DIGITIZING PRODUCTS:
CREATING DEMONSTRATORS
FOR FUTURE EDUCATION

digi demo

Chair telemetry transponder

Demonstrator classification and documentation

Aalborg, February 2021

Co-funded by the Erasmus+ Programme of the European Union

Project consortium

UCN

Fagskolen i Viken

Universitatea Babeș-Bolyai Cluj-Napoca, Centrul Universitar UBB Reșița

FHV Vorarlberg University of Applied Sciences

ésta school of business & technology Belfort

Dissemination level

Code	Access granted to	
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Legal Disclaimer

About the DigiDemo project

Environmental challenges and digital transformation are two of the main drivers changing the world and the way business will be is done in the future. Therefore, it is essential to enable future employees to address these drivers. The skills and competences needed to develop digitalized products and awareness of the environmental challenges are therefore crucial for the European workforce and industry to continue being competitive in a future green economy and to maintain jobs across Europe.

The DigiDemo project addresses these challenges by developing demonstrators especially for higher education allowing to improve mainly mechanical engineering studies by integrating skills and competences allowing them to understand, develop and commercialise connected products. The results will be publicly available and can be used by every institution interested in integrating this type of training in their cursus.

1 Contents

Document authors

	First name Last name	Institution
Key author	Torben Momme Holm	UCN
Further authors	Steffen Rahbek Vutborg	UCN
	Mogens Holm Iversen	UCN
	Esben Skov Laursen	UCN

Revision history

Version	Date	Author(s)	Description
1.0	2021-02-16	Holm	Initial draft
1.1	2021-02-19	Vutborg	Adaptation to template and elaboration of technical setup
1.2	2021-08-24	Vutborg, Iversen	Description of Chair Telemetry Transponder added
1.3	2022-02-04	Holm	Sustainability awareness expanded.
1.4	2022-12-02	Iversen	Sustainability expanded with information from Transnational Meeting in Dornbirn
1.5	2023-03-20	Vutborg	Finalising electronics part.

Document status

Status description
Final version

Abbreviations

ESTA	ESTA Belfort (France)
FHV	Fachhochschule Vorarlberg (Austria)
FIV	Fagskolen i Viken (Norway)
UCN	University College Nordjylland (Denmark)
UBB	Universitatea "Babes-Bolyai" din Resita (Romania)
IMU	Intertial Measurement Unit
SBC	Single Board Computer
IoT	Internet of Things
DOF	Degree of Freedom

List of figures

List of tables

1 Introduction

This demonstrator focuses on collecting data on the use of office chairs, specifically how often and how rigorous the user moves about when using the chair.

It is hoped that the demonstrator in the future can generate data, that could have two distinct uses.

The first use could be to remind office workers to move around more, or to stand up from time to time, to avoid muscular fatigue and stress from prolonged positioning in the same place.

The second use could be in the primary school, to detect if some pupils move around significantly more than average. It is proposed, that this energetic moving about could be an early sign that the pupil is not feeling comfortable in the learning situation. Whether this moving about indicates simple loss of focus because the lesson has been too long, or if this could be an indicator for a medical or psychological condition, needs to be investigated further by experts in this field.

