Developing a comprehensive Augmented Reality-based Software Platform for Educational Embedded Systems

Vlad Bartolomei
dept. of Computer Science
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
vladbartolomei@gmail.com

Victor Bâcu
dept. of Computer Science
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
victor.bacu@cs.utcluj.ro

***Abstract*—As the number of embedded system-devices has an exponential growth in the recent years, the works in-behind are still hard to teach and illustrate by teachers. 2010s have brought augmented reality in the game, a new technology whose main qualities are visualisation enhancement in real environment and interactability with the objects. We propose combining of two of the most used technologies in today's world by using for illustration purposes the best documented board from Arduino by visually expanding the datasheet: the Uno R3. We developed a software platform based mainly on visualisation with apps for both teacher and student so that a sense a collaboration between these two entities would be still possible.**

***Index Terms*—Augmented Reality, Model Target, Image Target, Vuforia, Engineering Education, Arduino Uno**

DOI: 10.37789/icusi.2025.19

I. Introduction

During the last years a significant technological advancement was met, resulting in an increased number of IoT (Internet of Things) devices in a range of different domains. This devices are called *embedded systems* and represent a hardware computer running a software program designed to fulfil a very specific task. Embedded systems can be seen everywhere, e.g. routers, printers or smart light bulbs. One can tell them apart from computing systems (i.e. computers, laptops or even smartphones) whose main characteristic is versatility with respect to tasks that can be fulfilled (they are general-purpose devices); while nowadays computers run on 64 bit processors, microprocessors powering the embedded systems can be even simpler, running on 8- or 16-bit architecture (32- or 64-bit processor architectures are nevertheless possible) [1].

In 2024, number of smartphone users is estimated around **4.48 billion** people, while the number of embedded systems reaches, estimatively, **18.8 billions** users [2], [3]. According to [4], "Today, there are billions of embedded system devices used across many industries including medical and industrial equipment, transportation systems, and military equipment. Many consumer devices from digital watches to kitchen appliances and automobiles also feature them. Embedded systems are small, fast, powerful, and designed for very specific use cases. While general purpose systems can perform multiple functions, they can be too costly for many applications, and may also fail to measure up to embedded systems' reliability, low power consumption, minimal size, and other functional and performance features". Specifically due to this presence in our world, embedded system architecture and programming are taught and studied in specialty faculties, alongside with project development based on these systems.

Alongside with the technological advancement comes the shift in learning paradigm which is headed towards digitalisation and evolves from professor-centric to student-centric. Students' behaviour changed under the influence of electronic devices, which became more versatile and, therefore, started to be used as main learning tools [5]. Students learn through experimenting, and Augmented Reality technology (in the following referred as *AR*) is today capable of offering this **visualising** and **interaction** frame. According to [6], AR technology is capable of offering mostly in scientific domains enthralling learning experiences. This study focuses on the effect of the AR-based learning model to demonstrate a synergy approach between professor and its students by visually extending the Arduino anatomy's documenting.

A. Augmented Reality

AR is a technology by which virtual elements are brought in the real world. AR is part of the *mixed reality*'s family combining in different proportions pieces of reality with virtuality. The surroundings are completely natural, and "classical computers are isolated to their physical environment. They only need electrical power and instructions given by human operators on keyboards" [7].

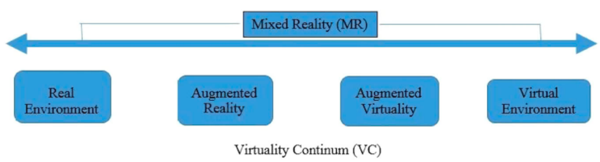

Fig. 1. Milgram's virtuality and reality continuum (taken from [5])

To classify a system as Augmented Reality, it has to check three conditions: (1st) **combining** of real and virtual information; (2nd) real-time **interactivity**; (3rd) **3D** space operability [7].

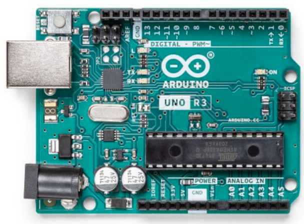

Fig. 2. Arduino Uno R3 microcontroller (taken from [8])

In the 2010s AR became more and more present, as an increasing number of devices evolved in this direction as well. More and more domains are using AR technology as medicine, industry, entertainment, retail. At the same time, education makes the perfect environment for AR using: "low-cost devices, setup, and learning environments developed to teach students from kindergarten to university level". In the same decade, following a study, by using augmented reality in education environments a series of imporvements are observable, such as "learning gain, student motivation, improve spatial ability, and user experience with the help of visualization and animation of AR" [5].

