

IMU

IMU é um dispositivo, portanto, um conjunto de componentes que tem como intenção final indicar a posição/velocidade angular de um mecanismo. Seu nome pode ser traduzido como dispositivo de medição inercial.

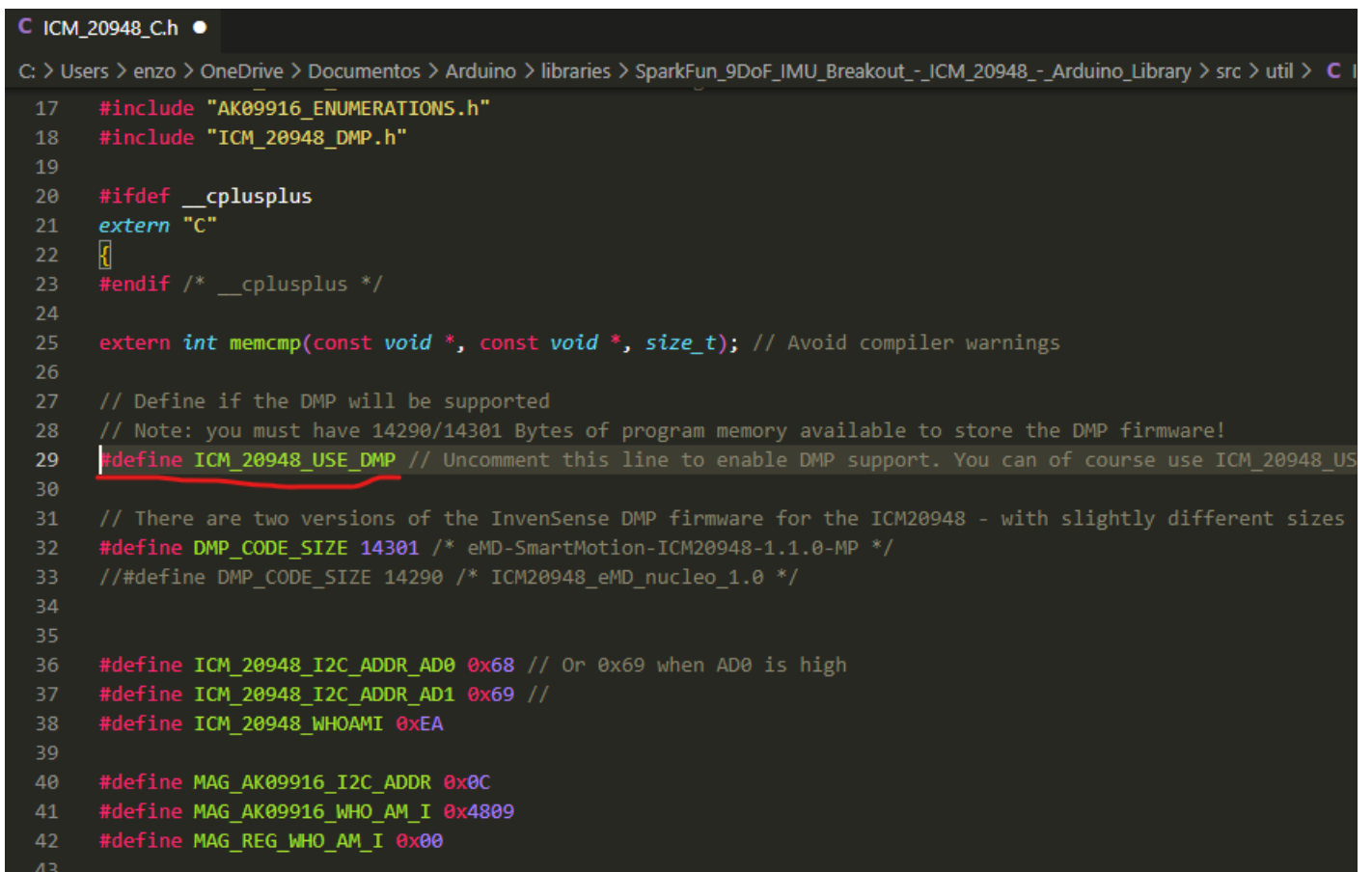
Para utiliza-lo é necessário editar um arquivo da biblioteca do Arduino IDE.

Faça o seguinte caminho em seu computador:

**C:\Users\“User”\Documentos\Arduino\libraries\SparkFun_9DoF_IMU_Breakout_-
ICM_20948-Arduino_Library\src\util**

Abra o arquivo ICM_20948_C.h, caso ele peça algum aplicativo para abrir, use o editor de texto do windows.

Por fim, descomente (remova os caracteres //) da seguinte linha.



```
C ICM_20948_C.h
C: > Users > enzo > OneDrive > Documentos > Arduino > libraries > SparkFun_9DoF_IMU_Breakout_-ICM_20948-Arduino_Library > src > util > C I
17 #include "AK09916_ENUMERATIONS.h"
18 #include "ICM_20948_DMP.h"
19
20 #ifdef __cplusplus
21 extern "C"
22 {
23 #endif /* __cplusplus */
24
25 extern int memcmp(const void *, const void *, size_t); // Avoid compiler warnings
26
27 // Define if the DMP will be supported
28 // Note: you must have 14290/14301 Bytes of program memory available to store the DMP firmware!
29 #define ICM_20948_USE_DMP // Uncomment this line to enable DMP support. You can of course use ICM_20948_US
30
31 // There are two versions of the InvenSense DMP firmware for the ICM20948 - with slightly different sizes
32 #define DMP_CODE_SIZE 14301 /* eMD-SmartMotion-ICM20948-1.1.0-MP */
33 // #define DMP_CODE_SIZE 14290 /* ICM20948_eMD_nucleo_1.0 */
34
35
36 #define ICM_20948_I2C_ADDR_AD0 0x68 // Or 0x69 when AD0 is high
37 #define ICM_20948_I2C_ADDR_AD1 0x69 //
38 #define ICM_20948_WHOAMI 0xEA
39
40 #define MAG_AK09916_I2C_ADDR 0x0C
41 #define MAG_AK09916_WHO_AM_I 0x4809
42 #define MAG_REG_WHO_AM_I 0x00
43
```

Agora em nosso código podemos fazer o seguinte.

```
#include <Arara.h>
```

```
IMU imu;
```

```
void setup() {
```

```
  // put your setup code here, to run once:
```

```
  imu.init();
```

```
}
```

```
void loop() {
```

```
  // put your main code here, to run repeatedly:
```

```
  Serial.print("Pitch: ");
```

```
  Serial.println(imu.getPitch());
```

```
  Serial.print("Roll: ");
```

```
  Serial.println(imu.getRoll());
```

```
  Serial.print("Yaw: ");
```

```
  Serial.println(imu.getYaw());
```

```
}
```

Revisão #1

Criado 19 agosto 2024 12:07:56 por Enzo Coutinho

Atualizado 19 agosto 2024 12:15:14 por Enzo Coutinho