

## Trabalho Prático 3 – Aplicação para um Sistema Embarcado

### 1. Descrição Geral

Este trabalho visa o desenvolvimento de uma aplicação para um sistema embarcado, através de uma linguagem de programação de baixo nível. O principal objetivo é promover ao grupo a experiência de programação em um dispositivo embarcado, de forma que diversos elementos da aplicação (tais como tratamento de interrupções, comunicação serial, etc.), deverão ser considerados e tratados para que o sistema funcione de forma adequada.

A aplicação será executada em um simulador, chamado PICsimLab (PIC Simulator Laboratory), o qual permite a execução de aplicações para diferentes modelos de placas. Mais informações sobre o PICsimLab podem ser encontradas no link a seguir: <https://www.embarcados.com.br/picsimlab-simulador-para-pic/>

### 2. Informações Importantes

- O trabalho deve ser feito em duplas ou trios, podendo ser discutido entre os colegas desde que não haja cópia ou compartilhamento do código fonte.

- A data limite de entrega será especificada através de uma tarefa no Moodle.

- Os trabalhos poderão ser entregues até as 23:55 do dia especificado para a entrega. O horário de entrega deve respeitar o relógio do sistema Moodle. Haverá uma tolerância de 5 minutos de atraso, de forma que os alunos podem fazer a entrega até às 0:00. A partir desse horário, os trabalhos já estarão sujeitos a penalidades. A fórmula para desconto por atraso na entrega do trabalho prático é:

$$\text{Desconto} = 2^d / 0.32 \%$$

onde  $d$  é o atraso em dias **corridos**. Note que após 5 dias **corridos**, o trabalho não pode ser mais entregue.

- O trabalho deve ser implementado obrigatoriamente em linguagem C.

- O código-fonte deve ser passível de compilação nas máquinas Windows e Linux do DCC, e posteriormente executado na plataforma de simulação disponibilizada ao grupo.

- Deverá ser entregue uma documentação contendo todas as decisões de projeto que foram tomadas durante a implementação, sobre aspectos não contemplados na especificação, assim como uma justificativa para essas decisões. Esse documento não precisa ser extenso (mínimo 3 e máximo de 6 páginas). A documentação deve indicar o nome dos alunos integrantes do grupo.

- Todas as dúvidas referentes ao trabalho deverão ser esclarecidas por meio do fórum disponível no ambiente Moodle da disciplina.

- O trabalho a ser entregue deverá estar contido em um único arquivo compactado, em formato “.zip”, com o nome no formato “tp3\_aluno1\_aluno2\_aluno3.zip”

- O arquivo .zip a ser enviado deverá ter três pastas:
  - “*application*”: Essa pasta deverá conter o código-fonte do programa implementado para o sistema embarcado (OBS.: Não devem ser incluídos quaisquer outros arquivos nessa pasta).
  - “*doc*”: Essa pasta deverá conter o arquivo da documentação, em formato PDF. Caso o grupo julgue necessário incluir quaisquer outros arquivos a parte, esses deverão ser justificados em um arquivo texto nessa mesma pasta com o nome README.
  - **Atenção:** Trabalhos que descumprirem o padrão definido acima serão penalizados.

### 3. Descrição da Plataforma de Simulação: PICsimLab

A plataforma de simulação PICsimLab (PIC Simulator Laboratory) é um software desenvolvido pelo Professor Luiz Claudio Gambôa Lopes (CEFET-MG) e que foi lançado em 2010. Seu propósito principal é ser uma alternativa em testes preliminares com aplicações de sistemas embarcados, as quais envolvem a utilização de microcontroladores PIC.

Atualmente o software conta com cinco opções de placas, as quais variam na complexidade e na gama de aplicações possíveis de serem desenvolvidas.

A plataforma de simulação PICsimLab também possui suporte à IDE MPLAB X (<http://www.microchip.com/mplab/mplab-x-ide>), e permite receber aplicações dessa IDE através da configuração de portas de comunicação serial virtuais para recepção do arquivo a ser carregado, simulando assim a transferência de uma aplicação pré-compilada para a placa, onde será executada. No entanto, também é possível carregar o arquivo binário proveniente de outras IDEs manualmente (em formato HEX), através do próprio PICsimLab.

Além de configurar uma porta de comunicação serial para simular o carregamento da aplicação, o PICsimLab também permite configurar uma porta de comunicação para envio de dados da placa, simulando assim a recepção de dados no PC. Na prática, outro programa em execução (um terminal) estará escutando a porta configurada, receberá os dados do simulador e irá exibir na tela.

