



LoRa2HA

MLA30

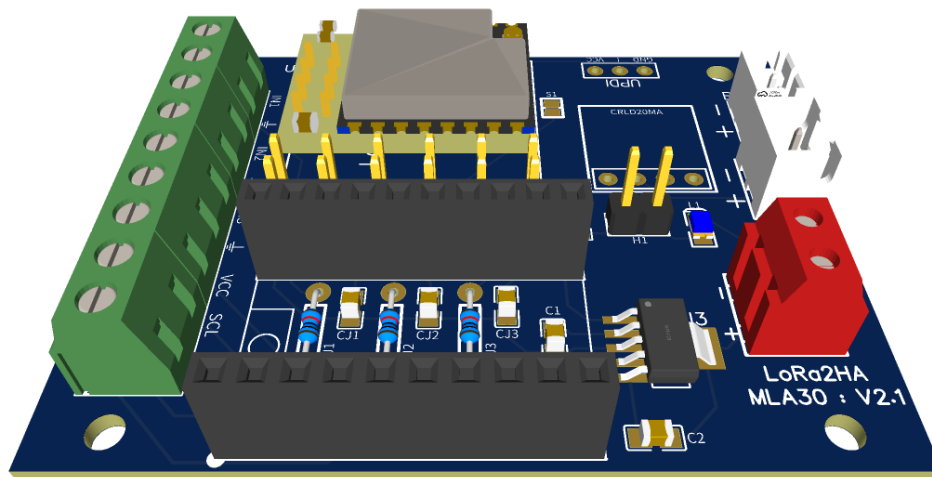


Table des matières

- 1 Description.....3
- 2 Caractéristiques techniques.....3
 - 2.1 Entrées.....3
 - 2.2 Alimentation.....4
 - 2.3 Module LoRa.....4
 - 2.4 LED.....5
 - 2.5 UPDI.....5
- 3 Montage de la carte.....5
 - 3.1 Schéma.....5
 - 3.2 Nomenclatures.....6
- 4 Programmation.....7
- 5 Utilisation.....7

1 Description

Le module MLA30 est principalement destiné à la mesure de capteurs, plus particulièrement des contacts, par exemple pour le suivi d'activité d'une boîte à lettres.

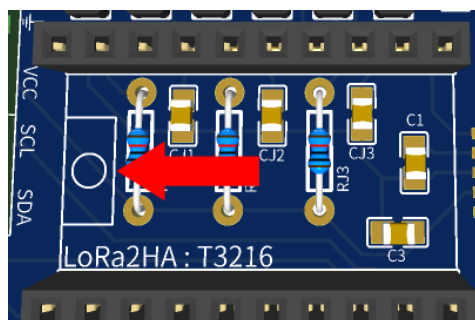
Il fonctionne sur batterie, il est donc autonome et n'a pas besoin d'une alimentation externe distante, seule une cellule solaire peut venir recharger la batterie.

2 Caractéristiques techniques

Le module doit fonctionner avec une carte d'extension **MLT3216**, c'est elle qui possède le microprocesseur.

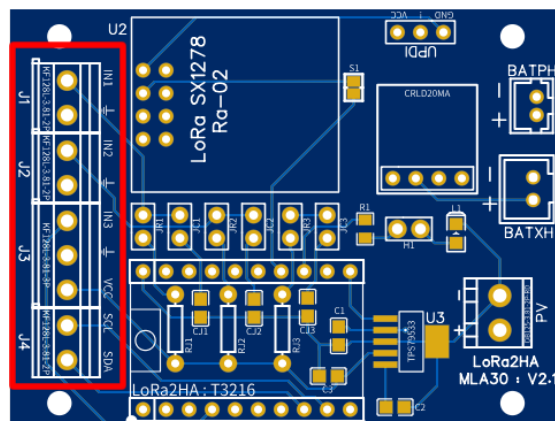
Pour les caractéristiques techniques du microprocesseur, voir la documentation de la carte MLT3216.

Attention de monter la carte MLT3216 dans le bon sens, en suivant la position du bouton CFG (un montage à l'envers n'est pas destructif, mais cela marchera beaucoup moins bien).



2.1 Entrées

La carte comporte 4 connecteurs à vis, permettant de brancher les différents capteurs.



- **J1, J2** : capteur binaire ou analogique
- **J3** : capteur binaire ou analogique, avec alimentation 3.3 V
- **J4** : capteur I2C

Les entrées J1, J2 et J3 peuvent fonctionner avec un « pont résistif » pour certains capteurs analogiques. L'activation du pont résistif se fait en positionnant un « jumper » sur les connecteurs **JR1, JR2, JR3**.

