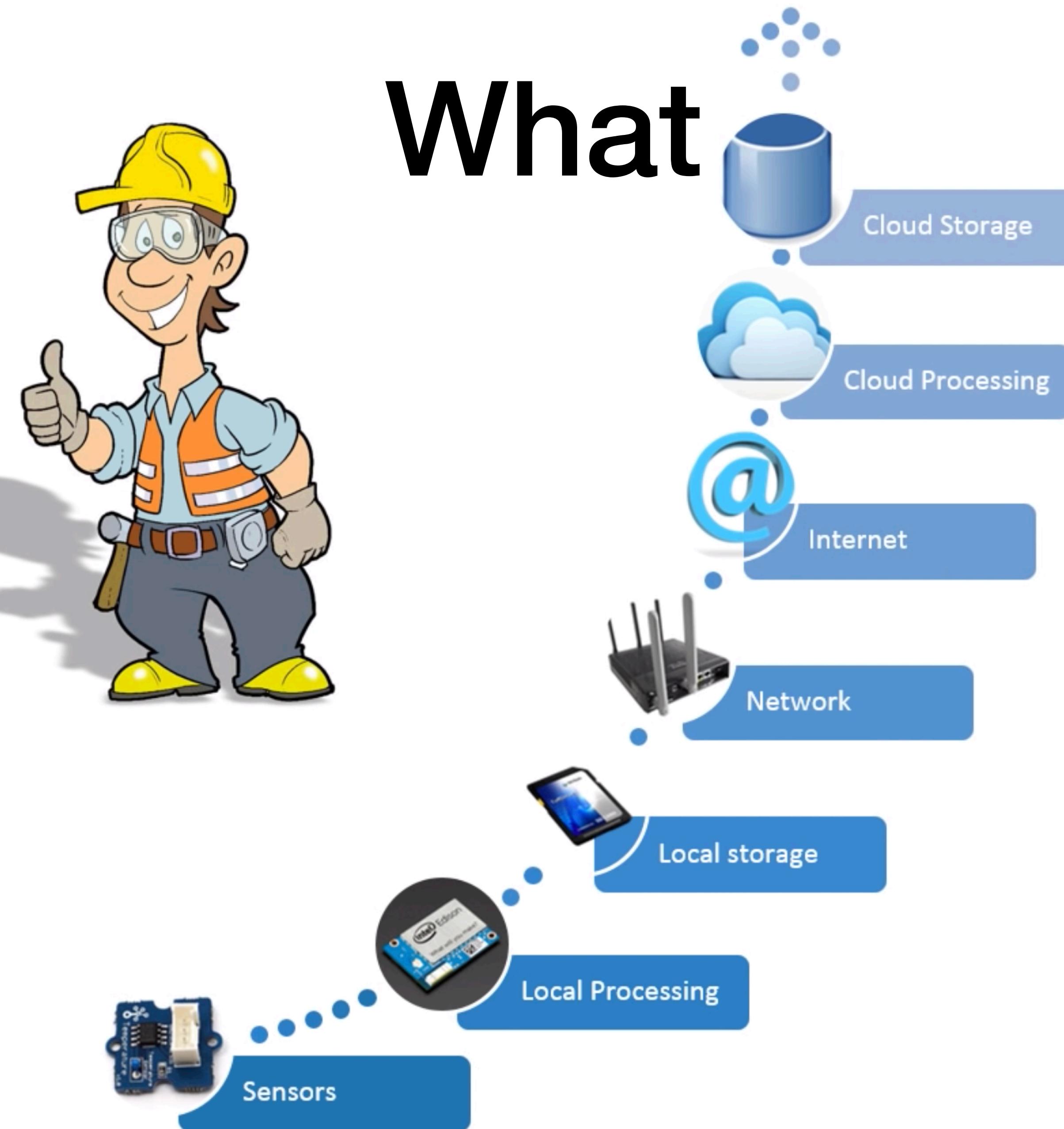


Image source: <https://www.iotforall.com/iot-product-managers-guide-iot-technology-stack/>

What



What



What

Internet of Things

00

What

Internet of Things

00

What

Internet of Things

00



Introducing
Google Home

<https://www.youtube.com/watch?v=KECfgrw02fU>

Introducing
Google Home

A Google Home smart speaker is the central focus, positioned on a light-colored wooden surface. To its right is a single red apple. In the background, a person's legs and feet are visible, wearing dark trousers and brown boots, suggesting they are walking away from the camera. The overall scene is casual and domestic.

<https://www.youtube.com/watch?v=KECfgrw02fU>



<https://www.ratp.fr/en/groupe-ratp/engineering/fully-automated-century-old-metro-lines>



<https://waymo.com/>



**"The Internet of everything will have five to 10 times
the impact on society as the Internet itself"**

Cisco CEO John Chambers



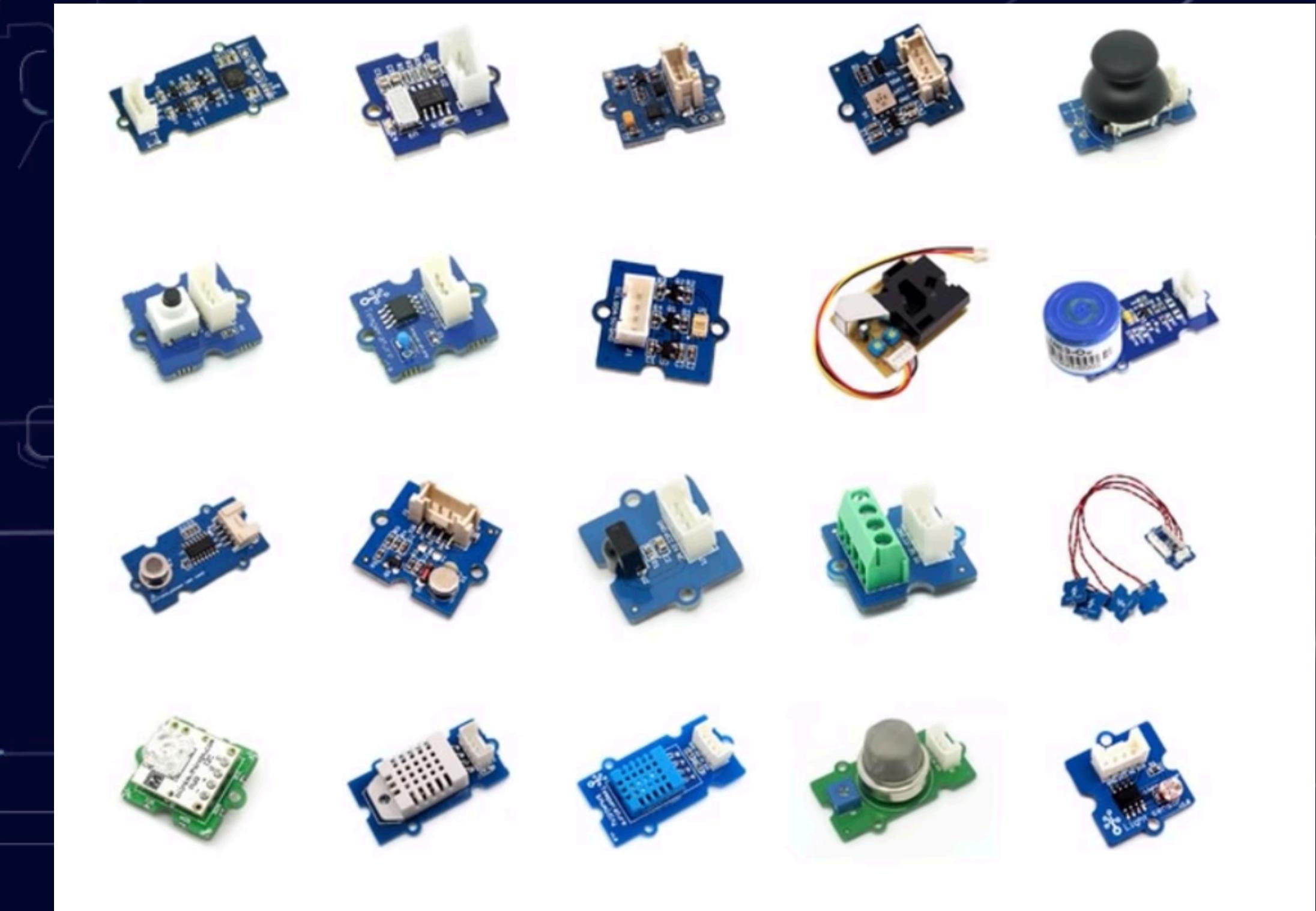
**"The Internet of everything will have five to 10 times
the impact on society as the Internet itself"**

Cisco CEO John Chambers

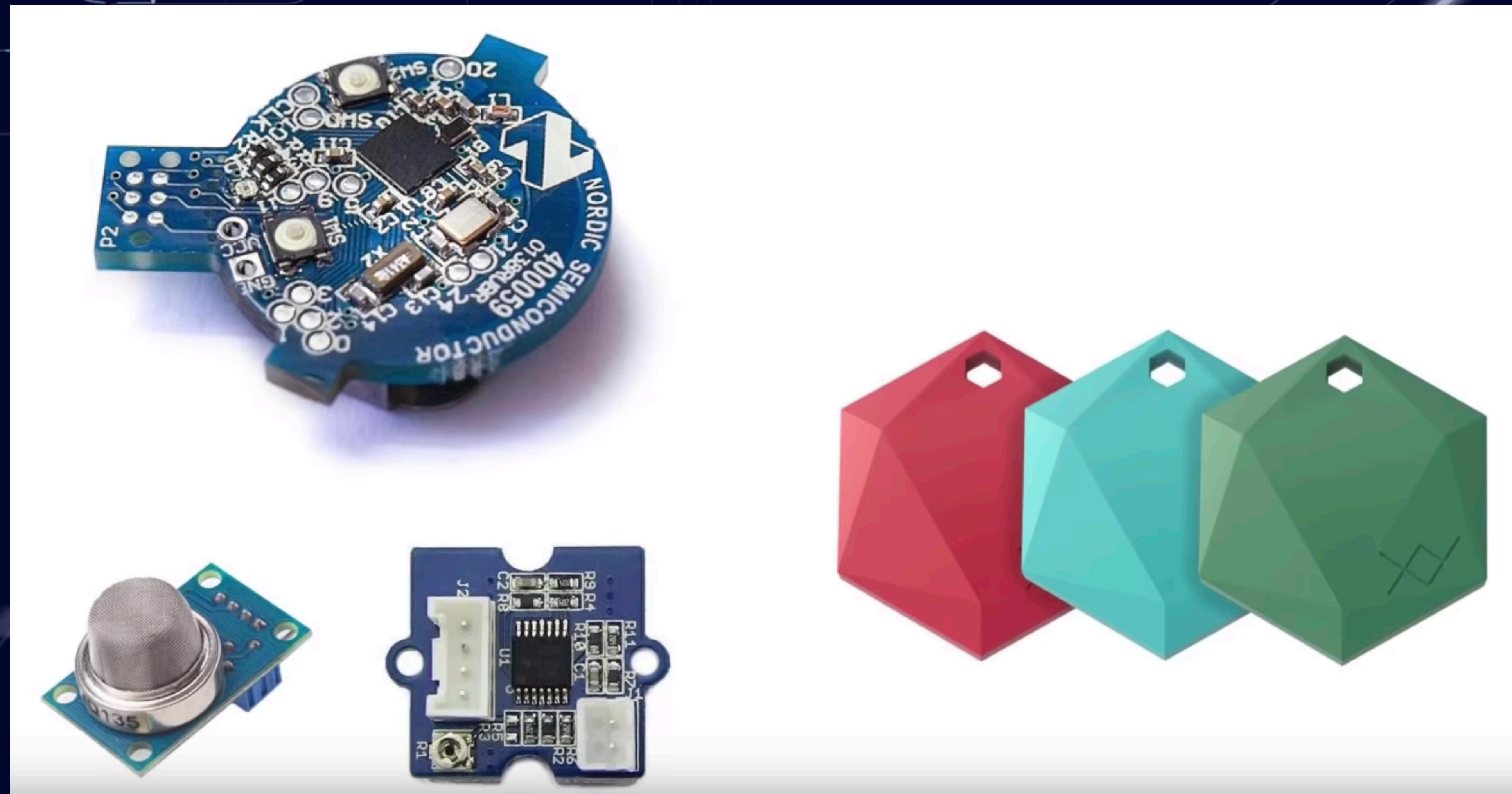
\$19 trillion in economic benefit and value over the next decade.

Sensors

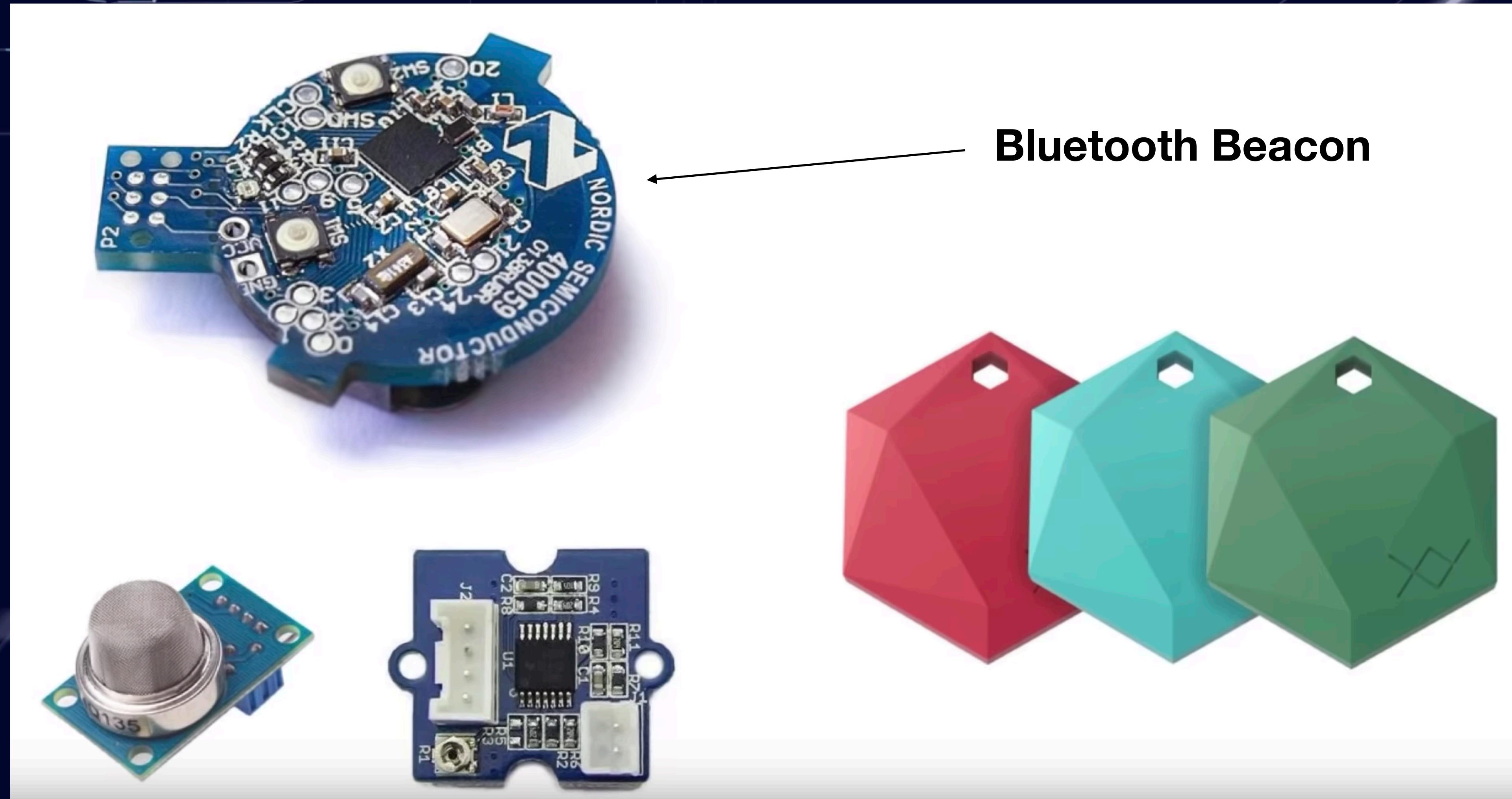
- Measure values.
- Send raw data.
- Low power.
- Almost no maintenance needed.



