



runlinc Intermediate Project 2: Radiation Counter (E32W Version)

Contents

Introduction.....	1
Part A: Design the Circuit on runlinc	4
Part B: Build the Circuit.....	5
Part C: Program the Circuit	8
Expected Result	11
Summary.....	12

Introduction

Problem

When we are in an environment with radiation, we want to make sure the level of the radiation is in the safe range. What can we do to determine the radiation level?

Background

When we think about radiation, we often think of things like nuclear weapons and accidents like those at the Chernobyl and Fukushima nuclear reactors. However, low levels of radiation are all around us every day, from the sun to the building materials and even in our food! Brazil nuts and bananas are the two foods quite well known for being slightly radioactive. Don't worry about overeating of such foods though; you could not eat enough of them to cause any health problems (You should always try and eat a good amount of fruit anyway!).

Radiation is also very useful in many applications, many smoke detectors use a tiny amount of radioactive Americium to detect the smoke. Radiation is often used in medicine as well, including in the treatment of some cancers, viewing the inside of the body by using X-rays (radiography), radio-opaque dyes and nuclear medicine procedures. Radioactive substances can also be used in scientific research. Radiography is used in many industries for quality control or to analyse the internal working of machinery. We also use radioactive elements to energise nuclear power plants, submarines and satellites. The Curiosity Mars rover is not only powered by radioactive elements but also uses them for analysing rock and soil samples it finds on the red planet.

However, radiation can be hazardous for living things. The effects on the human body due to receiving too much radiation vary, depending on which areas of the body were exposed and how much radiation was absorbed. Radiation damages and kills the cells that make up our bodies, and may, therefore, cause burns (sunburn is caused by UV radiation), loss of white blood cells, haemorrhaging (bleeding), fatigue and weakness and even cancer. If radiation damages vital organs like the brain, heart, lungs, digestive tract etc., then the effects will be more severe than if only the surface of your skin or your hand was affected.

Despite these dangers, people must sometimes enter dangerous radioactive zones. For example, people have been cleaning up the areas around the Fukushima nuclear reactor in Japan or when people are working in the places where radioactive compounds are used or stored such as a research centre or even in outer space. They often carry measuring devices like Geiger counters to help them measure how much radiation they have been exposed to. It is crucial to consider the level of radiation they have been exposed to, as well as the length of time they were exposed. If there was only a low radiation level, you could stay in the area for a longer time without any negative effects than if the radiation level was higher. For today's project, we will implement a counter to keep track of the total amount of radiation we have been exposed to. We will be using a safer type of radiation though, simply measuring the amount of light.

Note: Yes, (visible) light is a type of radiation. Much like radio waves, microwaves, infrared and x-rays are all radiation and are part of the electromagnetic spectrum. The danger in radiation generally comes with how short the wavelength for the type is. Types with longer wavelengths such as radio waves are safer than gamma waves or x-rays, which have significantly shorter wavelengths and have more energy per packet of light.

Ideas

How can we measure the radiation (light) level? How can the microchip and runlinc tell the user if the level is high or low? How can we make our device count up faster the more radiation is detected? Is a visual warning on a webpage enough, or would the additional method be advantageous? What kind of thresholds might be good? Is there any other information that might be good to display on our Webpage?

Plan

We have Photoresistor Light Sensor (PLS) in our kits which we can use to check the light intensity of a possible fire, and a web input to change the light intensity risk.

