

HSHL HACKATHON 2025



HOCHSCHULE
HAMM-LIPPSTADT



Time table

- **19.03.2025 Wednesday 12:00-19:00** -
 - First day of the event
 - Getting to know the hardware
- **20.03.2025 Thursday 10:30-19:00** -
 - Starting with the tasks
- **21.03.2025 Friday 10:30-17:00** -
 - Finishing the tasks
 - 17:00-18:00
 - Counting of the points
 - Announcement of the winner team.



Tools
Materials
Controllers
Sensors

Each team has access to:

1. Common expendable building materials.
2. Common building materials and tools that we expect to get back
3. **THE BOX.**



Common expendable building materials

While **you don't need to give them back**, we still ask you to be somewhat conservative for your and your colleges sake.

In abundance, but limited among all groups:

- Carton
- Sticky tape
- Two sided sticky tape
- Insulating tape
- Stationary rubber bands
- Resistors
- Wires

Common expendable building materials





Building materials and tools we expect to get back

- Metallic parts
- Pens
- Markers
- Scissors
- Screws and screwdrivers
- Glue gun



HOCHSCHULE
HAMM-LIPPSTADT

Building materials and tools we expect to get back





THE BOX

- Each team is given one **BOX** of components listed further and on **THE BOX** itself
- The Hackathon competition is comprised of **5 Tasks**
- You have enough parts in **THE BOX** to complete any task, but not every task at the same time.
- **Plan accordingly.**



THE BOX parts

**THE BOX contains parts, for
which you are responsible as a
team:**

- Arduino Uno Microcontroller x2
- Breadboard x1
- Numpad x1
- Servo motor 9g x3
- Servo Motor 360° x1
- Button x2
- Photoresistor x3
- USB Wire For Arduino x2
- RGB Diode x1
- Piezo Speaker x1
- IR sensor x3
- Display x1
- Variable resistor x3
- Radio Module x2



HOCHSCHULE
HAMM-LIPPSTADT

THE BOX

parts





Regarding the Tools and THE BOX

Your team is responsible for the equipment you have.

At the end of competition, we expect to get the box and the tools as they were given out.

- Egregious material hogging
- malicious destruction of parts, tools and materials
- negligent destruction of parts, tools and materials
- repeated destruction of parts, tools and materials

May result in

- stern talking's to,
- points deduction,
- long and uncomfortable gazes,
- walks of shame,
- expulsions,
- heavy fines,
- etc.

Jokes aside, just be careful and immediately speak to one of the organisers if something is bended, broken, glued shut, fuming or generally not performing like you would expect it to.



There are five tasks in total

You can work on any task in any order.

You have enough parts in **THE BOX** to complete any task, but **NOT** every task at the same time.

That doesn't mean you cannot work on separate tasks in parallel.

Plan accordingly.



For each of those five tasks

- You can get up to a **hundred points.**
- **Each has three difficulty levels,** with a certain number of points associated with completing them. You choose how complicated to make it.

Any device you build must not only be reliable in its function, but be safe, usable and aesthetically pleasing.

The construction must **NOT** “fall apart” if touched or moved, have unreliable electrical or mechanical junctions. Every device must have a frame, covering the non-functional parts. **The sturdiest and the prettiest device gets 20 additional points from the jury in the end.**



Task 1 Post terminal.

You need to assemble an autonomous post terminal, which will optimise issuance of equipment to employees.



<https://upload.wikimedia.org/wikipedia/commons/thumb/f/fe/PB-SM-2.jpg/800px-PB-SM-2.jpg>



Task 1

Post terminal.

Equipment from **THE BOX™**:

Arduino uno x1

Breadboard x1

Button x2

Numpad x1

RGB LED x1

Photoresistor x3

Servo motor 9g x3

A. Basic level (60 points max)

You need to assemble a device, **specifically a cabinet with three sections**. Doors to those sections are being controlled by **a servo motor**. It should have **a Numpad** and **two buttons**: one to store in and one to take equipment out of a section.