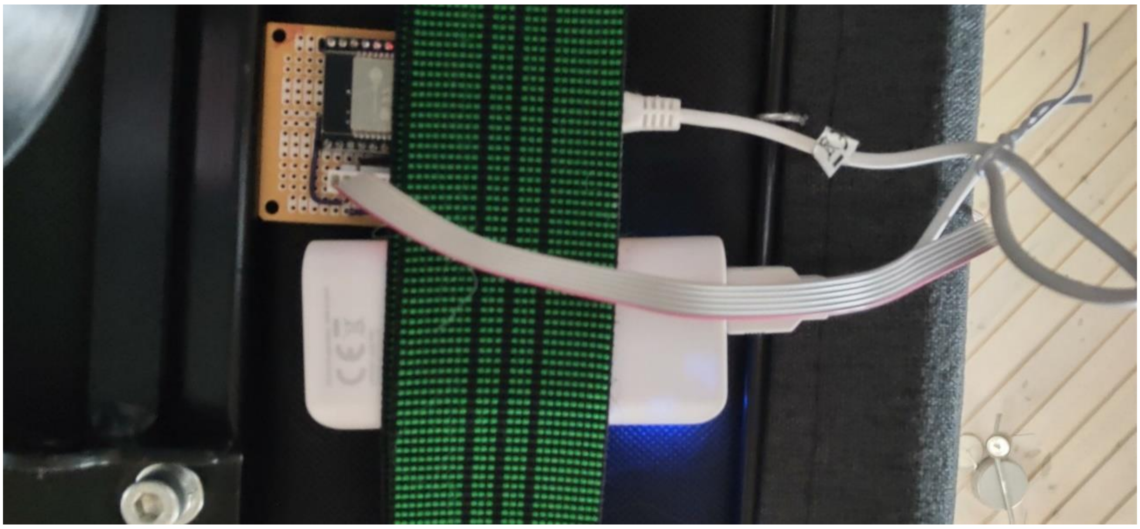
2 Overview

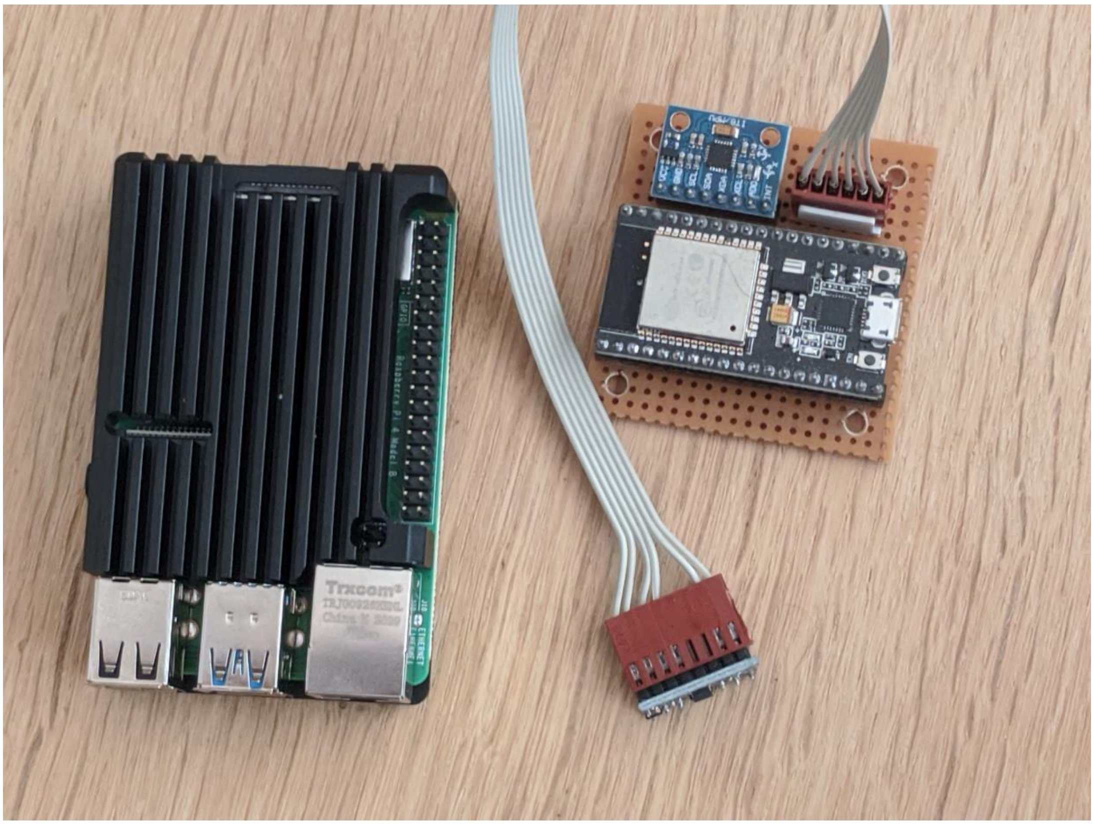
The key properties of the focus project are:

Table 1: Specification of key properties of the focus project

Key Property	Value
EQF level	5 (Technician)
Year of study	2
Domain	Mechatronics
Objective	Hands-on
Workload	5+ ECTS (depending on setup)
Keywords	Internet of Things (IoT), Inertial Measurement Unit (IMU), Micro Controller Unit (MCU), Distributed Programming, Embedded Programming