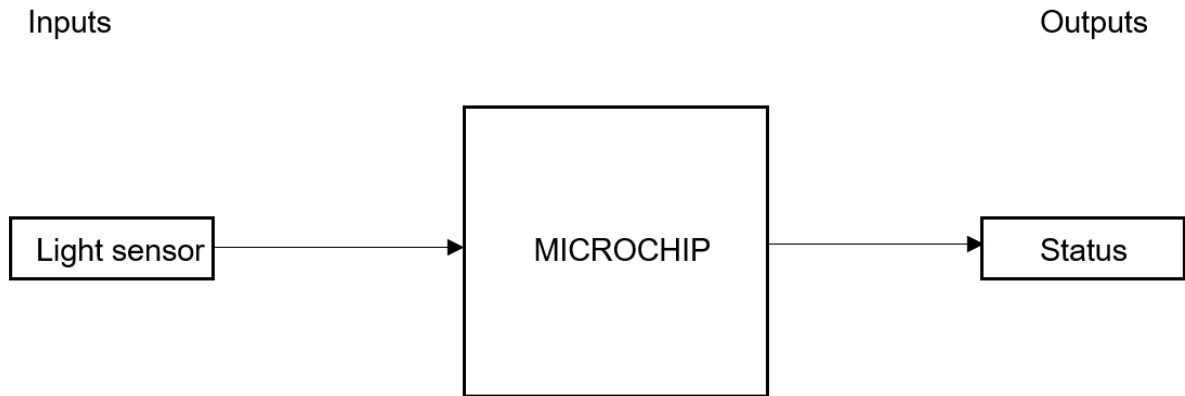


Figure 1: Block diagram of Microchip outputs

runlinc Background

runlinc is a web page inside a Wi-Fi chip. The programming is done inside the browsers compare to programming inside a chip. The runlinc web page inside the Wi-Fi chip will command the microchips to do sensing, control, data logging Internet of Things (IoT). It can predict and command.

Part A: Design the Circuit on runlinc

Note: Refer to runlinc Wi-Fi Setup Guide document to connect to runlinc

Use the left side of the runlinc web page to construct an input/output (I/O).

For port D33 name it RAD_Sensor and set it as ANALOG_IN.

In our circuit design, we will be using the light sensor. We happen to have this in our kits, so these can be used on our circuit design, as per the plan.

D23	DISABLED		
D25	DISABLED		
D26	DISABLED		
D27	DISABLED		
D32	DISABLED		
D33	ANALOG_IN	RAD_Sensor	4
D34	DISABLED		

Figure 2: I/O configurations connections

Part B: Build the Circuit

Use the STEMSEL E32 board to connect the hardware. For this project we are using both the left and right I/O ports, with **negative port (-ve)** on the outer side, **positive port (+ve)** on the middle and **signal port (s)** on the inner side (as shown below).

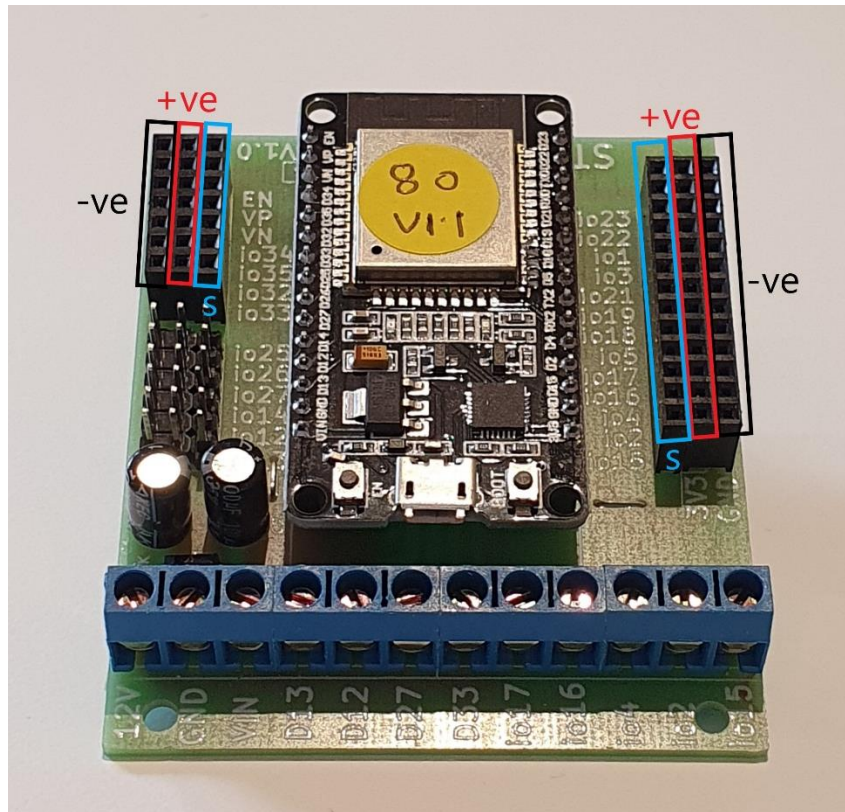


Figure 3: Negative, Positive and Signal port on the E32 board

There is one I/O part we are using for this project, a Photoresistor Light Sensor (PLS) module (KY-018), their respective pins are shown in the figure below.