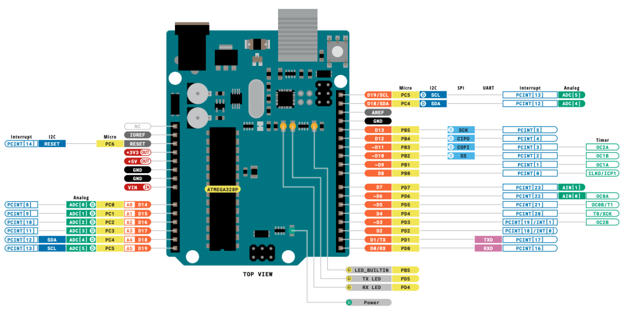

Fig. 3. Advanced pinout of Arduino Uno R3 (taken from [9])

B. *Arduino Uno R3*

Arduino is an italian company that designes, manufactures and supports electronic devices and software. Arduino products are simple, yet powerful, meeting the needs of a wide range of people, from students to professional developers. Among the selection of boards Arduino produce, the model *Uno R3* is the most robust, used and documented of them all. It's based on the **ATmega328P** chip. The board also has, among others, 14 input/output digital pins, 7 analog input pins, a 16 MHz ceramic resonator for clock generation, USB connection and an AC-to-DC electricity converter. If anything, the **ATmega328P** is cheap and easy to replace, as it is not soldered to the board. A characteristic the board has is the *program memory* working with EEPROM (*Electrically Erasable Programmable Read-Only Memory*) technology which remains unerased even when the power cable is plugged out [5], [8], [10].

Arduino provides in [8] an extensive documentation, schematics document, datasheet, CAD (*Computer Aided Design*) files containing 3D models of the board, getting-started tutorials as "Analog Input" or "Controlling PWM Output with a Potentiometer via MATLAB", compativbel shields and pieces of software necessary for project development.

II. Related Work

The use of Augmented Reality (AR) in educational contexts has seen a significant increase in recent years, particularly in STEM (Science, Technology, Engineering, and Mathematics) education. In [6] there has been a highlight of the benefits of AR for enhancing students' motivation, engagement, and conceptual understanding.

In the context of engineering education, AR has been successfully integrated with physical components such as microcontrollers to improve students' understanding of hardware architecture. One such example is presented in [5], where students interact with a virtual representation of a microcontroller while manipulating real-world sensors and actuators. The study emphasizes the advantages of visualization and interactivity provided by AR systems in understanding abstract concepts.

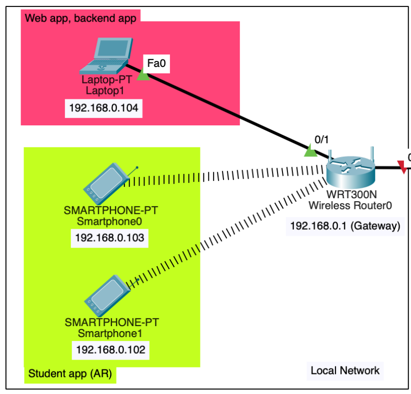

Fig. 4. Local newtork in which a laptop and two mobile stations communicate in infrastructure mode, by router, based on IP addresses

Furthermore, platforms such as Unity and Vuforia have been employed in various academic projects to develop AR applications that visualize electronics components or simulate circuit behavior. For instance, [7] presents a framework where users interact with an AR-enhanced environment to explore circuit design and signal flow in real time.

While existing works focus on general visualization of hardware or educational games, few of them target a structured and pedagogically integrated AR platform for component-level exploration. This paper proposes an AR system dedicated to supporting professors in presenting electronic components—specifically, the Arduino Uno R3—through a combination of model targets and image targets, offering both spatial and contextual understanding.