The demonstrator is centred around an embedded microcontroller connected to one or two 6-axis IMUs, and a LI-ion battery pack, as to not hinder the mobility of the char on which it is mounted with a tether. The embedded unit should be encased in an enclosure that will enable easy retrofitting to most existing office chairs. The microcontroller then collects acceleration and rotation data from the IMUs at a given sample interval, filters and pre-processes the collected data and transmits it to a Raspberry Pi[1] single board computer (SBC) over Wi-Fi. The raspberry pi functions as a central hub and handles connections to several embedded units, as well as collating and further analysing the data. The raspberry pi can also, if needed, retransmit to a cloud solution for intense processing, bulk storage or for connecting simultaneous datasets from several geographical locations. The demonstrator is thought to be interdisciplinary from mechanics over electronics to SW development.




Trxcom

3 Description of fulfilment of keywords/characteristics

The demonstrator will improve teaching, by showing digitalization and value-adding of a well-known object.

The demonstrator will be replicable, as most of the hardware will be based on readily available modules, and further instructions on system fitting will be provided.

The demonstrator will be interdisciplinary, covering the following fields: sensors for pickup of movement – electronics – edge devices for pre-processing data from gyroscopes – data transmission to cloud storage and further processing - mechanics for durable containment with focus on low investment.

Table 2: Description of fulfilment of demonstrator characteristics for the focus project

Characteristic	Description
Teaching improvement	The focus project allows the students to go through the entire process of development of a mechatronic product starting with requirements, through design, implementation, test and integration of both mechanical and electronic components with software. They work in interdisciplinary teams.
Sustainability awareness	In this project the sustainability awareness is seen in a broader context, relating to UN World goal 3, Good health and well-being, and to World goal 4, Quality education.
Replicability	The demonstrator is implemented using off the shelf components, which are easy to replicate.
Industry needs	An ever-increasing percentage of work is done from a desk, sitting on a chair for prolonged periods. This position is not optimum for the human body. By helping to reduce pain and malfunction in muscles and skeleton, this could be a useful tool, that could be implemented for less than the price of one sick-day.
Interdisciplinarity	The demonstrator requires both electrical and mechanical engineers to cooperate in implementing a mechatronic system. Software components are also part of the project, both PLC and microcontroller programming.

4 Classification according to the dimensions

The size of the setup will make it possible to be mounted to the underside and/or backside of most generic office chairs. Therefore, in the teaching and evaluation phase, fixing will be generic, most probably based on straps or tape, to allow for moving to a different chair within a few minutes.

Table 3: Classification of the focus project according to the dimensions

Dimension	Property	Value
Value chain	development	☑
	production	☑
	sales	☐
	after-sales-support	☐
	end-of-life	☐
Chain of technology	mechanical structure	☑
	sensors	☑
	electronic circuits	☑
	edge device	☑
	data transmission	☑
	cloud	☐
Sustainability	energy reduction	☑
	material reduction	☐
	better materials	☐
	better production	☐
	reparability	☐
	recycling	☐
Physicality	physical setup	☑
	simulation	☐
Degree of student freedom	demonstrated	☐
	guided	☑
	coached	☑
	autonomous	☐
Transportability	fixed	☐
	transportable	☐
	portable	☑
Costs (implementation)	EUR	300 – 1500

Chair telemetry transponder

Dimension	Property	Value
Costs (operation)	EUR	NA
Workload (implementation)	Hours	10 – 30
Workload (operation)	Hours	30 – 60
Size	m	< 1
Weight	kg	1-5
Special requests	no/yes, if yes: which	no

5 Educational information

5.1 Electronics Development

The demonstrator is to be used by students with a minimum of one semesters experience in low/mid-level programming, as well as knowledge about digital electronics. The following description assumes that C++ is used for programming the embedded ESP32 based unit. The learning goals are written based on Biggs work[2].

