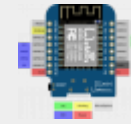


Table des matières

- Cube Guard Board Sensor** 2
 - Que fait il?** 2
 - Datagramme D1 Mini** 2
 - Cube Guard** 3
 - Schéma de Câblage** 4
 - GPIO Paramètre** 4
 - Sensor Paramètre** 5
 - Vue Générale 5
 - Paramètre Oled 5
 - Paramètre BME280 7
 - Les Rules** 8
 - Paramètre Contrôleur** 9
 - Activer P2P Network** 9
 - P2P Port Network** 10
 - P2P View** 10
 - Les Evolutions** 11
 - Câblage du relais** 11
 - La Rules 11
 - Explication** 12



Cube Guard Board Sensor

Que fait il?



C'est Simple Objet Connecté qui a pour rôle de mesurer la température et le taux d'humidité dans une baie réseau. Ceci afin de s'assurer que les équipements présent ne surchauffent pas.

Le tout est relié à un serveur Domoticz qui graphe les constantes.

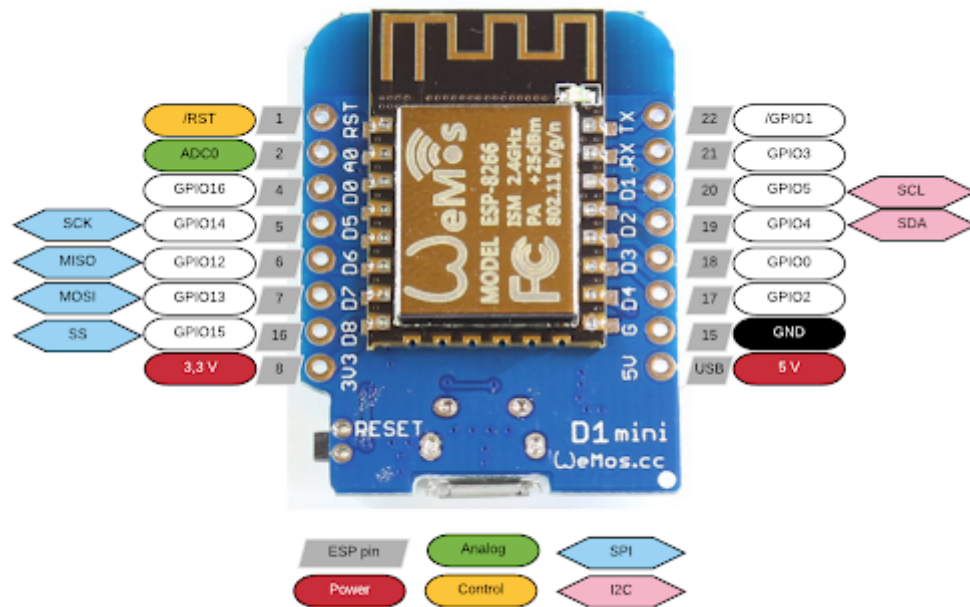
Ce même serveur Domoticz est lui interconnecté avec l'interface d'Alex de Amazon ou Google Home afin de pouvoir être interrogé Vocalement.

Datagramme D1 Mini

Cela reprend le Schéma de câblage pour le D1 Mini

Résumé de l'appareil

- Microcontrôleur: Tensilica 32 bits RISC CPU Xtensa LX106
- Tension de fonctionnement: 3,3 V
- Tension d'entrée: 7-12 V
- Broches d'E / S numériques (DIO): 11
- Broches d'entrée analogique (ADC): 1
- UART: 1
- SPI: 1
- I2C: 1
- Mémoire flash: 4 Mo
- SRAM: 64 Ko
- Vitesse d'horloge: 80 Mhz
- Wi-Fi: IEEE 802.11 b / g / n:
- Commutateur TR, balun, LNA, amplificateur de puissance et réseau d'adaptation intégré
- Authentification WEP ou WPA / WPA2 ou réseaux ouverts

