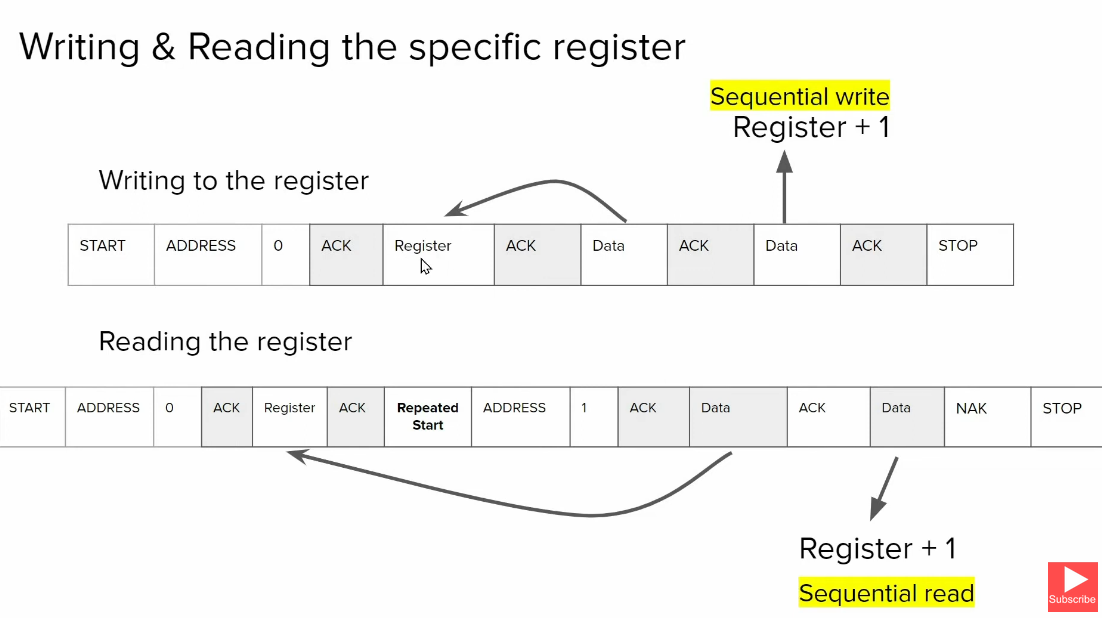
ODHS I2C connection with MPU6050

# Problem statement

*Read from the who am I register of the gyroscope(mpu6050) using I2C protocol using STM32F407VGT6*

# theory of i2c read and write

in i2c generally sequential write and read is supported (as it is in the case of mpu6050).



## To write to a specific register:-

* start bit
* address of slave
* R/W bit (=0 to write)
* Get Ack from slave
* Send 1st register address to be written to
* Send data (the data is written to the said register)
* If any following data is sent, it is sequentially written onto register+i th register
* Stop bit

## To read data from a register:-

* Start bit
* Address of slave
* R/W bit =0
* Get ack from slave
* Send 1st register address to be read
* Ack from slav
* Then resend start bit
* Address of the slave followed by R/W bit =1 (to read mode)
* Slave send ack and sends the data on that register
* If master sends back ACK then the slave sends the data in register +1 . this keeps happening till when master keeps sending ack
* To stop the conversation, master send a NAK followed by stop bit

# CPAL

CPAL in STM32 refers to the Communication Peripheral Abstraction Layer. It's a software layer provided by STMicroelectronics, specifically designed to abstract the low-level details of communication peripherals, such as I2C (Inter-Integrated Circuit), SPI (Serial Peripheral Interface), and UART (Universal Asynchronous Receiver-Transmitter), in STM32 microcontrollers.

The CPAL simplifies the development process by providing a standardized API (Application Programming Interface) that allows developers to interact with various communication peripherals without needing to worry about the intricate details of register settings and low-level configurations. Instead, developers can focus on higher-level application logic.

By using CPAL, developers can write portable code that can easily be adapted to different STM32 microcontrollers without significant modifications. This abstraction layer enhances code reusability and accelerates the development process for applications that require communication with external devices or other microcontrollers.

