

# EA871 – Laboratório de Programação Básica de Sistemas Digitais

## Atividade 04

Profs. Levy Boccato, Rafael Ferrari e Tiago Tavares – 2º semestre de 2019

### 1. Objetivos

- Familiarização com o conjunto de instruções do microcontrolador ATmega328P e com sua programação em linguagem simbólica (*Assembly*).

### 2. Resumo da Atividade

Elabore uma rotina em linguagem *Assembly* para o ATmega328P que introduza um atraso temporal. Em outras palavras, as operações executadas dentro da rotina não são propriamente relevantes; porém, desejamos que, ao final, um determinado intervalo de tempo (ou, equivalentemente, um número pré-estabelecido de ciclos de relógio) tenha sido consumido.

O uso de programação em *Assembly* neste cenário é particularmente pertinente pelo fato de podermos controlar de forma direta o número total de ciclos de relógio relacionados à execução da rotina, alcançando, assim, uma boa precisão com respeito ao tempo consumido por ela.

Nesta atividade, teremos o cuidado de preparar a sub-rotina de atraso de tal maneira que ela possa ser futuramente incorporada a projetos desenvolvidos em linguagem C sem maiores dificuldades. Por isso, seguiremos a convenção de passagem de parâmetros adotada pelo compilador C do Atmel Studio. No caso, a sub-rotina deve receber como entrada um valor de 16 bits (sem sinal) que especifica o tempo de atraso em milissegundos (e.g., um valor igual a 1000 deve levar a rotina a produzir um atraso de 1 segundo). Este valor deve ser recebido pela sub-rotina através do par de registradores r25:r24.

Por fim, escreva um programa em *Assembly* que faça com que um LED pisque com um período de 1 segundo. Ou seja, o LED deve permanecer aceso por 0,5 segundo, ficando apagado por 0,5 segundo em seguida. Por simplicidade, vamos utilizar o próprio LED incorporado presente no Arduino UNO, o qual pode ser acionado pelo bit 5 da porta B (PB5).

**Sugestão:** utilizando um osciloscópio, observe a forma de onda gerada no pino 5 da porta B e confira o valor de seu período.

### Observações:

- Considere que o sinal de relógio da operação normal do Arduino UNO é de 16 MHz (faça esta alteração no Atmel Studio para que a medida de tempo fique correta).
- É fundamental que **a sub-rotina salve o conteúdo de todos os registradores que venha a modificar**, restaurando seus valores antes do retorno.
- Uma forma de comprovar o funcionamento do programa consiste em obter analiticamente a **relação matemática** entre o total de ciclos consumidos pela sub-rotina e o valor do parâmetro de entrada especificado em r25:r24.

### Materiais de consulta

Para esta atividade, é recomendado consultar o manual de instruções de máquina do microcontrolador ATmega328P, assim como o *link* com a descrição das diretivas (pseudo-instruções) do montador.

### Instruções para a submissão do trabalho

- Nos comentários do código-fonte (main.c), justifique as operações e os valores carregados em todos os registradores.

- 2) Crie um projeto chamado 'atividade4' (letras minúsculas, sem espaço) no Atmel Studio e, ao final da atividade, salve o diretório completo em um arquivo no formato .zip (Aviso: não use .tar.gz nem .rar), com nome 'seu\_ra.zip' (Exemplo: 025304.zip).
- 3) Faça o *upload* da sua solução da atividade 4 no Google Classroom.