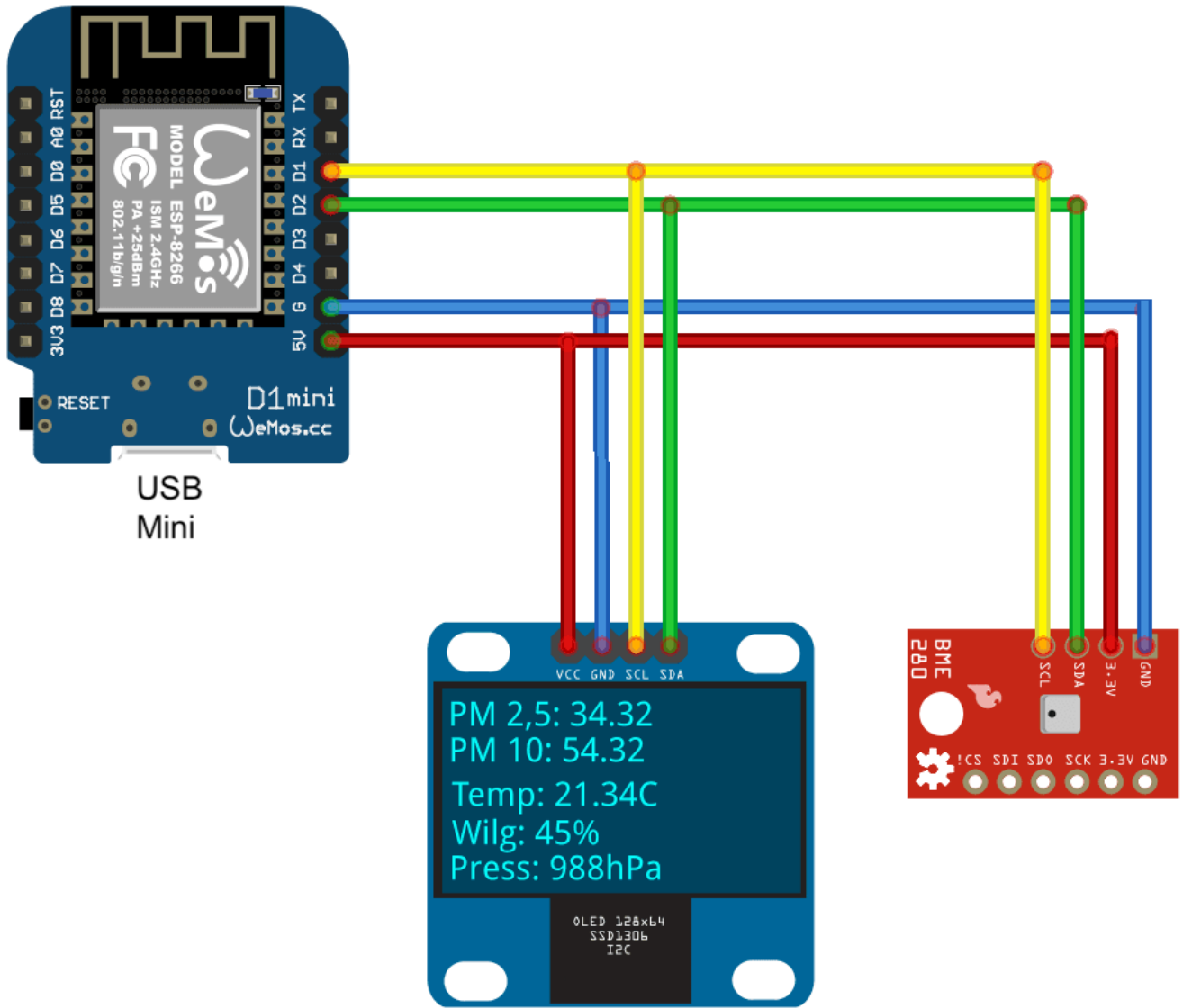


Cube Guard

Aperçu du module Auto suffisant grâce à son port mini USB, muni d'un module écran et d'un sensor Bosh BME-280 (Alti-Baro-Temp)



Schéma de Câblage



GPIO Paramètre

ESP Easy Mega: Cube.Guard

- △Main
- *Config
- ☰Controllers
- 📌Hardware**
- 📶Devices
- ⇒Rules
- 🔔Notifications
- 🔧Tools

Hardware Settings ?

Wifi Status LED

GPIO → LED:

Inversed LED:

Note: Use 'GPIO-2 (D4)' with 'Inversed' checked for onboard LED

Reset Pin

GPIO ← Switch:

Note: Press about 10s for factory reset

I2C Interface

GPIO ⇄ SDA:

GPIO → SCL:

Clock Speed: [Hz]

Note: Use 100 kHz for old I2C devices, 400 kHz is max for most.