A plataforma de simulação PICsimLab está disponível para ambientes Windows e Linux, é gratuita e pode ser baixada em sua página no SourceForge no link: <https://sourceforge.net/projects/picsim/>

### 4. Descrição da Aplicação a ser Desenvolvida

Deverá ser desenvolvida 1 (uma) aplicação para ser executada na plataforma de simulação disponibilizada. O grupo terá livre escolha para definir o propósito da aplicação, bem como qual das placas disponíveis no simulador será utilizada. No entanto, espera-se que sejam explorados ao menos 3 (três) aspectos de um sistema embarcado, a saber:

- Tratamento de interrupções
- Leitura de um sensor via conversor analógico-digital
- Uso de atuadores (Ex.: Ventoinha e relé)
- Envio de dados via serial para exibição no computador
- Indicação de qualquer ação ou evento via leds

- Execução de operações ao pressionar um botão
- Uso de timers internos do microcontrolador
- Exibição de informações no display LCD

**As propostas deverão ser enviadas em um tópico a ser criado no fórum da disciplina**, para análise e aprovação do professor e do monitor, e resolução de quaisquer conflitos entre grupos. A proposta deve conter o nome dos integrantes do grupo e uma breve descrição da aplicação, delineando quais dos aspectos acima do sistema serão exercitados. Caso os alunos proponham algo que exercite algum aspecto não contemplado na lista acima, o mesmo deverá ser detalhado na proposta.

As propostas serão analisadas com base na sua complexidade e no montante de aspectos do sistema envolvido.

**Importante: As propostas devem ser enviadas até o dia 15/11/2016.** Os grupos que enviarem suas propostas além desse prazo, além de serem prejudicados com o tempo reduzido para desenvolvimento, terão penalizações na nota final. Além disso, o professor e o monitor poderão identificar propostas muito ou pouco complexas, de forma a ajustar o escopo.

Abaixo seguem algumas sugestões de aplicação.

- **Semáforo de pedestres:** Deverá haver um led vermelho aceso constantemente. Ao apertar um botão, o led vermelho deverá apagar e o led verde ficará aceso durante um tempo (simulando assim a passagem de pedestres), e em seguida o led vermelho voltará a ficar aceso (botão, interrupções, timers e leds)
- **Cálculo de reflexo:** Cálculo de tempo para bater em um botão após um LED piscar. Essa aplicação é baseada em um caso real: uma aplicação de iPhone identifica se a pessoa pode possuir Parkinson via o padrão em que ela bate o dedo na tela do telefone (botão, interrupções, timers, leds e transmissão serial ou display LCD para exibição do tempo de reflexo)
- **Termômetro:** O usuário aperta um botão e o valor da temperatura é enviado via serial ou exibido no display LCD (botão, interrupções, leitura de sensor de temperatura e transmissão serial ou display LCD para exibição do valor lido)
- **Cronômetro:** O usuário aperta um botão. Após um tempo arbitrário, o mesmo botão é apertado, e o tempo decorrido (em segundos) é enviado via serial ou exibido no display LCD (botão, interrupções, timer, leds e transmissão serial ou display LCD para exibição do tempo decorrido no PC)
- **Tap Hero:** O usuário tem que pressionar o botão seguindo um certo ritmo, ao estilo Guitar Hero. No final, deve-se exibir no computador a quantidade de “taps” corretos, ou de pontos obtidos (botão, timers, leds e transmissão serial ou display LCD para exibição do resultado)
- **Código morse:** O usuário pressiona o botão para digitar um texto em morse. A saída do texto, já em ASCII, é feita no PC ou no display LCD (botão, timers, leds e transmissão serial ou display LCD para exibição do resultado)
- **Calculadora:** Realização de uma sequência arbitrária de operações aritméticas básicas (teclado, interrupções e transmissão serial ou display LCD para exibição de resultados e outras informações)

## 5. Descrição do Ambiente de Programação

Como já mencionado anteriormente, o principal ambiente de programação suportado pela plataforma PICsimLab é a IDE MPLAB X (<http://www.microchip.com/mplab/mplab-x-ide>). No

entanto, também é possível utilizar outras IDEs para desenvolvimento da aplicação, uma vez que o PICsimLab exige apenas o arquivo em formato HEX gerado pelo compilador.

A segunda opção de IDE é o mikroC PRO (<http://www.mikroe.com/mikroc/pic/>), o qual possui uma versão destinada especialmente aos microcontroladores PIC, e é de fácil utilização. Existe um ótimo canal no YouTube com um tutorial completo de desenvolvimento nessa IDE. Por essa razão, sugere-se o uso dessa IDE. O link para os vídeos do tutorial segue abaixo:

[https://www.youtube.com/playlist?list=PLMFrcR3zyPe9wWfvBeWr2pn3ZCUdb38\\_P](https://www.youtube.com/playlist?list=PLMFrcR3zyPe9wWfvBeWr2pn3ZCUdb38_P)

## 6. Sobre a Documentação

- Deve conter todas as decisões de projeto.
- Deve conter informações sobre a IDE utilizada (dentre as opções apresentadas anteriormente) e instruções de como compilar a aplicação na mesma. **OBS.:** Caso o grupo deseje utilizar outra IDE, o monitor deverá ser informado previamente, para que o uso dessa IDE seja aprovado.
- Deve informar qual placa foi usada no PICsimLab, bem como a versão do PIC empregada.
- Deve conter detalhes do funcionamento da aplicação desenvolvida.
- Deve conter elementos que comprovem que a aplicação foi testada com sucesso na plataforma de simulação (Ex.: Imagens de telas com dados recebidos da plataforma de simulação via serial, bem como imagens da plataforma em funcionamento).
- O código fonte não deve ser incluído no arquivo PDF da documentação.