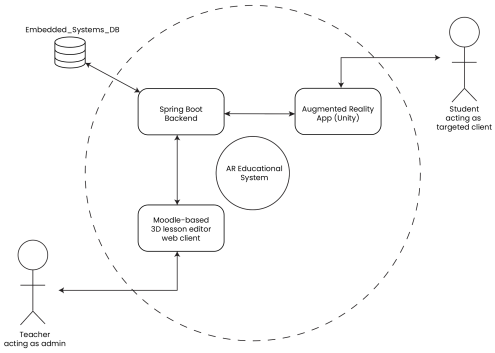

Fig. 5. Conceptual diagram of the platform

III. The Proposed Learning Platform

A. The problem

As extensive as Arduino Uno R3's documentation may seem to be, it has one issue that no one's addresing: it's esoteric and distributed. Take, for once, the pinout depicted in figure 3; while it mentions exactly technical specifications, you have to go through different other documents to determine the use of a certain pin. Moreover one could get confused with the different columns reffering the same pin.

Not to mention that Embedded Systems curricula tend to differ. Some instructors tend to be more inclined to teach the use of microcontrollers in a microprocessor-centric approach, meaning that students would have to control peripherals (for one) only to embedded-C like instructions [11]. For that professors will provide full documentation of used microprocessor. If we're talking about ATmega328P, the datasheet only has 294 pages [12], but for faculties that want to use a more performant microcontroller as Arduino Due is, you're looking at over 1400 pages of datasheet [13]!

B. Method. Concept development

We propose closing the gap of misunderstanding from both sides. Therefore we will develop a system consisting of four main components: a mobile system to run an AR-powered application for a student who wants to learn about microcontroller's components, a web system presenting the teacher with the same board that the student can visualise, allowing the instructor to select certain components and create specific lessons, a backend system whose task is to receive lessons from teacher and to persist them into the fourth component, a local database (DB); when asked by the student app, the backend will fetch the lessons from the DB.

The proposed platform will need no connection to the WAN (Wide Area Network, the Internet). Instead, all devices will need to run simultaneously in the same environment, that is one local network. In figure 4 a classroom/laboratory is depicted, with a router defining the local network in which devices run; in this case, the **laptop** represents the professor web app, the backend app and the database, while the smartphones represent student app. In order for students to access the resources, the teacher running *the laptop* needs to provide them with his/her IP address in the local network.

C. Design Prototyping

The next step following the idea outline is defining what each user can do with its application. Moreover, there has to be a definition for a lesson's format. For that we chose to create a JSON file, since it's easy to read and edit through code and there are several libraries that allow JSON parsing and editing, if not in-built mechanisms (the case of Typescript). Closely related to the lesson's JSON file format, we have to define the DB's structure so that lesson's persistance is allowed.

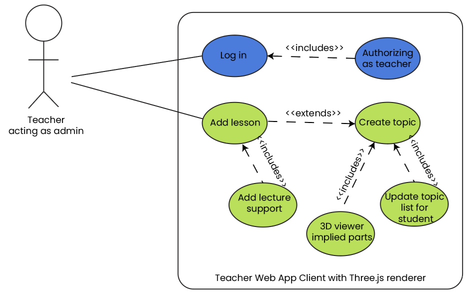

Fig. 6. Teacher use-case diagram for web application

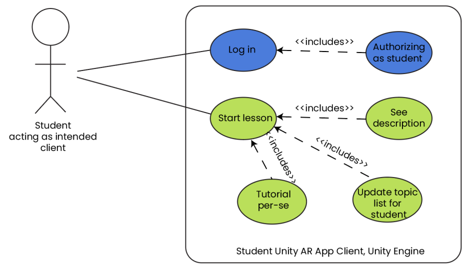

Fig. 7. Student use-case diagram for mobile AR application

The JSON file format for a lesson is an inch-perfect replica of the *Wizard* and *WizardPage* entities from our database schema, so that mapping in backend app would pose no problem:

```
{
    subtitle: "subtitle of the lesson;
        field is named subtitle due
        to future scalability reasons,
        since a lesson may have more
        subsections and, consequently,
        more subtitles",
    aim: "general aim of
        the current lesson",
    pages: {
```

```
        1: {
            target: "name of the described
                component on board",
            description: "custom text
                given by teacher to
                explain target",
            action: "one-word descriptive
                action, as ATTACH,
                SET and so on"
        },
        ...
        n: {...}
    }
}
```

This JSON will be created in web application, transmited to the backend server through an HTTP POST request (attached to its body) and persisted in the DB. The backend follows a *controller-service-repository* architecture, the usual choice for web application backend-side server.

IV. Results and Validation

A. Workflow

The platform is designed to be used in university environments, therefore every person that is allowed to access these resources has to be able to be accountable, i.e. access is granted by authenticating (see fig. 8).