Sensor Paramètre

Vue Générale

ESP Easy Mega: Cube.Guard

- △Main
- *Config
- ☰Controllers
- 📌Hardware
- 📶Devices**
- ⇒Rules
- 🔔Notifications
- 🔧Tools

	Task	Enabled	Device	Name	Port	Ctr (IDX)	GPIO	Values
Edit	1	✓	Display - OLED SSD1306/SH1106 Framed	OLED	I2C		SDA: 4 SCL: 5	
Edit	2	✓	Environment - BMx280	BME280	I2C	① (19) ② (25)	SDA: 4 SCL: 5	Temperature: 31.4 Humidite: 35 Pression: 1019

Paramètre Oled

ESP Easy Mega: Cube.Guard

- △Main
- *Config
- ☰Controllers
- 📌Hardware
- 🖱️Devices
- Rules
- 🔔Notifications
- 🔑Tools

Task Settings

Device: Display - OLED SSD1306/SH1106 Framed ? i

Name:

Enabled:

Display

Controller:

I2C Address:

Size:

Rotation:

Lines per Frame:

Scroll:

Display button:

Inversed Logic:

Display Timeout:

Contrast:

Content

Header:

Header (alternating):

Scroll long lines:

Wake display on receiving text:

Note: When checked, the display wakes up at receiving remote updates.

Line 1:	<input type="text" value="BAIE"/>
Line 2:	<input type="text" value="Watch"/>
Line 3:	<input type="text" value="v0.5"/>
Line 4:	<input type="text" value="%system_hm%"/>
Line 5:	<input type="text" value="[BME280#Temperature]°C"/>
Line 6:	<input type="text" value="[BME280#Humidite]%H"/>
Line 7:	<input type="text" value="Power"/>
Line 8:	<input type="text" value="By"/>
Line 9:	<input type="text" value="Me :o)"/>
Line 10:	<input type="text" value="[BME280#Temperature]°C"/>
Line 11:	<input type="text" value="Temp"/>
Line 12:	<input type="text" value="*Air*"/>

Interval: [sec]

Paramètre BME280

ESP Easy Mega: Cube.Guard

[Main](#)
[Config](#)
[Controllers](#)
[Hardware](#)
[Devices](#)
[Rules](#)
[Notifications](#)
[Tools](#)

Task Settings

Device: Environment - BMx280 ? i

Name:

Enabled:

I2C Address: [Detected: BME280]

Note: SDO Low=0x76, High=0x77

Altitude: [m]

Temperature offset: [x 0.1C]

Note: Offset in units of 0.1 degree Celcius (also correct humidity)

Data Acquisition

Send to Controller 1

IDX:

Send to Controller 2

IDX:

Send to Controller 3

Interval: [sec]

Values

#	Name
1	<input type="text" value="Temperature"/>
2	<input type="text" value="Humidite"/>
3	<input type="text" value="Pression"/>

Les Rules

Cette règle permet de transmettre au module N°6 (BoardGuard) via un réseaux propre P2P en UDP la valeur de sa température à afficher se ce module distant.

```
// SECTION SEND TEMP
On Clock#Time do
  SendTo 6,oled,5,1,RACKMATIC: [BME280#Temperature]0°
endon
```

Paramètre Contrôleur

Il sera important de relié votre module à votre serveur Domoticz si vous souhaitez grapher les mesures prises.

Procéder comme suit:

ESP Easy Mega: Cube.Guard

◁Main #Config **Controllers** Hardware Devices Rules Notifications Tools

	Nr	Enabled	Protocol	Host	Port
Edit	1	✓	Domoticz HTTP	192.168.0.30	8088
Edit	2	✓	Domoticz HTTP	192.168.0.30	8088
Edit	3	✓	ESPEasy P2P Networking	(IP unset)	0

Powered by Let's Control It community

Il vous faudra renseigner l'adresse ip, le port et le mot de passe du contrôleur.

ESP Easy Mega: Cube.Guard

◁Main #Config **Controllers** Hardware Devices Rules Notifications Tools

Controller Settings

Protocol: ?

Locate Controller:

Controller IP:

Controller Port:

Controller Queue

Minimum Send Interval: [ms]

Max Queue Depth:

Max Retries:

Full Queue Action:

Check Reply:

Client Timeout: [ms]

Credentials

Use Extended Credentials:

Controller User:

Controller Password:

Enabled:

Powered by Let's Control It community

Activer P2P Network

L'Activation de P2P Network vous permettra de transmettre des données entre vos modules.