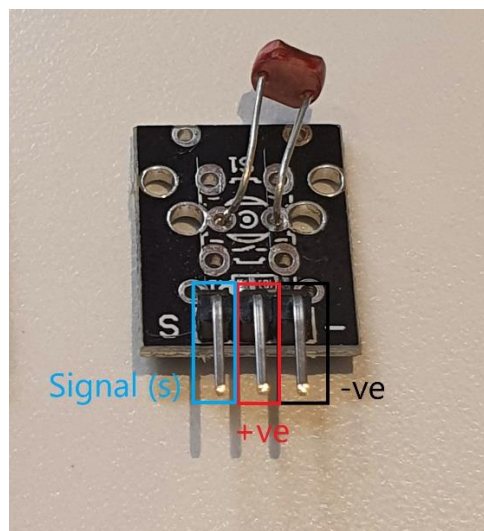


Figure 4: I/O part with negative, positive and signal pins indicated

Wiring instructions

- a.) Plug in the Photoresistor Light Sensor (PLS) to io33 on the E32 board.
- b.) Make sure the (-ve) pin are on the GND (outer) side of the I/O ports.

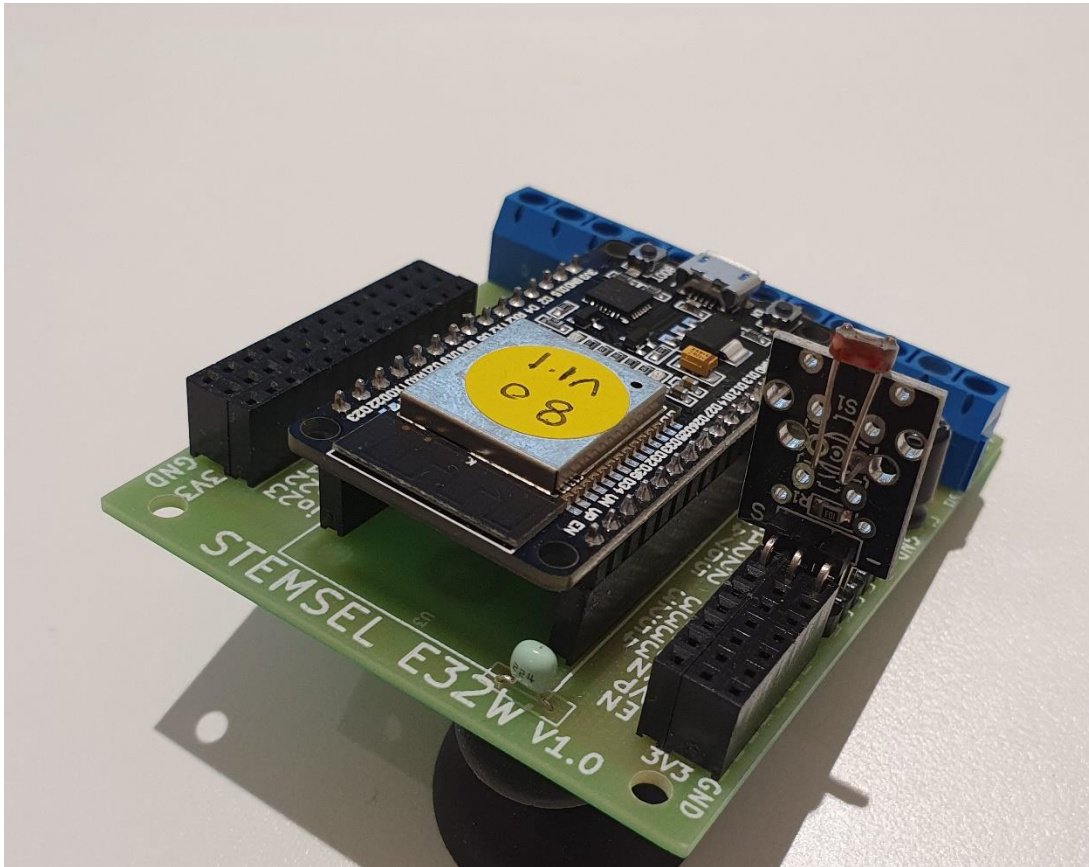


Figure 5: Circuit board connection with I/O part (side view)

