

Student: Mirt Florin - 1405B



Tema: Smart home

Rezumat:

Acest proiect are ca scop implementarea unui ansamblu de functionalitati, cum ar fi: monitorizarea temperaturii, a luminozitatii si a umiditatii, date care vor fi receptionate si prelucrate de un server.

Resurse hardware:

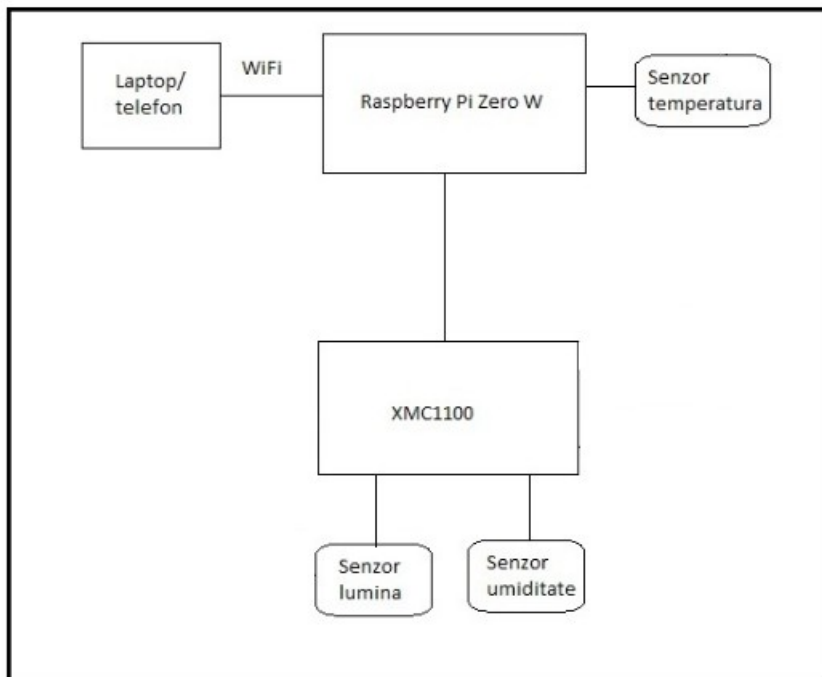
- Raspberry Pi Zero W
- XMC1100
- Senzori temperatura, umiditate, lumina
- Rezistori
- Fire conectoare

Descriere proiect:

Controlul se va face printr-un server gazduit de Raspberry Pi Zero W, conectata la o placuta XMC1100, de la care va primi datele de la senzori si va trimite comenzile catre XMC1100.

Partea de colectare a datelor de la senzori se va face printr-o placuta XMC1100, datele fiind trimise la Raspberry Pi.

Schita:



Documentatie:

<https://circuitdigest.com/microcontroller-projects/>

<http://embedac.ro/PM1/index.html>

<http://embedac.ro/SI/index.html>

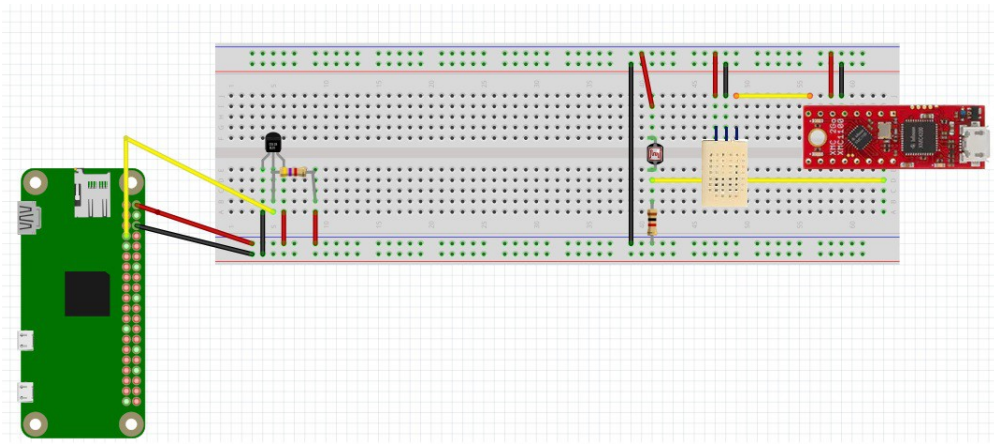
<https://www.hackster.io/projects>

Ciorna solutie:

