



VL53L0X : Capteur de temps de vol.

1 Introduction

Le capteur de distance VL53L0X est un capteur temps de vol développé par ST Microelectronics, vous trouverez plus d'information à cette adresse :

<https://www.st.com/en/imaging-and-photonics-solutions/vl53l0x.html>

ST fournit une couche de driver logiciel pour ces capteurs disponible à cette adresse :

https://www.st.com/content/st_com/en/products/ecosystems/stm32-open-development-environment

Dans ce document je vous propose d'en tirer que quelques fichiers pour obtenir des premières mesures du capteur rapidement, **il y a sûrement moyen d'optimiser les choses pour tirer de meilleurs performances du capteur.**

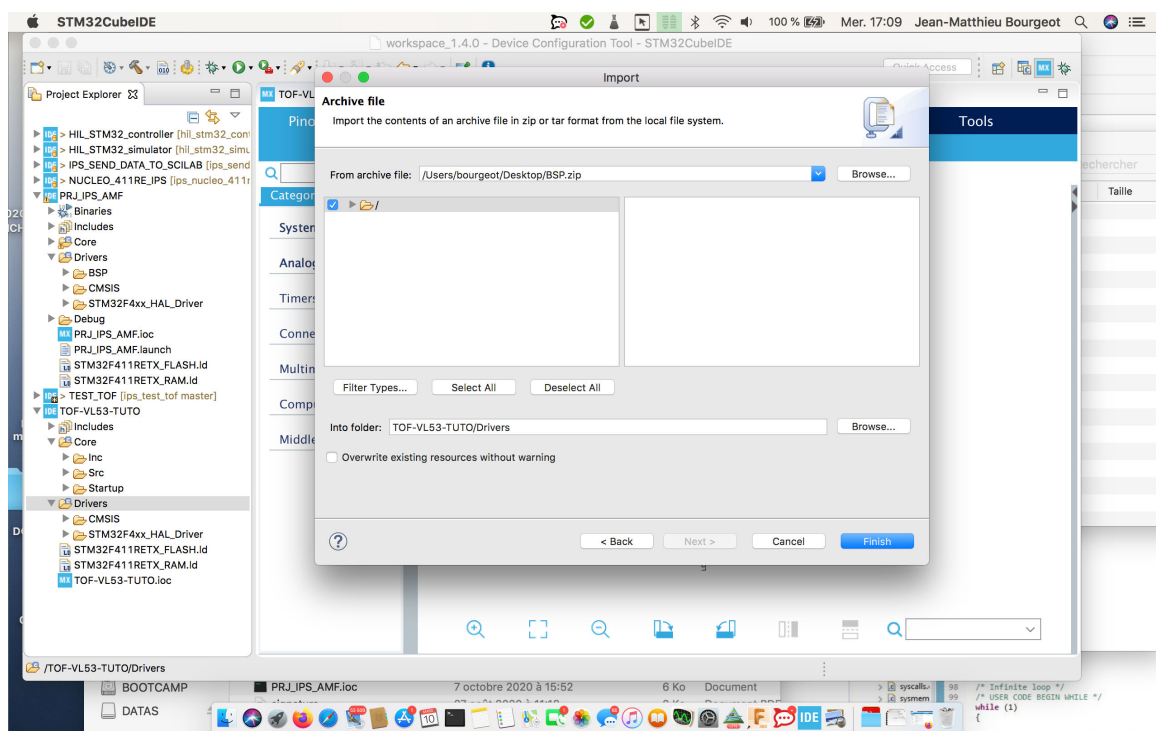
2 Procédure

Télécharger et décompresser l'archive disponible sur ma page personnel

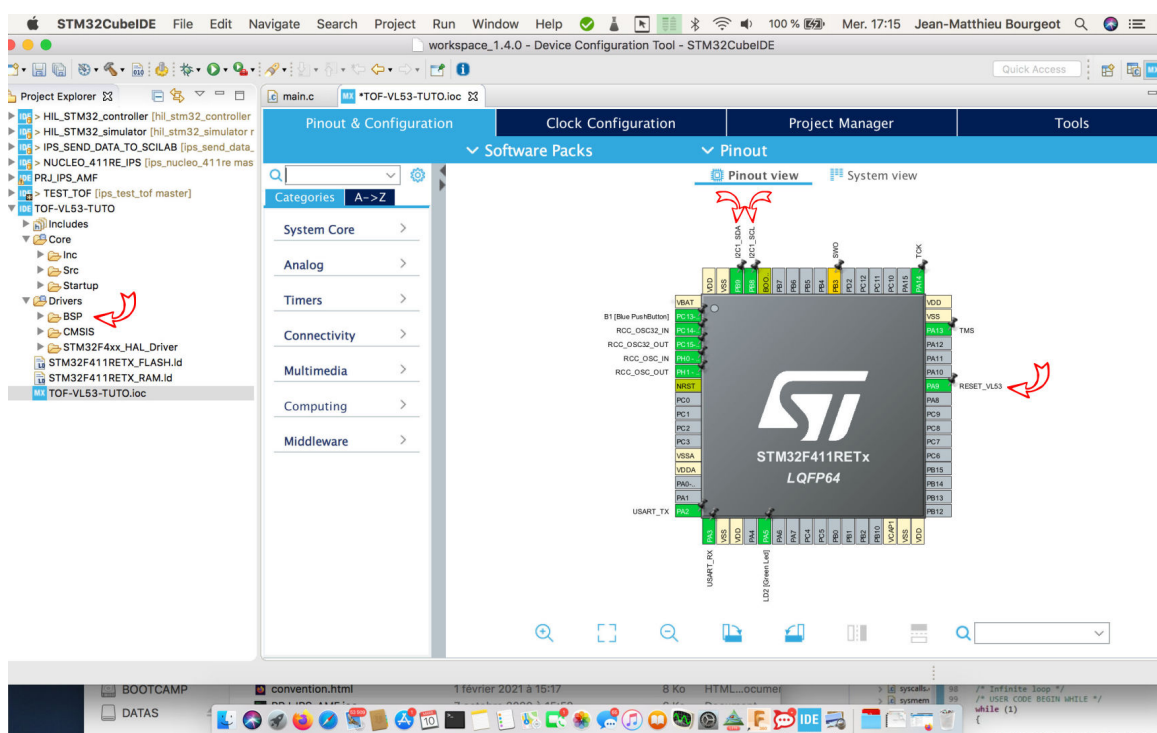
<https://www.enib.fr/~bourgeot/IPS/VL53L0X/>

Créer un projet pour notre carte NUCLEO-F411RE, dans les exemples des captures d'écran je l'ai nommé TOF-VL53-TUTO.

Dans l'explorateur de projet, sur le répertoire Drivers faites bouton droit Import -> Archive File, choisir l'archive BSPzip dans le répertoire de l'archive que vous venez de décompresser. (Voir figure ci dessous)