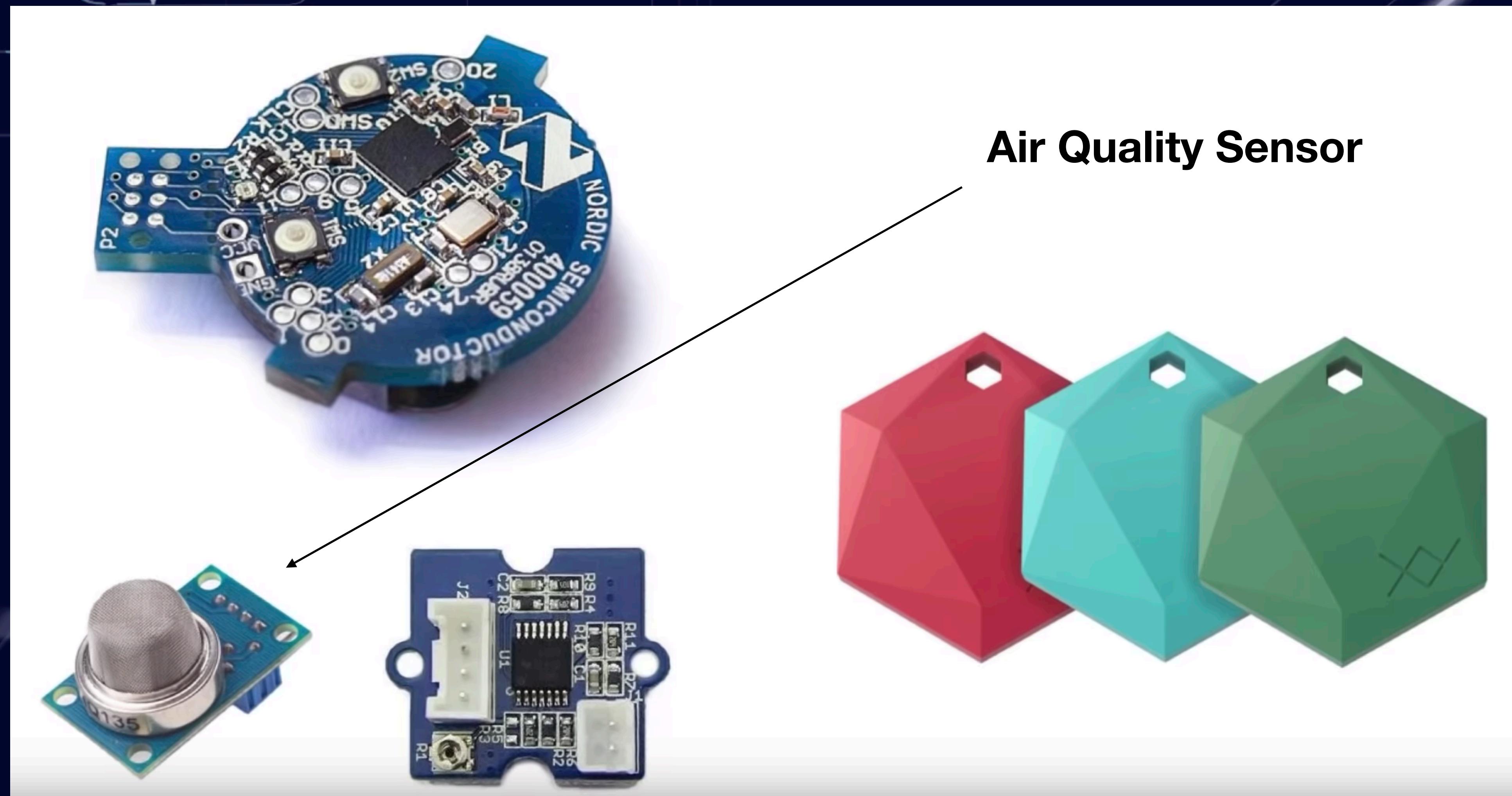
Sensors



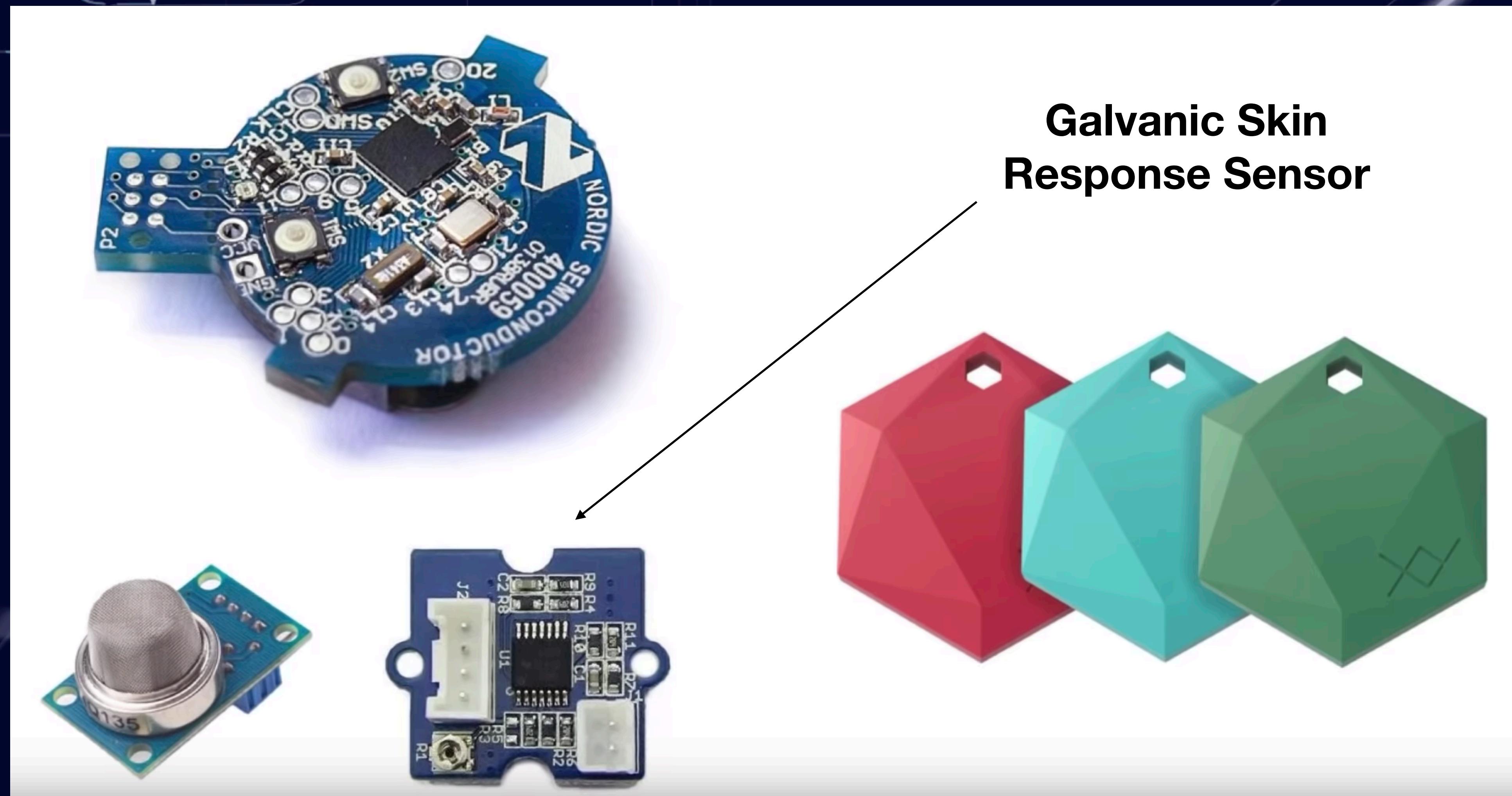
Sensors



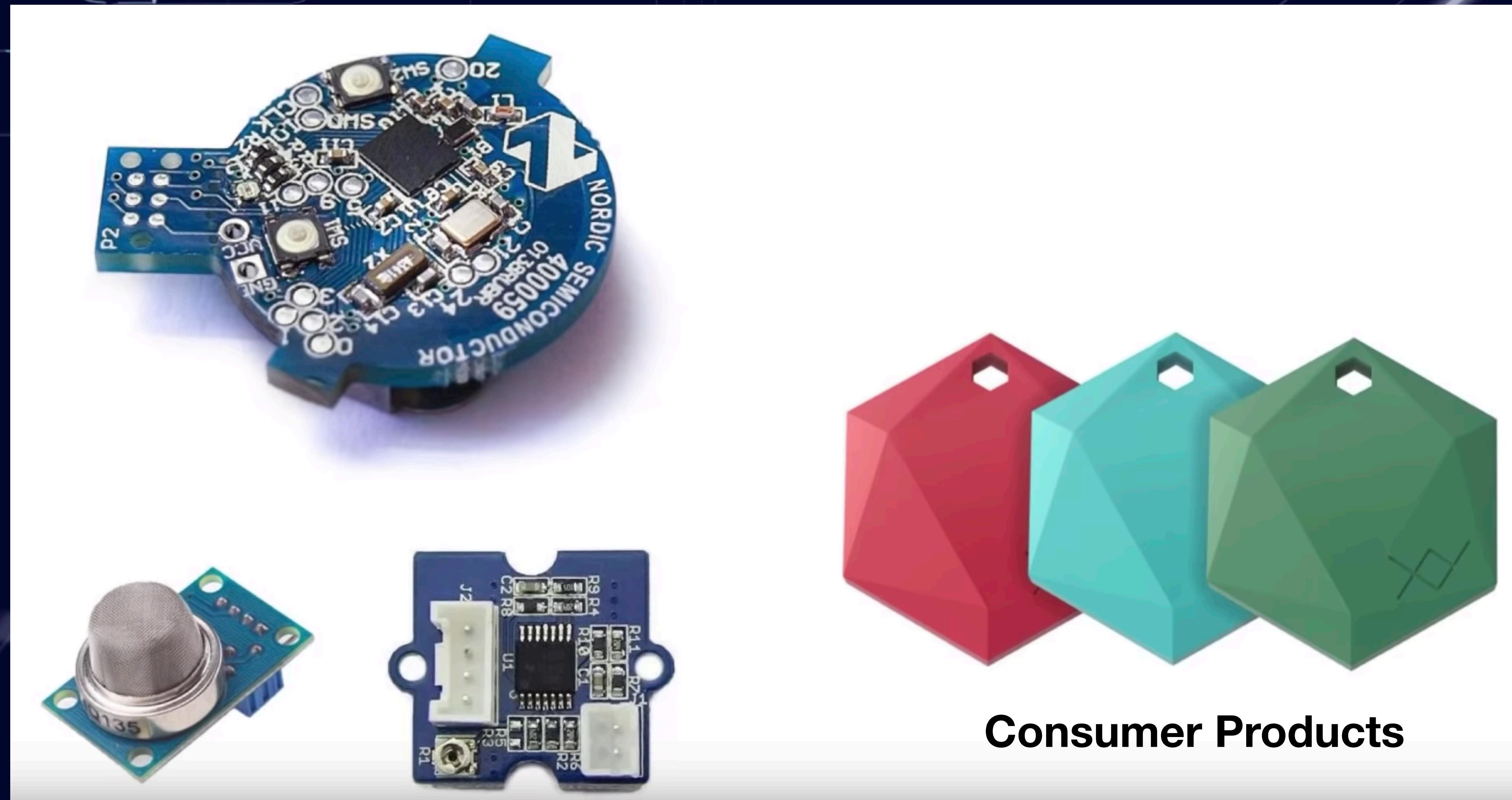
Sensors



Sensors



Sensors



Local Processing

- Collects sensor data.
- Light processing.
- Uploads data to the cloud.

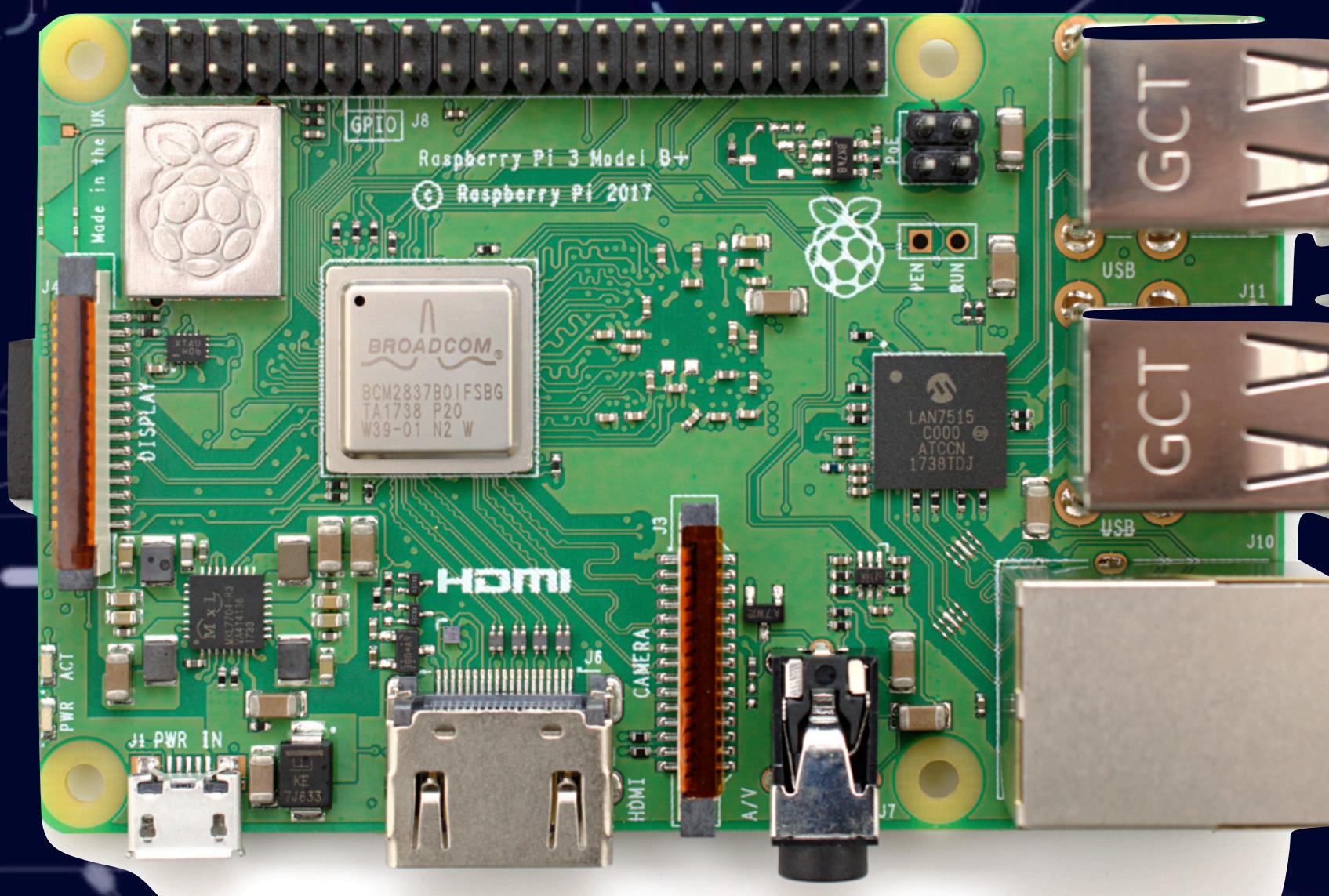


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>

Local Processing

- Collects sensor data.
- Light processing.
- Uploads data to the cloud.

Edge/Fog Computing

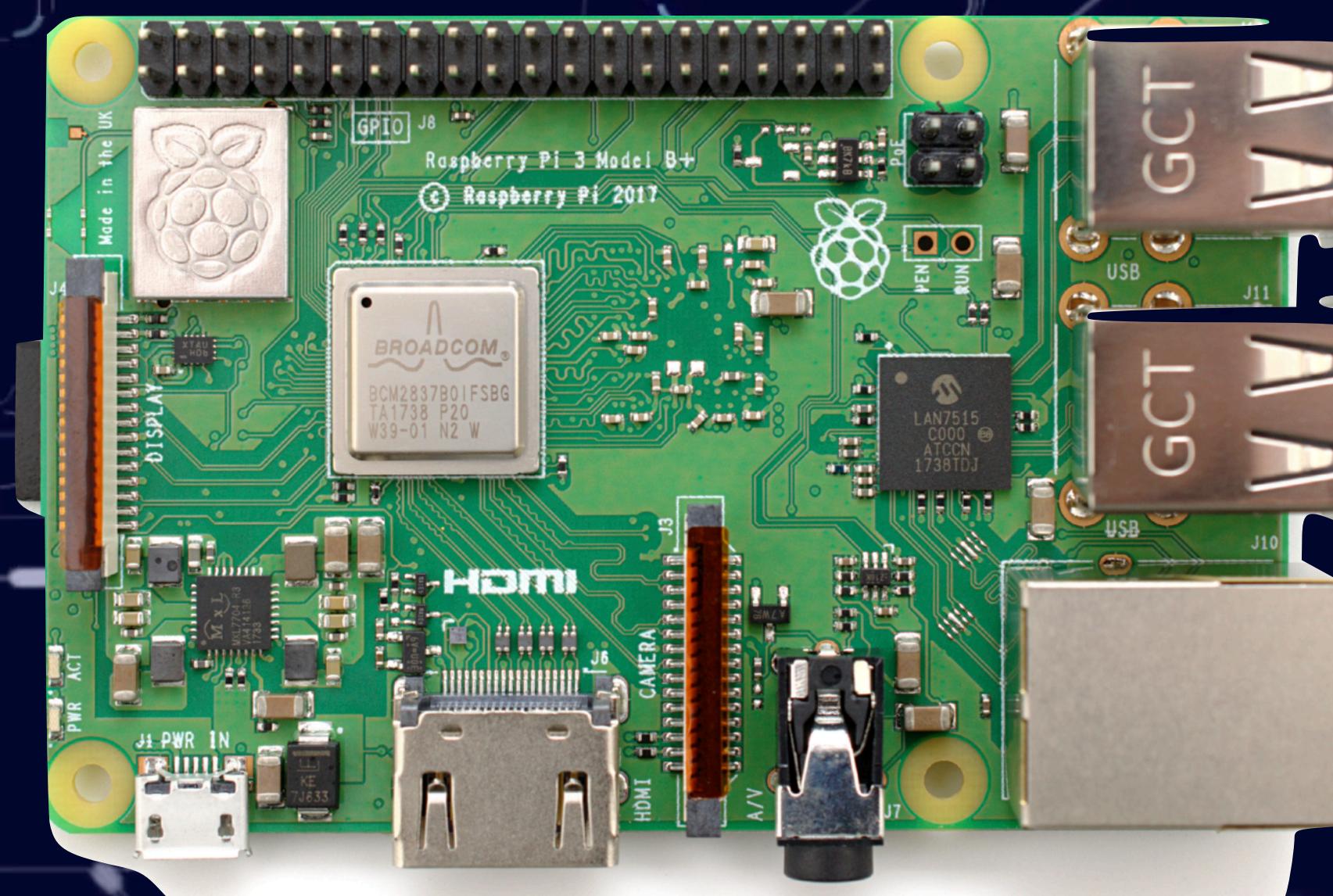


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>

Local Processing

- Collects sensor data.
- Light processing.
- Uploads data to the cloud.

Edge/Fog Computing

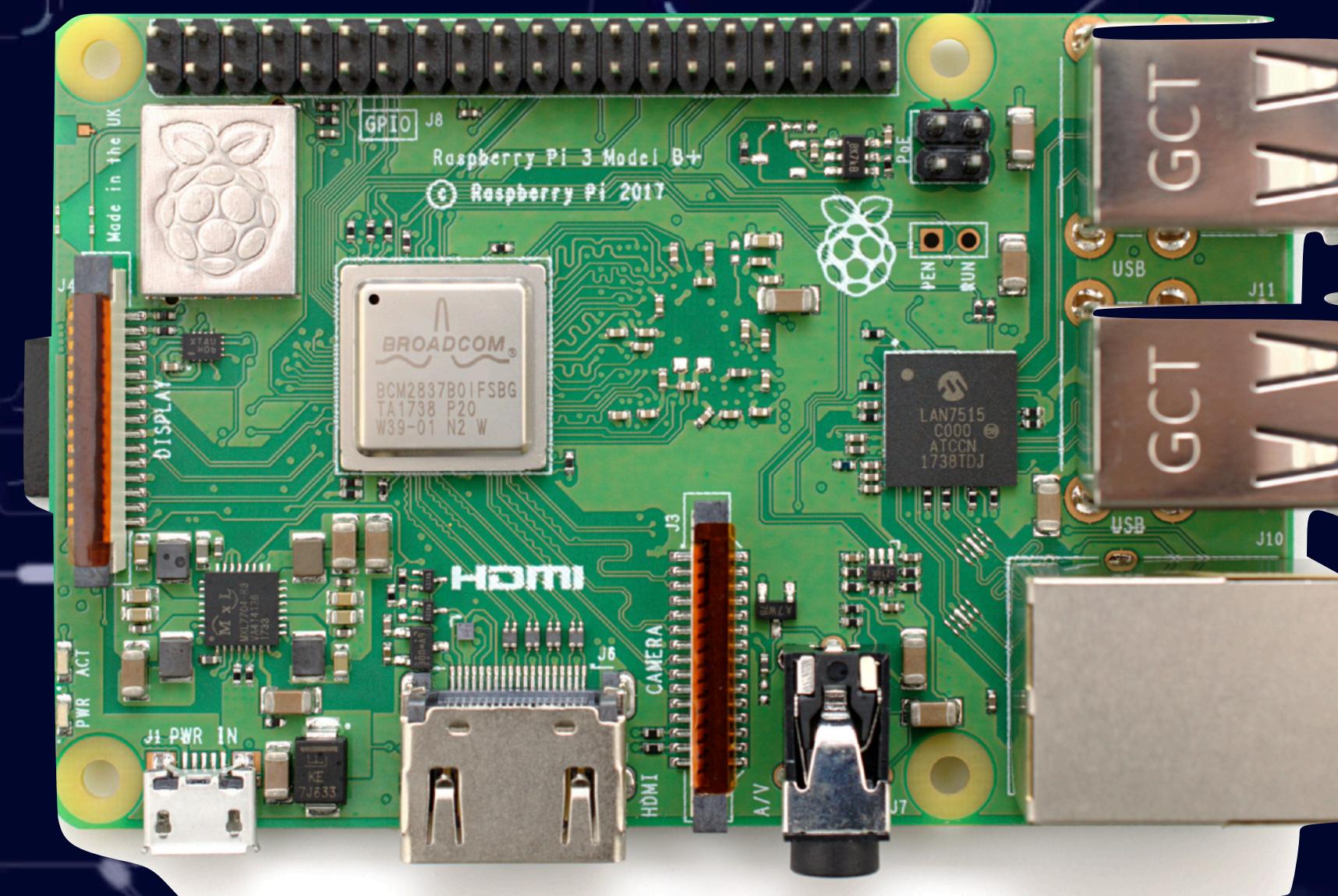
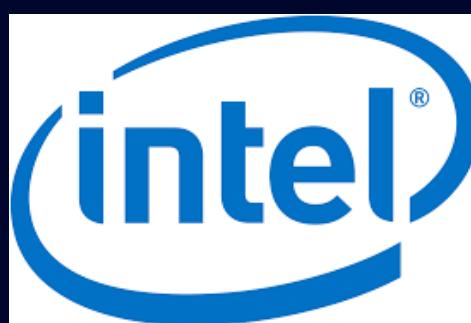


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>



Local Processing

- Collects sensor data.
- Light processing.
- Uploads data to the cloud.

Edge/Fog Computing

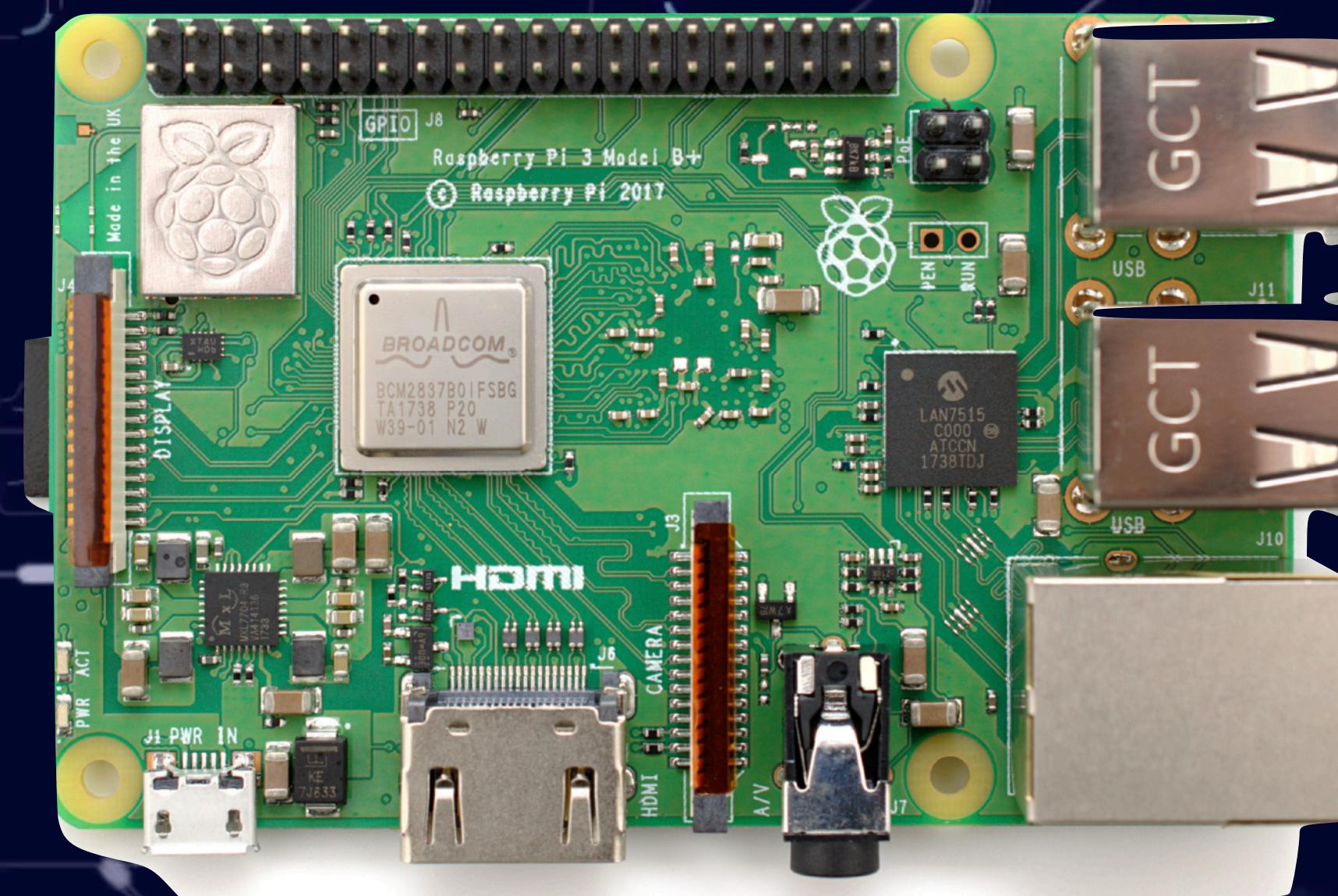
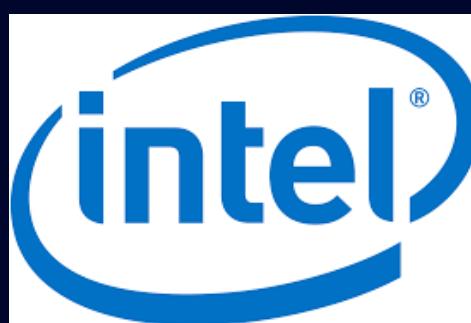


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>



Local Processing

- Collects sensor data.
- Light processing.
- Uploads data to the cloud.