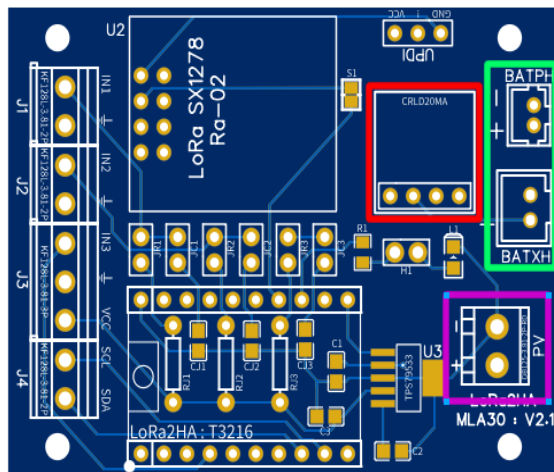
Les valeurs des résistances sont respectivement 4.7K Ω , 4.7K Ω , 1K Ω , connectée au VCC.

Exemple : Une thermistance CTN de 5K Ω est raccordée à J2, il lui faut un pont résistif afin de pouvoir mesurer la tension aux bornes de celle-ci ; il faut donc mettre un jumper sur JR2.

Dans le cas de certain contact, il peut exister un phénomène de « rebond » ; un petit condensateur de 100nF peut régler le problème ; il est activé en positionnant un jumper sur JC1, JC2 ou JC3.

Exemple : Un contact de porte est positionné en J1, si un « rebond » est détecté régulièrement, il faut positionner un jumper sur JC1 ?

2.2 Alimentation



L'alimentation s'effectue au travers de la carte **CRLD20MA** , celle-ci gère la charge de la batterie.

Connectez une batterie Lithium sur l'un des connecteurs **BATXH** ou **BATPH** , selon le type de connecteur qui équipe la batterie. Si la batterie n'a pas de connecteur JST (XH ou PH), il est possible de souder un connecteur à vis (pas 2.54) sur BATXH.

Une cellule solaire de 5V (maxi 500mA) peut être branchée au connecteur PV afin d'assurer la recharge de la batterie, lors de montages à forte consommation.

→ **Attention**, si vous avez besoin de relancer le module en débranchant la batterie, il faut laisser la batterie débranchée durant plusieurs secondes (> 30) car le module consomme tellement peu quand il est en veille, qu'il arrive à s'auto-alimenter durant plusieurs secondes par les composants internes.

Lors du branchement de la batterie, le LED doit faire un flash ; si il n'y a pas de flash c'est que le module est toujours en « veille », il faut laisser débrancher plus longtemps.

2.3 Module LoRa

Le module LoRa est un RA-02, en 433MHz, il est conseillé de lui connecter une antenne adaptée à la distance souhaitée ; généralement une antenne supérieure à 3 dBi permet de couvrir une distance d'au moins 500m.

2.4 LED

La led L1 permet de contrôler le fonctionnement du module, cela est utile lors de la mise en service mais totalement inutile par la suite et cela consomme de l'énergie supplémentaire. Pour que la led fonctionne, il faut positionner le jumper **H1**.

Lors de la mise sous tension, la LED doit faire 1 seul Flash ; si elle fait 3 flashes consécutifs c'est qu'il y a un défaut sur le module LoRa.

2.5 UPDI

C'est le connecteur qui permet la programmation du module, référez vous au chapitre « Programmation ».

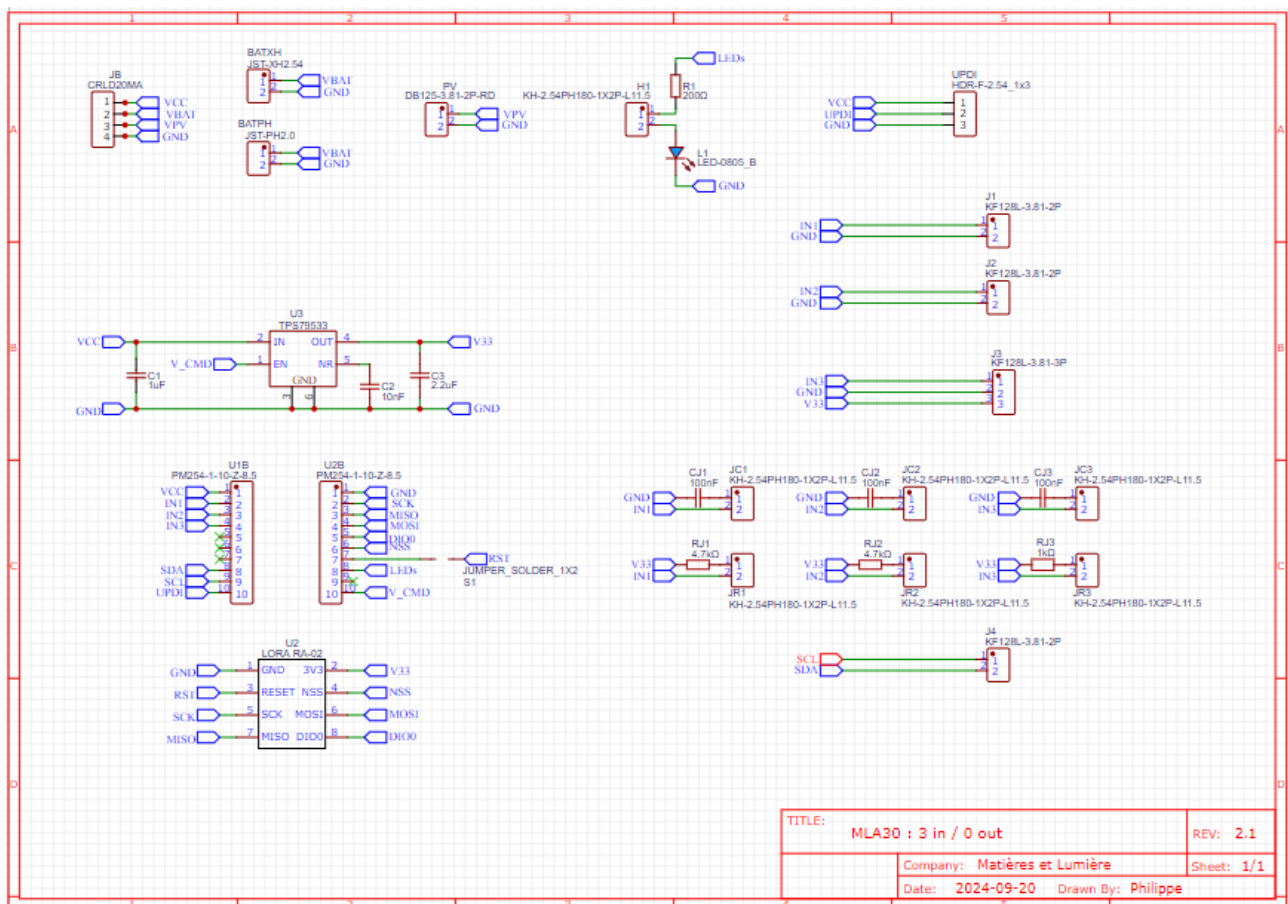
☒ Il faut débrancher l'alimentation principale lors de la programmation.

