

# Various controllers in comparison

Jan Weber  
*Oachkatzlschwoaf*

*Elektronik und Technische Informatik*

*HTBLuVA*

St. Pölten, Austria  
jan.weber@htlstp.at

Flora Ledermüller  
*Oachkatzlschwoaf*  
*Maschinenbau*

*HTBLuVA*

St. Pölten, Austria  
flora.ledermueller@htlstp.at

**Abstract**— The aim of this paper is to compare between the technology used on our robot and other related projects. This example compares the Wombat, the ESP32 Dev Module and the Arduino, listing the advantages and disadvantages for building a robot. Technical facts as well as features are referred to. The ESP32 Dev Module can be used for the control of sensors because it is small and cheap. For robots the Wombat controller is more suitable. It offers performance and is easy to use with motors and sensors, which results in a higher price.

**Keywords**—Wombat, Arduino, ESP32, Controller

## I. INTRODUCTION

There are many different controllers such as the Wombat controller used on our robot, or as other examples: the Arduino and the ESP32. Due to the different controllers, there are various advantages and disadvantages between them. The different application possibilities are also interesting and will be explained. Among other things, the following are compared: mode of operation, performance, design...

## II. CONTROLLER EXPLANATION

To be able to build a robot, you also need technical hardware apart from the components. The so-called "controller" serves as the control unit of the entire robot and brings it to life. They are not only powerful enough, but also have many other advantages and programming options, more on this in the point Wombat Controller.

## III. THE PERFECT CONTROLLER

Each controller has advantages and disadvantages, and the choice is large. So, it is usually not easy to make the right choice. Even though the differences are usually not great, not everyone is equally suitable. First, you should think about the purpose for which the controller is needed. Does it have to be able to do something specific, for example control sensors?

All this plays an important role in the assembly of the robot. Since 2020, the ECER has been using the "Wombat Controller", for more information see chapter "V. / A. Wombat Controller".

## IV. THE DIVERSITY OF ROBOTS

Robots are not only getting better and better but are also a topic for the future. It is often hard to imagine the economy without them and they can do incredible things. They are perfectly programmed to do what they are supposed to do and are already replacing jobs. [1]

They are to be seen in different areas, one sees them in the medicine or in the car industry, they play an important role by their variety. To what extent this will look in the future and what advantages or disadvantages this will have remains to be seen. The fact is, robots will become an important part of our future. Not only that, but robots can also be used for various competitions, for example at the Open Competition at the European Conference on Educational Robotics (ECER). This will engage students to improve their STEM-skills resulting in long-term improvements in this field of research.

## V. COMPARISON OF THE CONTROLLERS

### A. Wombat Controller

#### 1) Explanation

The Wombat is a Linux-based controller based on a Raspberry PI 4 with 4GB Ram and an ARM architecture. It has been used as a controller for the robots at ECER since 2020. This controller has to be used for the Botball competition. You can see the Wombat with the display in Figure 1.

#### 1) Advantages of the Wombat Controller

On the one hand, it has a powerful processor despite its compactness, which allows the robot to implement commands as quickly as possible. The other advantage is the user-friendliness of the controller, as it enables easier and faster programming in conjunction with the display and the pins on the controller. Thus, settings can be made directly, for example, on various sensors. The controller allows to set motor values manually by using the touch display. Sensor values can easily be seen on the screen. [2]

## 2) Disadvantages of the Wombat Controller

As great as the controller is, it also has some drawbacks. The biggest disadvantage is the high price. Another is the size including the display. While this has the advantage of easier programming, it can become a disadvantage when the robot has to be compact. Another disadvantage is that additional sensors are difficult to include.

