Fachhochschule Nordwestschweiz
Hochschule für Technik

IoT Hardware for CS Bachelor Students

[CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) thomas.amberg@fhnw.ch, 24.01.2019

Abstract

Options and thoughts around Internet of Things hardware for computer science bachelor students.

Introduction

The following options were collected during the evaluation of IoT hardware for the course *IoT Engineering* [0] at FHNW, the University of Applied Sciences and Arts Northwestern Switzerland.

[0] https://www.fhnw.ch/de/studium/module/9280188

A simple reference model

A simple IoT reference model [1] is used to identify the main technical parts of connected products:

Device (with Sensors & Actuators) <-> (Edge) Gateway <-> (Cloud) Backend <-> Client (Apps)

[1] http://www.tamberg.org/iotmark/2018/ConnectedProductReferenceModel.pdf

Learning objectives

The course *IoT Engineering* [0] introduces bachelor students to IoT, covering the following topics:

- Architecture of IoT systems and Internet-connected products
- Prototyping with beginner hardware, sensors and actuators
- Connectivity options for short and long-distance data transmission
- Communication protocols and data formats in the context of IoT
- Local gateways and "edge computing" with Raspberry Pi
- IoT cloud platforms, Web dashboards and client apps
- Integration with surrounding systems, rule-based and voice control

IoT Hardware Qualities

The following qualities are considered relevant when choosing IoT hardware for CS bachelor students:

- Easy to get started, as many students are new to embedded programming and hardware
- Simple IDE for Windows, macOS and Linux, ideally open source, independent of 3rd party clouds
- Substantial community, for "Googleability" of code snippets, wiring diagrams and project ideas
- Broad library support for flexibility, extensibility and applicability re sensors/actuators/connectivity
- Available and affordable for schools and individuals, to allow direct "real world" use of learned skills
- "Future-proof", re form factor, connectivity, supported programming languages and suppliers
- Inspectable, documented, ideally open source, and open to expert tools and custom firmware

Programming Languages

The course requirements state *Object Oriented Programming I* [2], a Java course, as a precondition.

Language	*Advantage*	*Disadvantage*
C	+ Supported on all microcontrollers + Many (Arduino) examples	- Rather an "expert" language
Java	+ Available on Raspberry Pi + Already known to students (oop1) + Used in software eng./arch. courses	- Not widely used on microcontrollers (possible exception: feature-phones)
Javascript	+ Popular on Raspberry Pi + Used in Web programming courses	- Not widely used on microcontrollers
Python	+ Popular on Raspberry Pi + Upcoming on microcontrollers + Mainstream in data science	- Not used in Web programming courses at FHNW

Note that selecting a single language is merely a simplification, not strictly necessary. E.g. the Arduino language uses a subset of C that looks reasonably familiar to Java programmers.

From an engineering perspective, the languages *Go* and *Rust* might become interesting alternatives in the future, if they gain broader adoption on Raspberry Pi and Microcontrollers.

[2] https://www.fhnw.ch/de/studium/module/6007993

IoT Gateway Hardware

IoT gateway hardware (Wi-Fi routers, ZigBee bridge, LoRaWAN gateways) is often based on Linux.

Raspberry Pi

For compatibility with other courses, Raspberry Pi hardware is set here. A minimal setup would be the Raspberry Pi Zero W [3] connected to students laptops via USB [e.g. 4, p.18 ff], or accessed via Wi-Fi.

As the Raspberry Pi Zero W has Wi-Fi and BLE connectivity, it can serve as an example of a simple "BLE to Internet" gateway. There are BLE Libraries for Javascript [5], Python [6] and possibly Java. Used in groups of two or more, the Raspberry Pi can play central and peripheral roles in a BLE setup.

[3] https://www.raspberrypi.org/products/raspberry-pi-zero-w/
[4] http://www.tamberg.org/fhnw/2018/Syspr00Einfuehrung.pdf
[5] https://github.com/noble/noble
[6] https://learn.adafruit.com/install-bluez-on-the-raspberry-pi/installation

IoT Device Hardware

IoT devices usually are relatively small, low power devices with sensors and (local) connectivity. Below is a selection of specific boards, for details about development ecosystems, microcontroller models and prototyping hardware form factors, see *Appendix B*.

While the Raspberry Pi is regularly used as an IoT device in IoT projects, from a teaching perspective, there is still a substantial difference between a full-fledged Linux computer and a microcontroller. If the budget permits, teaching both, microcontrollers (for IoT devices) and Linux (for IoT gateways and IoT edge devices) promises the most complete experience of IoT systems and connected products.

Selection criteria: Arduino-programmable, Feather-compatible, ARM or ESP controller.

Arduino Uno (for reference)

Classic board for teaching "physical computing", but ATmega328P is too limited for IoT applications.