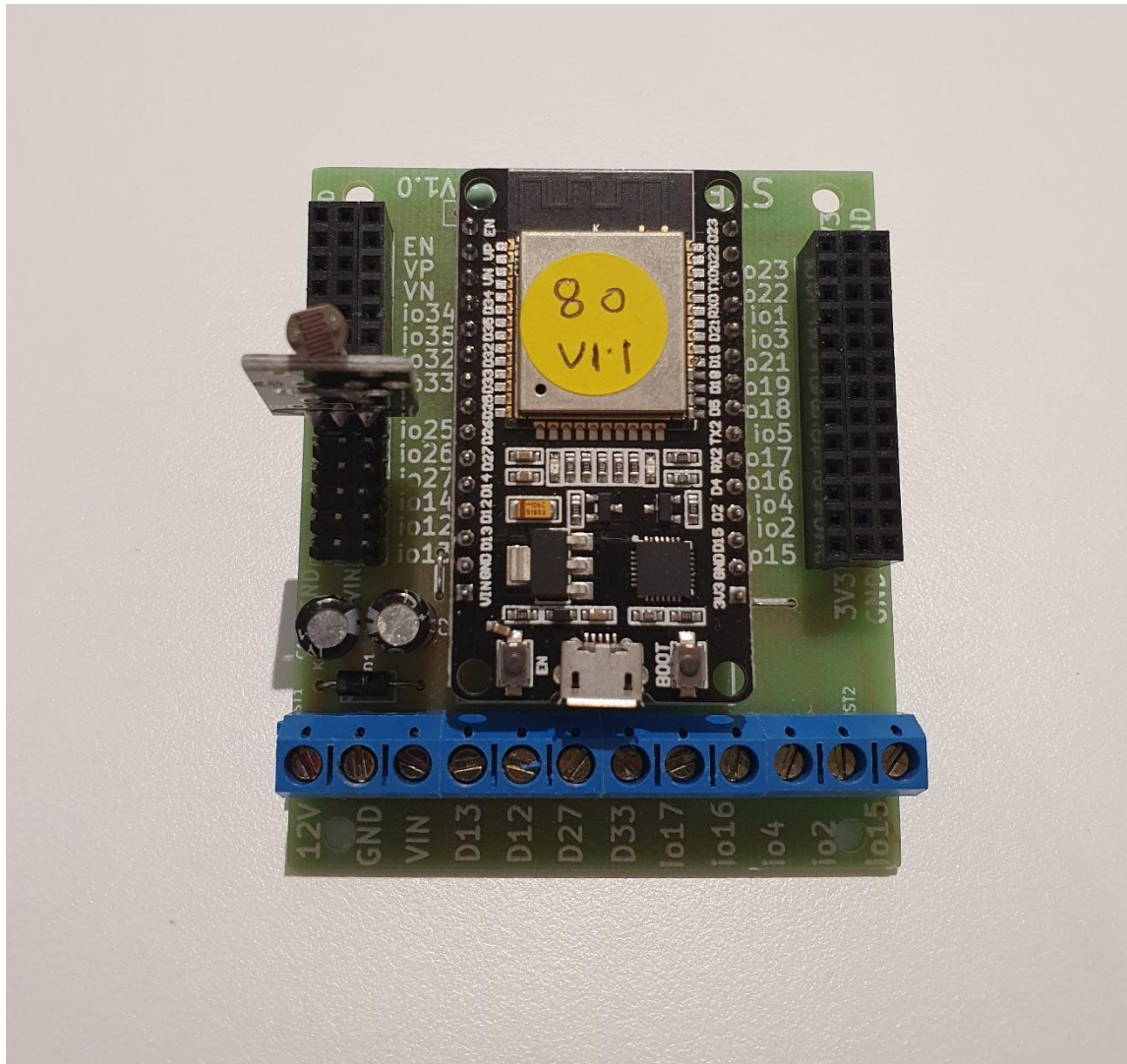


Figure 6: Circuit board connection with I/O part (top view)

Part C: Program the Circuit

HTML:

Now we can start to program the functions of the radiation counter. To do this, HTML and JavaScript Loop are needed.

For **HTML** box type the following:

```
<h2>runlinc Radiation Counter</h2>  
Radioactivity:  
<font id=RadNow></font> Curie
```

This will set up our webpage to receive the Information from the RAD_Sensor.

