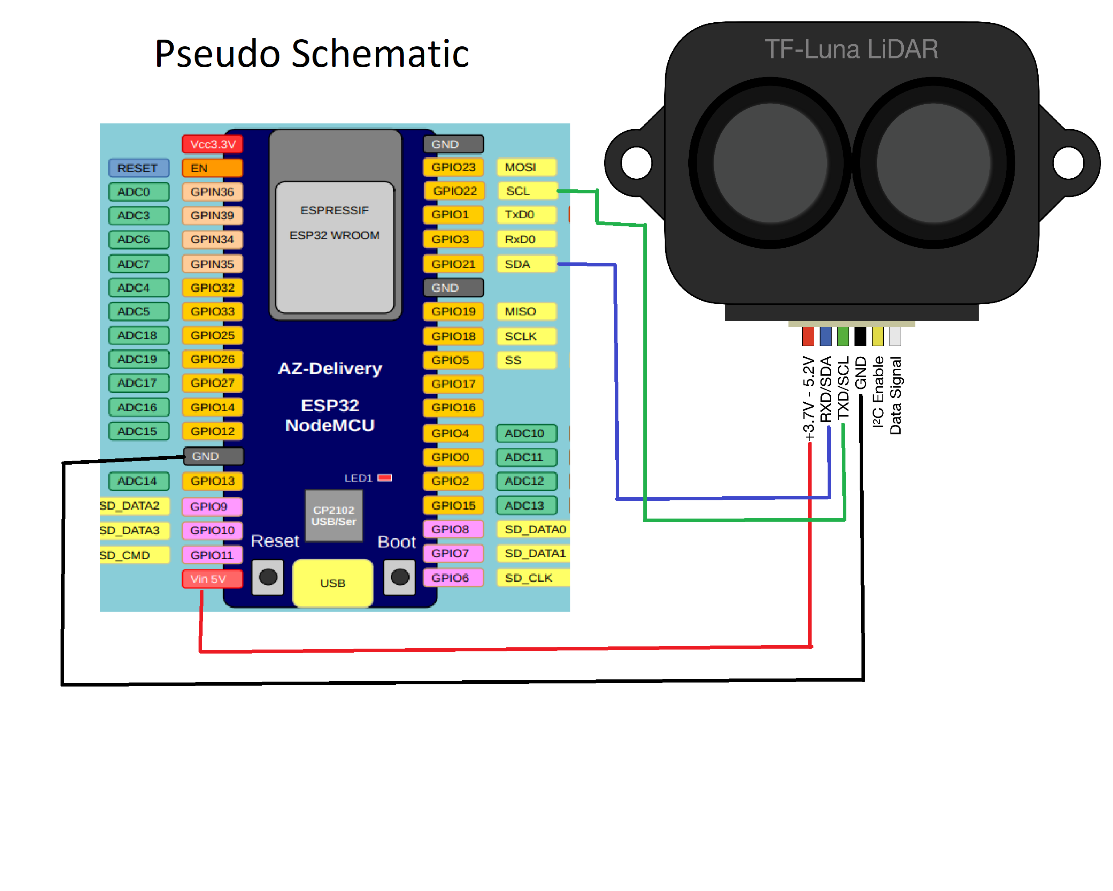
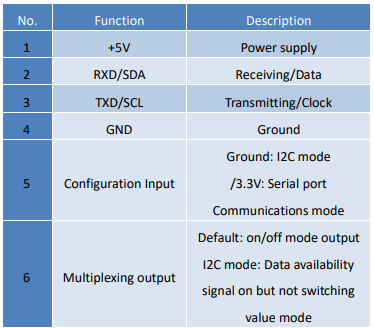
# **Résumé détection d’obstacle**

1. **Lidar TF-Luna – I2C**
2. La tension d’alimentation est 5V
3. La tension de communication est 3.3V – I2C (adresse 0x10)
4. Code + Circuit



