



Universidade do Minho
Escola de Engenharia

MESTRADO INTEGRADO EM ENGENHARIA DE TELECOMUNICAÇÕES E INFORMÁTICA

LABORATÓRIOS DE TELECOMUNICAÇÕES E INFORMÁTICA II

SISTEMA DE MONITORIZAÇÃO DE ATIVIDADE FÍSICA FASE A

Grupo 2:

David José Ressurreição Alves - A79625

José Pedro Afonso Rocha - A70020

Luís Pedro Lobo de Araújo - A73232

Guimarães, 28 de Junho de 2019

Índice

1	Diagrama de blocos da arquitetura do sistema desenvolvido na fase A	3
2	Diagramas dos formatos das tramas do protocolo	4
3	Referências às bibliotecas e aos códigos exemplo	6

Lista de Figuras

1	Diagrama de blocos da arquitetura do sistema	3
2	Definição da trama DATA.	4
3	Definição da trama ERROR.	5
4	Definição da trama START.	5
5	Definição da trama STOP.	5

1. Diagrama de blocos da arquitetura do sistema desenvolvido na fase A

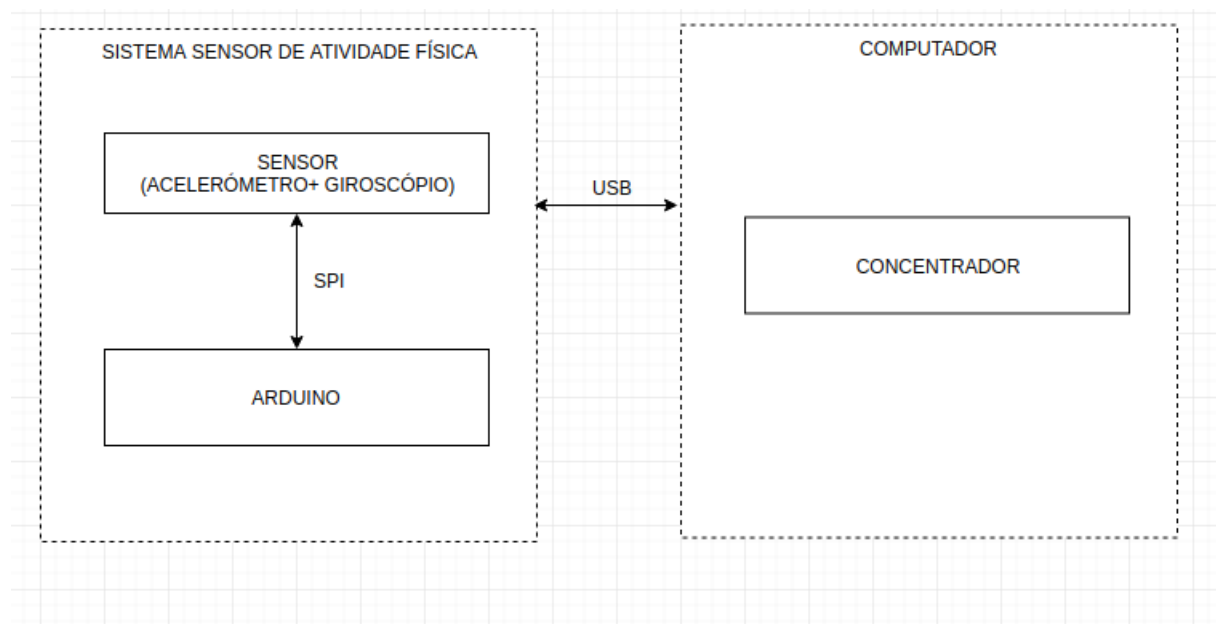


Figura 1: Diagrama de blocos da arquitetura do sistema

Para desenvolver o código do concentrador utilizámos funções presentes numa biblioteca encontrada *online* [1].

Para desenvolver o código Arduino baseamos-nos num código encontrado *online* [2] da biblioteca MPU6050_tockn [3].

2. Diagramas dos formatos das tramas do protocolo

- **DATA** - mensagem enviada do Arduino para o concentrador com os valores das amostras. Tal como se pode observar, definimos 1 byte(8 bits) para identificar o tipo de mensagem que esta a ser recebida/enviada, neste caso para o tipo DATA, definimos o valor de 0. Os restantes campos da trama foram definidos de acordo com o tipo e tamanho dos dados que estão a ser enviados/recebidos.