5.1.1 Intended Learning Outcomes

The demonstrator is thought to be used over several teaching sessions with exercises. These sessions consist of 4 modules of 45 minutes each. It is expected that the student after the demonstrator sessions session can

- Explain the workings of the I2C protocol
- Read a technical datasheet and extract information
- Implement driver functions to communicate with a digital sensor over I2C
- Explain how to connect to a Wi-Fi network and implement automatic connection in code
- Design, order and produce a PCB suited for electronics module integration
- Design and produce code for low power applications

The students are taught I2C protocol, data filtering and encoding, socket programming, sleep states in C++, as well as PCB design, layout and production and are then asked to solve the following exercises:

1) Initial design, testing and validation

a) Connect two Accelerometer modules the I2C bus with the AD0 connection pulled high for one, and low for another. As shown in the schematic:

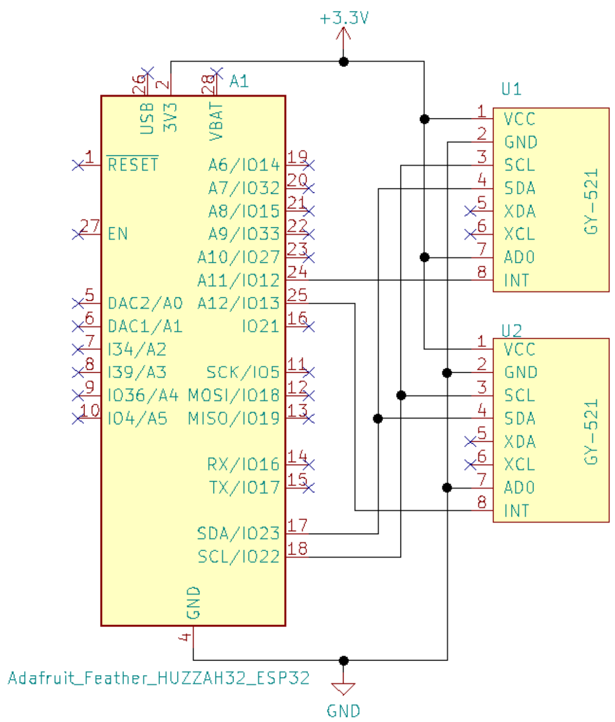


Figure 1: Electronics schematic

b) Connect USB cable to PC with Arduino IDE installed and using the Wire.h (I2C) library, and the MPU-6050 datasheet, program the microcontroller to read accelerometer and gyroscope data from the MPU-6050's internal registers and at a regular interval print these to the serial port. Verify that the data changes as the electronics is moved around.

c) Format the read data so that it will properly display as graphs for each Degree of Freedom (DoF); meaning - three axes of linear accelerations as well as angular velocities around the same axes.
Example shown below:

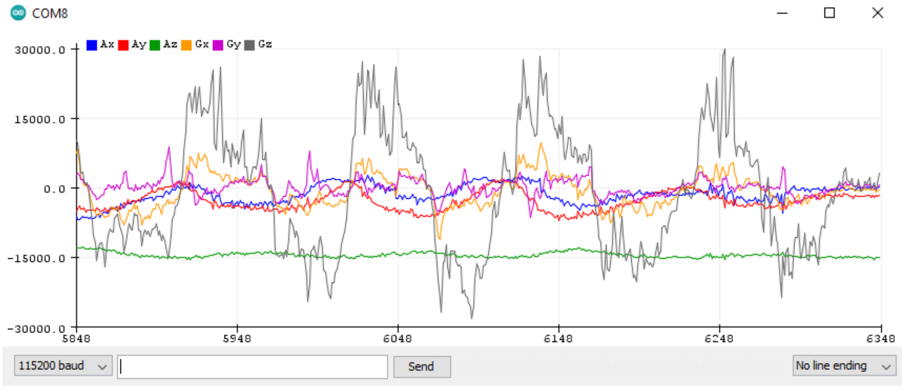


Figure 2: Graph of accelerations and angular velocities

d) Expand your code to handle both IMUs and verify that both appears to work as expected.

e) Using the WiFi.h library, add a function to your code that connects to a network. Verify that the device is connected.

f) With the Raspberry PI, connected to the same network, and with the host application running, write a function that connects to the host socket. When connected, send accelerometer data as a byte stream terminated by a newline character, at a fitting regular interval.

2) Prototype design

a) Using your preferred PCB design layout software, create a carrier board for your ESP32 board and one IMU, and add an off-board connector for connecting the other IMU, so that it may be placed independent of the carrier board. Take into consideration the physical dimension constraints that might exist from mounting/enclosure, battery placement etc. See the mechanical part of the project.

b) Generate production files e.g. Gerber files, and have your design produced. Once produced, verify that your existing code still works.