One big problem is the usage of servos. If servos are blocked they draw too much current resulting in triggering the fuse. If this happens too often, the resettable fuse has to be replaced. [2]

## 3) Software & Programming

The "KISS WEB IDE" (KIPR Instructional Software System Web Integrated Development Environment) is used for programming. This is developed by the "KISS Institute for Practical Robotics" for robotics training and offers an optimal possibility to bring the robot to life through its simple and understandable development environment. In addition, direct settings can be made via the display. The programming language for the entire robot can be either C or Python, this offers especially beginners in textual programming a good introduction to programming. [3]

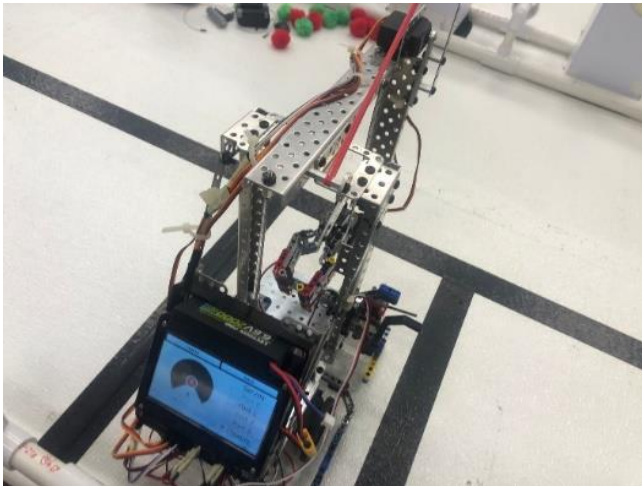


Fig. 1: Wombat controller on a robot

## B. ESP32 Dev Module

### 1) Explanation

The ESP32 Dev Module is compact and relatively powerful for its size. In addition, it is a low-cost-controller and has a low power requirement, which means that it is mostly used for smaller applications. [4]

The ESP32 has several variants, in this case the "ESP32 Dev Module" is explained.

## 2) Technical Data

The ESP32 Dev Module has several technical features: [5]

- Dual high performance Xtensa 32-bit LX6 CPU cores, clocked up to 240MHz and 512Kilobyte SRAM
- WiFi with 2.4GHz
- Bluetooth
- Multiple peripherals
- Low power consumption

For many it is the first choice because of the WIFI and Bluetooth modules.

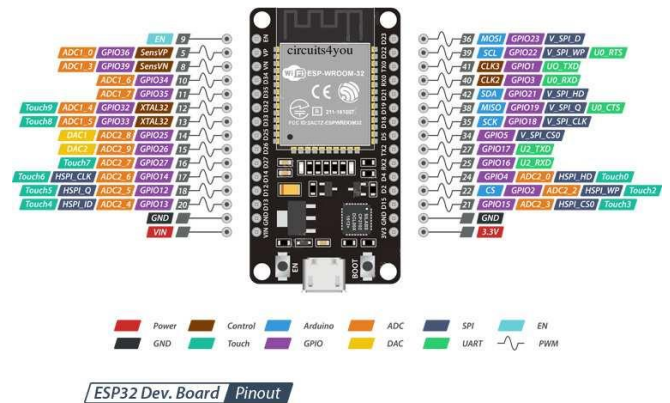


Fig. 2: ESP32 Dev Module Pinout [6]

- In total, the ESP32 DEV Module has 34 GPIO pins



Fig. 3: ESP32 Chip with WIFI and Bluetooth

### 3) Advantages of the ESP32 Dev Module

Not only is the ESP32 Dev Module small and compact, but it is also powerful enough for many applications.

As already mentioned, it has Bluetooth and WIFI integrated, so it has many application possibilities and is often in use. Furthermore, it also offers the perfect entry into electronics, as it can be individually configured through its GPIO pins.

#### 4) Disadvantages of the ESP32 Dev Module

As great as the microcontroller is, it also has some disadvantages. As already mentioned, it is small and basically comes without a case, which is not an advantage for some applications. However, the much bigger point is that it is not powerful enough for larger projects which need more calculation power and more pins. Therefore, it is often used for smaller electronic projects.

#### 5) Software & Programming

Programming can be done differently with the ESP32 Dev Module, but mostly in the C or C++ programming language. How it is programmed concretely is up to everyone. In some cases it is the software "Arduino IDE" or "Visual Studio Code".