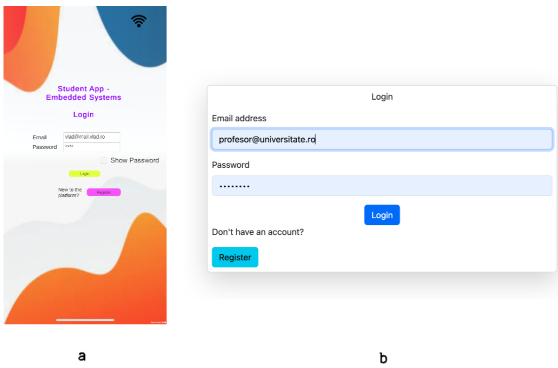

Fig. 8. Login page for (a) student application and (b) teacher application

After login, both teacher and student are granted the access to their respective apps. The student can see all the lessons persisted in the local database and the teacher can review (the same) lessons already created (see fig. 9).

The teacher will then create a new lesson. He has access to a page where he can see the Arduino Uno R3 board. By point-and-click, he can select any component on the board he likes and add it to the left-hand side of the page to a step creator. Each step is represented as a block (similar to Scratch) and more steps can be connected and reordered (see fig. 10).

On the other hand, the student can choose a lesson to follow. He can click on the refresh button to check if there are any new lessons, as there is no self-updating mechanism implemented.

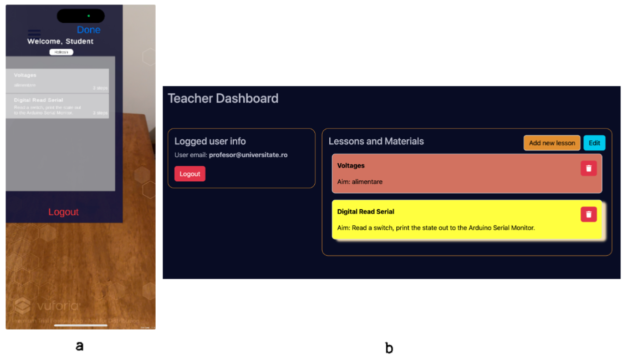

Fig. 9. Lesson listing for (a) student application and (b) teacher application

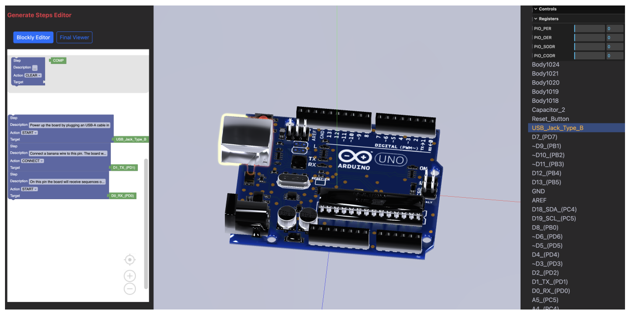

Fig. 10. Teacher web app's 3D lesson creator

But first he has visualise the Arduino Uno R3 board. For that he is given two alternatives (fig. 12):

- either track a physical Arduino Uno R3 board – typically this happens in a laboratory environment, where each working table has a board prepaired
- or in absence of a physical board, detect the QR code (see fig. 11) – this feature guarantees the student accesibility to the board.

Fig. 11. QR code used for marker-tracking as Image Target in Student App.

B. Usability

When writing the project proposal of this platform, five SMART objectives were formulated. Among them one is relevant for this subsection: *platform testing with the help of at least two persons verifying the features and user experience during a week.* Later on I created an usability questionnaire (accessible here:

Fig. 12. Student is following a tutorial. Left: real board visible. Right: virtual board spawned after scanning the QR code

https://forms.gle/zHmbnpmuZZags2Rf7), available in the native language of the authors only (romanian), which I asked my persons to complete. The survey consisted of four sections – demographic data, web application, mobile application, open feedback. Sections two and three contained 20, respectively 15 system's ease of use confidence on a scale on 1 to 5, "Strongly Disagree" to "Strongly Agree"; there were also a few open-answer questions.