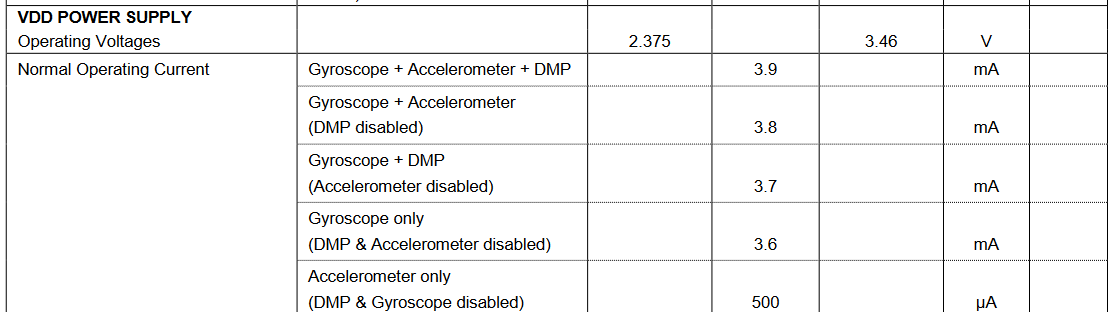
To download CPAL files required for this follow the link:-

<https://docs.google.com/document/d/1XgmuZdV3H02oNoKyO8WgNbg7TIMane0wnFkh-nzFZkI/edit>

# MPU6050

The MPU-60X0 is an integrated 6-axis MotionTracking device that combines a 3-axis gyroscope, 3-axis accelerometer, and a Digital Motion Processor™ (DMP) and a dedicated I2C sensor bus.

Electrical specifications:-

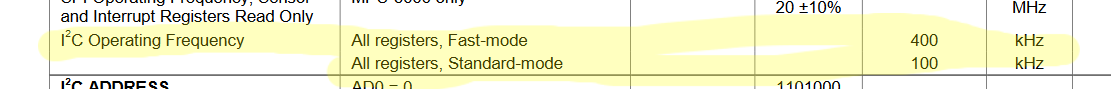


Part 6.3 of MPU6050 Datasheet

* VDD can vary from 2.375V to 3.46V and is Thus tied to the 3V output of the STM32F4.. board

## I2C specifications:-

### Operating Frequency

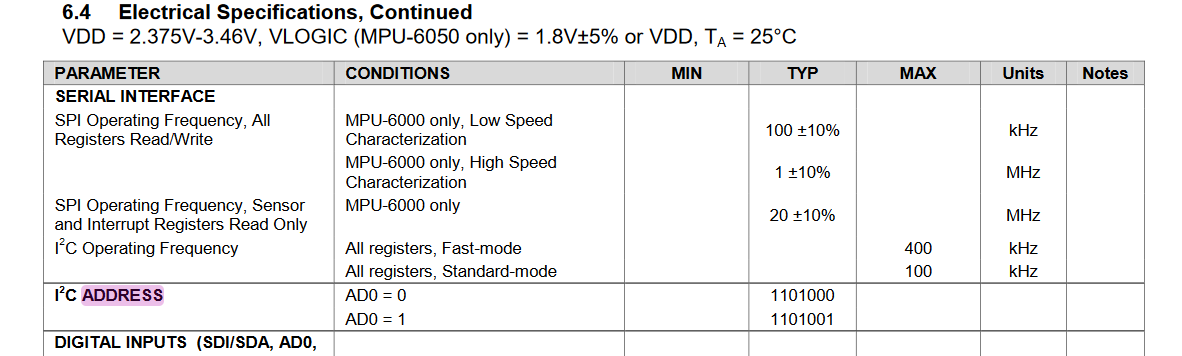


In fast mode the MPU can do upto 400kHz and in standard mode it can do upto 100kHz

* For this project standard mode is used at a frequency of 100000 Hz i.e. 100kHz

### Address of MPU6050

The address of MPU6050 is :-



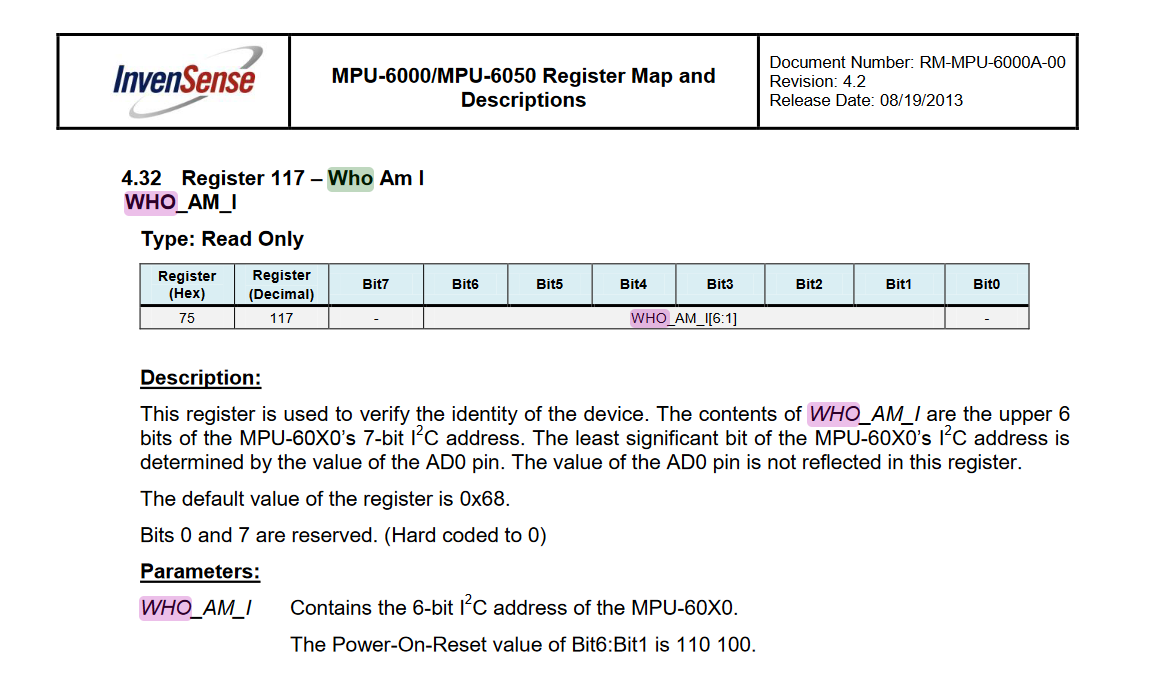
When AD0=0 address= 1101000 (using this)