Adafruit Feather HUZZAH ESP8266

ESP8266 with Wi-Fi connectivity, works with Arduino, Feather form factor, not fully pin compatible.

[7] https://www.adafruit.com/product/3213 (~20$)

Adafruit Feather HUZZAH ESP32

ESP32 with Wi-Fi/BLE connectivity, works with Arduino, Feather form factor, BLE library not ready.

[8] https://www.adafruit.com/product/3405 (~20$)

Adafruit Feather nRF52840 Express

ARM Cortex M4 with BLE connectivity, works with Arduino (or CircuitPython), Feather compatible.

[9] https://www.adafruit.com/product/4062 (~25$)

Micro:bit

ARM Cortex M0, nRF51822 BLE connectivity and sensors, works with Arduino, custom form factor.

[10] https://microbit.org/ (~15$)

#IoT Octopus

ESP8266 with Wi-Fi connectivity and on-board sensors, works with Arduino, Feather-compatible.

[11] https://www.tindie.com/products/FabLab/iot-octopus-badge-for-iot-evaluation/ (~35$)

IoT Edge Device Hardware

IoT edge devices are an emerging category of IoT devices with substantial computing resources for data processing and machine-learning applications, as well as memory intensive sensors like cameras.

Raspberry Pi

The Raspberry Pi Zero W, which is set as IoT gateway hardware already, can serve this role, with reasonable processing power for simple machine learning applications [12] and a camera connector.

[12] https://www.tensorflow.org/lite/rpi

IoT Hardware Setups

Selected options for IoT hardware setups, best choice might vary depending on region, taxes, budget.

1) Raspberry Pi + Feather nRF52840 Express + Breadboard or Grove

Use nRF5280 for microcontroller, BLE and Raspberry Pi for IoT (edge) gateway, Wi-Fi use cases.

2) Raspberry Pi + Micro:bit + on-board Sensors or Grove

Use Micro:bit for microcontroller, BLE and Raspberry Pi for IoT (edge) gateway, Wi-Fi use cases.

3) Raspberry Pi + Feather HUZZAH ESP8266 + Breadboard or Grove

Use ESP8266 for microcontroller, Wi-Fi and Raspberry Pi for BLE, IoT (edge) gateway use cases.

4) Raspberry Pi + Grove

Use Raspberry Pi for BLE, Wi-Fi, IoT (edge) gateway use cases. No hands-on with microcontrollers.

Modules for LoRaWAN / Wi-Fi / BLE Connectivity

Depending on the setup, one or more of the options in *Appendix A* might allow additional use-cases.

Conclusion

Evaluation of IoT hardware for computer science education becomes more manageable if the various dimensions like development ecosystems, microcontroller models, form factors, wiring standards and language support are understood separately and cross-checked with desired objectives and qualities.

Note: The proposed setups should be reevaluated every year, as availability and best practice change.

Appendix A - Cost (01.2019)

IoT Hardware Setups

Approximate cost of selected options, best choice might vary depending on region, taxes, budget.

1) Raspberry Pi + Feather nRF52840 Express

Use nRF5280 for microcontroller, BLE and Raspberry Pi for IoT (edge) gateway, Wi-Fi use cases.

a) With Grove Sensors (~100$)

https://www.raspberrypi.org/products/ (Zero W, incl. SD Card w/ Reader, USB cable, ~30$)
https://www.adafruit.com/product/4062 (Feather nRF52840 Express, incl. BLE, ~25$)
https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-Zero-p-3187.html (connector, ~9$)
https://store.particle.io/products/grove-starter-kit?variant=13981942612037 (conn. + sensors, ~35$)

Sensors: Button, Rotary Sensor, Ultrasonic Ranger, Temp. & Hum. Sensor, Light Sensor v1.2,
Actuators: Chainable RGB LED, Buzzer, 4-Digit Display

b) With Breadboard Sensors (~100$)

https://www.raspberrypi.org/products/ (Zero W, incl. SD Card w/ Reader, USB cable, ~30$)
https://www.adafruit.com/product/4062 (Feather nRF52840 Express, incl. BLE, ~25$)
Breadboard, Wires, Sensors (~45$)

- https://www.adafruit.com/product/64 (Breadboard, ~5$)
- https://www.adafruit.com/product/169 (Servo, ~6$)
- https://www.adafruit.com/product/189 (PIR, ~10$)
- https://www.adafruit.com/product/375 (Door Switch, ~4$)
- https://www.adafruit.com/product/758 (Jumper Wires, M/M, ~4$)
- https://www.adafruit.com/product/385 (DHT22 Temp, Hum, ~10$)
- https://www.adafruit.com/product/1766 (Vibration, ~1$)
- https://www.adafruit.com/product/160 (Buzzer, ~1$)
- https://www.adafruit.com/product/161 (LDR, ~1$)
- https://www.adafruit.com/product/159 (Tricolor LED, ~2$)
- Resistors etc. (~1$)