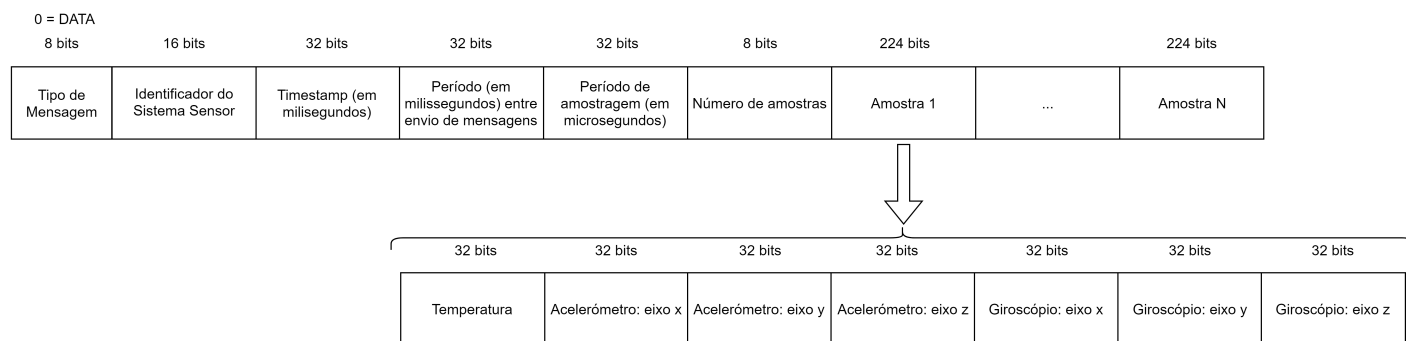


Figura 2: Definição da trama DATA.

- **ERROR** – mensagem enviada do Arduino para o concentrador a indicar uma condição de erro. Para esta trama, definimos 1 byte(8 bits) para identificar o tipo de mensagem que esta a ser recebida/enviada, neste caso para o tipo ERROR, definimos o valor de 1. Os restantes campos da trama foram definidos de acordo com o tipo e tamanho dos dados que estão a ser enviados/recebidos.

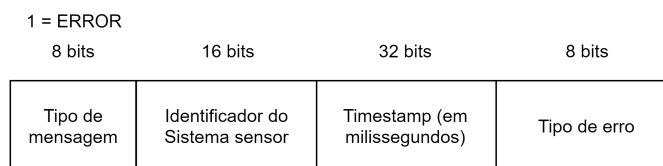


Figura 3: Definição da trama ERROR.

- **START** – mensagem enviada do concentrador para o Arduino a pedir o início da recolha e envio das amostras. Para esta trama, definimos 1 byte(8 bits) para identificar o tipo de mensagem que esta a ser recebida/enviada, neste caso para o tipo START, definimos o valor de 2. Os restantes campos da trama foram definidos de acordo com o tipo e tamanho dos dados que estão a ser enviados/recebidos.

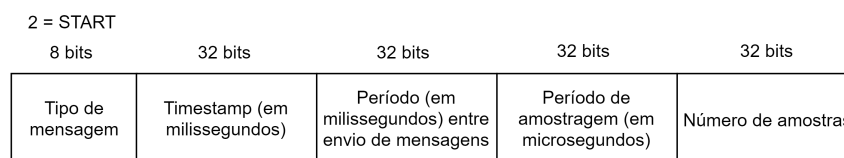


Figura 4: Definição da trama START.

- **STOP** – mensagem enviada do concentrador para o Arduino a pedir o fim da recolha e envio das amostras. Para esta trama, definimos 1 byte(8 bits) para identificar o tipo de mensagem que esta a ser recebida/enviada, neste caso para o tipo STOP, definimos o valor de 3. Os seguintes campos da trama foram definidos de acordo com o tipo e tamanho dos dados que estão a ser enviados/recebidos.



Figura 5: Definição da trama STOP.

3. Referências às bibliotecas e aos códigos exemplo

[1] **"Marzac/rs232", GitHub,**

Disponível em: <https://github.com/Marzac/rs232>. [Acedido em 16 de março 2019].

[2] **"tockn/MPU6050_tockn", GitHub,**

Disponível em: https://github.com/tockn/MPU6050_tockn/blob/master/examples/GetAllData/GetAllData.ino. [Acedido em 16 de março 2019].

[3] **"tockn/MPU6050_tockn", GitHub,**

Disponível em: https://github.com/tockn/MPU6050_tockn. [Acedido em 16 de março 2019].