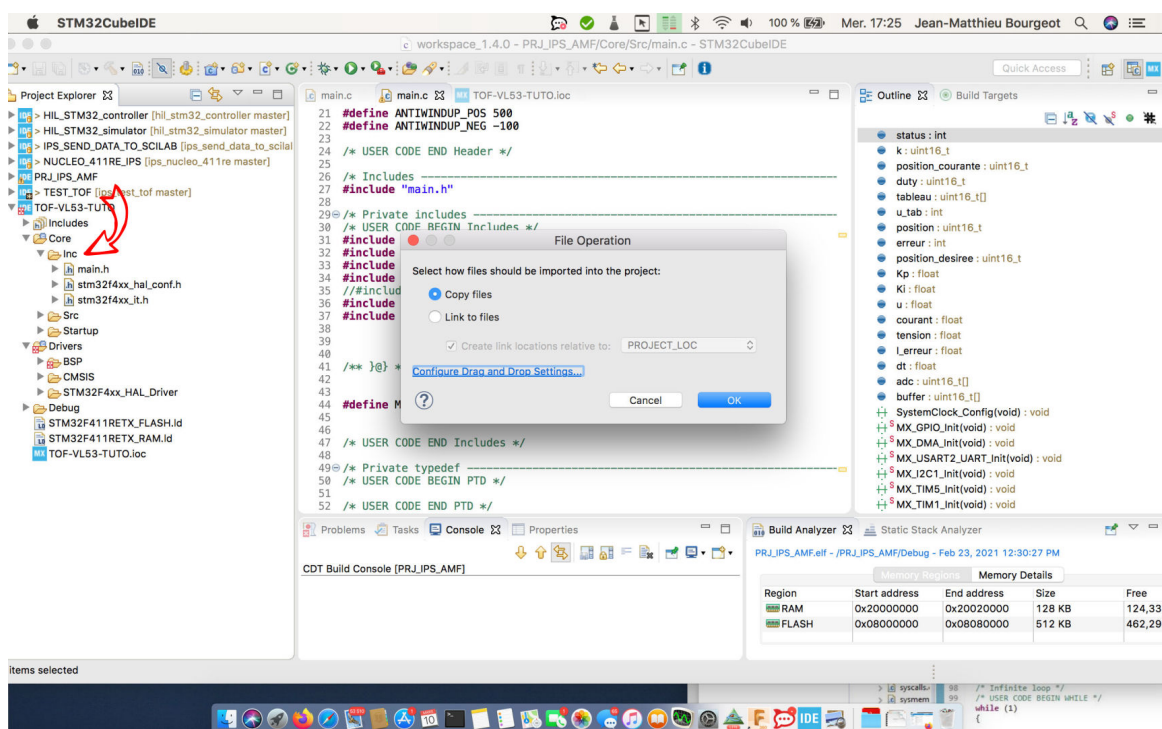
Un nouveau répertoire BSP apparaîtra dans le repertoire Drivers
Configurer les connexions de votre STM comme sur la capture d'écran ci contre.



- PA9 GPIO_output , user label : RESET_VL53
- Connectivity -> I2C1 -> I2C , choisir les pattes PB9 pour I2C1_SDA et PB8 pour I2C1_SCL

Puis générer le squelette de votre code Project - > Generate Code

Copier les fichiers mxconstants.h et stm32xxx_hal.h dans le repertoire Inc du project explorer (voir capture ci dessous).