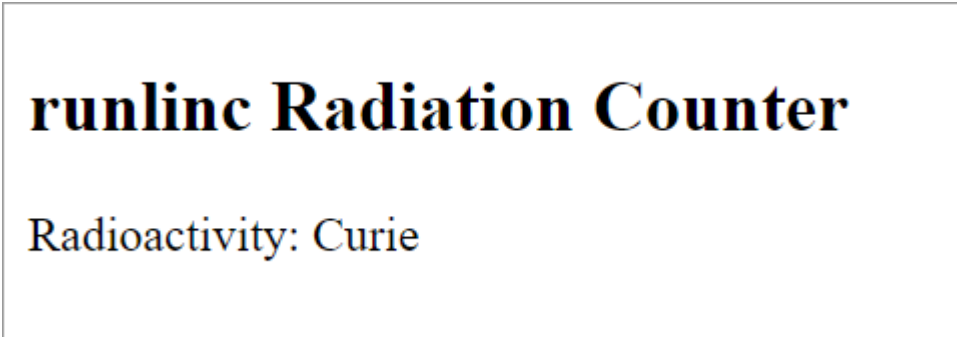
For **JavaScript Loop** box type the following code:

```
sense_num = analogIn(RAD_Sensor);  
sense_num = 255-sense_num;  
sense_num = sense_num / 51;  
Curie = sense_num.toFixed(2);  
document.getElementById('RadNow').innerHTML = Curie;
```

Here, a variable is initialised with it containing the light level read from the light sensor. Then, we are sending it to the HTML webpage so it can be displayed.

Since our Photoresistor Light Sensor have a **higher reading when its surrounding is darker**, and a lower reading when its surrounding is brighter, '**sense_num = 255-sense_num**' is added to reverse the result, ensuring a logic and clear output reading.

At this stage, save and run the code and you should have something similar to **Figure 7** displayed on your webpage. If it doesn't work, go through your code and check for any errors.



The screenshot shows a webpage with a large heading 'runlinc Radiation Counter' in a bold, black, serif font. Below the heading, the text 'Radioactivity: Curie' is displayed in a smaller, blue, serif font. The entire content is enclosed in a thin black border.

Figure 7: runlinc webpage screenshot 1

Now we can expand on what has been done to now display the radiation level. To do this, we will need to add to the HTML script and the JavaScript Loop as well as the JavaScript.

runlinc Intermediate Project 2: Radiation Counter (E32W Version)

For **HTML** box, typing the following under what has already been done:

```
<br>  
<font face="verdana" id=TotRads></font> Rads
```

For the **JavaScript** box, typing the following:

```
let sense_num = analogIn(RAD_Sensor);  
let total_rads = 0;
```

For the **JavaScript Loop** box, typing the following under what has already been done:

```
total_rads = total_rads + sense_num;  
await mSec( 500 );  
document.getElementById('TotRads').innerHTML = total_rads.toFixed(2);
```

Now Save and run the code and see what's new, you should have something similar to Figure 8 at this stage. If it's not working double check your code for any errors, barring any errors before refresh the page and load in your program to try again.

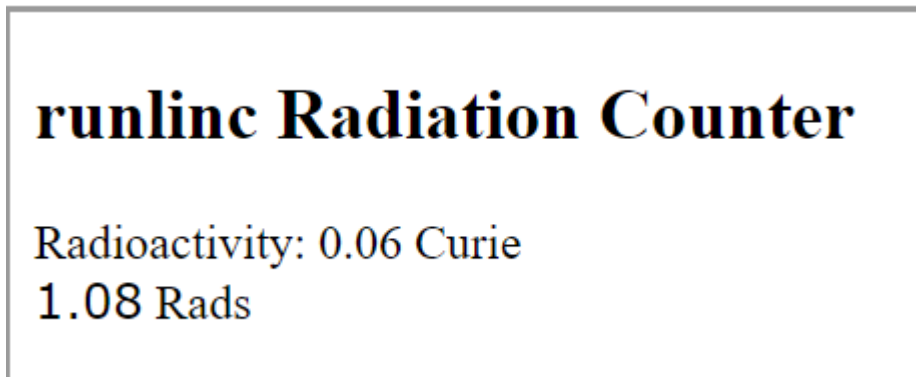


Figure 8: runlinc webpage screenshot 2

Then we're going to create the alerts, to do this we're going to add extra content to the HTML box and the JavaScript Loop box.