The device can read via numpad digit Codes which have three types:

1. **Wrong code**, red LED shines, no doors can be opened.
2. **Correct user** code to take equipment out from a “full” section. One code input grants user one door opening. All three sections start full. We count section empty if the door was opened.
3. **Admin code**. Admin can either take equipment from a full section or restock empty ones. There are **two buttons**: first one opens empty sections one after another in case of subsequent presses of the button. Second button lets the admin take equipment out of the full section.

During the issuance of equipment a section must stay **open for 10 seconds**, during refill – **15 seconds**.



Task 1

Post terminal.

Equipment from **THE BOX™** :

Arduino uno x1

Breadboard x1

Button x2

Numpad x1

RGB LED x1

Photoresistor x3

Servo motor 9g x3

B. Intermediate level (80 points max)

There must be a way to know if there is equipment in the storage in case the section was open, but no equipment was taken.

Also, there must be a LED, which is ON when the section is opened and start blinking when there is only 3 seconds left until the section closes.

C. Advanced level (100 points max)

Using serial connection, log to the console in a readable and understandable format:

- 1. When was equipment taken, and by who (user or admin)**
- 2. When was a section refilled**
- 3. How much equipment is left**
- 4. Which sections are empty**



Task 2

Sunflower.

Sunflowers have an ability to follow the sun movement, thanks to a mutation in their “internal clock”. They control growth of their cells, making the inflorescence move from east to the west during the day. **We will assemble an electronic variant, which will follow a light source.**



[https://www.etsy.com/listing/1326388782/
/metal-sunflower](https://www.etsy.com/listing/1326388782/metal-sunflower)

Task 2

Sunflower.

Equipment from **THE BOX™**:

- Arduino uno x1
- Breadboard x1
- Button x1
- Servo Motor 360 deg x1
- Piezodynamic x1
- Photoresistor x2

A. Basic level (60 points)

Make a rotary platform with a sunflower arrowhead pointing on top.

In **ten seconds** after the system has been activated with a press of a **button** the sunflower most point in the **60-degree sector** of a light source. (The difference between the direction the arrowhead is pointing, and the light source should be no more than **30 degrees**).



Task 2

Sunflower.

Equipment from **THE BOX™**:

Arduino uno x1

Breadboard x1

Button x1

Servo Motor 360 deg x1

Piezodynamic x1

Photoresistor x2

B. Intermediate level (80 points max)

Both the sector and time to find it shrink. Sector becomes **20 degrees** (The difference between the direction the **arrowhead** is pointing and the **light source** should be no more than **10 degrees**). This sector must be found within **5 seconds**. If no **light source** was found – sound a signal for **3 seconds**.

C. Advanced level (100 points max)

Sunflower inflorescence must follow the **light source in real time**.



Task 3

CakeSort.

Sushi type conveyer belt carries **4 types of cake**: Tiramisu, Cheesecake, Pumpkin pie and Apple pie.

Before sending them to clients they must be sorted.

You will have to create the device that can sort and separate them automatically.



<https://hypebae.com/2018/9/conveyor-belt-cake-restaurant-japan>



Task 3

CakeSort.

Equipment from **THE BOX™**:

- Arduino uno x1
- Breadboard x1
- Servo Motor 9g x3
- Infrared Sensor x3
- Display x1

A. Basic level (60 points)

You will have to make an inclined line, on which the “**cake boxes**” have to slide. The **box** itself is marked with **three-bar codes** which correspond to cakes:

Black-White-Black - Tiramisu

White-Black-White - Cheesecake

Black-Black-White – Pumpkin pie

White-White-Black - Apple pie



At some point of an incline the **cake** must be stopped by a **servo**, read and let run past again. **Your task is to count the cakes and display the quantity of each kind passed. Cakes are loaded one at the time.**



Task 3

CakeSort.

Equipment from **THE BOX™**:

- Arduino uno x1
- Breadboard x1
- Servo Motor 9g x3
- Infrared Sensor x3
- Display x1

B. Intermediate level (80 points max)

You must implement an automatic sorting station – **2 types** must go in one pile and **2 other** in the other.

C. Advanced level (100 points max)

It is now possible for “**Rogue cakes**” to appear, with markings that are **NOT** consistent with any of the **known cakes**. **They must be sorted outside of the two piles.**



We must not let them pass.