Copier les morceaux de code suivant dans le `main.c` généré.

```

/* USER CODE BEGIN Includes */
#include "uart_trace.c"
#include "vl53l0x_api.h"
/* USER CODE END Includes */

...
/* USER CODE BEGIN PV */
int status=0;
uint16_t position_courante=0;
/* USER CODE END PV */

...
/* USER CODE BEGIN 1 */
RangingConfig_e RangingConfig = LONG_RANGE;
/* USER CODE END 1 */

...
/* USER CODE BEGIN 2 */
XNUCLE053L0A1_hi2c = hi2c1;
char msg2[]="main code";
F411_SetDisplayString(msg2);
ResetAndDetectSensor(0);
F411_SetDisplayString("debut while");
/* Reset and Detect all sensors */
ResetAndDetectSensor(0);
SetupSingleShot(RangingConfig);
/* USER CODE END 2 */

...
/* USER CODE BEGIN 3 */
/* only one sensor */
/* Call All-In-One blocking API function */
status = VL53L0X_PerformSingleRangingMeasurement(&VL53L0XDevs[0],
                                                    &RangingMeasurementData);
position_courante=RangingMeasurementData.RangeMilliMeter;
if( status ==0 ){

    trace_printf("\r\n%d", position_courante);
    // Sensor_SetNewRange(&VL53L0XDevs[0],&RangingMeasurementData);
}
}
/* USER CODE END 3 */

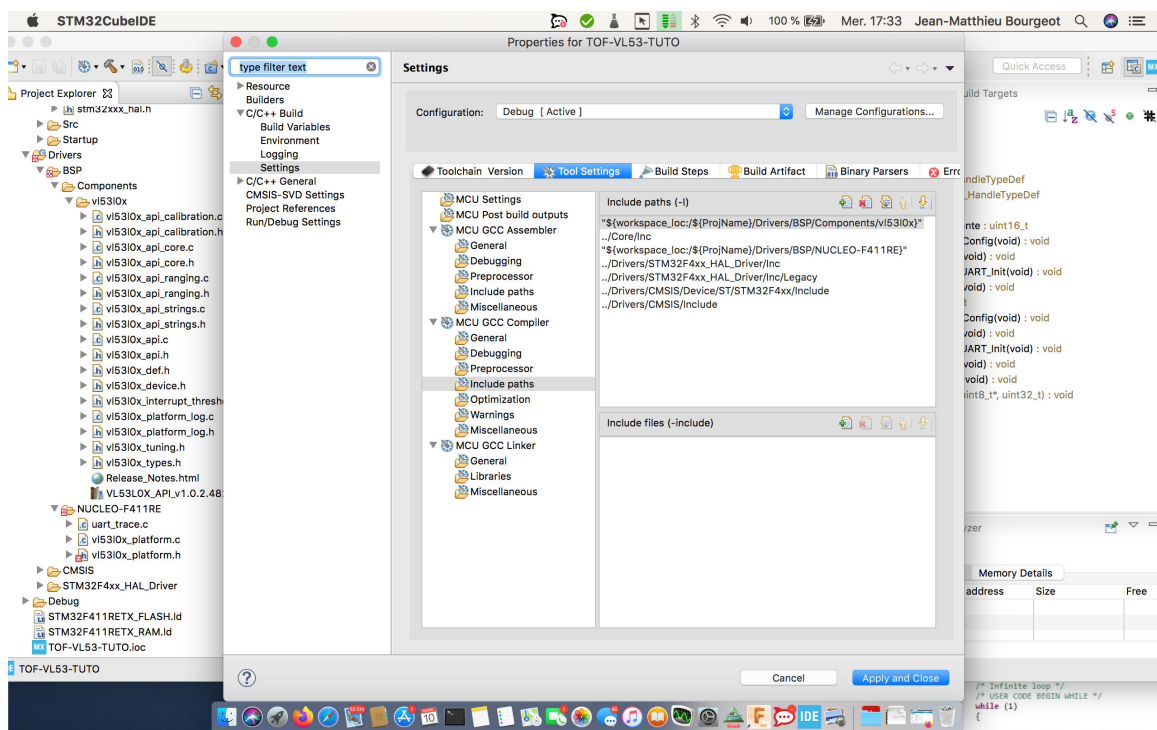
```

Ajouter les fichiers du driver à votre *path* pour la compilation

Project -> Properties -> C/C++ Build -> Setting -> Tool Settings ->

MCU GCC Compiler -> Include paths ,

de type Workspace pour les répertoire BSP/Components/vl53l0x et BSP/NUCLEO-F411RE (voir capture ci dessous).



Connecter physiquement le capteur à votre carte STM32 NUCLEO F-411RE

Coté Capteur	Coté NUCLEO
VCC	3V3
GND	GND
SCL	PB8
SDA	PB9
GPIO1	n.c.
XSHUT	PA9

Pour tester le programme en mode debug vous pouvez visualiser la variable `position_courante` dans l'onglet Live Expressions (en cliquant sur *Add new expression*). Si la fenêtre Live expression n'est pas disponible il suffit de la faire apparaître avec Windows -> Show View -> Live Expression.