ESP Easy Mega: Cube.Guard

[Main](#)
[Config](#)
[Controllers](#)
[Hardware](#)
[Devices](#)
[Rules](#)
[Notifications](#)
[Tools](#)

Controller Settings

Protocol: [?](#)

Enabled:

P2P Port Network

Valider le port 65500 pour les communications en UDP.

ESP Easy Mega: Cube.Guard

[Main](#)
[Config](#)
[Controllers](#)
[Hardware](#)
[Devices](#)
[Rules](#)
[Notifications](#)
[Tools](#)

Serial Settings

Enable Serial port:

Baud Rate:

Inter-ESPEasy Network

UDP port:

P2P View

Lorsque vous activez les réseaux P2P cela vous permet d'avoir un visuel de tous vos modules depuis l'interface Web de n'importe lequel de vos modules.

ESP Easy Mega: Cube.Guard

[Main](#)
[Config](#)
[Controllers](#)
[Hardware](#)
[Devices](#)
[Rules](#)
[Notifications](#)
[Tools](#)

System Info

Unit Number: 3
 Git Build: mega-20200608
 Local Time: 2020-07-10 23:34:51
 Uptime: 0 days 0 hours 16 minutes
 Load: 13.13% (LC=5685)
 Free Mem: 11472 (32032 -)
 Free Stack: 3680 (3824 -)
 IP Address: 10.200.60.8
 RSSI: -77 dB (IoT.2G)

Node List	Name	Build	Type	IP	Age
Unit 0	Iron.Guard	20107	ESP Easy Mega	10.200.60.8	0
Unit 1	Home.Guard	20107	ESP Easy Mega	10.200.60.9	1
Unit 2	Pool.Guard	20107	ESP Easy Mega	10.200.60.10	1
Unit 3	Cube.Guard	20107	ESP Easy Mega	10.200.60.11	1

Les Evolutions

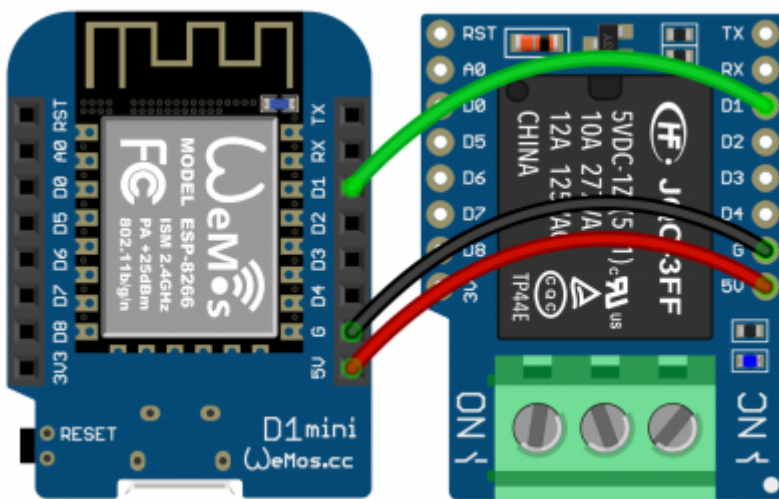


Évolution possible:

Rajouter un relais pour actionner ou non un extracteur d'air chaud en fonction du niveau de température intérieur.

L'ajout de Rules sera nécessaire pour contrôler le relais en fonction des paliers de température.

Câblage du relais



Peut aussi être détourné pour l'utilisation d'une VMC ou même un VELUX déjà Motorisé.

La Rules

Voici le type de Rules à mettre en place si sur votre module vous disposer d'un capteur de température type BME280 (alti/baro/temp).

```
on BME280#Temperature>25 do
GPIO,5,1
endon
on BME280#Temperature<=20 do
GPIO,5,0
endon
```



Votre relais n'a pas besoin d'être déclaré dans la partie devices pour être



commandé.

Explication



Si mon capteur de température BME280 dépasse 25° alors actionner le relais câblé sur le GPIO5. Si mon capteur de température BME280 repasse à 20° ou en dessous alors couper le relais câbler sur le GPIO5.

Cette règles est valide si votre capteur est nommé BME280 et si votre relais est bien câbler sur le GPIO5. A vous d'adapter en fonction de vos câblage sur votre module et déclaration dans ESPEasy.

— [sylvain](#) 2020/07/10 19:16

From:

<https://wiki.mazinger.fr/wiki/> - **My Personal Wiki**

Permanent link:

<https://wiki.mazinger.fr/wiki/doku.php?id=arduino:esp8266:ibss4>

Last update: **2024/03/03 12:56**