**Code d’exemple « LUNA-TEST-I2C.ino » (sans modification)**

**Code d’exemple « LUNA-I2C.ino » (avec fonctions générales)**

**Code de temps d’éxecution « LUNA-temps-execution.ino » (programme de temps d’éxecution)**

1. **Ecriture des fonctions pour le programme général**

**Initialisation**

**TFLI2C luna;** //crée un objet TF-Luna appelé « luna »

**int16\_t tfAddr = TFL\_DEF\_ADR;** // default I2C address – uint8\_t

**uint16\_t tfFrame = TFL\_DEF\_FPS;** // default frame rate

**Setup**

**Wire.Begin() ;**

**luna.Soft\_Reset(adr)** // Reset, reboot et restart

**if ( luna.Soft\_Reset(adr) ) { LUNA\_OK }** //Détection d’erreur provenant du TF Luna

**else luna.printStatus() ;** //permet de connaitre l’état du TF Luna

**luna.Get\_Time( tfTime, adr);**  //Horloge du TF Luna sur un entier 16 bits

**luna.Set\_Frame\_Rate( tfFrame, adr) ;** //Envoie le nombre de frame/s, entier 16 bits

**tfFrame = FPS\_1 2 3 4 5 6 7 8 9 10 35 50  100  125 250 ou TFL\_DEF\_FPS=100**

**luna.Get\_Frame\_Rate( tfFrame, adr) ;** // Lecture du nombre de frame/s

**luna.Set\_Trig\_Mode(adr) ;** //Mode trigger du Luna

**luna.Set\_Cont\_Mode(adr) ;** //Mode continu du Luna

**luna.Save\_Settings(adr) ;**//Nécessaire après avoir changé le Frame-rate ou le Trig/Cont\_Mode ou

l’adresse I2C

**luna.Set\_I2C\_Addr(new adr, adr) ;** //Permet de changer l’adresse I2C du Luna

**Acquisition**

**luna.getData(dist, flux, temp, adr) ;** //Lit le Luna et change la valeur de la distance, du flux et de la

température qui sont des entier signés 16 bits.

Il faut diviser la température reçue par 100.

La fonction renvoie 0 si la mesure s’est bien passé (si l’adresse I2C de retour est OK), on doit faire if( GetData){ affectation des mesures } Else { réagir au problème }

**luna.getData(dist, adr) ;** //Comme précédemment mais seulement pour la distance

1. **Pins**

|  |  |
| --- | --- |
| 1. **+5V** | **+5V** |
| 1. **SDA** | **21** |
| 1. **SCL** | **22** |
| 1. **GND** | **GND** |
| 1. **Config I2C/Uart** | **GND** |
| 1. **Multiplex output** | **rien** |

**Librairies à utiliser**

|  |  |  |
| --- | --- | --- |
| **#include <Arduino.h>** | **#include <Wire.h>** | **#include <TFLI2C.h>** |

**Initialisation TF-Luna**

**TFLI2C luna;**

**int16\_t luna\_addr = TFL\_DEF\_ADR;**

**uint16\_t luna\_frame = FPS\_35;**

**int16\_t luna\_dist = 0 ;**

**int16\_t luna\_flux = 0 ;**

**int16\_t luna\_temp = 0 ;**

**Initialisation LED**

**#define led\_verte 32**

**#define led\_rouge 33**

**bool vert = HIGH;**

**Fonctions à utiliser**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom fonction** | **Rôle** | **Pire temps d’exécution** | **Meilleur temps d’exécution** |
| Luna\_Setup() | Setup du TF-Luna | **42ms/**62ms | **23ms** |
| Luna\_Measure() | Mesure de la distance, flux, et température | **23ms** | **20ms/**3ms |
| Luna\_OK() | Inverse l’état de la LED verte | **<1ms** | **<1ms** |
| Luna\_ERREUR() | Allume la LED rouge pendant 20ms | **21ms** | **20ms** |