Edge/Fog Computing

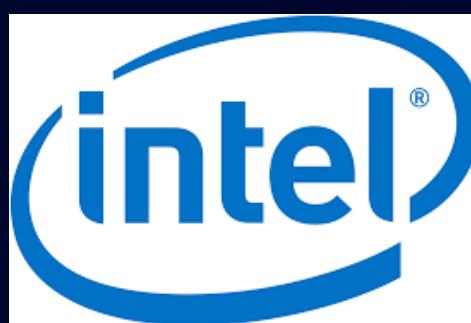


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>

Local Processing Persistence

- Collects sensor data.
- Light processing.
- Uploads data to the cloud.

Edge/Fog Computing

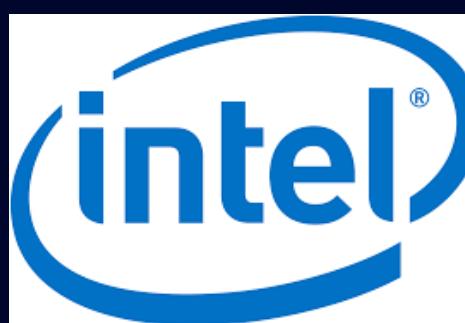


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>

Network & Internet

- IoT Gateways.
- Connects multiple sensors and local processing units.
- Protocols:
 - CoAP
 - MQTT
 - HTTP
 - XMPP



Image source: <https://commons.wikimedia.org/wiki/File:Linksys-Wireless-G-Router.jpg>

Network & Internet

- IoT Gateways.
- Connects multiple sensors and local processing units.
- Protocols:
 - CoAP <http://coap.technology/>
 - MQTT <http://mqtt.org/>
 - HTTP
 - XMPP



Image source: <https://commons.wikimedia.org/wiki/File:Linksys-Wireless-G-Router.jpg>

Network & Internet

- IoT Gateways.
- Connects multiple sensors and local processing units.
- Protocols:
 - CoAP <http://coap.technology/>
 - MQTT <http://mqtt.org/>
 - HTTP <https://www.w3.org/Protocols/>
 - XMPP



Image source: <https://commons.wikimedia.org/wiki/File:Linksys-Wireless-G-Router.jpg>

Network & Internet

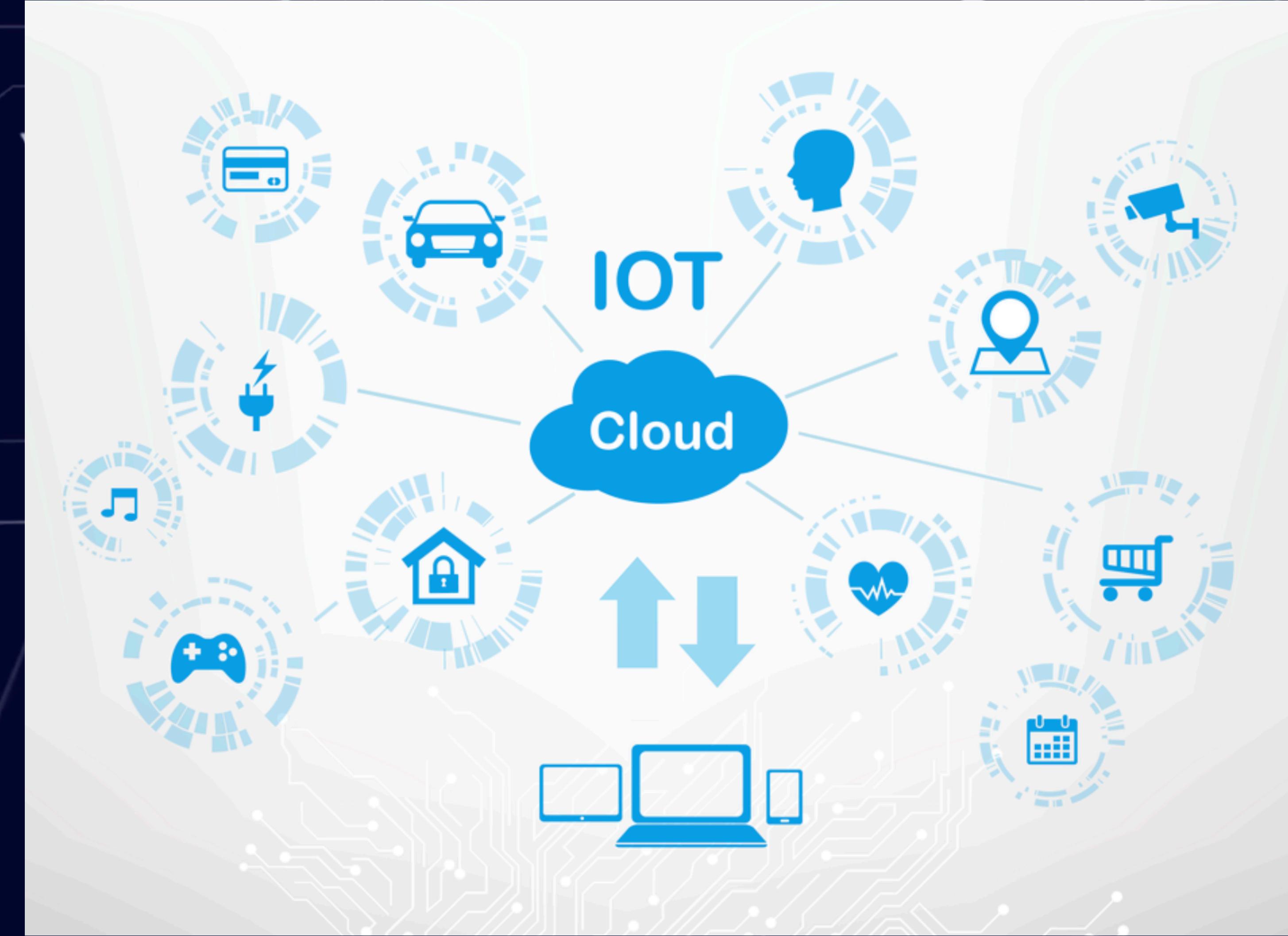
- IoT Gateways.
- Connects multiple sensors and local processing units.
- Protocols:
 - CoAP <http://coap.technology/>
 - MQTT <http://mqtt.org/>
 - HTTP <https://www.w3.org/Protocols/>
 - XMPP <https://xmpp.org>



Image source: <https://commons.wikimedia.org/wiki/File:Linksys-Wireless-G-Router.jpg>

Cloud Processing & Storage

- Aggregate.
- Store.
- Analyze.
- Predict.



How it started

- August 26th, 1997
- Bruce Perens
- <https://lists.debian.org/debian-announce/1997/msg00026.html>

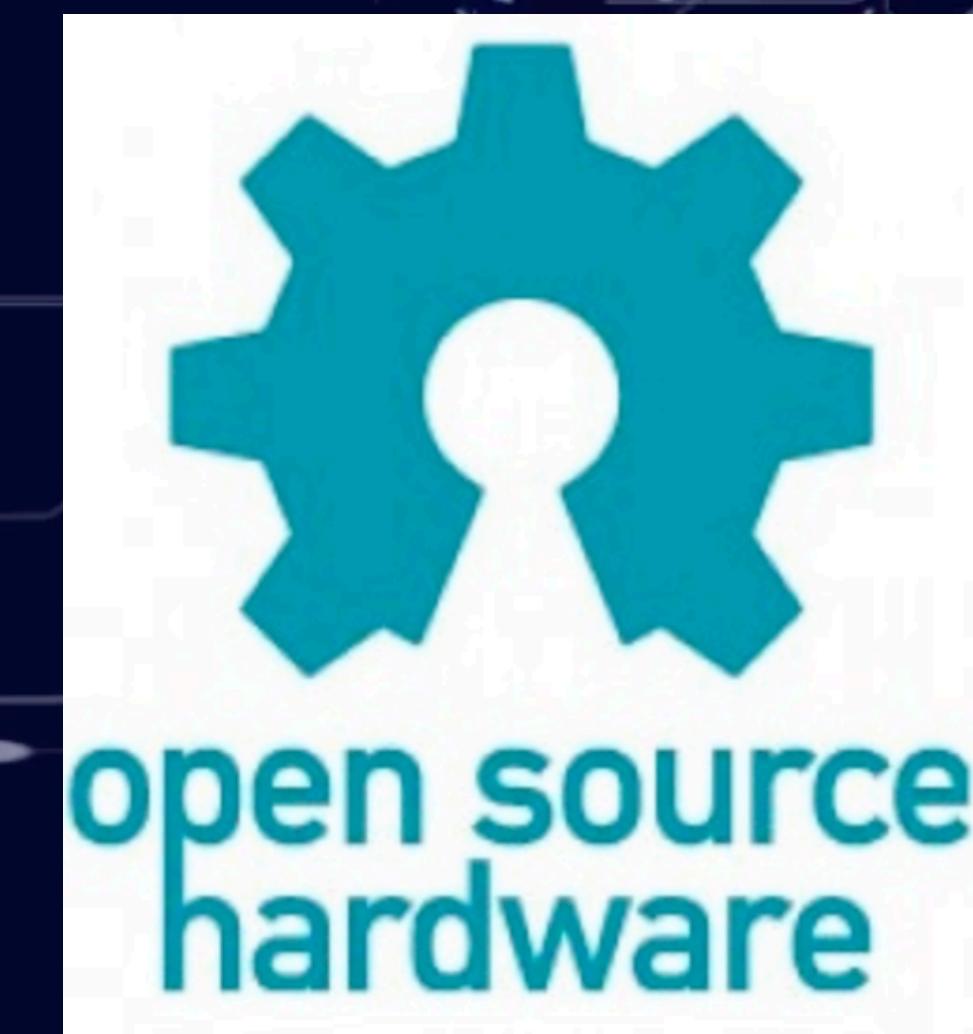
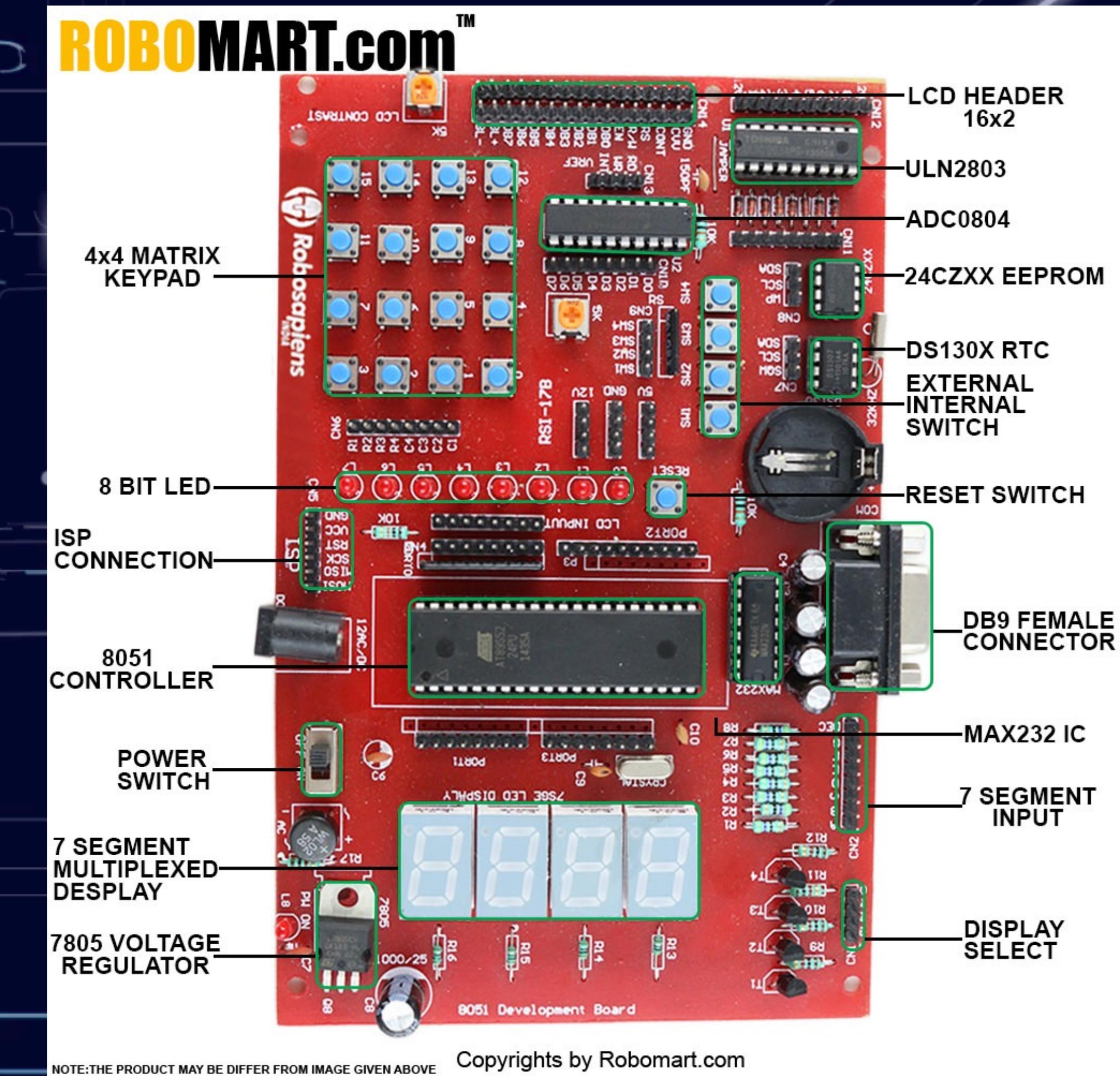


Image source: <https://www.oshwa.org/open-source-hardware-logo/>

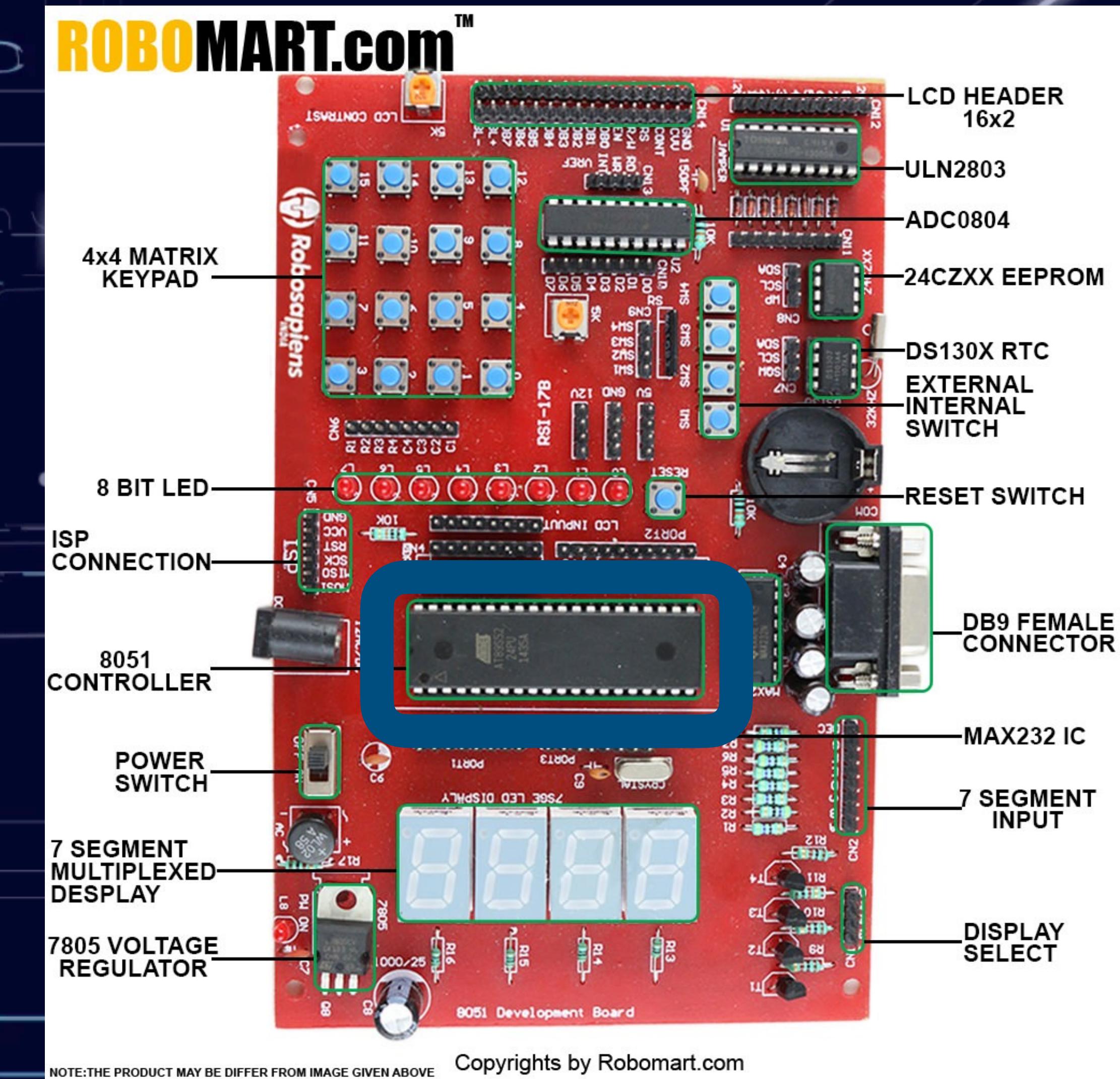
Microcontroller

- Small programmable devices.
- Easily connectable.



Microcontroller

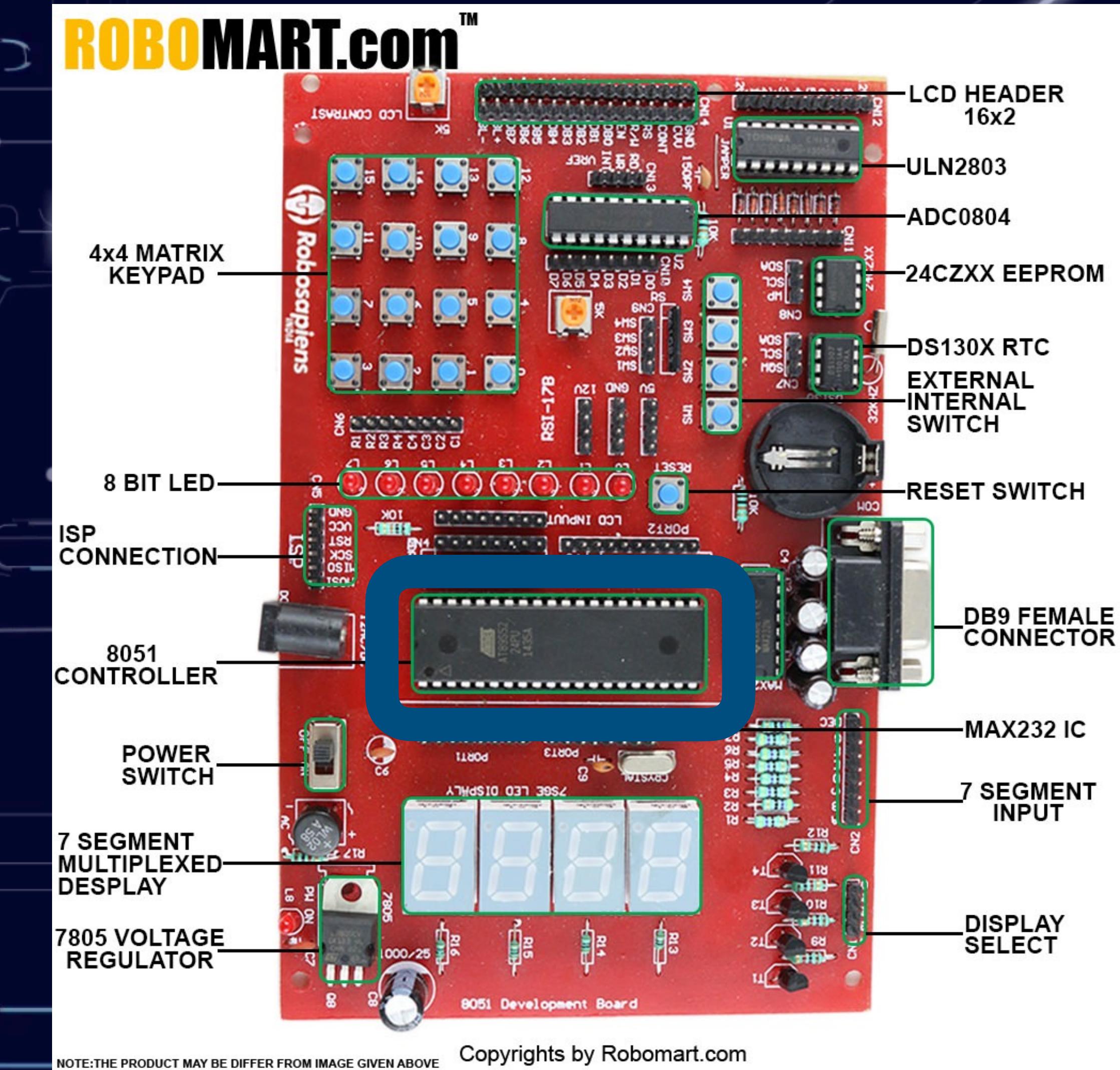
- Small programmable devices.
- Easily connectable.