When AD0=1 address= 1101001

* The AD0 pin is connected to GND to make it 0 and thus the Address of the MPU6050 = (0b110100<<1 +0)

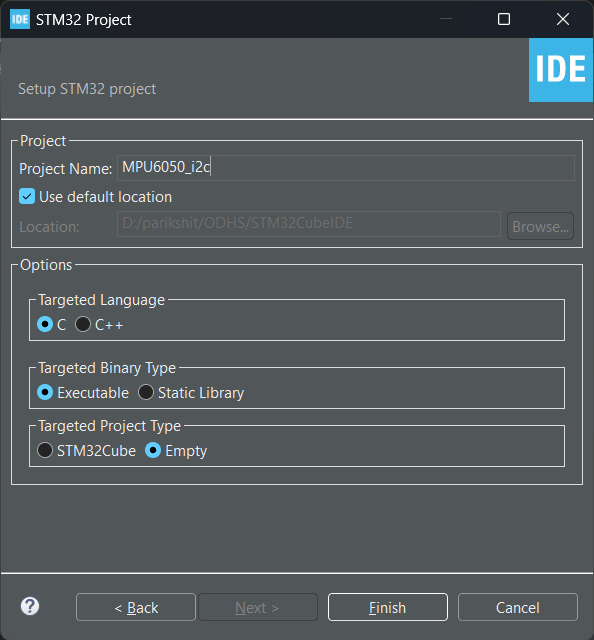
### Register to be accessed

Register 117- Who AM I is to be accessed



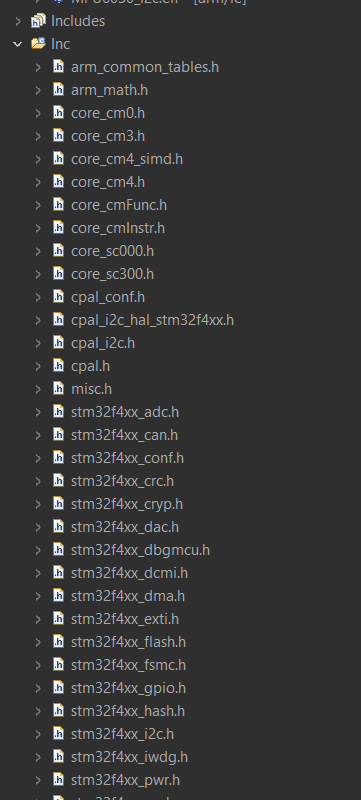
# Setting up the project

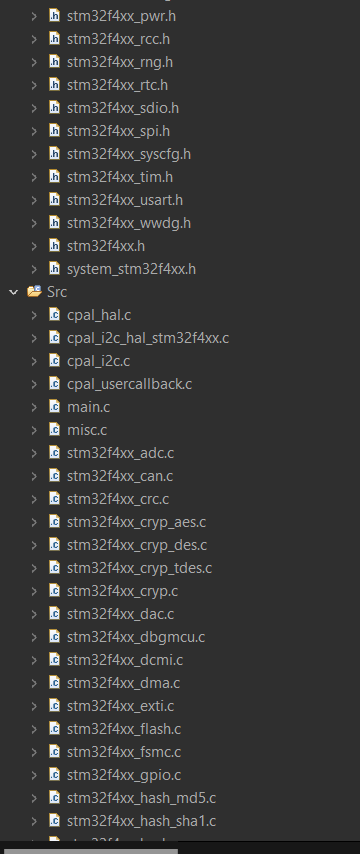
* In STM32cudeIDE select STM32F407VGT6 and create an EMPTY project