3) Software improvements

a) Implement a deep sleep state, when unable to connect to the host device, to save on battery life when not in use.

b) Implement initialisation functionality to setup the IMUs programmable low-pass filter and sample intervals, as to minimize network traffic.

c) Implement initialisation functionality to setup the IMUs digital motion processors to generate interrupts to wake the unit from deep sleep state when motion is detected and go into deep sleep when no motion has been detected for an appropriate period.

5.2 Software Development

The demonstrator is to be used by students with a minimum of one semesters experience in object-oriented programming (or equivalent experience if another programming language is used). The following description assumes that Java is used for programming the Raspberry Pi. The learning goals are written based on Biggs work[2].

5.2.1 Intended Learning Outcomes

The demonstrator is thought to be used over several teaching sessions with exercises. These sessions consist of 4 modules of 45 minutes each. It is expected that the student after the demonstrator sessions session can

- Explain how to set up a Socket Connection in Java
- Implement a Socket Connection in Java and receive data from the DigiDemo board
- Implement a Socket solution using Threads so it can be expanded with further DigiDemo boards
- Use separation of concern so the application can handle multiple calculations using parallelism.
- Explain the principles behind data storage on an embedded unit
- Implement a solution using data storage on an embedded unit in Java
- Explain simple principles for handling live data from sensors
- Implement a program doing as a minimum running time average on live data from the DigiDemo board

The students are taught socket programming, threads, gui, and data processing in java and are then asked to solve the following exercises

1. Communicate with the DigiDemo board. Take one DigiDemo board and connect it to Wi-Fi. Connect the student's computer to the same Wi-Fi net. The student must now establish a connection to the board using knowledge about threads and socket communication. This solution is without GUI and should just get data and print it to a console so it can be proved that there is a connection to the board. The solution to this exercise must be stored in a version control system so the student can use it for the coming exercises.
2. Read and display data from one DigiDemo board. Take one DigiDemo board and connect it to Wi-Fi. The actual IP address of the board is needed. The student will now log onto the same Wi-Fi net as the board. The Java application is compiled and started on the student's computer. This requires use of the application version where the socket connection is implemented. The task for the student is now to write the code that will allow the user to activate the socket and start to receive the data. Data must be received and handled using appropriate patterns. Minimum requirement is to show the data on the screen real time as they are sent from the DigiDemo board. The solution must be designed so it is relatively easy to add further boards to the application. Note that where exercise 1 has the nature of a quick hack to ensure connection this takes it the step further by using system design of the SW part.
3. Storing data from the board. The data transferred from the DigiDemo board must be stored somewhere before transferred to a cloud computing solution. One or more boards are now connected to the student's computer or even better a Raspberry Pi[1] which then must work as an edge computing unit. The student must find out which storage solutions are possible to use within their curriculum and select one for implementation. This have then to be stored in version control (Git or Mercury).

4. Challenge exercise: Scalable solution for storing data from the board. A typical usage scenario consists of 30 chairs connected to a single Raspberry Pi unit. The solution from exercise 3 needs to be made scalable. The following steps are suggested:

   4.1. Allowing connection to 2 DigiDemo boards running in parallel

   4.2. Allowing connection to 5 DigiDemo boards running in parallel

   Part of the implementation is to prove that the writing to the chosen storage solution can happen in real time.

5. DSP work on the edge computing unit. Before storage the data needs to be processed and compressed in a reasonable manner (reasonable manner is defined as what fits with the curriculum of the students encountering the DigiDemo project). The students must create a program that will collect data from each single DigiDemo board and process it. One example could be to create a rolling time average of the data before storing it, but a lot of other solutions are possible. By nature, this work needs to be done in threads using principles about parallelism taught in class. The solution developed in 3 and 4 must still work with the data processing and this must be part of the test criteria for the solution.

6. Send the stored data from the unit to a central server. The data stored in 5 (or 3/4 if the students didn't solve 5) must be sent to a cloud service for further processing/storage. The students will have to investigate the possibilities and then implement the selected solution. A student's PC can be an acceptable service and it will be up to the teacher to define the success criteria here. As a minimum the data from the Raspberry Pi have to be sent to the student's PC where it has to be stored in a local storage solution (a database of some kind). Next step can be to visualise the data on the student's PC. And the final step is a real time visualisation.

5.3 Mechanical development

This demonstrator has heavy focus on the hardware and software, and it has a more limited learning potential in the mechanical area. The primary focus will be on understanding how to adapt the mechanics to the hardware requirements, and to find universal ways of fastening the devices to various brands of chairs.