1972

Microcontroller

- Small programmable devices.
- Easily connectable.



1972

Microcontroller

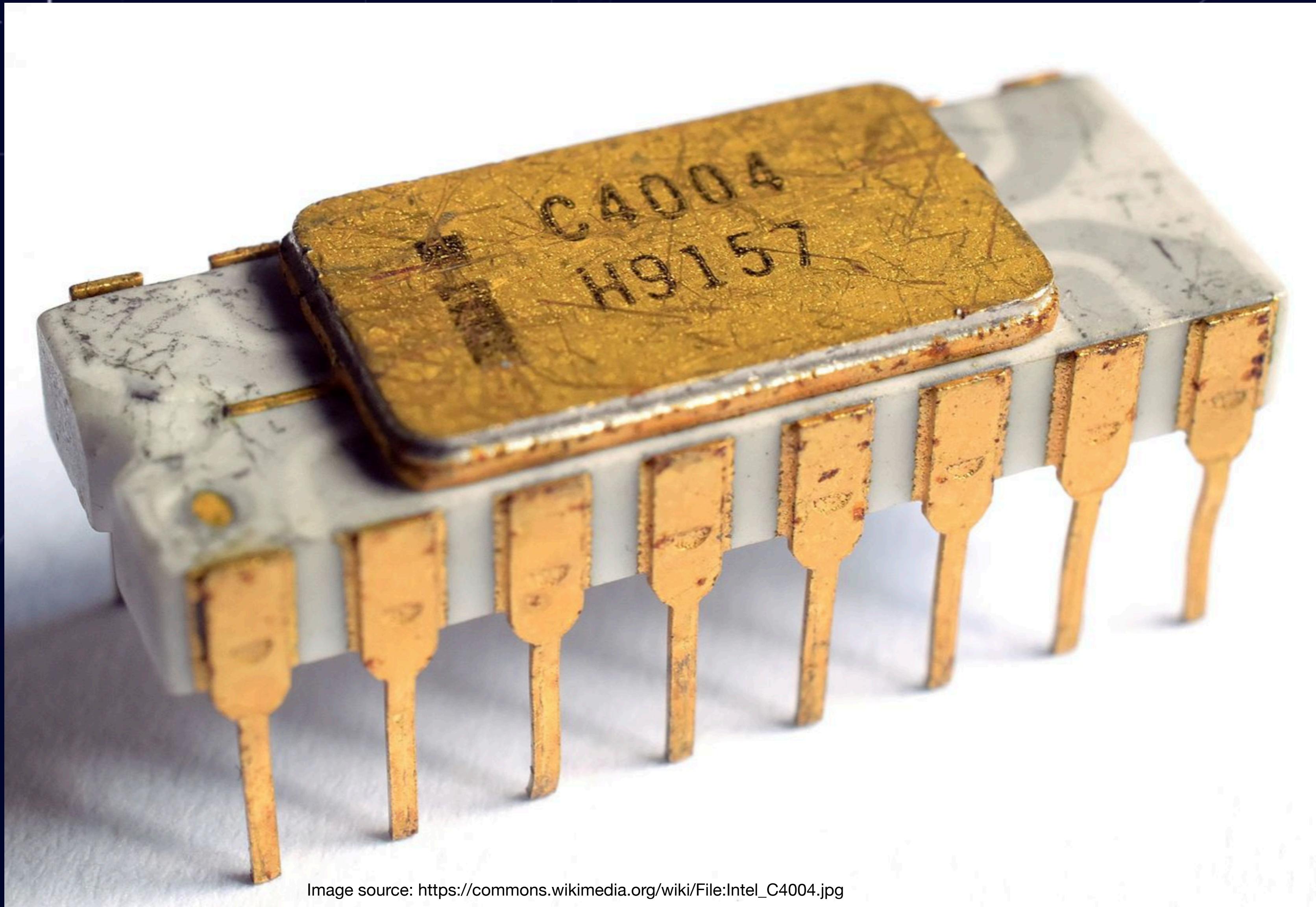


Image source: https://commons.wikimedia.org/wiki/File:Intel_C4004.jpg

1972 1982

First Internet Connected Device



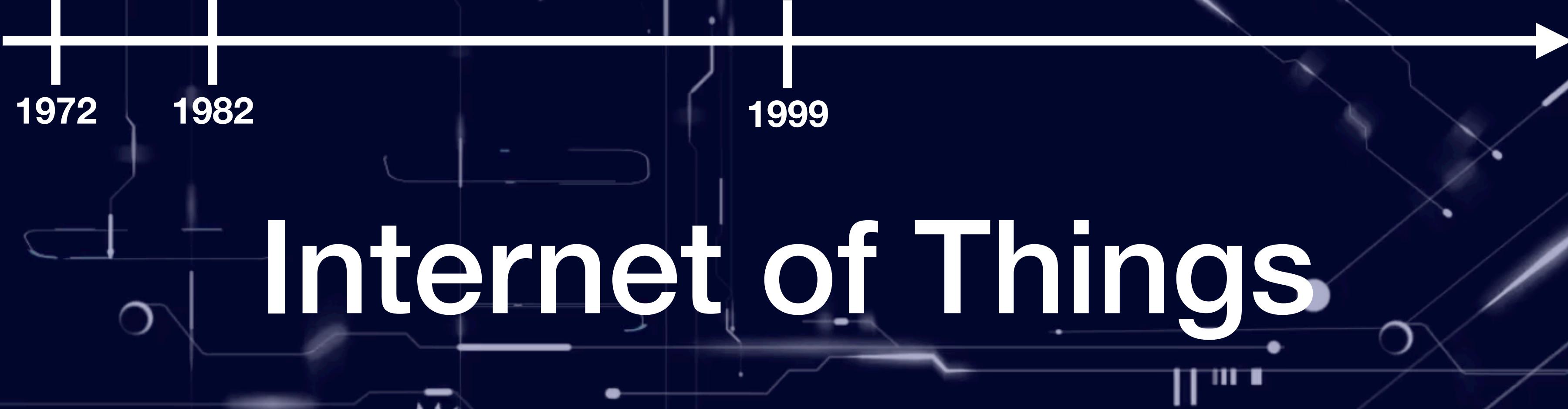
CMU's connected Coke machine

1972 1982 1999

Internet of Things



Kevin Ashton



Internet of Things

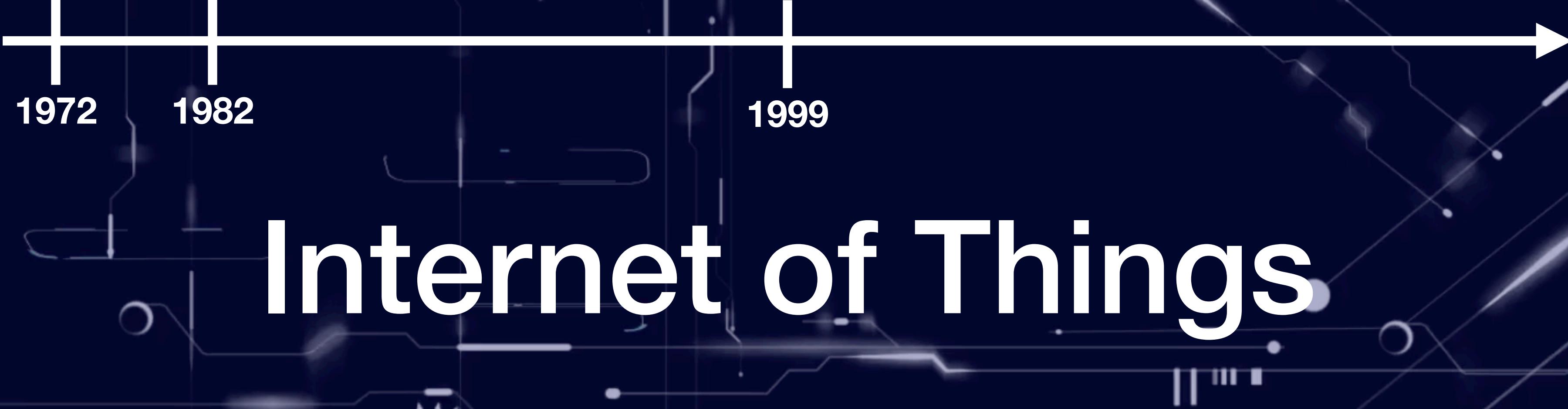
The image is a composite of several elements. On the left, the Tide logo is displayed prominently. In the center, there is a portrait of a man, identified as Kevin Ashton. To the right of the portrait, there is text for a report. At the very bottom, there is a partial view of another person's face.

Our Brands ▾ Our Impact ▾ Our Story ▾ Coupons ↗

P&G

Tide

Kevin Ashton **A FORCE FOR GOOD
A FORCE FOR GROWTH
2018 CITIZENSHIP REPORT ➔**



Internet of Things

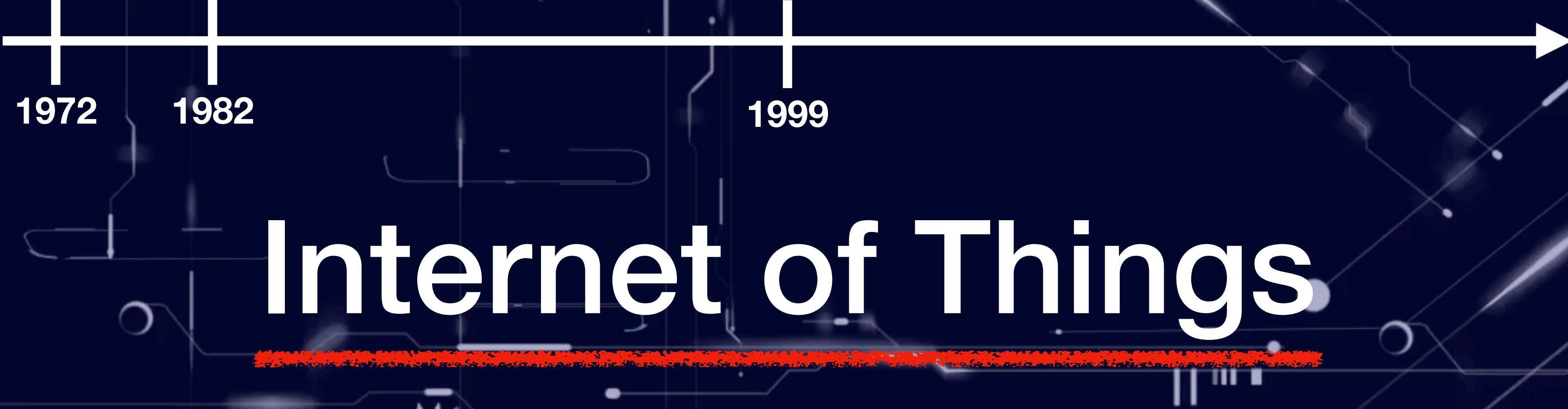
The image is a composite of several elements. On the left is the Tide laundry detergent logo, which consists of the word "Tide" in a stylized blue font inside a yellow circle with concentric red and yellow outlines. In the center is a portrait photograph of Kevin Ashton, a man with short brown hair and a serious expression, wearing a light-colored shirt. To the right of the portrait is a diagram of an RFID chip, shown as a square with concentric lines and the word "RFID" in the center. At the top of the image is a navigation bar for Procter & Gamble (P&G) with links for "Our Brands", "Our Impact", "Our Story", and "Coupons". Below the navigation bar, the text "Kevin Ashton A FORCE FOR GOOD A FORCE FOR GROWTH 2018 CITIZENSHIP REPORT" is displayed in white and yellow.

Our Brands ▾ Our Impact ▾ Our Story ▾ Coupons ↗

P&G

Tide

Kevin Ashton **A FORCE FOR GOOD
A FORCE FOR GROWTH
2018 CITIZENSHIP REPORT ➔**



Internet of Things

The image is a composite of several elements. On the left is the Tide laundry detergent logo, which consists of the word "Tide" in a stylized blue font inside a yellow circle with concentric orange and red outlines. In the center is a portrait photograph of a man with short brown hair, identified as Kevin Ashton. To the right of the portrait is a black square icon representing an RFID chip, with the word "RFID" in white text inside it. Below these elements, the text "Kevin Ashton" appears in white, followed by "A FORCE FOR GOOD" and "A FORCE FOR GROWTH" in large white capital letters, and finally "2018 CITIZENSHIP REPORT" in large yellow capital letters. At the very bottom of the image, there is a partial view of a woman's face. At the top of the image, there is a navigation bar with links: "Our Brands", "Our Impact", "Our Story", and "Coupons". The P&G logo is also present in the top right corner of the image area.

Our Brands ▾ Our Impact ▾ Our Story ▾ Coupons ↗

P&G

Tide

Kevin Ashton **A FORCE FOR GOOD
A FORCE FOR GROWTH
2018 CITIZENSHIP REPORT ➤**

1972

1982

1999

2003

Arduino

- Programmable device.
- Easily connectable.
- Open source.
- Simple to use software.

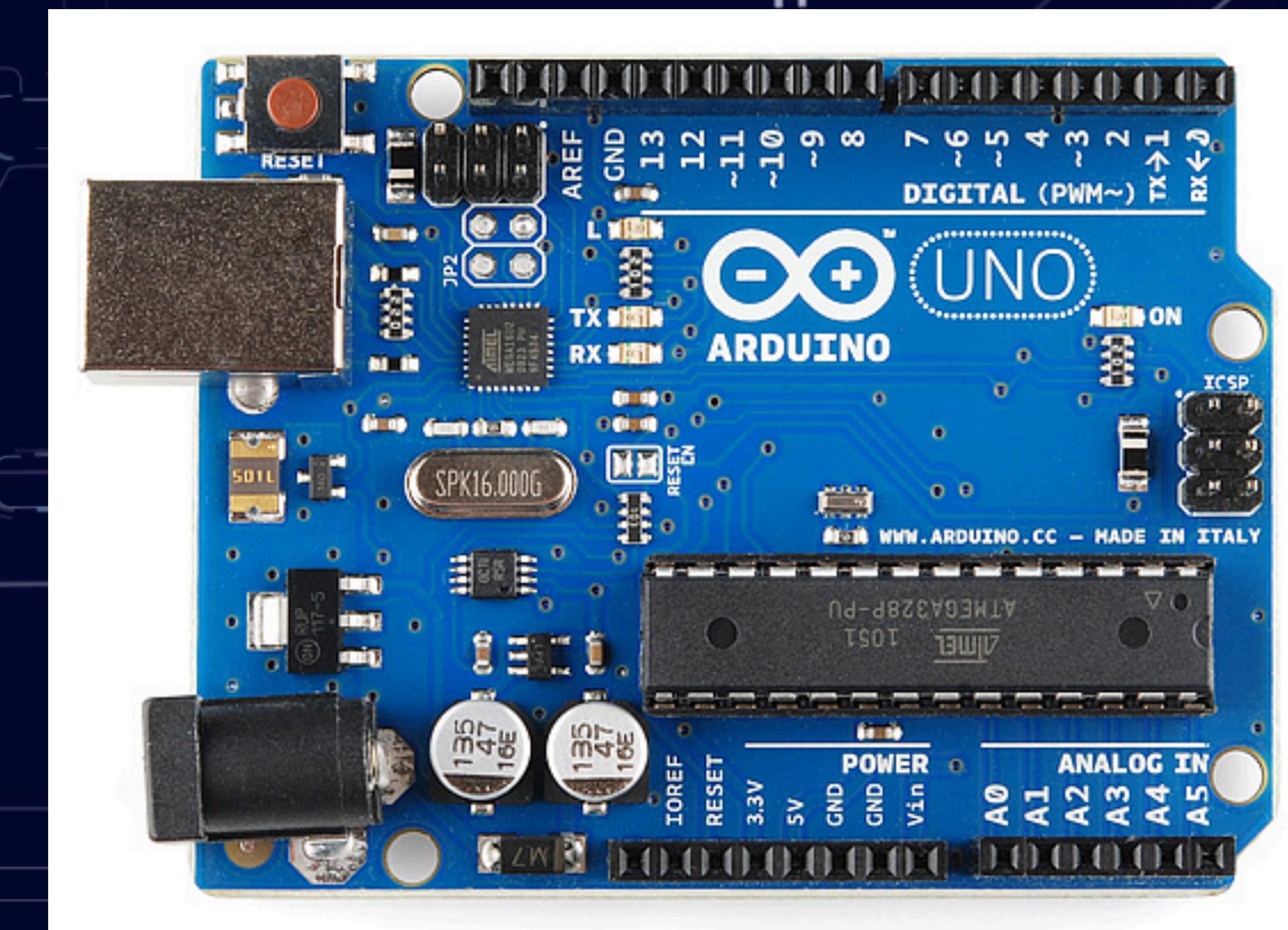


Image source: <https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>

1972

1982

199

200

Arduino

- Programmable device.
 - Easily connectable.
 - Open source.
 - Simple to use software