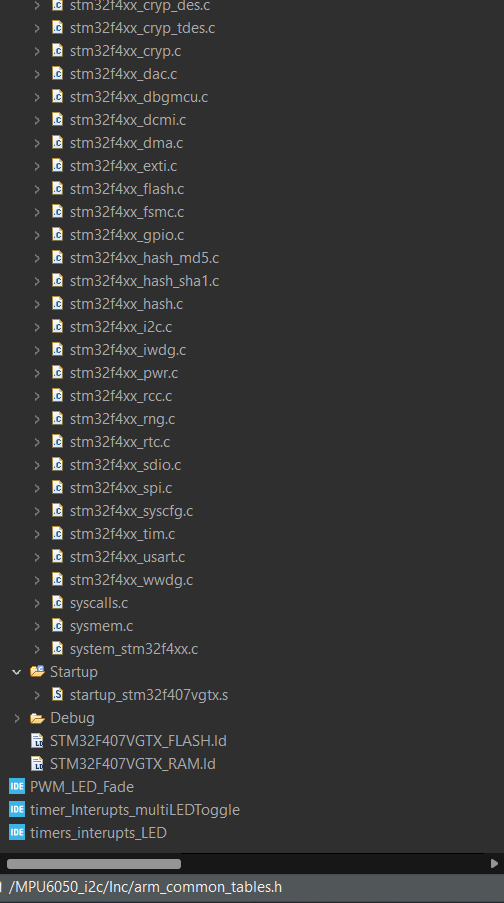


* Add the required CPAL Files, headers .h into inc, .c into src and .s into startup

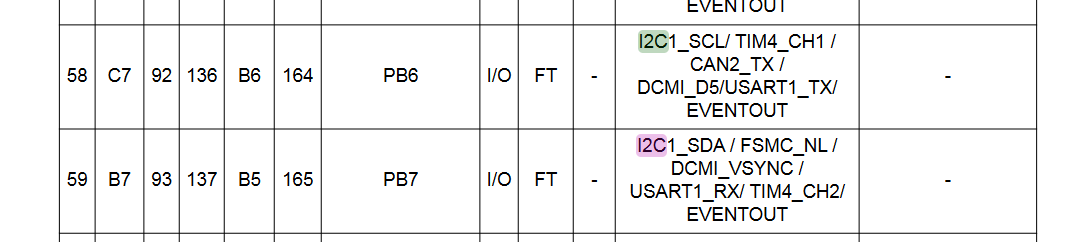
The file structure will look something like this :-

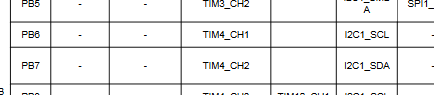






According to the datasheet of stm32f407xx (table 9) Alternate Function of pins PB6 and PB7





* Pin chosen for i2c communications:- PB6 = I2c1 SCL line & PB7 = i2c1 sda line

# Implementation

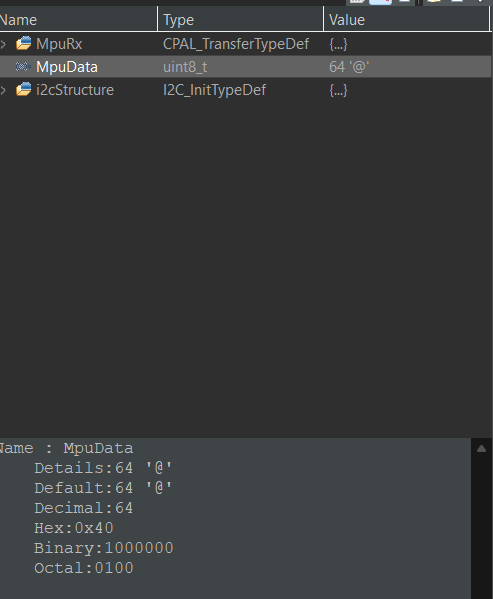
Note:- the main.c had this warning

**#if** !defined(\_\_SOFT\_FP\_\_) && defined(\_\_ARM\_FP)

**#warning** "FPU is not initialized, but the project is compiling for an FPU. Please initialize the FPU before use."

**#endif**

Explaination:- <https://community.st.com/t5/stm32cubeide-mcus/quot-warning-quot-fpu-is-not-initialized-quot-version-1-9-0/td-p/118382>



references:-

<https://www.youtube.com/playlist?list=PLmXXQ1iFwiyKKURU4wAeGT_d1HsIe1YCY>

datasheet:-

<https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>

register map:-

<https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Register-Map1.pdf>

CPAL documentation:-

<https://www.st.com/en/embedded-software/stsw-stm32127.html#documentation>