- Raspberry Pi Zero W
- XMC1100
- Senzori temperatura, umiditate, lumina

Implementare solutie

Placile de dezvoltare folosite in realizarea proiectului sunt XMC2Go si Raspberry Pi Zero W. Am utilizat comunicatia seriala pentru receptia si transmiterea de date intre cele doua placi. Pe Raspberry Pi am configurat un server Apache, iar placa XMC2Go a fost programata cu DAVE.



### Principiu de functionare

XMC-ul asteapta un caracter pe seriala. Daca este "1", el apeleaza o functie care preia data de la senzorul de lumina(fotorezistenta), si il trimite catre seriala; daca primeste "2", preia data de la modulul cu senzor de umiditate si il trimite catre seriala(cand valoarea senzorului depaseste un anumit prag, iesirea de la modul e 0, in caz contrar e 1).

Pe placa Raspberry Pi, am 3 fisiere Python in directorul /var/www cu care preiau datele de la senzori:

- fisierele temp.py preia temperatura de la senzorul DS18B20, care e conectat la Raspberry
- fisierele light.py preia luminozitatea de la fotorezistenta, prin intermediul comunicatiei cu placa XMC
- fisierele humidity.py preia umiditatea de la modulul cu senzor, prin intermediul comunicatiei cu placa XMC

temp.py

```
import time
from w1thermsensor import W1ThermSensor
sensor=W1ThermSensor()
temperature = sensor.get_temperature()
print (int(temperature))
```

light.py

```
import serial
xmc=serial.Serial('/dev/ttyACM0',baudrate=9600)
xmc.write("1")
a=xmc.read()
b=xmc.read()
c=xmc.read()
d=xmc.read()
print int(a+b+c+d)
```

XMC - main.c - functia cu care aflu luminozitatea

```
XMC_VADC_RESULT_SIZE_t result;
void Adc_Measurement_Handler()
{
uint32_t pin;
#if(UC_SERIES == XMC11)
result = ADC_MEASUREMENT_GetResult(&ADC_MEASUREMENT_0);
#endif
pin = result >> ((uint32_t)ADC_MEASUREMENT_0.iclass_config_handle-
>conversion_mode_standard * (uint32_t)2);
}

int main(void)
{
DAVE_Init();
ADC_MEASUREMENT_StartConversion(&ADC_MEASUREMENT_0);
Adc_Measurement_Handler();

return 0;
}
```

In fisierul index.php imi creez un obiect XMLHttpRequest care imi permite sa obtin date cu ajutorul unui request HTTP. Asa pot actualiza continutul paginii fara a reimprospata pagina de fiecare data cand vreau sa vad datele primite.

Initializez cererea cu numele fisierului getTemp.php, care imi trimite date de la cele 3 scripturi Python.

Fisierul getTemp.php:

```
<?php
$t = shell_exec('sudo python /var/www/temp.py');
$l = shell_exec('sudo python /var/www/light.py');
$h = shell_exec('sudo python /var/www/humidity.py');
echo sprintf('%d:%d:%d', $t, $l, $h);
?>
```

Partea de script a fisierului index.php:

```
<script>
var x=setInterval(mes,1000);
function mes(){
var request = new XMLHttpRequest();
request.onreadystatechange = function () {
if (request.readyState == 4 && request.status == 200) {
var ret=this.responseText.split(":");
document.getElementById("temp").innerHTML=ret[0];
document.getElementById("li").innerHTML=ret[1];
document.getElementById("humi").innerHTML=ret[2];
}
}
request.open('GET','getTemp.php', true);
request.send();
}

</script>
```

Rezultat:

**Data:**

Temperature: 23

Lightness: 894

Humidity: 0