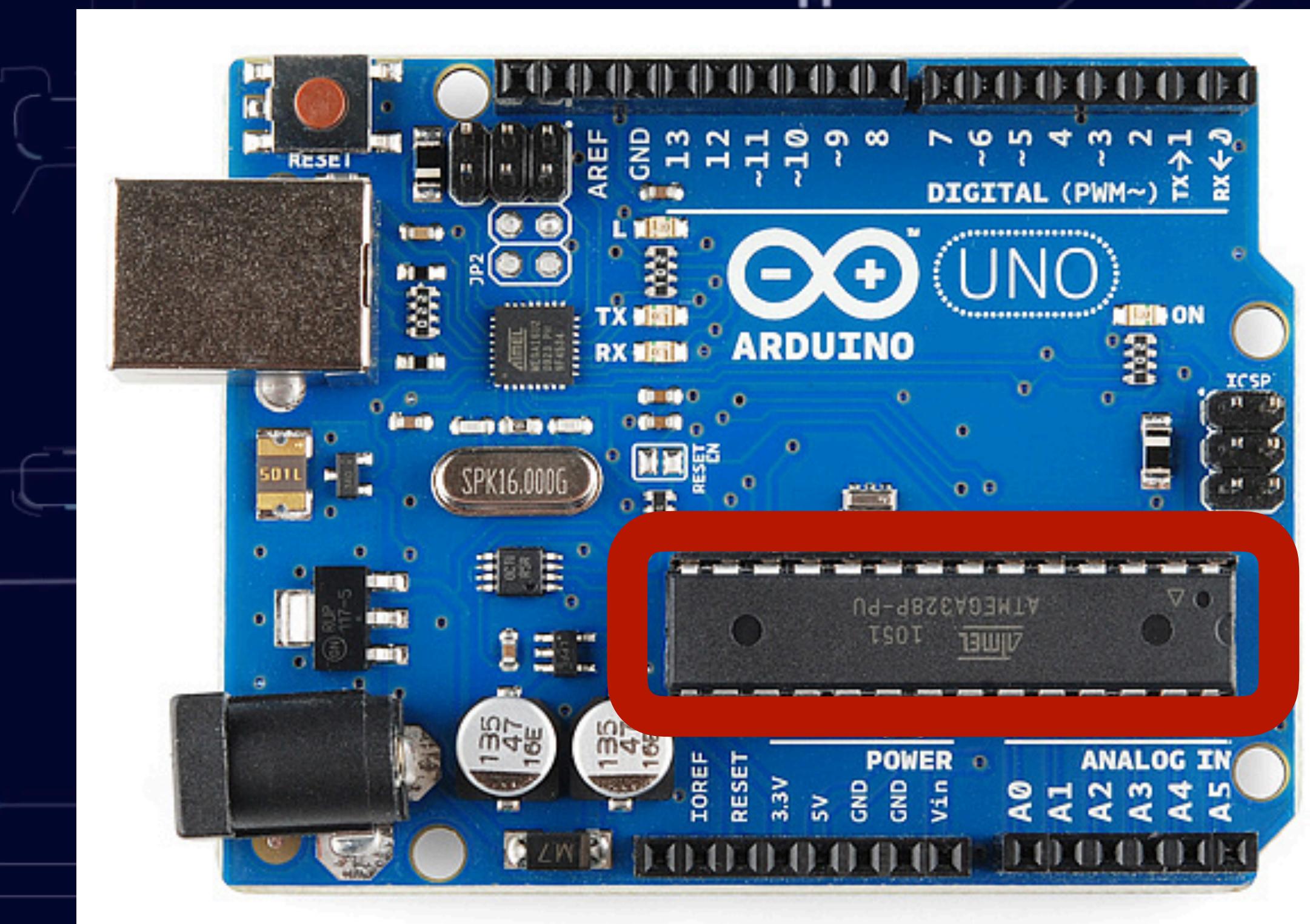


Image source: <https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>

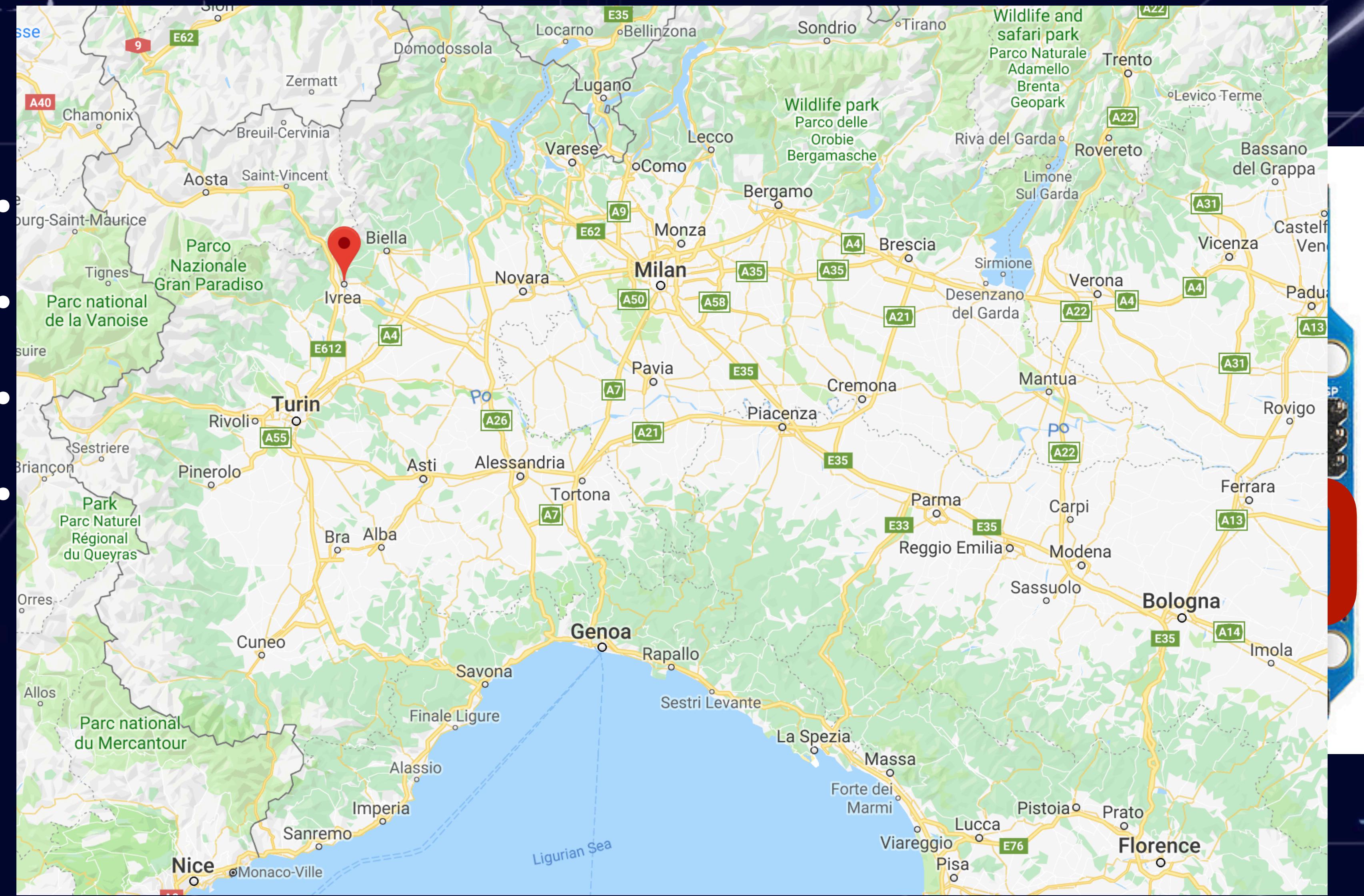
1972

1982

1999

2003

Arduino



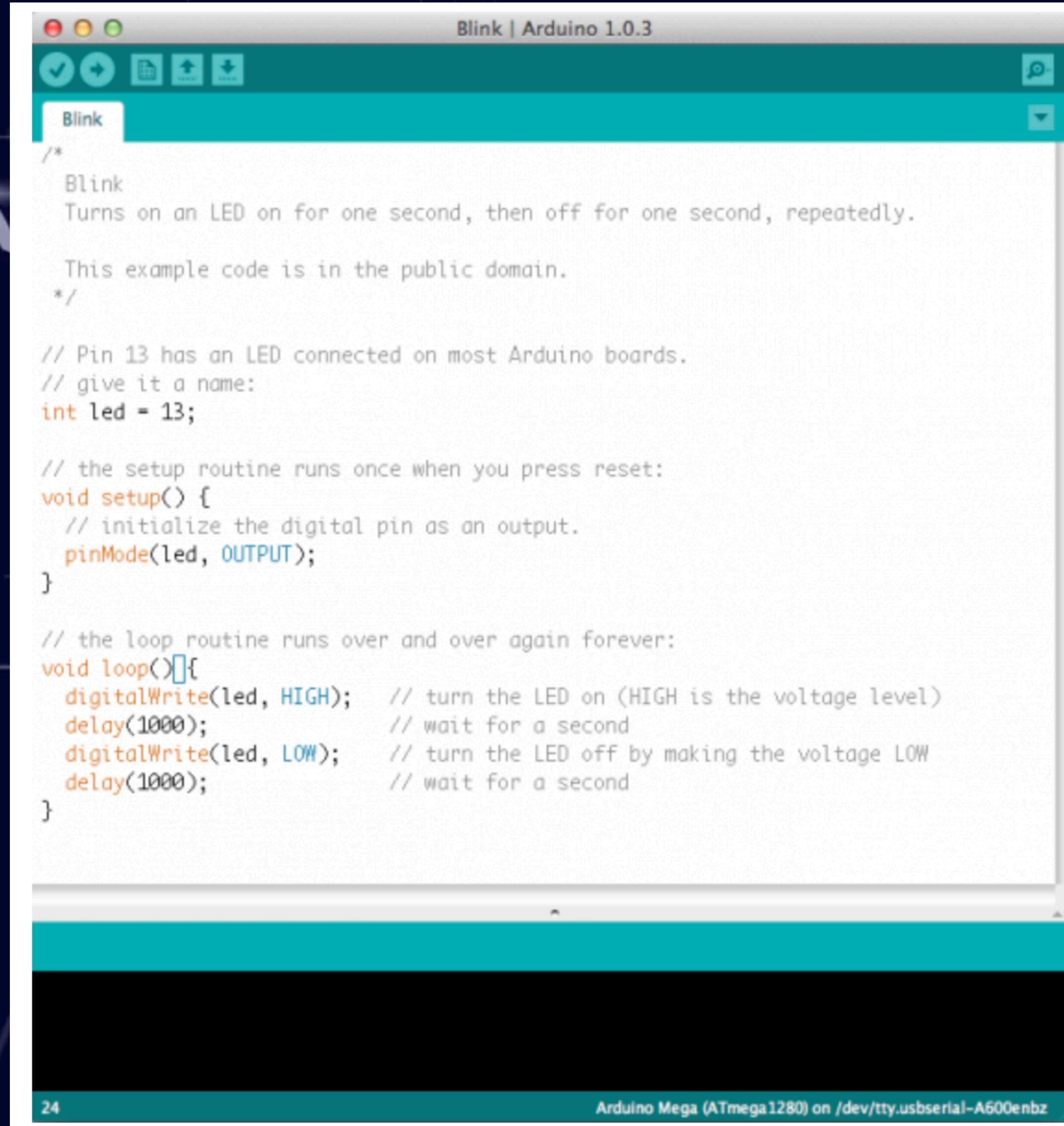
1972

1982

1999

2003

Arduino



The image shows a screenshot of the Arduino IDE version 1.0.3. The title bar reads "Blink | Arduino 1.0.3". The main window displays the "Blink" example sketch. The code is as follows:

```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);      // turn the LED on (HIGH is the voltage level)
  delay(1000);                // wait for a second
  digitalWrite(led, LOW);       // turn the LED off by making the voltage LOW
  delay(1000);                // wait for a second
}
```

At the bottom of the IDE window, there is a status bar with the text "Arduino Mega (ATmega1280) on /dev/tty.usbserial-A600enbz".



Arduino Ethernet

- Programmable device.
- Easily connectable.
- Open source.
- Simple to use software.
- Has network connectivity.



1972

1982

1999

2003

2012

Raspberry Pi

- Computer.
- Linux inside.
- Programming oriented.
- Full network stack.

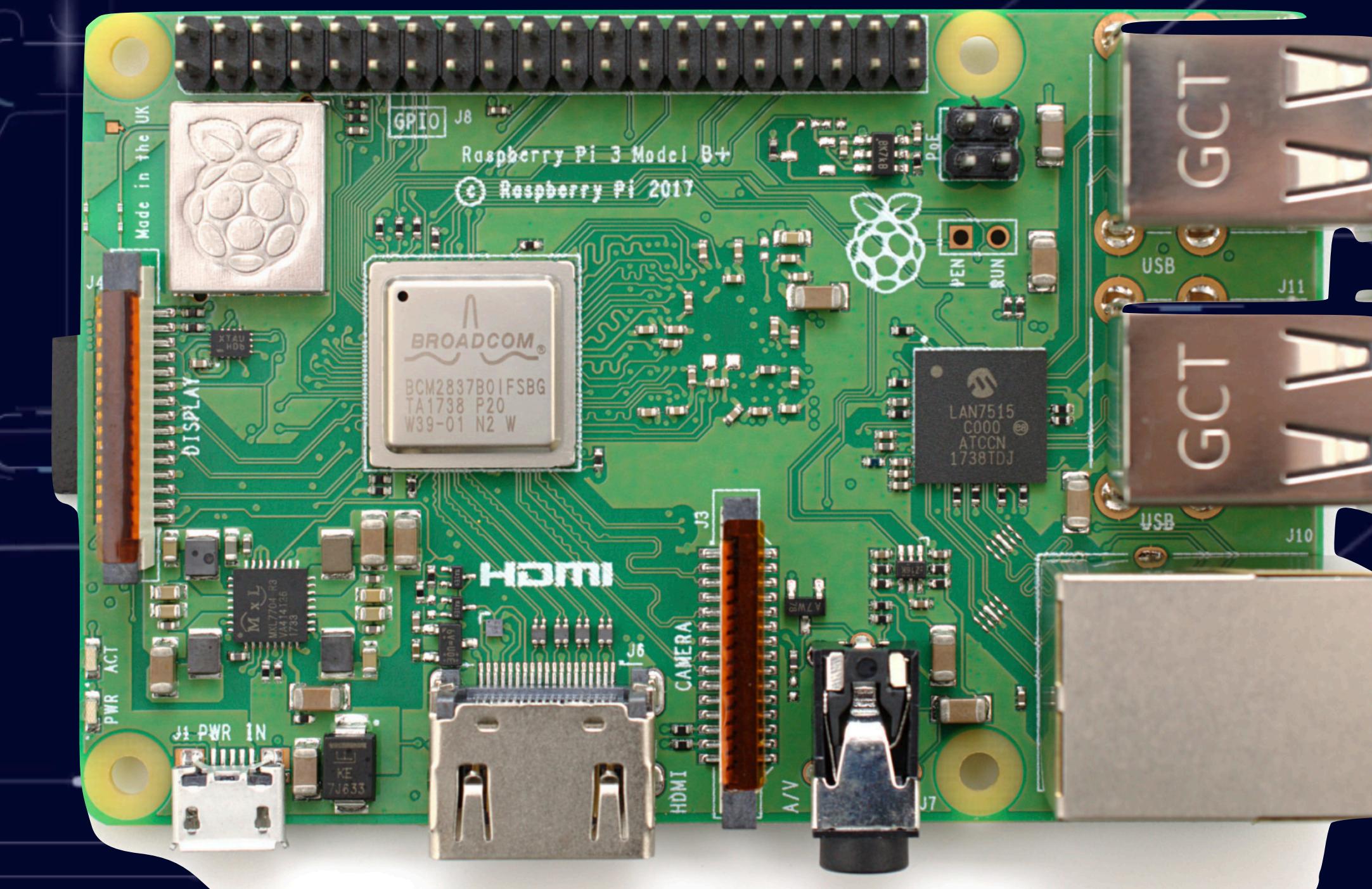


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>

1972

1982

1999

2003

2012

Raspberry Pi

- Computer.
- Linux inside.
- Programming oriented.
- Full network stack.

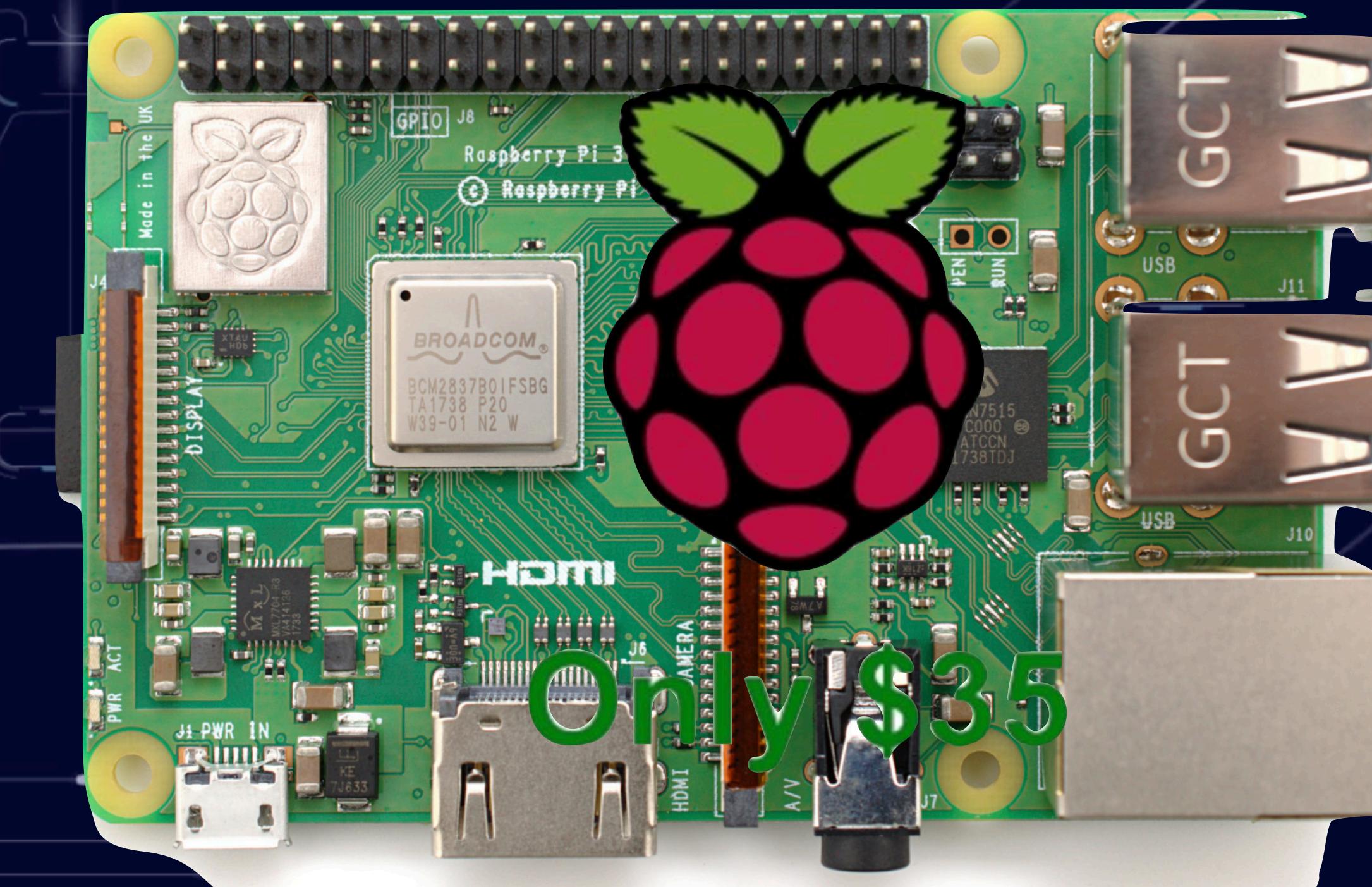


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>

1972

1982

1999

2003

2012

Raspberry Pi

February 29th, 2012

- Computer.
- Linux inside.
- Programming oriented.
- Full network stack.

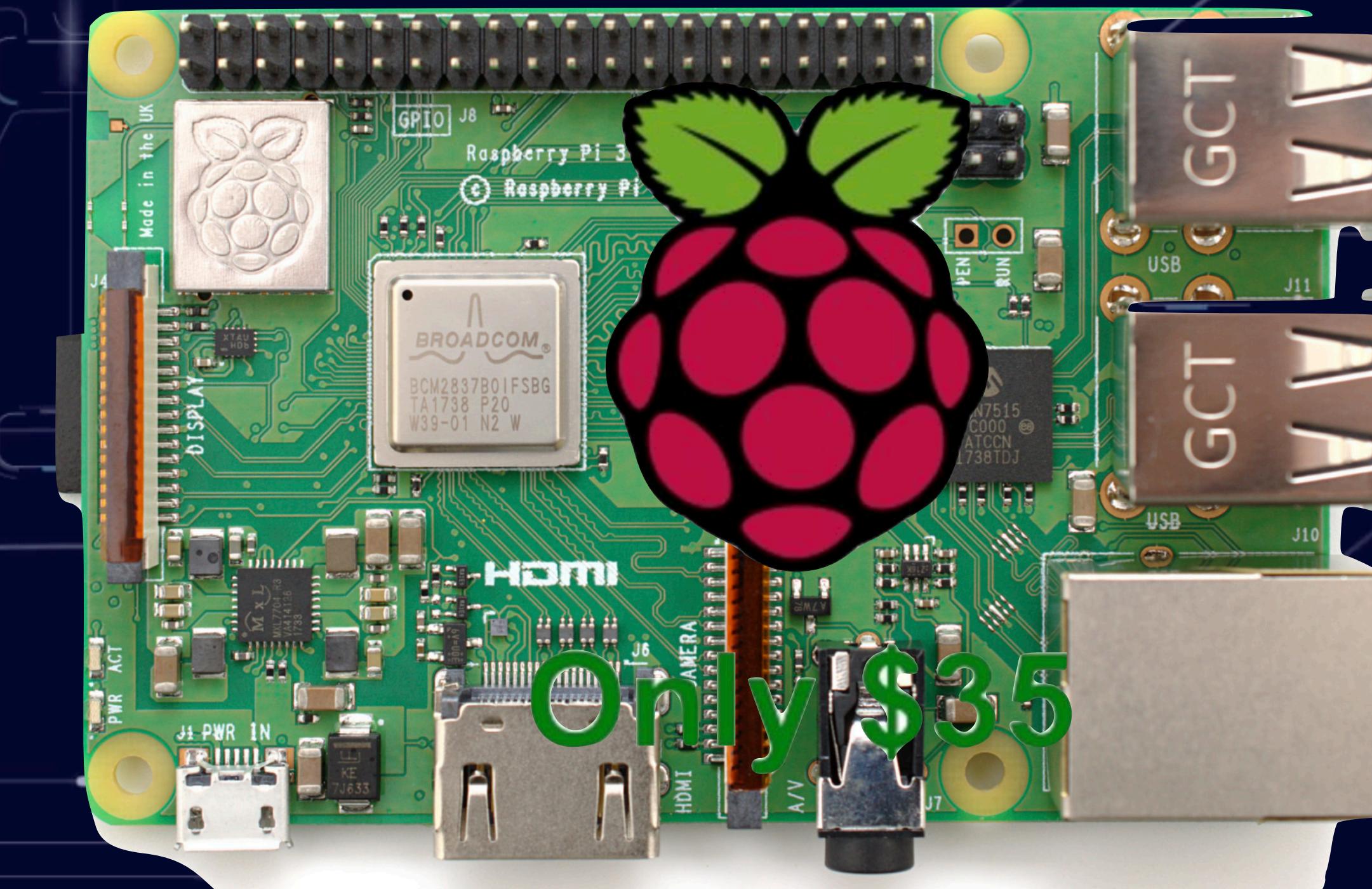


Image source: <https://www.flickr.com/photos/120586634@N05/39906369025/>



Raspberry Pi Zero

- A Broadcom BCM2835 application processor
- 1GHz ARM11 core
 - 40% faster than Raspberry Pi 1
- 512MB of LPDDR2 SDRAM
- A micro-SD card slot
- A mini-HDMI socket for 1080p60 video output
- Micro-USB sockets for data and power
- An unpopulated 40-pin GPIO header
- Small form factor, at 65mm x 30mm x 5mm