2) Raspberry Pi + Micro:bit

Use Micro:bit for microcontroller, BLE and Raspberry Pi for IoT (edge) gateway, Wi-Fi use cases.

a) With Micro:bit on-board sensors only (~50$)

https://www.raspberrypi.org/products/ (Zero W, incl. SD card w/ reader, USB cable, ~30$)
https://microbit.org/resellers/ (Micro:bit, incl. sensors, BLE, ~20$)

b) With Grove Sensors (~120$)

https://www.raspberrypi.org/products/ (Zero W, incl. SD card w/ reader, USB cable, ~30$)
https://microbit.org/resellers/ (Micro:bit, incl. sensors, BLE, ~20$)
https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-Zero-p-3187.html (connector, ~9$)
https://www.seeedstudio.com/Grove-Inventor-Kit-for-micro-bit-p-2891.html (conn. + sensors, ~60$)
Sensors: Rotary Angle Sensor, Speaker, Ultrasonic Ranger, Light Sensor v1.2, Gesture,
Actuators: Red LED, WS2812 Waterproof LED Strip - 30 LEDs 1 meter, 4-Digit Display

3) Raspberry Pi + Feather HUZZAH ESP8266

Use ESP8266 for microcontroller, Wi-Fi and Raspberry Pi for BLE, IoT (edge) gateway use cases.

a) With Grove Sensors (~95$)

https://www.raspberrypi.org/products/ (Zero W, incl. SD Card w/ Reader, USB cable, ~30$)
https://www.adafruit.com/product/3213 (Micro:bit microcontroller, incl. Wi-Fi, ~20$)
https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-Zero-p-3187.html (connector, ~9$)
https://store.particle.io/products/grove-starter-kit?variant=13981942612037 (conn. + sensors, ~35$)
Sensors: Button, Rotary Sensor, Ultrasonic Ranger, Temp. & Hum. Sensor, Light Sensor v1.2,
Actuators: Chainable RGB LED, Buzzer, 4-Digit Display

b) With Breadboard, Sensors (~75 to 95$)

Either https://www.raspberrypi.org/products/ (Zero W, incl. SD Card w/ Reader, USB cable, ~30$)
https://www.adafruit.com/product/2680 (Special offer, ESP8266 microcontroller + sensors, ~38$)
https://www.adafruit.com/product/64 (Breadboard, ~5$)
https://www.adafruit.com/product/592 (USB cable, ~3$)

Or https://www.raspberrypi.org/products/ (Zero W, incl. SD Card w/ Reader, USB cable, ~30$)
https://www.adafruit.com/product/3032 (Special, ESP8266 incl. Wi-Fi, breadboard, sensors, ~45$)

Or https://www.raspberrypi.org/products/ (Zero W, incl. SD Card w/ Reader, USB cable, ~30$)
https://www.adafruit.com/product/3213 (ESP8266 incl. Wi-Fi, ~20$)
Breadboard, Wires, Sensors & Actuators (~45$)
https://www.adafruit.com/product/64 (Breadboard, ~5$)
https://www.adafruit.com/product/169 (Servo, ~6$)
https://www.adafruit.com/product/189 (PIR, ~10$)
https://www.adafruit.com/product/375 (Door Switch, ~4$)
https://www.adafruit.com/product/758 (Jumper Wires, M/M, ~4$)
https://www.adafruit.com/product/385 (DHT22 Temp, Hum, ~10$)
https://www.adafruit.com/product/1766 (Vibration, ~1$)
https://www.adafruit.com/product/160 (Buzzer, ~1$)
https://www.adafruit.com/product/161 (LDR, ~1$)
https://www.adafruit.com/product/1734 (Neopixel Color LED, ~1$)
Resistors etc. (~2$)

4) Raspberry Pi only

Use Raspberry Pi for BLE, Wi-Fi, IoT (edge) gateway use cases. No hands-on with microcontrollers.

a) With Grove Sensors (~80$)

https://www.raspberrypi.org/products/ (Zero W, incl. SD Card w/ Reader, USB cable, ~30$)
https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-Zero-p-3187.html (connector, ~9$)
Sensors (~18$)
https://www.seeedstudio.com/Grove-Button--p-1243.html (~2$)
https://www.seeedstudio.com/Grove-Light-Sensor-p-1253.html (~3$)
https://www.seeedstudio.com/Grove-Temperature-Humidity-Sensor-DHT1-p-745.html (~5$)
https://www.seeedstudio.com/Grove-PIR-Motion-Sensor-p-802.html (~8$)
Actuators (~22$)
https://www.seeedstudio.com/Grove-Buzzer-p-768.html (~2$)
https://www.seeedstudio.com/Grove-Red-LED-p-1142.html (~2$)
https://www.seeedstudio.com/Grove-Chainable-RGB-Led-V2-0-p-2903.html (~6$)
https://www.seeedstudio.com/Grove-4-Digit-Display-p-1198.html (~6$)
https://www.seeedstudio.com/Grove-Servo-p-1241.html (~6$)

