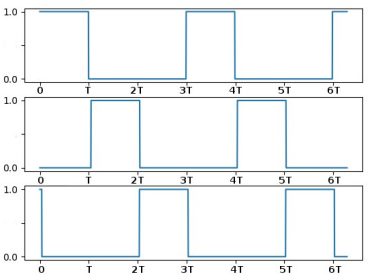
# TRABALHO 1 DE MICROPROCESSADORES

ALUNO: Francilândio Lima Serafim (472644)

## PROJETOPT2 – Resolução

Para a realização do projeto foi pedido que, a partir de um código em Assembly feito no MPLABX fosse implementado em um microprocessador PIC18F4550 a geração de ondas quadradas em três pinos distintos, essas ondas deviam seguir o modelo representado na Figura 1.

Figura 1 – Formas de onda a serem geradas.



No gráfico é possível notar que a escala de tempo tem intervalos de medida T, essa constante foi dada pela fórmula

Onde N é o último dígito da matrícula do aluno, e como ela é 472644, N = 4 e T que é dado em microssegundos vale 500 us.

Ainda sobre o gráfico da Figura 1, é possível notar que as formas de onda se repetem a partir de 3.T, ou seja, isso possibilita explorar um padrão para a geração de ondas e a partir dessa análise foi idealizado o uso de uma estrutura de repetição no código.

Para a resolução, usou-se uma única variável CONT para ser o contador de um laço e iniciando o programa, com a instrução MOVLW para colocar o valor 11111000 no W e depois usada MOVWF para colocar o valor de W em TRISD, o que significa setar os pinos que serão as saídas, ou seja, RD0, RD1 e RD2.

Na parte seguinte do código foi definido uma estrutura de repetição com rótulo REPETE, e nessa estrutura estão definidos os três instantes citados anteriormente. O primeiro período inicia com a instrução BCF que faz o bit 0 de PORTD ir para 0, e a seguir a instrução BSF seta o bit 2 de PORTD, daí vem a primeira chamada ao atraso com a instrução CALL finalizando o primeiro período. O segundo período inicia com a instrução BCF que zera o bit 2 de PORTD e depois a instrução BSF seta o bit 1 de PORTD e aí faz-se a segunda chamada ao atraso, finalizando o segundo período. O terceiro e último período inicia-se com a instrução BCF que zera o bit 1 de PORTD e depois a instrução BSF zera o bit 0 de PORTD, daí é chamado o atraso de modo a terminar o terceiro período e depois é chamada a instrução GOTO para tornar a repetir as instruções após o REPETE.

Sobre a rotina de atraso citada anteriormente, trata-se de uma parte do código onde as instruções a partir do rótulo ATRASO são executadas todas as vezes que CALL ATRASO ocorre no código. As instruções presentes em ATRASO foram utilizadas pensando nos tempos de execução que cada uma necessita para que cada chamada a essa rotina fosse concluída em 500 us, seguindo os tempos mostrados na Tabela 1.

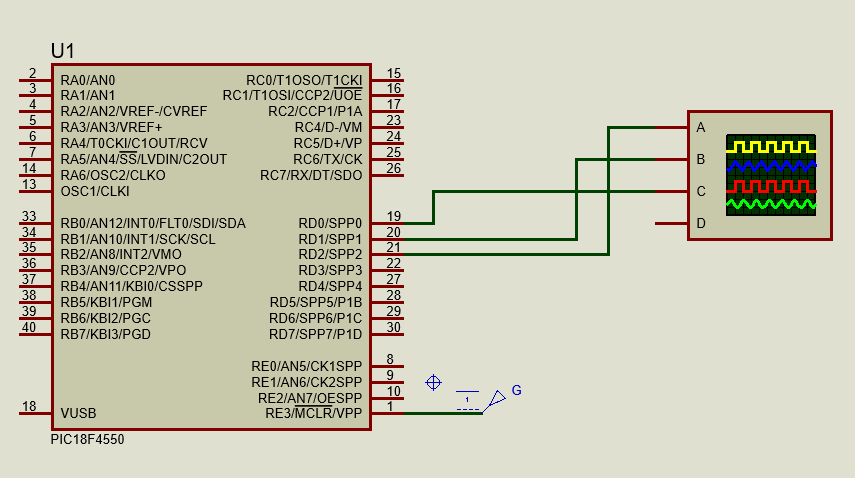
Tabela 1 – Tempos de execução das instruções.

|  |  |
| --- | --- |
| **Instrução** | **Tempo de execução (us)** |
| CALL | 2 |
| MOVLW | 1 |
| MOVWF | 1 |
| NOP | 1 |
| DECFSZ | 1 |
| GOTO | 2 |
| RETURN | 2 |

As instruções fora do loop presente na rotina de atraso são: CALL, MOVLW, MOVWF, NOP(4) e RETURN. O que resulta em um tempo de 10 us. Já dentro do loop presente no atraso, tem-se NOP(2), DECFSZ e GOTO. Daí considerando que o loop roda x vezes, dependendo do valor setado para o contador do laço, tem-se 5x microssegundos. Ou seja, são 10 + 5x microssegundos, e para descobrir o valor de x fazemos 10 + 5x = 500, o que resulta em x = 98, e sendo assim esse é o valor atribuído para a variável CONT.

Após compilar o código e gerar seu executável, foi copiado o caminho desse código para anexar ao PIC18F4550 no programa PROTEUS, e assim, foi possível conectar software e hardware a partir da simulação usando o osciloscópio, assim como mostra a Figura 2 com circuito montado.

Figura 2 – Circuito montado no Proteus.



Nesse momento, fazendo alguns ajustes no osciloscópio, foi feita a simulação para observar as ondas dos três pinos mostradas nos gráficos A, B e C. As formas de onda estão registradas nas figuras 3 e 4.

Figura 3 – Ondas geradas.

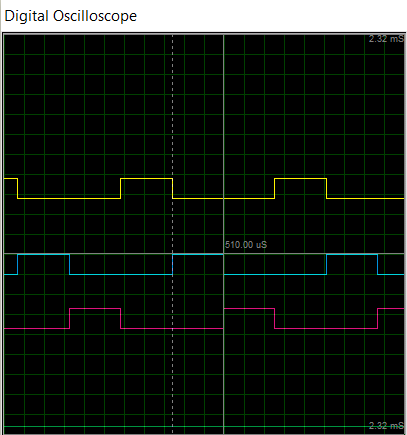
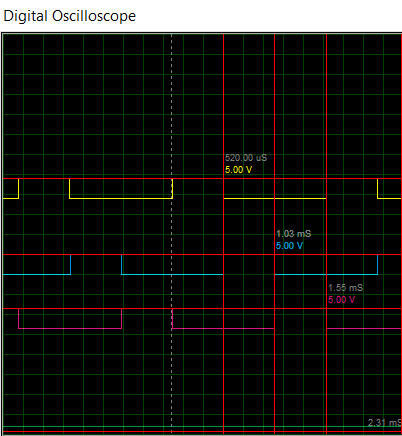


Figura 4 – Ondas geradas e registro dos tempos.



Fazendo analogia ao gráfico da Figura 1, os gráficos mostrados nas figuras 3 e 4 mostram os intervalos de tempos T, 2T e 3T, sendo que T oscilou em algumas execuções entre 510 us e 520 us (devido a algum bug do Proteus ou configuração errada).

Os códigos da resolução se encontram em prints a seguir.

Figura 5 – Código da resolução.