It also offers additional connection options through its GPIO pins, these can be easily controlled by the software. [7]



Fig. 4: ESP32 Dev Module

### C. Arduino Uno R3

#### 1) Explanation

The Arduino is in principle quite similar to the ESP32 Dev Module, it is also a microcontroller with some features. Essentially, they differ in the technical data and features.

Meanwhile there are many variants from different manufacturers to the Arduino boards. For comparison we take the Arduino UNO R3 from Arduino.

For many, "Arduino" is well known, at the latest when you are more involved with microcontrollers and programming.

It is the perfect board to get started in electronics, plus they have their own software and documentation on programming. [8]

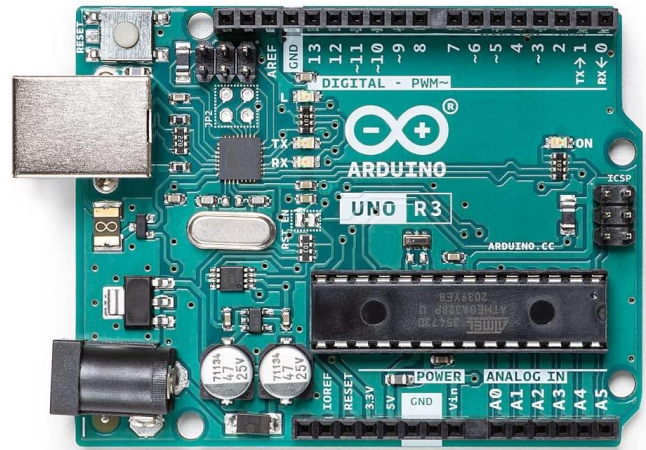


Fig. 5: Arduino Uno R3

#### 2) Technical Data

The Arduino R3 is based on an 8-bit ATmega328 processor. The CHIP can be easily replaced because it is not soldered to the board. Another special feature is the 1kb of EEPROM, this is not erased when the power is turned off.

In addition, the microcontroller can be easily powered with a 9V battery.

It has a total of 14 GPIO pins, 6 of which can be used as PWM outputs, other features:

Memory data:

- AVR CPU with up to 16 MHz
- 32KB Flash
- 2KB SRAM
- 1KB EEPROM

Peripherals:

- 2x 8-bit Timer/Counter with a dedicated period register and compare channels
- 1x 16-bit Timer/Counter with a dedicated period register, input capture and compare channels
- 1x USART with fractional baud rate generator and start-of-frame detection
- 1x controller/peripheral Serial Peripheral Interface (SPI)
- 1x Dual mode controller/peripheral I2C
- 1x Analog Comparator (AC) with a scalable reference input
- Watchdog Timer with separate on-chip oscillator
- Six PWM channels
- Interrupt and wake-up on pin change

#### 3) Arduino IDE & Programming

The Arduino is not only the finished board, but also offers an additional extensive software, the "Arduino IDE". About it, the microcontroller can be programmed quite easily, without installing additional things before. In addition, the software has another advantage: Libraries and sample codes. Through

the extensions of libraries that anyone can create and add themselves, the software offers a comprehensive range. Furthermore, there are sample codes to facilitate the programming process.

As with the ESP32 Dev Module, the programming language is mostly C or C++. It does not have to be the software from Arduino, the microcontroller can also be programmed with Visual Studio Code or other programs and therefore offers coding in teams.

#### 4) *Advantages the Arduino UNO R3*

A big advantage to many other microcontrollers is the replaceable chip, you do not have to trigger it before, but can easily take it out. Another plus is the easy supply with a 9V battery, so it can be used everywhere very easily. .

Finally, Arduino has a big community offering support in many questions or tasks.

#### 5) *Disadvantages of the Arduino UNO R3*

By default, the Arduino Uno R3 has no Wi-Fi or Bluetooth, although this can be retrofitted with an "Arduino Shield", but that causes extra costs.

It is larger than the ESP32 Dev Module and is therefore not suitable for some projects and because it has a smaller memory, other microcontrollers are used more often. Strong motors cannot be operated directly with the Arduino board, this is only possible when the Arduino gets more power additionally with a "Shield" or other external solutions.

