

### Trabalho Prático 3 (*quatro aulas*) (2º Trabalho Prático de Avaliação)

**Software adicional para desenvolver este trabalho:**  
GCC (MinGW), ...

Pretende-se que cada grupo acrescente funcionalidades aos sistemas AQC1 e AQC2 do trabalho anterior e que desenvolva uma aplicação de Monitorização e Registo (MR) em C para correr no PC.

#### **O sistema AQC1 (PIC16) do trabalho anterior deverá:**

- transmitir periodicamente (de um em um minuto) pela UART (9600bits/s) de uma mensagem XML com os valores dos LDR (deverá utilizar um *timer*, associado a uma rotina de atendimento de *interrupts*, para contar o tempo);
- alterar do estado do sistema (para normal ou *standby*) sempre que um botão de pressão, associado a uma rotina de atendimento de *interrupts*, seja pressionado;

#### **O sistema AQC2 (PIC24) do trabalho anterior deverá, sempre que muda do modo *standby* para normal, comunicar por I2C com um sistema externo (um *slave* com o endereço 0x0A):**

- o AQC2 deverá enviar o valor 0x07 para pedir o valor do “Limite de Diferença” de luminosidade entre os LDR;
- o *slave* envia um byte com um valor entre 20 e 200 (com o limite da diferença de luminosidade), ou um valor fora desse intervalo se ERRO;
- ver página 25 do datasheet “24-I2C 39702b.pdf”;

#### **A aplicação de MR (no PC), será responsável por:**

- receber as mensagens XML enviadas pelo sistema AQC1 de um em um minuto;
- apresentar as mensagens recebidas na consola;
- reencaminhar as mensagens recebidas para um Servidor Remoto (<http://193.136.120.133/~sad/>), através de *HTTP POST*. As mensagens recebidas pelo servidor são registadas num ficheiro com a data do próprio dia (<http://193.136.120.133/~sad/2021-MM-DD.log>);
- (é provável que o antivírus do PC bloqueie a aplicação de MR; se bloquear será necessário dar permissão)

#### **Cada grupo tem liberdade para decidir:**

- se pretende melhorar as funcionalidades propostas, implementar funcionalidades extra ou apenas parte...

#### **Entrega e discussão do trabalho:**

Este trabalho deverá entregue em formato ZIP (projectos e relatório em PDF) na página da disciplina no Moodle até ao dia 17 de Junho. O trabalho será apresentado e discutido em Junho, de acordo com um calendário a publicar na página da disciplina no Moodle.

### Informação adicional

[1] Exemplo de código C (para Windows) que envia dados pela porta COM (o objectivo neste trabalho não é receber, é enviar), <https://batchloaf.wordpress.com/2013/02/13/writing-bytes-to-a-serial-port-in-c/>

[2] Exemplo de código C (para Windows) onde é feito um HTTP GET (o objectivo neste trabalho não é fazer GET, mas sim POST), <https://www.binarytides.com/winsock-socket-programming-tutorial/>. Para gerar o executável deste exemplo de código, utilizando o MinGW, através da linha de comandos é necessário inserir "-lws2\_32", por exemplo: gcc main.c -o main.exe -lws2\_32

[3] Online REST & SOAP API Testing Tool, <https://reqbin.com/>