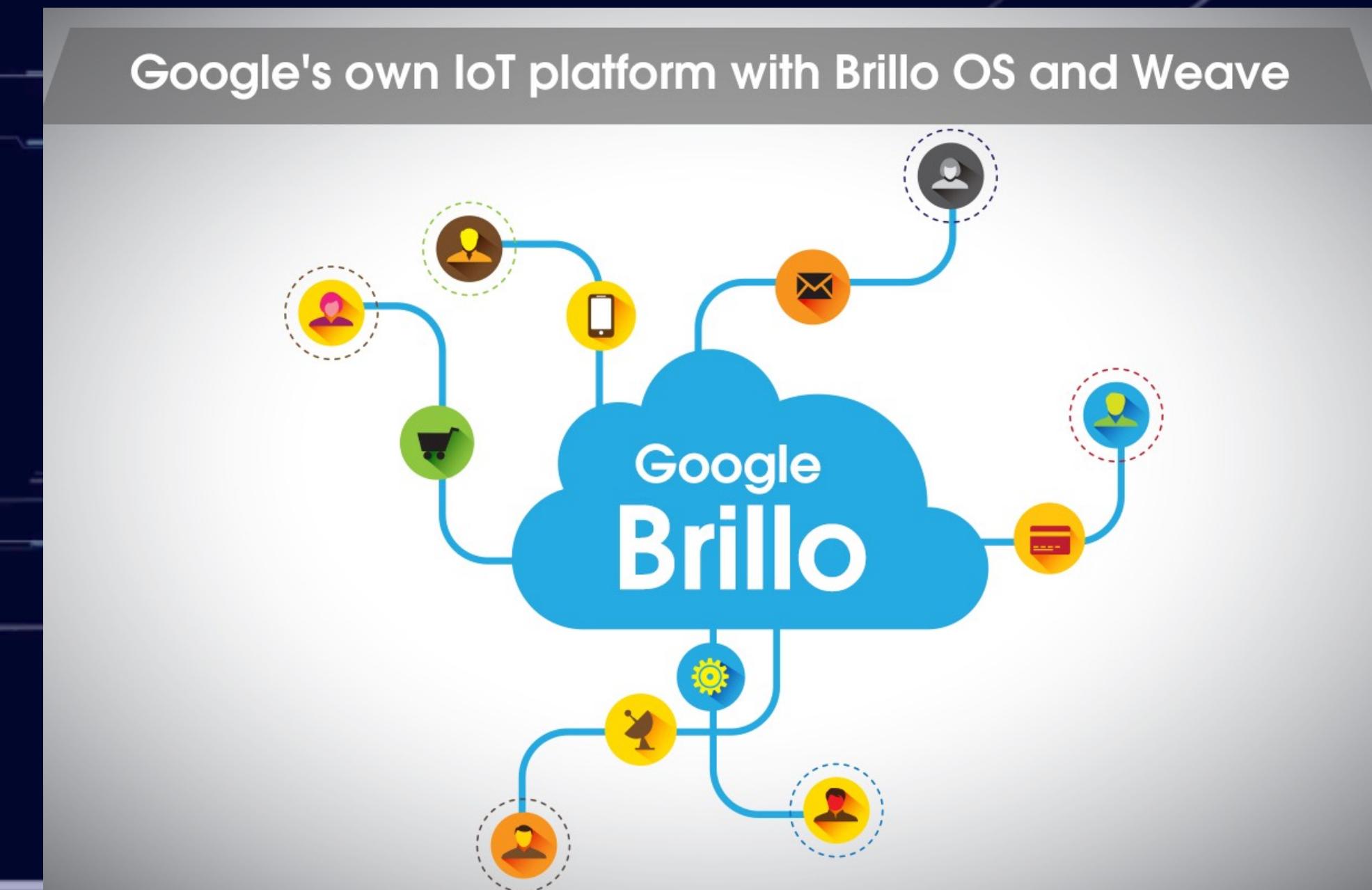
1972 1982

1999 2003

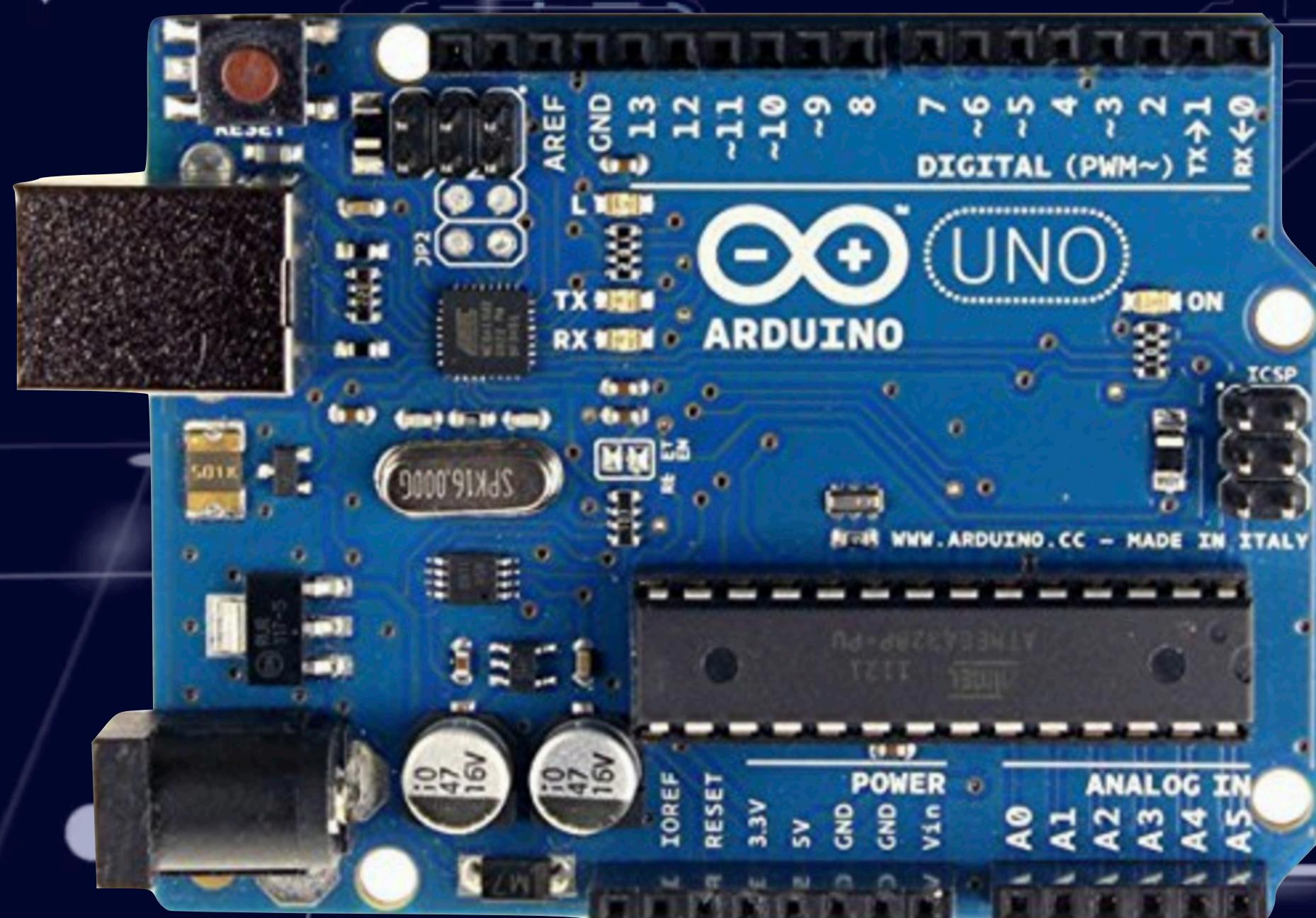
2012 2015 2018

Android Things

- Android Things lets you experiment with building devices on a trusted platform, without previous knowledge of embedded system design:
- Develop using the Android SDK and Android Studio.
- Access hardware such as displays and cameras natively through the Android framework.
- Connect your apps with Google services.
- Integrate additional peripherals through the Peripheral I/O APIs.
- Use the Android Things Console to push over-the-air feature and security updates.