Task 4

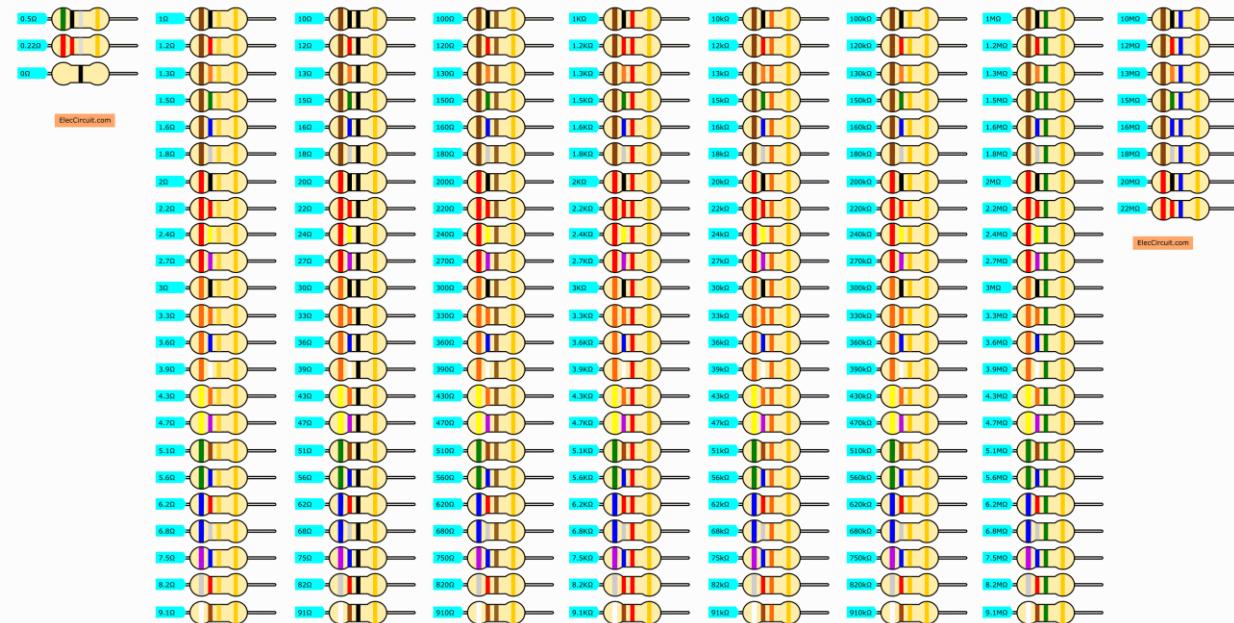
MegaOhm

It is often the case that aspiring electro engineers must spend too much time in **the resistors markings table** to find the value.

It is as hard for teachers to explain how to convert coloured lines into numbers.

Today you can help them both, by creating a learning device!

Standard resistor values 5% tolerance





Task 4

MegaOhm

Equipment from **THE BOX™**:

Arduino uno x1

Breadboard x1

Button x1

RGB diode x1

Variable resistor x3

Display x1

Basic level (60 points)

Three potentiometers with cool handles. Each potentiometer has ten equal sections, coloured in ten different colours. The user can choose a colour on each of the **three potentiometers**. The corresponding value of a resistor in accordance to the **table** must be displayed.

Intermediate level (80 points max)

Attach two electrical contacts, to which you can connect a **resistor** with unknown markings. Display the value in Ohms of the connected resistor under the cylinder value.

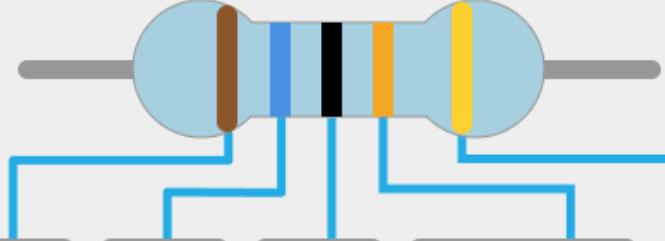
Advanced level (100 points max)

Add a check for equality. If the connected **resistor** has a value in Ohms within **10 percent** of the cylinder one – light **green**. If it falls outside of that – **red**.



