1. PERIFÉRICOS

Real Time Clock (RTC) é um periférico construído com circuitos integrados tendo a finalidade de gerenciar o controle do tempo de um dispositivo. A vantagem de utilizá-lo em vez de um software ou função se dá pelo fato do RTC apresentar baixo consumo de energia, aliviar o sistema para gerenciamento de outros processos e possuir uma melhor precisão. O RTC do microcontrolador da Atmel que está sendo estudado necessita de um oscilador externo com frequência de 32,768kHz para funcionamento correto do periférico.

O Timer Counter (TC) é um periférico capaz de realizar a medição de frequências, contagem de eventos, medição de intervalos, geração de pulsos, atraso de tempo e trabalhar com sinais PWM. No microcontrolador da Atmel existem 6 canais TC com possibilidade de serem programados de forma independente. O TC é frequentemente utilizado para processos que utilizam interrupções do processador.

Os endereços reservados para os periféricos são:

- 0x400E0E00 (PIOA).
- 0x400E1000 (PIOB).
- 0x40040000 (ACC).
- 0x400E0600 (UART0).
- 0x400E0800 (UART1).

2. PIO

- **Verifique no datasheet do uC os pinos físicos do uC associados aos I/O: PA01, PB22 e PC12.**

  PA1: PWMH1, TIOB0, A18 e WKUP1;

  PB12: PWML1 e ERASE;

  PC12: NCS3 e AD12.

Obs.: os I/Os PB vão apenas até o I/O PB14.

- **Verifique quais periféricos podem ser configuráveis nos I/Os : 1. PC20 / 2. PB3.**

PC20: A2 e PWMH2. Nesse I/O pode ser configurado o LED Vermelho (Power);

PB3: UTXD1, PCK2 e AD7. Nesse I/O pode ser configurado o User Push-Button

2.1 CONFIGURAÇÕES

Muitas vezes ruídos aleatórios provenientes de contatos metálicos interferem na contagem ou identificação de um sinal. Quando se aciona uma chave, esses ruídos geram um estado transitório podendo ser interpretado no microcontrolador como diversos acionamentos da mesma. Para solucionar este problema existe uma técnica chamada *deboucing*; esta técnica certifica que o botão apertado apenas uma vez seja interpretado corretamente.

Um algoritmo que implementa o deboucing pode ser realizado com a utilização de delay e estruturas condicionais:

```
if(botao == 1)
 {
   delay_ms(50);
   if(botão == 1)
       {
          Printf("Botão acionado");
       }
 }
```

2.3 SET/CLEAR

Race conditions ocorrem quando mais de um "atuador" podem utilizar o mesmo "bit". Já que o bit será modificado algumas vezes, seu real estado só será definido no final, quando ocorrer a última alteração. Para modificar o valor dos bits manipulam-se registadores diferentes, habilitando pinos ao colocar o nível 1 na porta do registrador, e deste modo, é possível saber que registrador está atuando no bit, havendo maior controle.

2.4 CONFIGURANDO UM PINO EM MODO DE OUTPUT

Determinando se a linha de I/O será atribuída a uma função de periférico, o drive do I/O será controlado, dependendo dos valores do PIO_ABCDSR3 E PIO_ABCDSR2, pelos periféricos A, B ou C.

Já quando a linha de I/o é controlada pelo controlador do próprio PIO, pode-se configurar o mesmo para ser somente de entrada, ou de saída; e os valores setados no I/O podem setar ou apagar os registros de dados do mesmo. Existem configurações especificas para cada valor (0 ou 1) setado nos pinos dos registradores.