**1)**Um periférico RTC é um relógio com bateria que mantém o tempo mesmo com o corte de energia, fornecendo informações gerais sobre o tempo (horário, calendário, …).

O TC pode ser utilizado para contar pulsos regulares de clock, tornando-se um temporizador (timer), ou para contar pulsos irregulares de clock, tornando-se um contador (counter). Ele realiza contagem de eventos, frequência, intervalos, geração de pulsos.

**1.2**

PIOA: 0x400E0E00, 0x400E0E04, …, 0x400E0F64

PIOB: 0x400E1000, 0x400E1004, …, 0x400E1164

ACC: 0x40040000, 0x40040004, 0x40040024, 0x40040028, 0x4004002C, 0x40040030, 0x40040094, 0x400400E4, 0x400400E8.

UART0: 0x400E0600, 0x400E0604, 0x400E0608, 0x400E060C, 0x400E0610, 0x400E0614, 0x400E0618, 0x400E061C, 0x400E0620.

UART1: 0x400E0800, 0x400E0804, 0x400E0808, 0x400E080C, 0x400E0810, 0x400E0814, 0x400E0818, 0x400E081C, 0x400E0820.

**2)**

PA1 – pino 72 do uC

PB22 - os PB vão de 0-11, sendo assim, não existe o PB22.

PC12 – pino 23 do uC.

PC20 – pode controlar o led vermelho da placa.

PB3 – pode receber sinal de um “User push button 1”

**2.11** – O que é debouncing?

Metodo Debouncing:

O método debouncing pode ser utilizado para certificar que uma chave (ou botão e etc) esteja acionada, verificando se a mesma está acionada por um determinado tempo, também pode ser utilizado para ativar comandos.

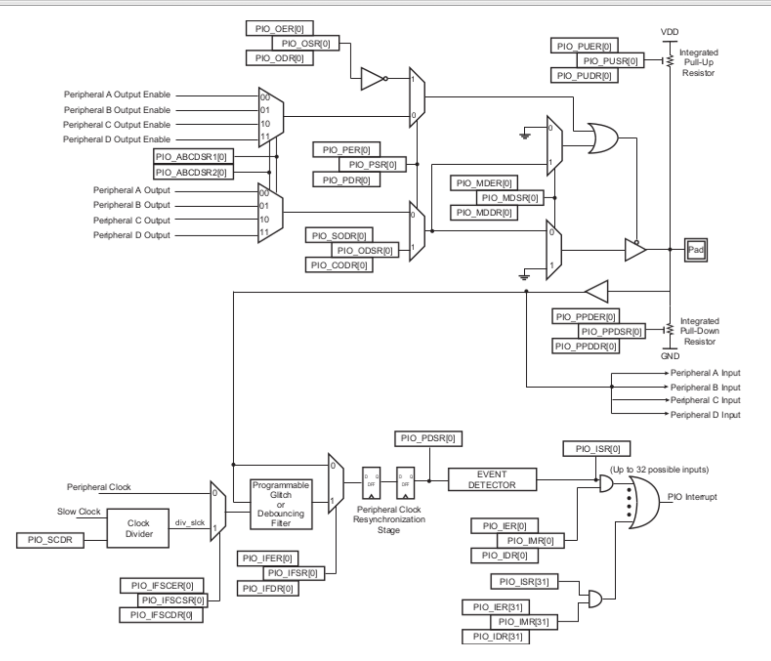
É uma técnica que nos dá certeza de que o botão apertado apenas uma vez seja interpretado corretamente como apenas UM aperto de botão.

**2.12**-Descreva um algoritmo que implemente um debouncing?

If (botao = 1)  
 {  
 delay\_ms(100);  
 if (botão = 1)  
 {  
 // Botão acionado.  
 }  
 else if(botão = 1)  
 {  
 // nao esperou tempo suficiente  
 }  
 {

**2.4.** Configurando um pino em modo de saída

Explique o trecho anterior utilizando o diagrama "I/O Line Control Logic"



A linha de Entrada e saída (I/O) pode ser utilizada de algumas formas, tanto como serviço de funções periféricas que controla os drivers, quanto pelo PIO Controller.

Caso a linha de entrada e saída esteja sendo controlada pelo PIO Controller, será através dos registradores PIOOER e PIOODR e o seu nível será determinado pelo Set e o Clear (ativo e não ativo). Estes dois registradores informarão o PIOODSR se ele irá trabalhar de forma controlada pelo PIO Controller (nível lógico 1) ou se será somente uma entrada com nível lógico 0.

Caso a linha de entrada e saída seja utilizada para funções periféricas, isso será feito deixando em o nível lógico do PIOSDR em 0, após isso, os registradores PIOABCDSR1 e PIOABCDSR2 vão definir quais periféricos irão trabalhar com as I/O.

Em geral, os registradores PIOODR e PIOOED tem a função de configurar a forma com que os pinos disponíveis irão ser utilizados para I/O, seja para funções periféricas ou para o PIO Controller.