Good with Sensors

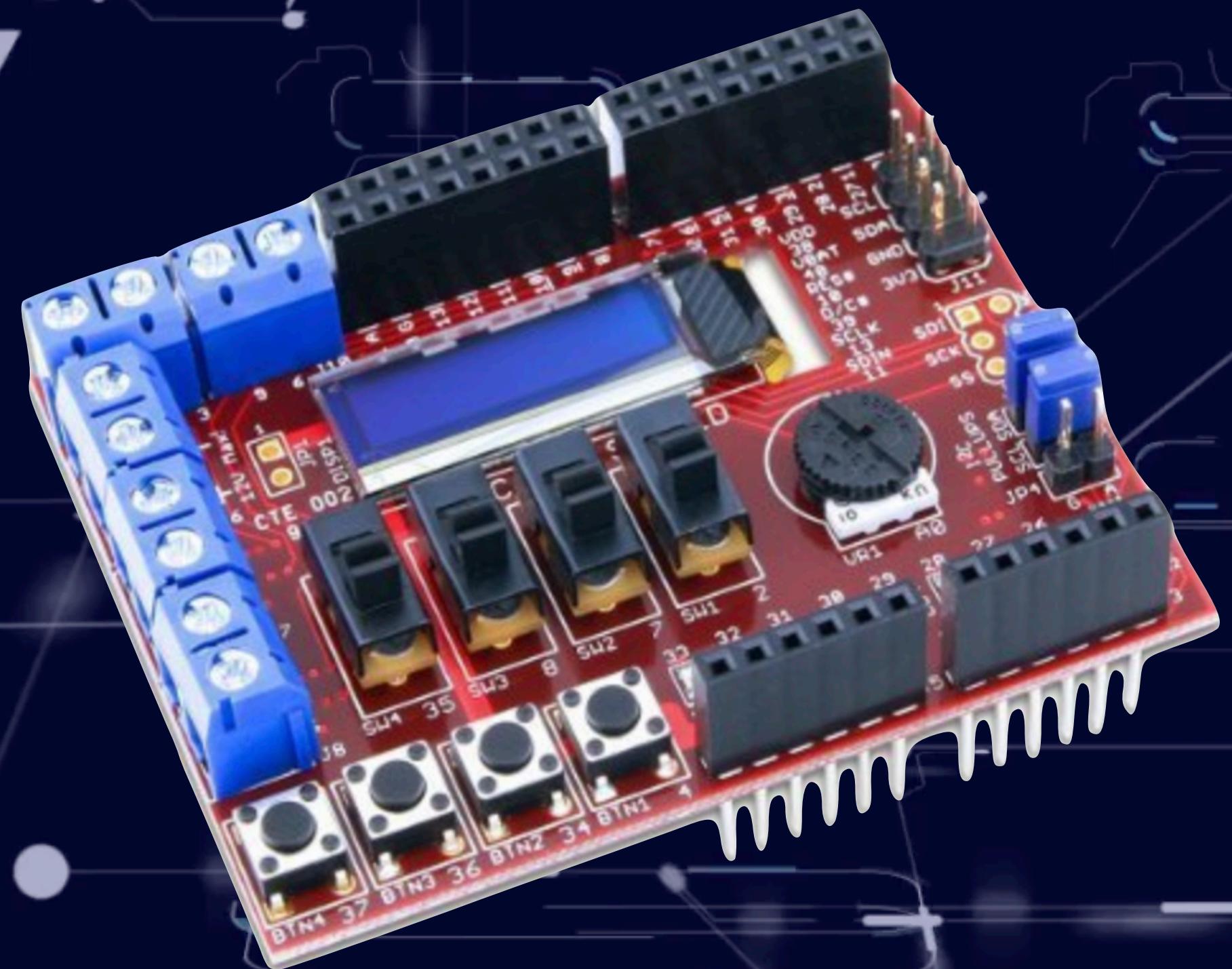


Arduino Uno

20\$

ATmega328P

Good with Sensors

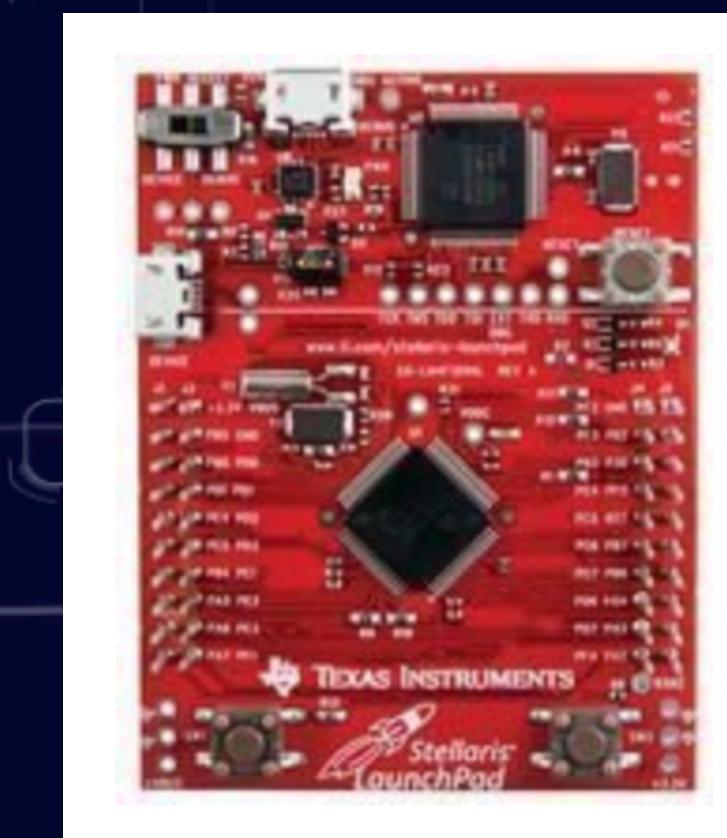


chipKIT Basic I/O Shield

40\$

PIC

Good with Sensors

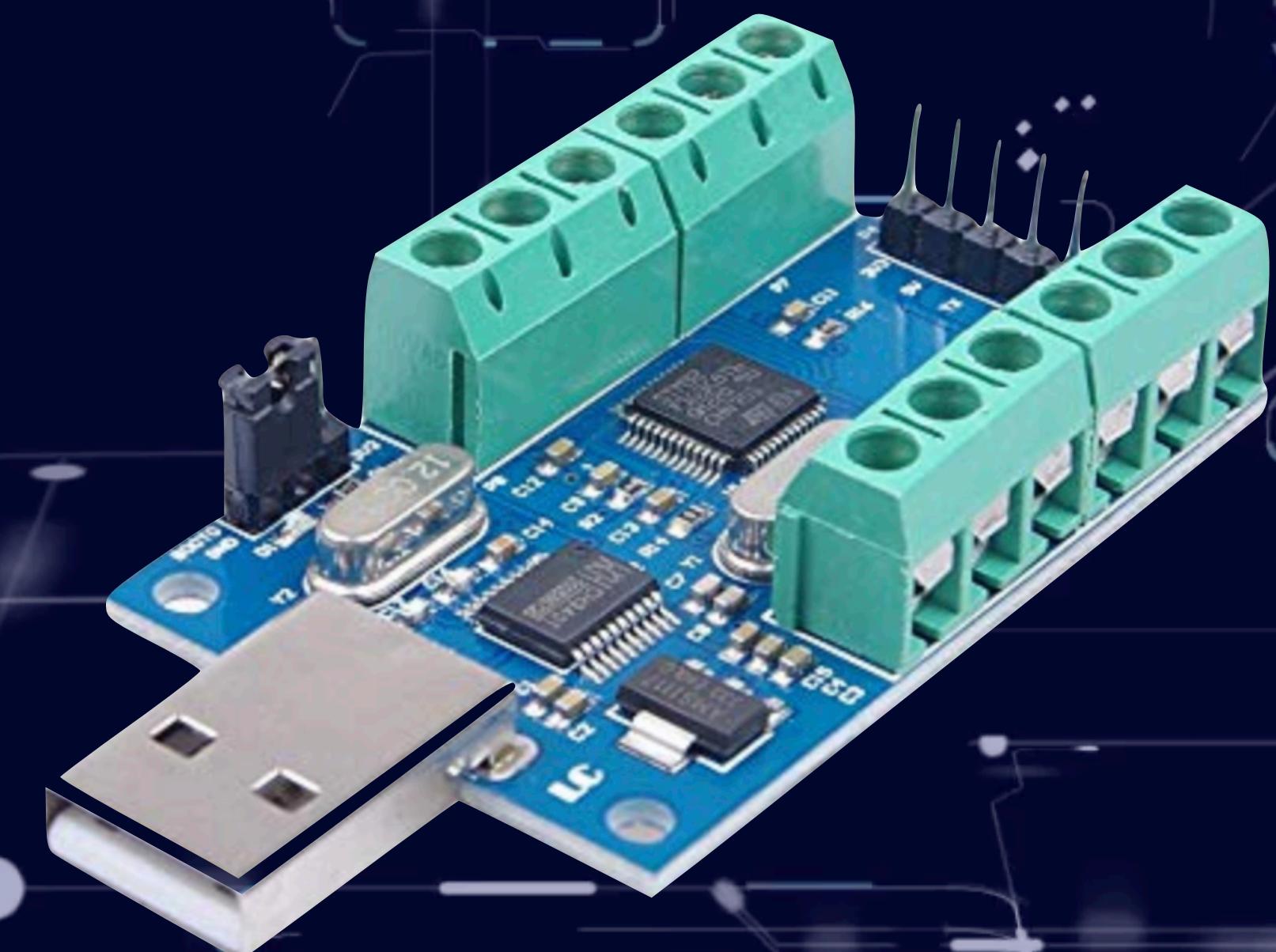


LAUNCHPAD

25\$

LM4F120

Good with Sensors and for Processing

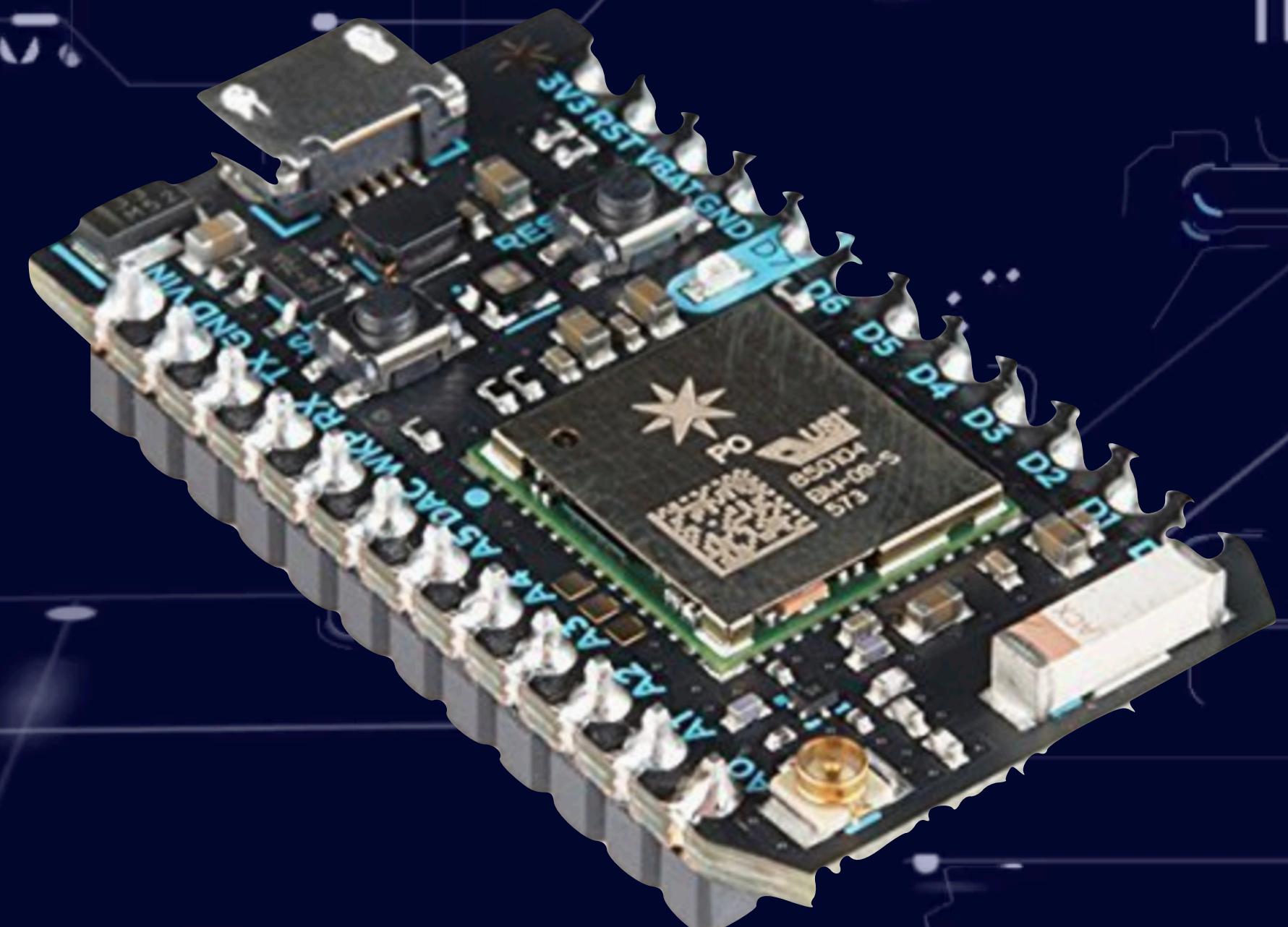


STM32

13\$

CH340 ADC Module

Good with Sensors and for Processing



Particle

20\$

ARM

Good with Sensors and for Processing

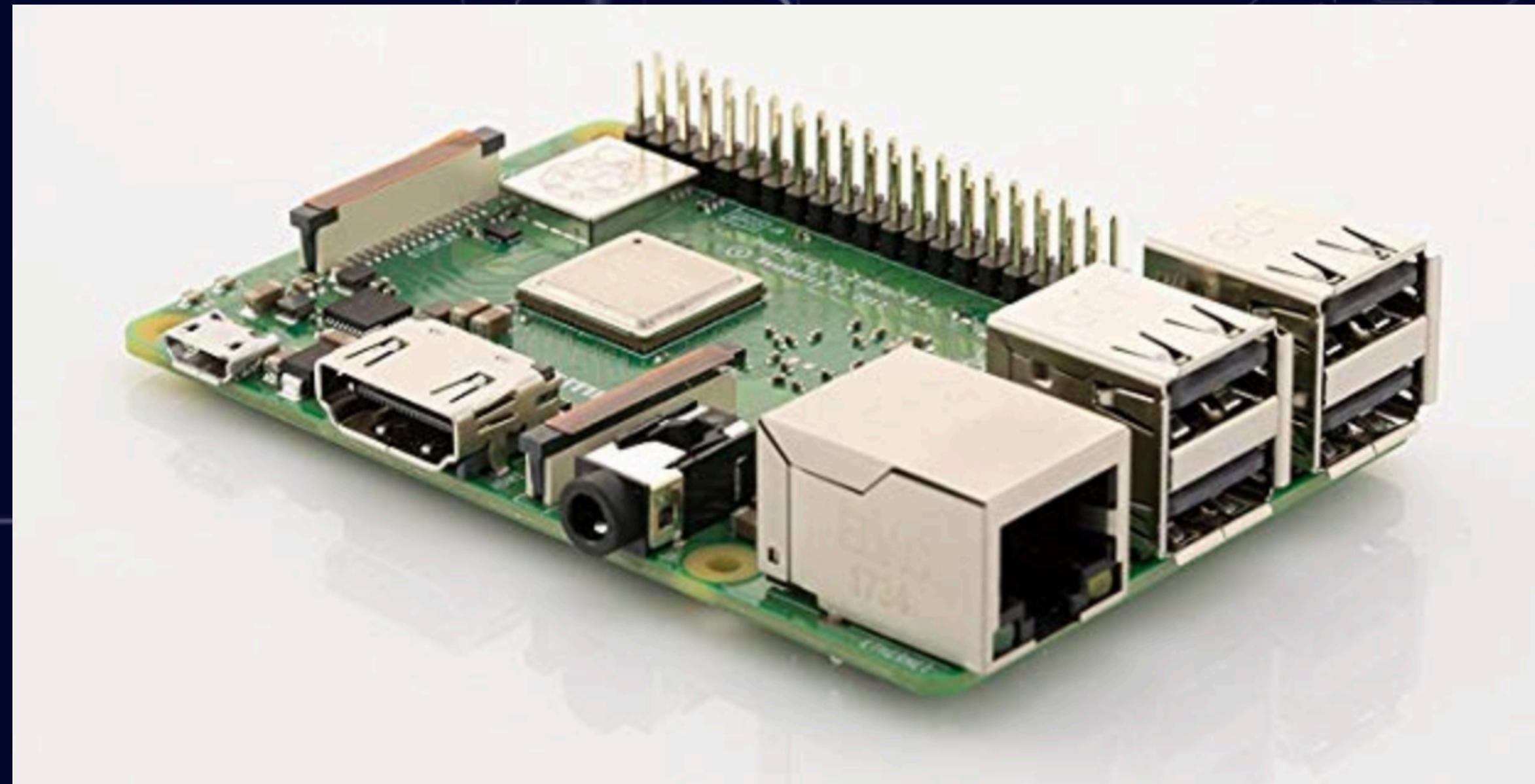


Raspberry Pi Zero

5\$

ARM

Good for Processing and Network

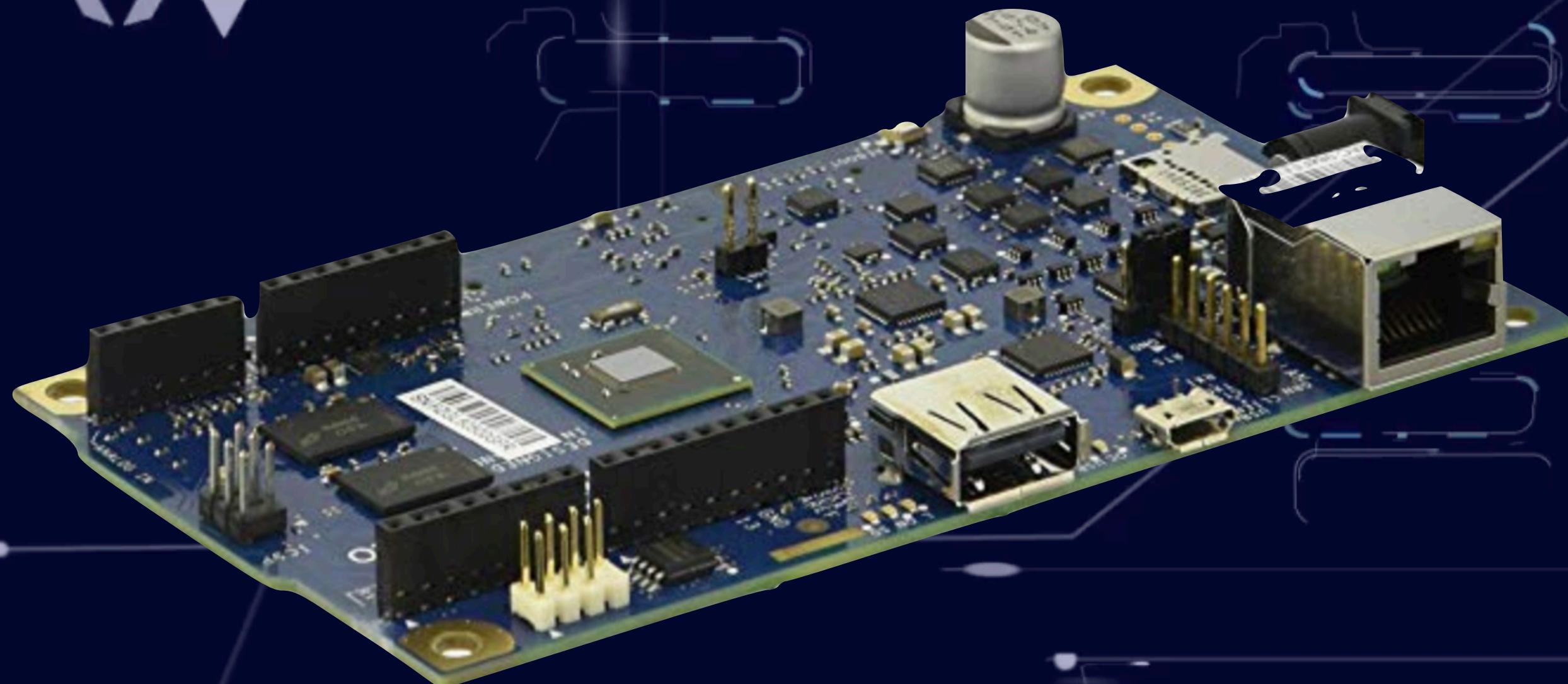


Raspberry Pi 3 B+

35\$

ARM - 1GB RAM

Good for Processing and Network



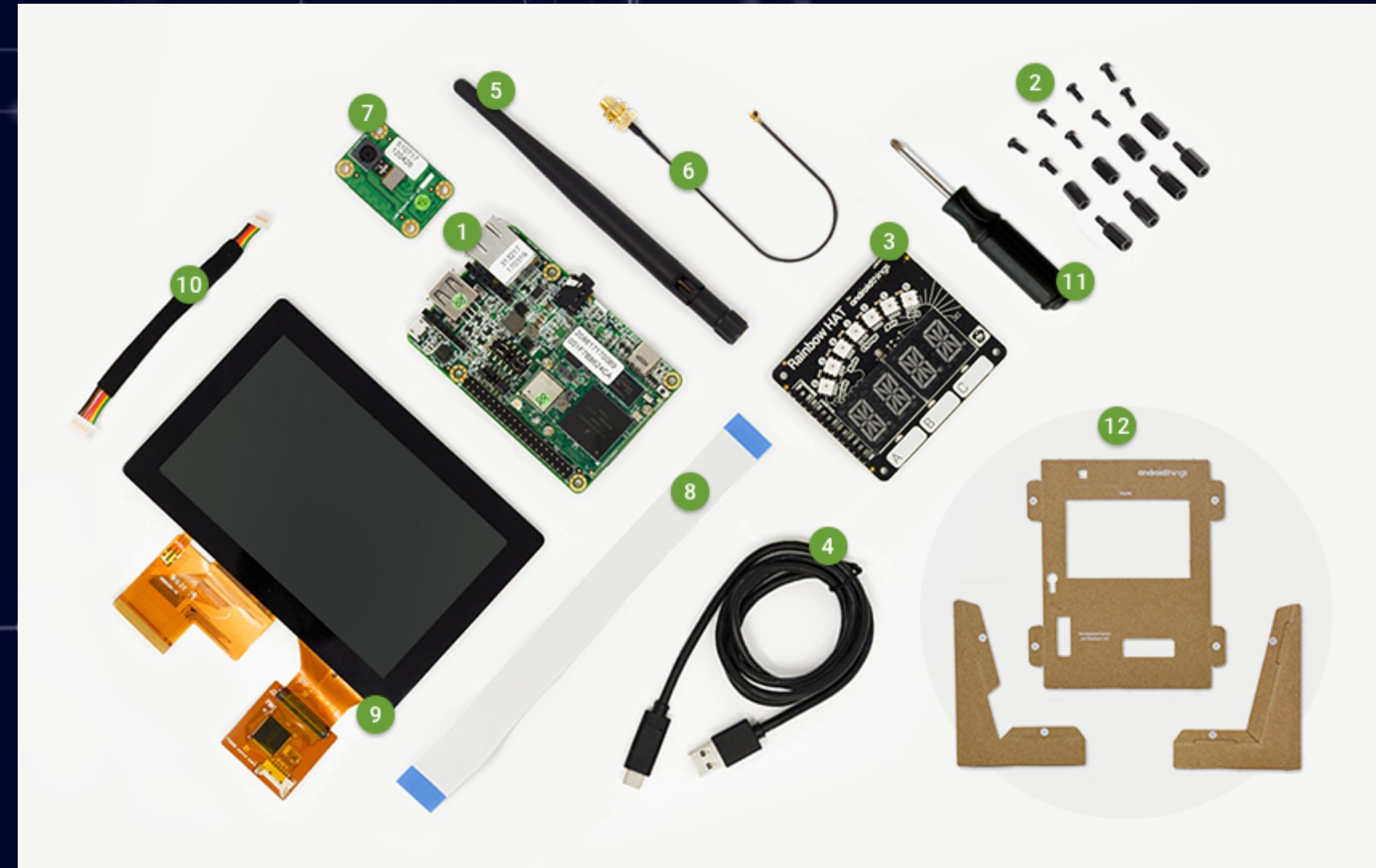
Intel Galileo Gen 2P

50\$

Quark SoC - 400MHz - 256MB RAM

Android Things

NXP i.MX7D Starter Kit



ARM Cortex-A7 + M4

200\$

<https://shop.technexion.com/pico-pi-imx7-startkit-rainbow-hat.html>

Android Things Raspberry Pi Kit



ARM Cortex A53

100\$

<https://androidthings.withgoogle.com/#!/kits/raspberry-pi-3-starter-kit>

IDE Options

The screenshot shows the Arduino IDE Sketchbook interface. On the left, a sidebar menu titled 'EDITOR' lists options: Sketchbook (selected), Examples, Libraries, Monitor, Help, and Preferences. A large 'Sketchbook' icon is also present. The main area displays a list of sketches in the 'Sketchbook'. One sketch, 'sketch_feb24a', is selected and shown in the code editor. The code editor window has tabs for 'sketch_feb24a.ino' and 'ReadMe.adoc'. At the top of the code editor, there are buttons for 'NEW SKETCH', file operations ('CREATE', 'OPEN', 'SAVE'), and a search bar labeled 'SEARCH SKETCHBOOK'. To the right of the code editor, there are buttons for 'Select Board or Port', '...', and 'SHARE'. Below the code editor, a message encourages importing sketches to the online Sketchbook.

sketch_feb24a

✓ → -- Select Board or Port -- ... SHARE

sketch_feb24a.ino ReadMe.adoc

```
1  /*
2  */
3 
4 
5 void setup() {
6 }
7 }
8 
9 void loop() {
10 }
11 
12 
```

SEARCH SKETCHBOOK

ORDERING BY LAST MODIFIED

sketch_feb24a

Import your sketches to your online Sketchbook and access them from any device!

EDITOR

Sketchbook Examples Libraries Monitor Help Preferences

NEW SKETCH CREATE OPEN SAVE

sketch_feb24a.ino ReadMe.adoc

Select Board or Port ... SHARE

sketch_feb24a

✓ → -- Select Board or Port -- ... SHARE

sketch_feb24a.ino ReadMe.adoc

```
1  /*
2  */
3 
4 
5 void setup() {
6 }
7 }
8 
9 void loop() {
10 }
11 
12 
```

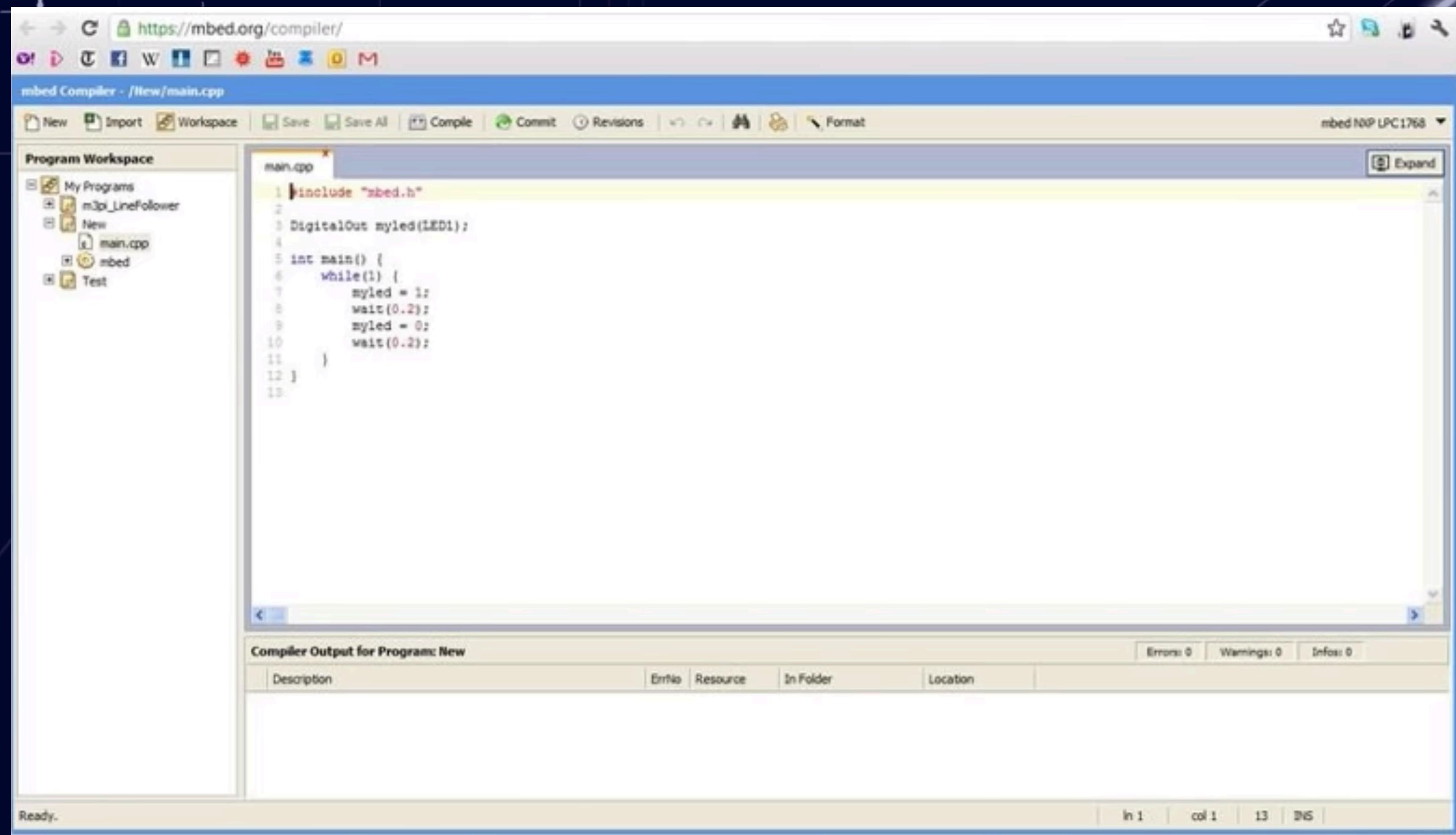
SEARCH SKETCHBOOK

ORDERING BY LAST MODIFIED

sketch_feb24a

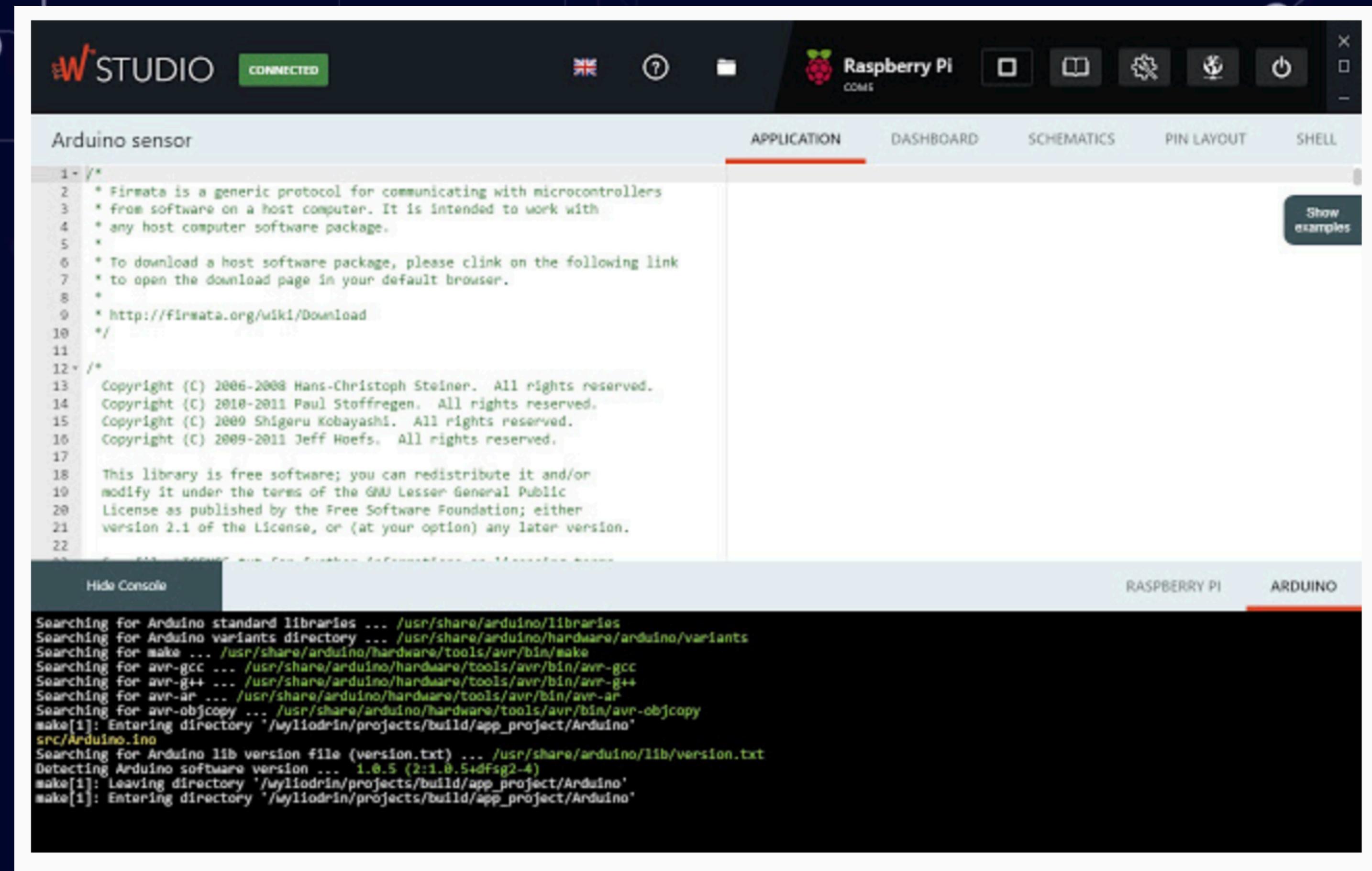
Import your sketches to your online Sketchbook and access them from any device!