For **HTML** box, typing the following under what has already been done:

```
<br>  
<font id=Message></font>
```

For the **JavaScript Loop** box, typing the following under what has already been done:

```
if(sense_num < 2)  
  document.getElementById('Message').innerHTML = "Too LOW";  
else  
  document.getElementById('Message').innerHTML = "Too HIGH";
```

runlinc Intermediate Project 2: Radiation Counter (E32W Version)

Once again save the code and run it. If it doesn't work, check your code for any errors if no errors refresh the page and reload your code then try and rerun it. Your webpage window should look similar to **Figure 9**.

Final Code:

The final code for **HTML** block:

```
<h2>runlinc Radiation Counter</h2>
Radioactivity:
<font id=RadNow></font> Curie
<br>
<font face="verdana" id=TotRads></font> Rads
<br>
<font id=Message></font>
```

The final code for **JavaScript** block:

```
let sense_num = analogIn(RAD_Sensor);
let total_rads = 0;
```

The final code for **JavaScript Loop** block:

```
sense_num = analogIn(RAD_Sensor);
sense_num = 255-sense_num;
sense_num = sense_num / 51;
Curie = sense_num.toFixed(2);
document.getElementById('RadNow').innerHTML = Curie;
total_rads = total_rads + sense_num;
await mSec( 500 );
document.getElementById('TotRads').innerHTML = total_rads.toFixed(2);
if(sense_num < 2)
  document.getElementById('Message').innerHTML = "Too LOW";
else
  document.getElementById('Message').innerHTML = "Too HIGH";
```

Expected Result

runlinc Radiation Counter

Radioactivity: 4.94 Curie
 29.67 Rads
 Too HIGH

Figure 9: Expect webpage screenshot 3

runlinc V1.2 Copyright and International Patent Pending. All rights reserved.

File

Load File

Save

Board

Send

Get

Run Code

Board IP:

ESP32

PORT	CONFIGURATION	NAME	STATUS
D2	DISABLED	<input type="text"/>	
D4	DISABLED	<input type="text"/>	
D5	DISABLED	<input type="text"/>	
D12	DISABLED	<input type="text"/>	
D13	DISABLED	<input type="text"/>	
D14	DISABLED	<input type="text"/>	
D15	DISABLED	<input type="text"/>	
RX2	DISABLED	<input type="text"/>	
TX2	DISABLED	<input type="text"/>	
D18	DISABLED	<input type="text"/>	
D19	DISABLED	<input type="text"/>	
D21	DISABLED	<input type="text"/>	
D22	DISABLED	<input type="text"/>	
D23	DISABLED	<input type="text"/>	
D25	DISABLED	<input type="text"/>	
D26	DISABLED	<input type="text"/>	
D27	DISABLED	<input type="text"/>	
D32	DISABLED	<input type="text"/>	
D33	ANALOG_IN	RAD_Sensor	3
D34	DISABLED	<input type="text"/>	

CSS

HTML

```
<h2>runlinc Radiation Counter</h2>
Radioactivity:
<font id=RadNow></font> Curie
<br>
<font face="verdana" id=TotRads></font> Rads
<br>
<font id=Message></font>
```

JavaScript

```
let sense_num = analogIn(RAD_Sensor);
let total_rads = 0;
```

JavaScript Loop

```
sense_num = analogIn(RAD_Sensor);
sense_num = 255-sense_num;
sense_num = sense_num / 51;
Curie = sense_num.toFixed(2);
document.getElementById('RadNow').innerHTML = Curie;
total_rads = total_rads + sense_num;
await mSec( 500 );
document.getElementById('TotRads').innerHTML = total_rads.toFixed(2);
if(sense_num < 2)
  document.getElementById('Message').innerHTML = "Too LOW";
else
  document.getElementById('Message').innerHTML = "Too HIGH";
```

Figure 10: Expect runlinc result screenshot

Summary

We are surrounded by radiation every day, and although it is useful, radiation can cause severe and ongoing health problems. We need to be careful of how much radiation we are exposed to, so people working in areas with high radiation need to monitor their exposure with devices similar to the one we made in this project.