3 Montage de la carte

Il est conseillé de souder en premier les composants CMS puis ensuite les traversants.

☒ La carte processeur MLT3216 peut être soudée directement mais il est conseillé de la mettre sur support afin de faciliter la programmation de celle-ci lorsque le montage est finalisé et en production.

3.1 Schéma



3.2 Nomenclatures

Repère	Valeur/référence	Qté
C1	1uF (CMS 0805)	1
C2	10nF (CMS0805)	1
C3	2.2uF (CMS0805)	1
CJ1,CJ2,CJ3	100nF (CMS0805)	3
H1,JC1,JC2,JC3,JR1,JR2,JR3	2 Pin header mâle 2.54	7
J1,J2,J4	KF128L-3.81-2P	3
J3	KF128L-3.81-3P	1
L1	LED-0805_B	1
PV	DB125-3.81-2P-RD	1
R1	200 Ω (CMS0805)	1
RJ1,RJ2	4.7k Ω	2
RJ3	1k Ω	1
U1B,U2B	10 pin header femelle 2.54	2
U3	TPS79533	1
CRLD20MA	CRLD20MA	1
BATXH	JST XH 2.54 mâle	1
BATPH	JST PH 2.0 mâle	1

4 Programmation

La programmation utilise « MegaTinyCore », un projet Github prévu pour tous les microprocesseurs de la gamme Mega/Tiny de la série 0/1/2.

<https://github.com/SpenceKonde/megaTinyCore>

Dans le projet on trouve toute la documentation afin de transformer un simple Arduino (Uno, Nano) en programmeur UPDI. Il est aussi possible d'utiliser une carte USB/TTL, avec une résistance montée entre TX et RX. Les 2 possibilités sont détaillées dans la documentation MegaTinyCore.

Toutes les bibliothèques utiles sont incluses dans le package MegaTinyCore ou l'IDE Arduino, sauf la librairie dédiée à LoRa2HA : **Radiolink** (version > 2.0 obligatoire).

<https://github.com/PM04290/RadioLink>

☒ Si vous avez installé une ancienne version (< 2.0), il faut la re-télécharger.

La bibliothèque **MLiotElements** permet de gérer facilement tous les capteurs/actionneurs des modules. Elle n'est pas indispensable si vous créez votre propre programme, mais obligatoire si vous recompilez les firmwares par défaut.

Depuis la version 1.0 des HUB, ces derniers permettent la programmation des ATtiny3216 à l'aide du connecteur UPDI, voir le chapitre « Programmation UPDI » de la documentation principale.

5 Utilisation

Avec les Firmwares fournis dans le projet, lors de la première utilisation, la carte MLT3216 va s'initialiser avec des valeurs par défaut :

- Numéro (UID) = 30
- Numéro du Hub (HUID) = 0

Pour appairer le module avec le HUB, il faut :

1. maintenir le bouton CFG lors de la mise sous tension ; la LED va clignoter (1Hz)
2. relâcher le bouton CFG avant 10s (si vous tenez appuyé durant plus de 10s, les données internes vont être ré-initialisées)
3. la LED va s'allumer durant la transmission des informations vers le HUB (il doit être en mode « Active pairing »)
4. après transmission, la LED clignote lentement
5. lors de la fin du paramétrage par le HUB (Finish Pairing), le module passe en mode « normal », la LED s'éteint.