The questionnaire was completed by two people. Both respondents of this questionnaire have ages between 18 and 26 years, are highly experimented with the technology and came across with at least one AR app before. Regarding the mobile applications, both the iOS and Android versions were tested. The *Score* column values are obtained by adding the respondents' values. Given that the number of respondents $R = 2$, the maximum value for each question would be $N * 5 = 10$, while the minimal value is $N * 1 = 2$. Therefore, computing the usability percent for each question (and for totals) will have to follow this formula:

$$usability_{\%} = \frac{sumOfAllRespondentsValues}{maximumValue} * 100 \quad (1)$$

Open questions revealed interesting pieces of information as well. When asked "*what purpose **do you think** the platform has*", respondents reported that the platform plays an educational role in learning the components of an Arduino board. While testing as a student, one of the respondents noted that by having this setup in place experimentation space doesn't limit to the laboratory room.

TABLE I
USABILITY SURVEY QUESTIONNAIRE ASSESSMENT

S.No	Questions	Score
	Web Application Usability Questions	
1	The **login process** is simple and quick	10
2	The **registration process** is simple and quick	9
3	**Viewing** lessons in the dashboard is simple and intuitive	10
4	**Adding** a lesson to the dashboard is easy to find and intuitive	10
5	**Editing** an existing lesson in the dashboard is easy to find and intuitive	10
6	**Deleting** an existing lesson in the dashboard is easy to find and intuitive	7
7	The dashboard interface is intuitive	9
8	The editor interface is clearly structured and intuitive	10
9	The central panel is easy to adjust in terms of camera position and distance to the object	10
10	The object is easy to view and there are no visual impediments	10
11	The scene (central panel) lighting is sufficient	9
12	The scene lighting does not disturb the user at all	10
13	The Arduino board components are easy to select by **clicking on the central panel**	9
14	The Arduino board components are easy to select by **clicking in the Inspector list**	8
15	A selected component is clearly identified and easy to view	10
16	**Creating** a new step is easy and intuitive	8
17	**Modifying** a step is easy and intuitive	9
18	**Deleting** a step is easy and intuitive	9
19	**Reordering** steps is easy and intuitive	10
20	The final script can be seen and matches what you have entered (subtitle, aim, steps)	10
21	The final script can be seen along the way, can be edited, and is responsive to each action	9
	Usability percent in total for web application - 93.33%	
	Mobile AR Application Usability Feedback	
22	The instructions and process for setting the correct IP address for the server are clear and easy to follow	9
23	The login form is clear and easy to use	7
24	You understand the purpose of the Register button, it is easy to locate and works as expected	9
25	At first glance, the interface is intuitive and easy to understand	8
26	Existing lessons are easy to preview and you know what to expect (goal) when clicking on one	8
27	If the teacher has created a new lesson and informs you, the refresh button applies the changes as expected	10
28	The application correctly identifies the real Arduino board and overlays a transparent 3D model smoothly	10
29	The application correctly identifies the QR code substitute for the Arduino board and overlays an opaque 3D model smoothly	10
30	When following a lesson, the steps are clear and easy to read	10
31	During a lesson (whether with a real or virtual board), the current component is clearly highlighted	10
32	You can easily find and use the button that temporarily hides the instructions ("minimize bar") and you can return to them	10
33	The logout process works as expected	10
34	Navigating through tutorial steps is comfortable	10
35	This application helps more than traditional methods	10
36	The application works technically without problems	10
	Usability percent in total for mobile application - 94.0%	
	Total Usability - 93.61%	

V. Future work

On the survey (see IV-B) a few open-ended questions as well as some 1-to-5 confidence questions raised some issues that have to be addressed: on mobile app (see 8-a) login only works if the correct IP address of the server is introduced and saved, WiFi button is a bit unintuitive, and the interface is static (both 8-a and 9-a), meaning that on some phone models text or icons might be cut off. On the web application the only obscure bit is how do you create a new step, as you have to right-click *in Inspector side* and then select add Target to steps editor. Survey revealed that in these cases more instructions should be displayed and, I would add, a name convention is to be established (target means component on the board).

In its current form, platform can only be used in the laboratory conditions as one needs a local network and teacher's lessons are persisted most probably on his/her laptop (which also has to run two applications – backend and web), which isn't helpful if a student wants to follow the lessons at home. Therefore the backend app (and web app) should be hosted online or at least on the university's servers.