5.3.1 Intended Learning Outcomes

In projects where hardware is used in combination with mechanic enclosures, it is vital for the students to understand that one must obtain clear and concise specifications. Besides physical measures, this also covers thermal considerations, access to hardware for maintenance, applied environment (water, vapor, chemicals, vibration levels, ambient temperature etc) and expected lifetime of the product.

In this demonstrator, the mechanical parts will primarily be 2 boxes for the hardware. These boxes should be prototyped for additive manufacturing, but should be considered for high volume plastic molding, with all the learning details in relation to that process.

Furthermore, it should be considered if a generally applicable form of fixing the boxes to chairs of different brand can be developed. This can show to be quite a tricky task.

5.4 Sustainability – questions for reflection

The questions below are meant as inspiration for the teacher using the demonstrator. The aim is to strengthen the student's awareness regarding sustainability through reflecting upon questions related to the demonstrator. The questions can be used during the lecture or as the starting point for individual or group work, connected to the lecture. The sustainability questions can be used in all teaching subjects (here Electronics Development and Software Development). It is suggested that the students in groups answer the questions and then present the result of the discussion on class.

The questions below are divided into three main categories, (1) general considerations, (2) recycling, and (3) product integrity, where the last two categories are addressing the specific demonstrator.

(1) General considerations

- Does the making or use of the product create pollution?
- Can the material and energy used to make or use this product be reduced?
- Can nature regenerate the resources used to make or use this product?

(2) Recycling

- What are the main materials used for the chair (look at office chairs in general)?
- Can all the materials be recycled – are they "pure"?
- How should the parts be connected in order to be reused?

(3) Product integrity

- What are the key aspects to obtain a durable product, in terms of construction and material?
- What are the most important components to be able to maintain (e.g., the seat or battery)?
- Do you foreseen the need to update any components in order to maintain the integrity of the product (e.g., the sensors)?
- Could the same functionality be obtained using a mobile phone (attached to the chair)?

Is it possible to design the sensors so they could relocate onto another chair?

6 Organizational information

6.1 Electronics Development

The following requirements apply

- Project duration: 4 sessions during a semester. Each session consists of 4 modules of 45 minutes per module.
- Group size: Ideally individual or in smaller groups, depending on the number of available microprocessor boards and IMUs, and to a smaller extend batteries.
- Prerequisites: Either preprogrammed Raspberry PI or an interdisciplinary group structure where the students also develop the Raspberry PI based data collection application.

6.2 Software Development

The following requirements apply

- Project duration: The SW part is thought to run over 4 sessions during a semester. Each session consists of 4 modules of 45 minutes per module.
- Group size: 2 to 4 students must team up and solve the exercises using the DigiDemo boards
- Prerequisites: Either preprogrammed DigiDemo boards or an interdisciplinary group structure where the students also have to assemble the board.

7 Description of the technology and the setup

7.1 Bill of Materials

The following table will outline the materials needed to produce the demonstrator. Note this is the equipment for one group of SW developers and must be multiplied with the number of groups.

Table 4: Classification of the focus project according to the dimensions

Quantity (Unit)	Description	Price (per Unit)	Price (total)
1 pc	Raspberry Pi 4B, 4 GB	64 €/pc	64
1 pc	Black Aluminum Case for Raspberry Pi 4B	30 €/pc	30
1 pc	16 GB micro SD card	12 €/pc	12
1 pc	USB type-c Powersupply	14 €/pc	14
5 pc	Embedded Unit (see below)	40 €/pc	200
			320

Embedded Unit:

Quantity (Unit)	Description	Price (per Unit)	Price (total)
1 pc	Adafruit HUZZAH32 - ESP32 Feather Board	18 €/pc	18
2 pc	GY-521 MPU-6050 breakout board	6 €/pc	12
1 pc	Lithium Ion Battery - 3.7v 2000mAh	10 €/pc	10
			40

Cables for connection the sensors are not included in the embedded unit, as they are custom to the specific setup.

8 References

[1] Raspberry Pi, “Teach, learn and make with raspberry pi,” 2021. [Online]. Available: https://www.raspberrypi.org/. [Accessed: 24-Feb-2021].

[2] John Biggs and C. Tang, *Teaching for Quality Learning at University*, 3rd ed. McGraw-Hill/Society for Research into Higher Education & Open University Press, 2007.