### **D. Comparison – The perfect choice for the Open-ECER robot**

In the robotics competition, a robot must be able to react fast and execute commands quickly. Thus, not every controller is eligible.

When we compare the controllers, we especially notice the difference in the technical data. In the technical data, the controller must have enough power to bring the robot to life.

Likewise, it must be possible for the robot to be programmed by several people.

Therefore, we have several operations that the controller must fulfill:

- enough power for the robot
- user-friendliness of the programming
- enough connections for sensors and motors

#### 1) *ESP32 Dev Module*

The Esp32 Dev Module offers a lot of possibilities. It brings directly WIFI and Bluetooth and is more powerful compared to other microcontrollers. For the ESP32 Dev Module speaks also the small size and low power consumption.

The big disadvantage of the microcontroller is, however, that it comes without a housing and is not particularly strong for a robot. Its features offer a good possibility for smaller projects, but not for a robot.

It is not designed for programming by more people at one time, because the software IDE is in most cases "Arduino IDE" and no direct sharing of the codes is possible.

In the future, it would not be advantageous to use this microcontroller for the main control of a robot, but for smaller auxiliary controls, such as controlling a specific sensor.

#### 2) *Arduino*

The Arduino is quite similar to the ESP32 Dev Module but does not have Wifi and Bluetooth integrated. As with the ESP, this can be easily powered, additionally directly with a 9V battery.

The greatest aspect is the heart of the controller, the chip is easily replaceable and does not need to be desoldered first.

A big advantage of the Arduino are expandable "shields" which are very easy to attach. The problem here is that the Arduino, like the ESP32, is more suitable for small projects and has little power for a robot.

The recommendation here is the ESP32 Dev Module for additional sensors before you reach for the Arduino.

#### 3) *Wombat & Future – The best combination*

The Wombat combines all aspects together and offers extensive features. Because it is based on Linux, the controller can be programmed by several people at the same time via the "KISS Web IDE". The Wombat is much more powerful and has enough GPIO pins to control multiple sensors at the same time. Sensors, not supported by KIPR are hard to implement. Concerning the power consumption, it needs much more power than the ESP32 Dev Module or the Arduino.

With an extra housing including a display, the Wombat has a big advantage, because it can be configured not only via WIFI, but also directly via the touch display. The biggest disadvantage is its high price. Another negative aspect is its size, but that usually doesn't matter that much.

In summary, the Wombat has a mixture of the ESP32 Dev Module and Arduino with much stronger performance, including a display. It is therefore not absolutely necessary to use an extra microcontroller to control some sensors. Currently the controller is the perfect choice.

## **VI. REFERENCES**

- [1] B. Costanich, "Article Microcontroller "The Future is Tiny"," 05 09 2018. [Online]. Available: <https://medium.com/@bryancostanich/the-future-is-tiny-44dea02e4517>. [Accessed 04 03 2023].
- [2] "KIPR Wombat "Libwallaby Docs"," [Online]. Available: <https://www.kipr.org/doc/index.html>. [Accessed 04 03 2023].

- [3] “KIPR Wombat "Hardware & Software",” [Online]. Available: <https://www.kipr.org/kipr/hardware-software>. [Accessed 04 03 2023].
- [4] “ESP32 Dev Module "Information & Datasheet",” [Online]. Available: <https://www.az-delivery.de/en/products/esp32-developmentboard>. [Accessed 04 03 2023].
- [5] “ESP32 Dev Module "Docs",” [Online]. Available: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/get-started-devkitc.html>. [Accessed 04 03 2023].
- [6] “ESP32 Dev Module "Pinout",” [Online]. Available: <https://circuits4you.com/2018/12/31/esp32-devkit-esp32-wroom-gpio-pinout/>. [Accessed 04 03 2023].
- [7] “Arduino IDE "Software",” [Online]. Available: <https://www.arduino.cc/en/software>. [Accessed 04 03 2023].
- [8] “Arduino UNO R3 "Information",” [Online]. Available: <https://docs.arduino.cc/hardware/uno-rev3>. [Accessed 04 03 2023].