Modules for LoRaWAN / Wi-Fi / BLE Connectivity

Depending on the setup, one or more of these options might allow additional use-cases.

a) Grove, via UART (+ ~20$)

https://www.seeedstudio.com/Grove-LoRa-Radio-868MHz-p-2776.html (RFM95W, ~20$)
https://www.seeedstudio.com/Grove-UART-WiFi-V2-ESP828-p-3054.html (ESP8285, ~14$)
https://www.seeedstudio.com/Grove-BLE-p-1929.html (HM-11, ~20$)

b) Feather Wing Extension (+ ~20$)

https://learn.adafruit.com/radio-featherwing (RFM95W, Feather wing, ~20$)
https://www.adafruit.com/product/3213 (ESP8266 incl. Wi-Fi, Feather replacement, ~20$)

c) Breadboard Breakout, via UART or SPI (+ ~20$)

https://www.adafruit.com/product/3072 (RFM95W, ~20$)
https://www.adafruit.com/product/2471 (ESP8266, ~10$)

Appendix B

Embedded Development Ecosystems

Microcontroller models are often classified into distinct "ecosystems", including an IDE and libraries.

Arduino

Arduino is a simple, open source prototyping hardware ecosystem, originally including ATmega328P controllers, an easy to use IDE, a C dialect called Arduino (language), and many libraries for sensors and actuators. Newer models are based on ARM Cortex M0 controllers, the IDE accommodates many 3rd party boards, most prominently the ESP8266 microcontroller family. Arduino is used worldwide.

CircuitPython

CircuitPython [13] brings Python programming to ARM based controllers, the IDE can be any text editor. Setup and deployment are easy, using the USB mass storage profile. Adoption is growing.

[13] https://learn.adafruit.com/welcome-to-circuitpython/what-is-circuitpython

mBed

mBed [14] is an embedded development ecosystem for C on ARM controllers, including an online IDE.

[14] https://www.mbed.com/

Makecode

Makecode [15] is a simple, kid-friendly, block or code based IDE for education, e.g. with the Micro:bit.

[15] https://www.microsoft.com/en-us/makecode

Microcontroller Models

ATmega328P

Popular through the Arduino Uno, this 8-bit controller has not enough resources for IoT applications.

ARM Cortex M0/M3/M4

The ARM Cortex M family is used for beginner hardware as well as commercial connected products.

ESP8266 and ESP32

The ESP controller family provides Wi-Fi connectivity at a low price and is popular with makers.

Prototyping Hardware Form Factors

Adafruit Feather

A small, breadboard-friendly form factor introduced by Adafruit. All feather boards include a JST LiPo battery connector and charging circuit. Many microcontroller models and connectivity options, as well as a number of hardware extensions, "feather wings" (or in general "shields") are available [16].

[16] https://www.adafruit.com/feather

Arduino MKR

A new Arduino form factor [17], similar, but not compatible to Adafruit Feather. Few available shields.

[17] https://store.arduino.cc/arduino-genuino/arduino-genuino-mkr-family

Arduino Uno

The iconic form factor for prototyping hardware, very accessible and breadboard-friendly, as well as allowing hardware extensions through shields. As a downside, shields can be rather expensive.

M5Stack

A new, modular, open source hardware device form factor based on ESP32 microcontrollers [18].

[18] https://www.m5stack.com/

Wemos

A small, simple form factor, stackable shield "standard" for ESP8266 NodeMCU microcontrollers.

Connectivity Options

The Raspberry Pi and many microcontroller boards come with built-in connectivity.

Personal Area: Bluetooth Low Energy (BLE)

Used in many current-day connected products together with smartphones and other BLE gateways.

Local Area: Wi-Fi Connectivity

Popular in current-day maker projects, but rather energy intensive for many IoT device applications.

Wide Area: LoRaWAN

Often added as an external module or shield, requires a LoRaWAN gateway in reach (2-15 km).

Sensor Wiring "Standards"

The sensor wiring "standards" below were considered in addition to device / extension form factors.

Breadboard and Jumper Wires

Ubiquitous, cheap and easy to use for temporary assembly with Fritzing-style [19] wiring instructions.

[19] http://fritzing.org/

Grove

Easier to use than a breadboard, permanent connections, but more expensive sensor modules [20].

[20] http://wiki.seeedstudio.com/Grove_System/

Screw Terminals

Permanent connections, less easy to use than a breadboard, requires additional hardware.