IDE Options



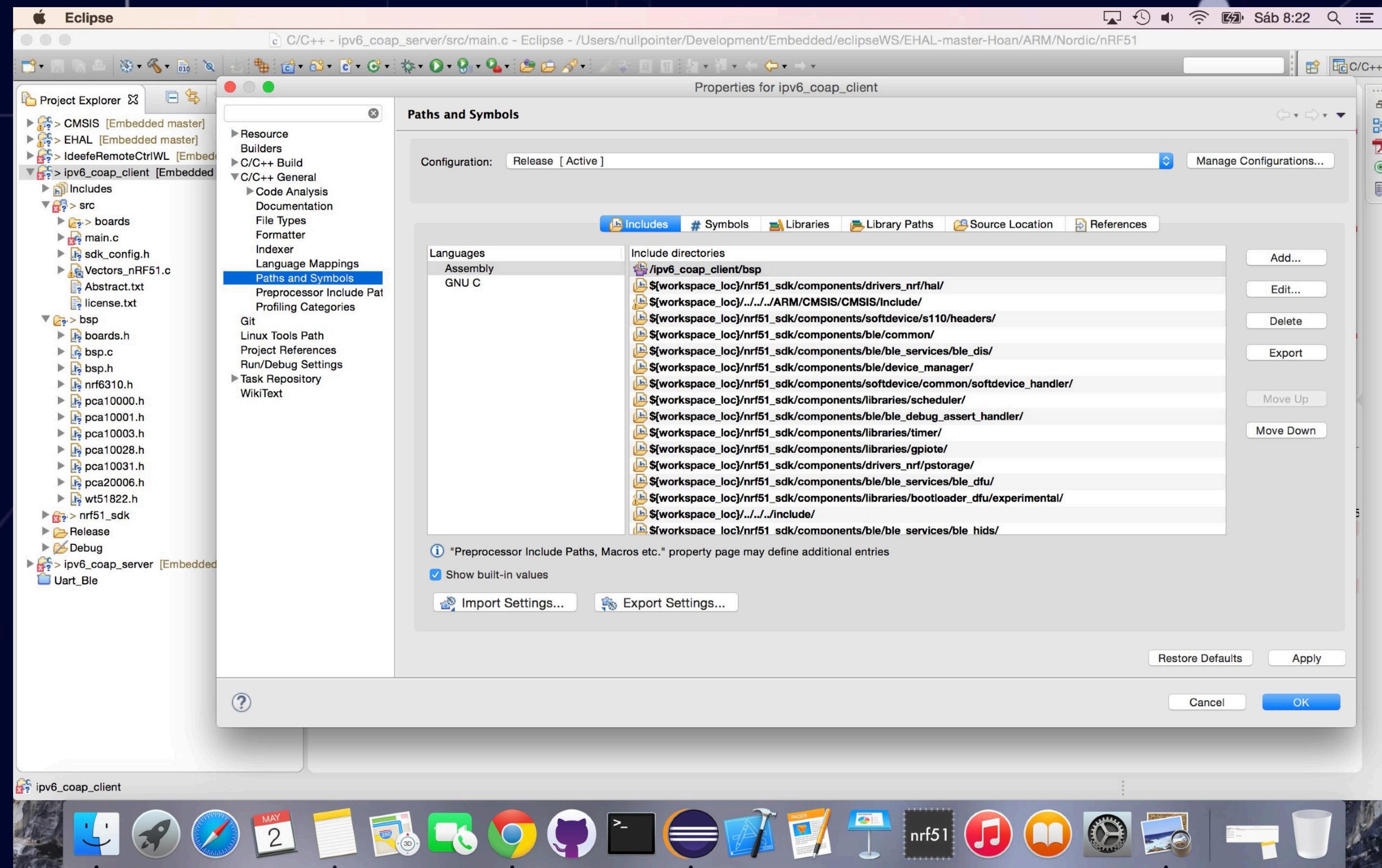
<https://os.mbed.com/>

IDE Options

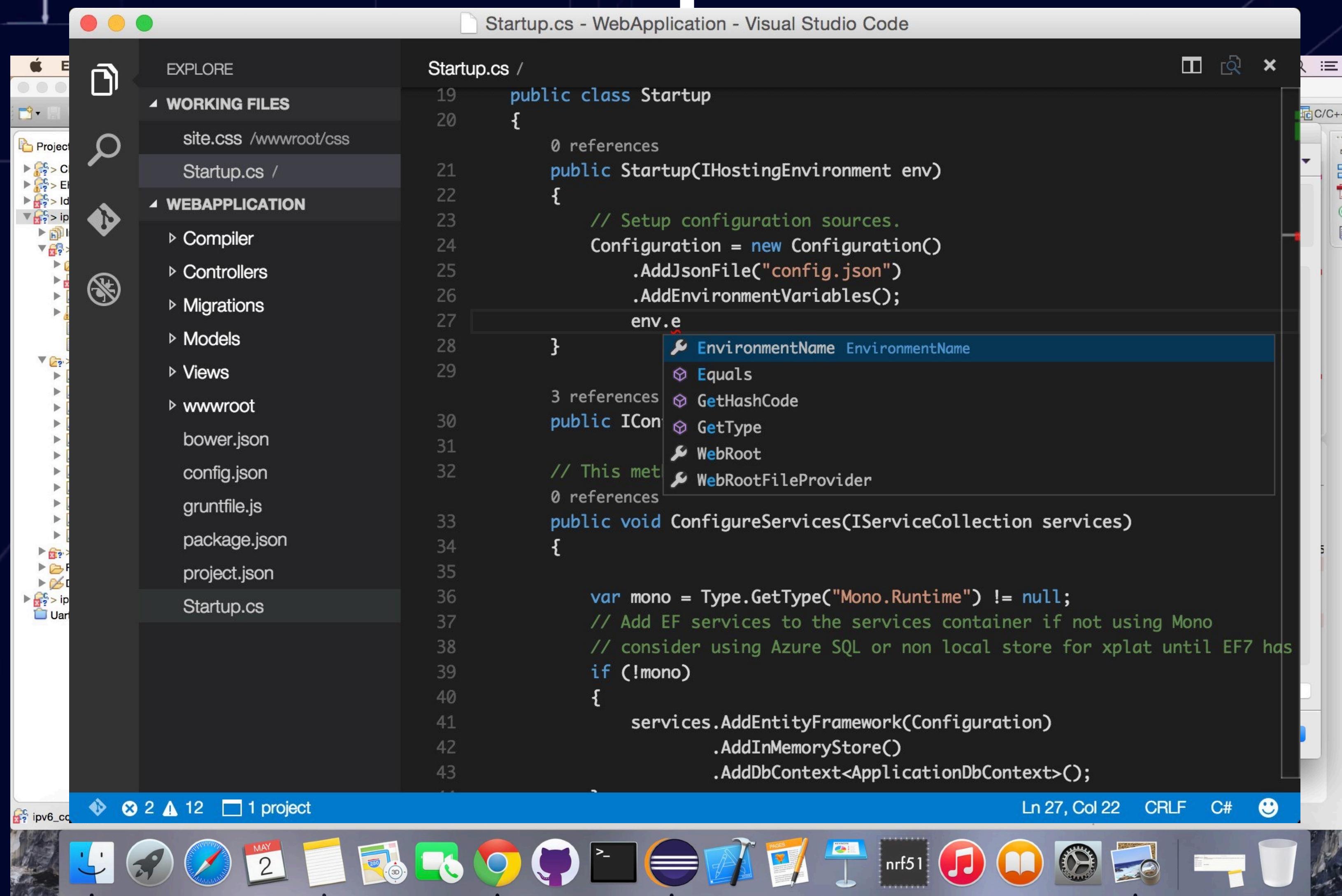


<https://wyliodrin.com/>

IDE Options



IDE Options



A screenshot of the Visual Studio Code interface on a Mac OS X desktop. The main window shows the file `Startup.cs` from a project named `WebApplication`. The code defines a `Startup` class with methods `ConfigureServices` and `Configure`. A code completion tooltip is open over the `env` variable, listing properties like `EnvironmentName`, `Equals`, `GetHashCode`, `GetType`, `WebRoot`, and `WebRootFileProvider`. The status bar at the bottom shows the file is 12 lines long and contains 1 project. The dock at the bottom has icons for various applications.

```
public class Startup
{
    public Startup(IHostingEnvironment env)
    {
        // Setup configuration sources.
        Configuration = new Configuration()
            .AddJsonFile("config.json")
            .AddEnvironmentVariables();
    }

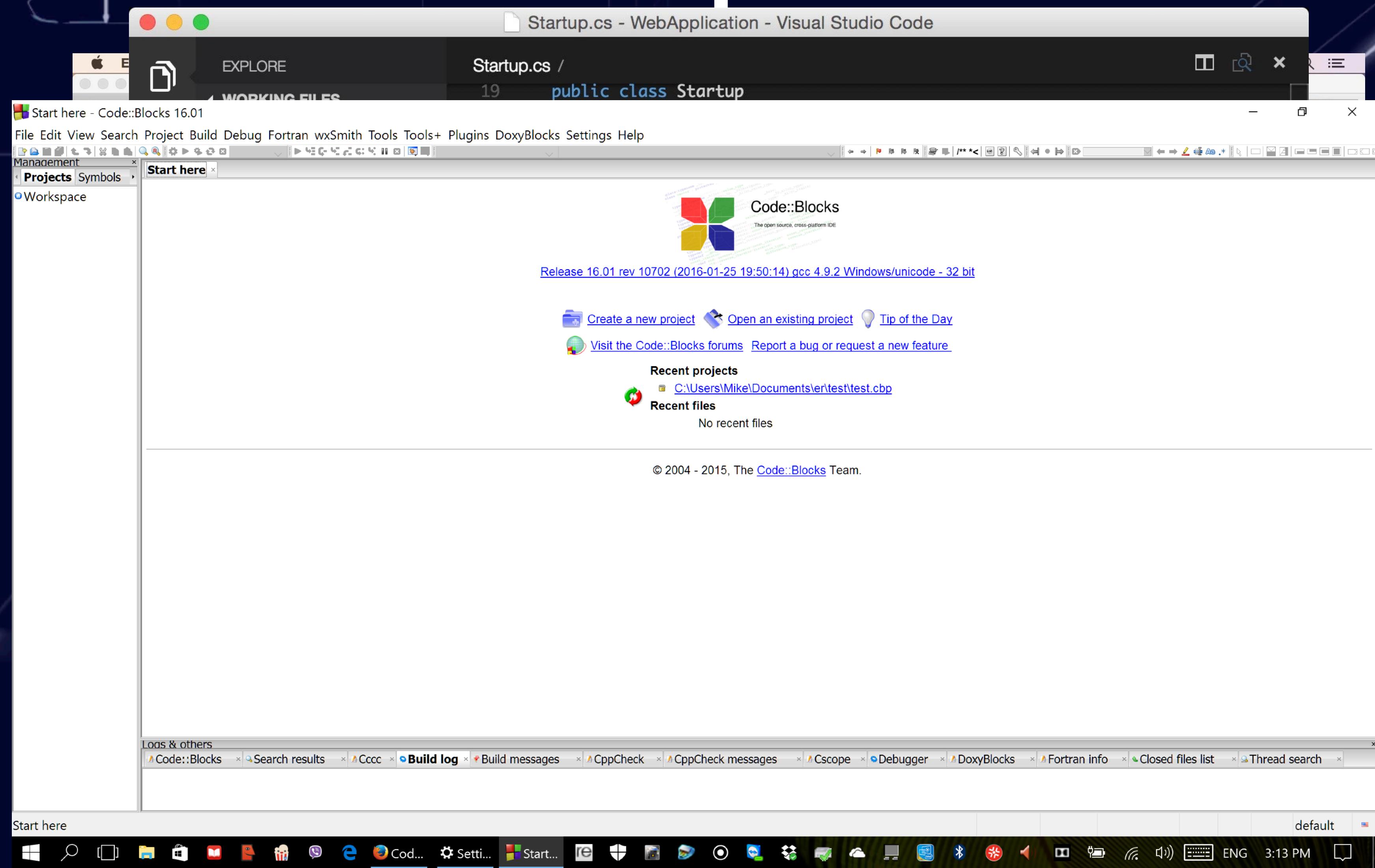
    public void ConfigureServices(IServiceCollection services)
    {

        var mono = Type.GetType("Mono.Runtime") != null;
        // Add EF services to the services container if not using Mono
        // consider using Azure SQL or non local store for xplat until EF7 has
        if (!mono)
        {
            services.AddEntityFramework(Configuration)
                .AddInMemoryStore()
                .AddDbContext<ApplicationContext>();
        }
    }

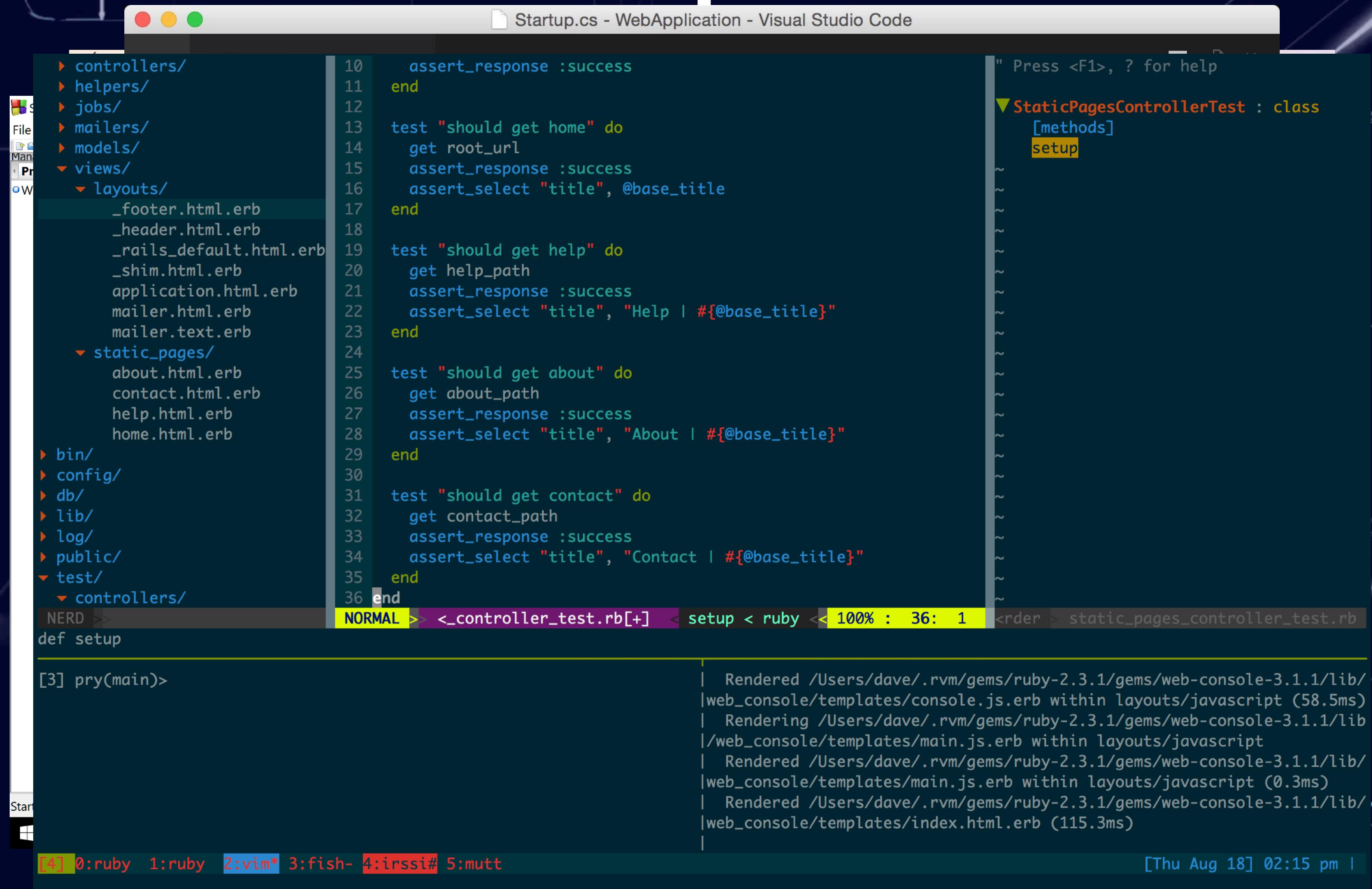
    public IHostingEnvironment Environment { get; private set; }

    public Configuration Configuration { get; private set; }
}
```

IDE Options



IDE Options



The screenshot shows a terminal window with a dark background and light-colored text. At the top, it displays the path: `File Manager / Project / WebApplication`. Below this is a file tree:

```
▶ controllers/
▶ helpers/
▶ jobs/
▶ mailers/
▶ models/
▶ views/
  ▶ layouts/
    _footer.html.erb
    _header.html.erb
    _rails_default.html.erb
    _shim.html.erb
    application.html.erb
    mailer.html.erb
    mailer.text.erb
  ▶ static_pages/
    about.html.erb
    contact.html.erb
    help.html.erb
    home.html.erb
▶ bin/
▶ config/
▶ db/
▶ lib/
▶ log/
▶ public/
▶ test/
  ▶ controllers/
    NERD >>
def setup
```

The main area of the terminal contains a Ruby script named `Startup.cs`:

```
10 assert_response :success
end

test "should get home" do
  get root_url
  assert_response :success
  assert_select "title", @base_title
end

test "should get help" do
  get help_path
  assert_response :success
  assert_select "title", "Help | #{@base_title}"
end

test "should get about" do
  get about_path
  assert_response :success
  assert_select "title", "About | #{@base_title}"
end

test "should get contact" do
  get contact_path
  assert_response :success
  assert_select "title", "Contact | #{@base_title}"
end
```

At the bottom of the terminal, there is a command history:

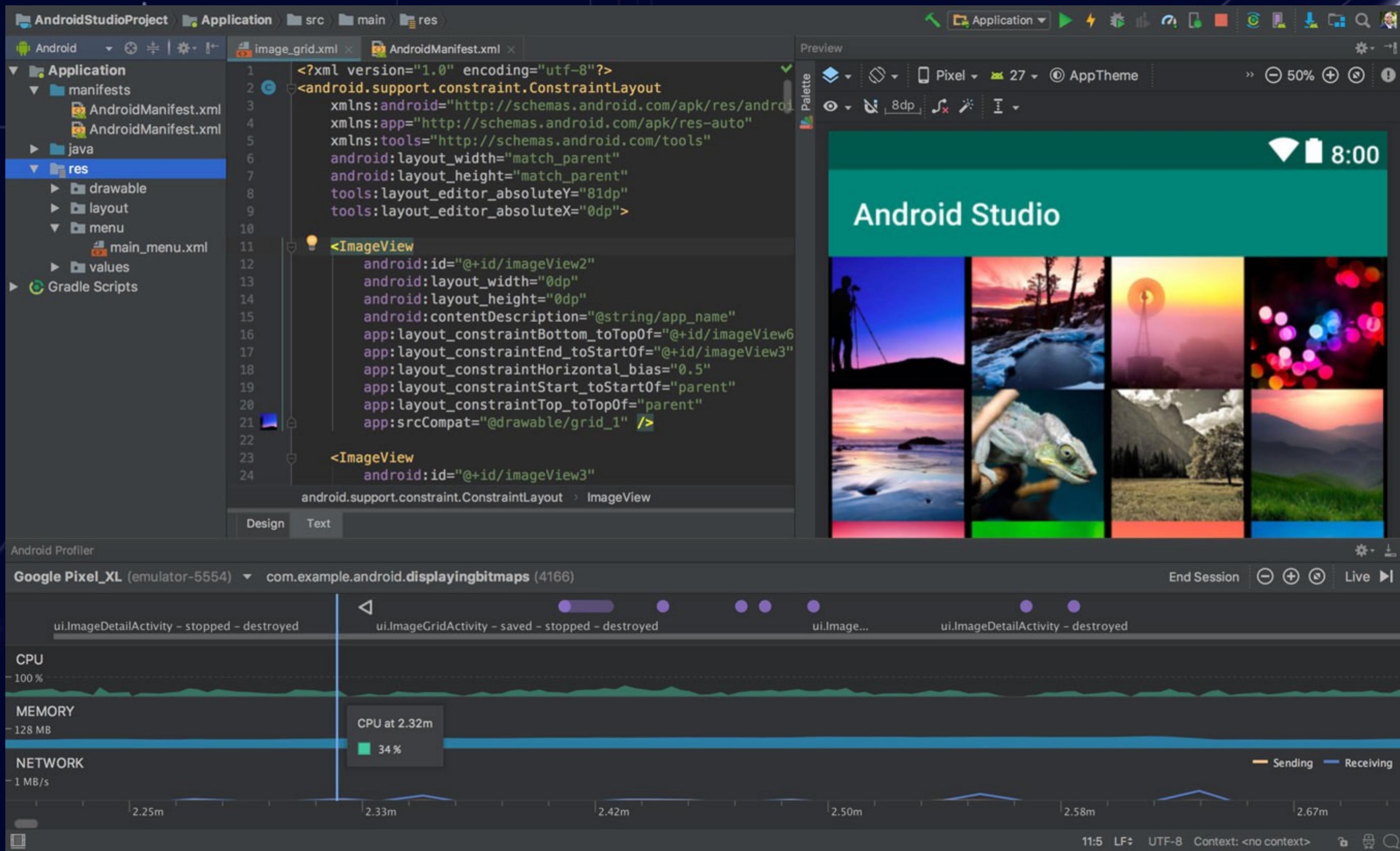
```
[4] pry(main)>
```

On the right side of the terminal, there is a vertical status bar with the following information:

- " Press <F1>, ? for help
- StaticPagesControllerTest : class
- [methods]
- setup

At the very bottom of the terminal, the status bar shows the date and time: [Thu Aug 18] 02:15 pm |

IDE Options



Laboratory Projects

- Two projects:
 - Individual project - 60% of the final grade.
 - Team project - 40% of the final grade.



Image source:
<http://inkawall.com>

Individual Project



Image source:
<http://ceelo.org/individual-leader/>

- A project similar to the samples available here:
 - <https://github.com/androidthings/doorbell/>
 - Or enhance an existing sample.

Individual Project

- A project similar to the samples available here:
 - <https://github.com/androidthings/doorbell/>
- Or enhance an existing sample.
- Due: Last laboratory before Easter holiday.
 - Either **April 21st** or **April 28th**, depending on your laboratory frequency.
- Deliverables:
 - The source code should be hosted in a github classroom repository.
 - A webpage presenting the project results, similar to:
 - <https://github.com/androidthings/doorbell/>
 - A short video presenting the results.



Image source:
<http://ceelo.org/individual-leader/>

Team Project



Image source:
<https://www.contractingbusiness.com>

- A team of 3 students to tackle a real-world problem.
- Either:
 - Choose an existing project proposed by the lab instructor.
 - Define a new one, together with the lab instructor.

Team Project

- A team of 3 students to tackle a real-world problem.
- Either:
 - Choose an existing project proposed by the lab instructor.
 - Define a new one, together with the lab instructor.
- Due: Last laboratory.
- Either **May 12h** or **May 19th**, depending on your laboratory frequency.
- Deliverables:
 - The source code should be hosted in a github classroom team repository.
 - A webpage presenting the project results, similar to:
 - <https://github.com/androidthings/doorbell/>
 - A short video presenting the results.
 - A companion mobile app to manage the IoT app.



Image source:
<https://www.contractingbusiness.com>

Lecture outcomes

- Understand the available hardware and software options.
- IDE options.
- Project details.