Task 4 – MegaOhm Table of resistances for reference

<https://developer.wildernesslabs.co/Hardware/Circuits/Components/Resistors/Reading/>



Color	1 st	2 nd	x th	Multiplier	Tolerance
Black	0	0	0	x1	
Brown	1	1	1	x10	±1%
Red	2	2	2	x100	±2%
Orange	3	3	3	x1k	±3%
Yellow	4	4	4	x10k	±4%
Green	5	5	5	x100k	±0.5%
Blue	6	6	6	x1M	±0.25%
Purple	7	7	7	x10M	±0.1%
Gray	8	8	8	x100M	
White	9	9	9	x1,000M	
Gold				x0.1	±5%
Silver				x0.01	±10%

Example resistor:
1 6 0 x1k = 160kΩ

} Common
Tolerances



Task 5

RadioMorse

The question about the authorship of radio communication persists to this day from 19th century. It is no surprise as this technology allows people to pass signals with the speed of light!

One of the earliest systems of encoding text to pass over long distances was Morse code, which is very resilient to interference.

Let's put that to the test ourselves!

International Morse Code

1. The length of a dot is one unit.
2. A dash is three units.
3. The space between parts of the same letter is one unit.
4. The space between letters is three units.
5. The space between words is seven units.

A	• -	U	• • -
B	- - . .	V	• - -
C	- - : -	W	• - - -
D	- - : :	X	- - - .
E	•	Y	- - - -
F	• . - -	Z	- - - . .
G	- - -		
H	• • . .		
I	• • :		
J	• - - -		
K	- . -	1	• - - - -
L	• - - .	2	• - - - -
M	- -	3	• - - -
N	- - .	4	• - - -
O	- - -	5	• - - -
P	• - - - .	6	• - - - - .
Q	- - - . -	7	• - - - - .
R	• - - .	8	• - - - - .
S	• • .	9	• - - - - .
T	-	0	• - - - - .

https://upload.wikimedia.org/wikipedia/commons/b/b5/International_Morse_Code.svg



Task 5

RadioMorse

Equipment from **THE BOX™**:

Arduino uno x2

Breadboard x1

Button x1

Radio Module x2

Display x1

A. Basic level (60 points)

You must create a device for a wired transmission of text using Morse code. It should include a **transmitter** and a **receiver**. The **transmitter** sends the elements of Morse by long or short press of a **button**. The **receiver** then gets the signal and writes the corresponding dots and dashes into the **display** as dots and dashes.

B. Intermediate level (80 points max)

Establish wireless connection.

C. Advanced level (100 points max)

Modify the system for extended reliability. The **transmitter** must count **the hash sum** and deliver it after the information. The **receiver** then counts **the hash sum** by the same algorithm and compare it to the received one. The **receiver** must be modified to output the deciphered code to the **display** as letters.



Task 5

RadioMorse

Equipment from **THE BOX™**:

Arduino uno x2

Breadboard x1

Button x1

Radio Module x2

Display x1

A. Basic level (60 points)

You must create a device for a wired transmission of text using Morse code. It should include a **transmitter** and a **receiver**. The **transmitter** sends the elements of Morse by long or short press of a **button**. The **receiver** then gets the signal and writes the corresponding dots and dashes into the **display** as dots and dashes.

B. Intermediate level (80 points max)

Establish wireless connection.

C. Advanced level (100 points max)

Modify the system for extended reliability. The **transmitter** must count **the hash sum** and deliver it after the information. The **receiver** then counts **the hash sum** by the same algorithm and compare it to the received one. The **receiver** must be modified to output the deciphered code to the **display** as letters.



During the competition:

- You can approach one of the Jury team for help or evaluation at any point.
- For us to register the task as complete, or partially complete, you must show it to us, before dismantling it for the next task.
- You will get a copy of this presentation for reference.
- **The goal is to get more points than other teams in a timeframe, without resorting to sabotage.**
- The points will be counted and the winning team announced on the last day.
- **If you have any questions about the rules or equipment, please approach us.**

LET THE GAMES BEGIN!