Regarding platform's features, a lot of upgrade opportunities arise. In the future, both the student and teacher could select what board to use and link lessons to a board by a board ID (persisting this information in the DB). A student could connect more embedded system components (like wires, water pumps, servo motors and so on) to simulate an embedded systems real-world application and each component would have its own 3D models, animations and behaviours. For that the `action` field (see subsection III-C – lesson's structure in JSON file format) would make a difference, as teacher can already indicate precisely how should components be manipulated; the client applications (web and mobile) could benefit of an inventory system where all these components would be stored and placed in the scene when needed, similar to *Cisco Packet Tracer* application. More in-depth behaviour of the microprocessors could be explored by simulating actions of the effects on modifying certain internal registers or Timer's manipulation could be possible, seeing in real time how the clock is oscillating and programming actions on certain moments (different rising/falling edges on different periods); something similar is depicted in the article [5]. As a follow-feature, the mobile application could generate the C-embedded code necessary to make the simulations work in a real life setup, customized for the used board.

VI. Conclusion

Development of the software platform for educational activities in computer programming and electronics has been a comprehensive endeavor that encompasses several key aspects. Through rigorous design and testing, we have created a user-friendly interface that facilitates hands-on learning experiences for students exploring embedded systems, particularly those utilizing Arduino boards. To encourage the learn-by-doing approach in educational environments we created alongside with the student app a professor app in order to allow having customized lessons and experiences. By that the teacher still has control over the presented concepts and ideas and he/she can use this platform as a tool to enhance concept illustration.

The usability survey has provided valuable insights into the platform's strengths and areas for improvement, which will be addressed in future updates. The findings from this study demonstrate the potential of our software platform to revolutionize the way students learn about embedded systems, providing a tangible connection between theoretical knowledge and practical application. By continuing to refine and expand its features, we aim to further enhance the learning experience, making it more accessible, effective, and enjoyable for students worldwide.

Ultimately, the success of this project relies on the continued support and engagement from educators, researchers, and enthusiasts who share our passion for promoting STEM education through innovative technology solutions.

References

[1] Arm Ltd., "Embedded systems," https://developer.arm.com/documentation/den0013/d/Introduction/Embedded-systems, n.d., [Accessed: 2025-06-04].

[2] "How Many People Have Smartphones in 2024?" https://www.oberlo.com/statistics/how-many-people-have-smartphones [Accessed 2025.06.14]. [Online]. Available: https://www.oberlo.com/statistics/how-many-people-have-smartphones

[3] "State of IoT 2024: Number of connected IoT devices growing 13% to 18.8 billion globally," https://iot-analytics.com/number-connected-iot-devices/ [Accessed 2025.06.04]. [Online]. Available: https://iot-analytics.com/number-connected-iot-devices/

[4] "What Is Embedded System Design (ESD)?" https://www.arm.com/glossary/embedded-system-design [Accessed 2025.06.04]. [Online]. Available: https://www.arm.com/glossary/embedded-system-design

[5] A. Kumar, A. Mantri, and R. Dutta, "Development of an augmented reality-based scaffold to improve the learning experience of engineering students in embedded system course," *Computer Applications in Engineering Education*, vol. 28, no. 2, pp. 320–331, 2020.

[6] R. Zuo, L. Wenling, and Z. Xuemei, "Augmented reality and student motivation: A systematic review (2013–2024)," *Journal of Computers for Science and Mathematics Learning*, vol. 2, no. 1, pp. 38–52, 2025.

[7] J. M. Ariso, Ed., *Augmented Reality: Reflections on Its Contribution to Knowledge Formation*, ser. Berlin Studies in Knowledge Research. Walter de Gruyter GmbH, 2017, vol. 11.

[8] "UNO R3," https://docs.arduino.cc/hardware/uno-rev3/ [Accessed 2025.06.29]. [Online]. Available: https://docs.arduino.cc/hardware/uno-rev3/

[9] "Arduino Uno R3 – Full Pinout," https://docs.arduino.cc/resources/pinouts/A000066-full-pinout.pdf [Accessed 2025.06.29]. [Online]. Available: https://docs.arduino.cc/resources/pinouts/A000066-full-pinout.pdf

[10] "About Arduino," https://www.arduino.cc/en/about/ [Accessed 2025.06.29]. [Online]. Available: https://www.arduino.cc/en/about/

[11] D. E. Bolanakis, "A survey of research in microcontroller education," *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, vol. 14, no. 2, pp. 50–57, 2019.

[12] Microchip Technology Inc., "Atmel ATmega328P – Datasheet," https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf, n.d., [Accessed: 2025-06-29].

[13] ——, "Atmel — SMART ARM-based MCU, SAM3X / SAM3A Series – Datasheet," https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-11057-32-bit-Cortex-M3-Microcontroller-SAM3X-SAM3A_Datasheet.pdf, n.d., [Accessed: 2025-06-29].