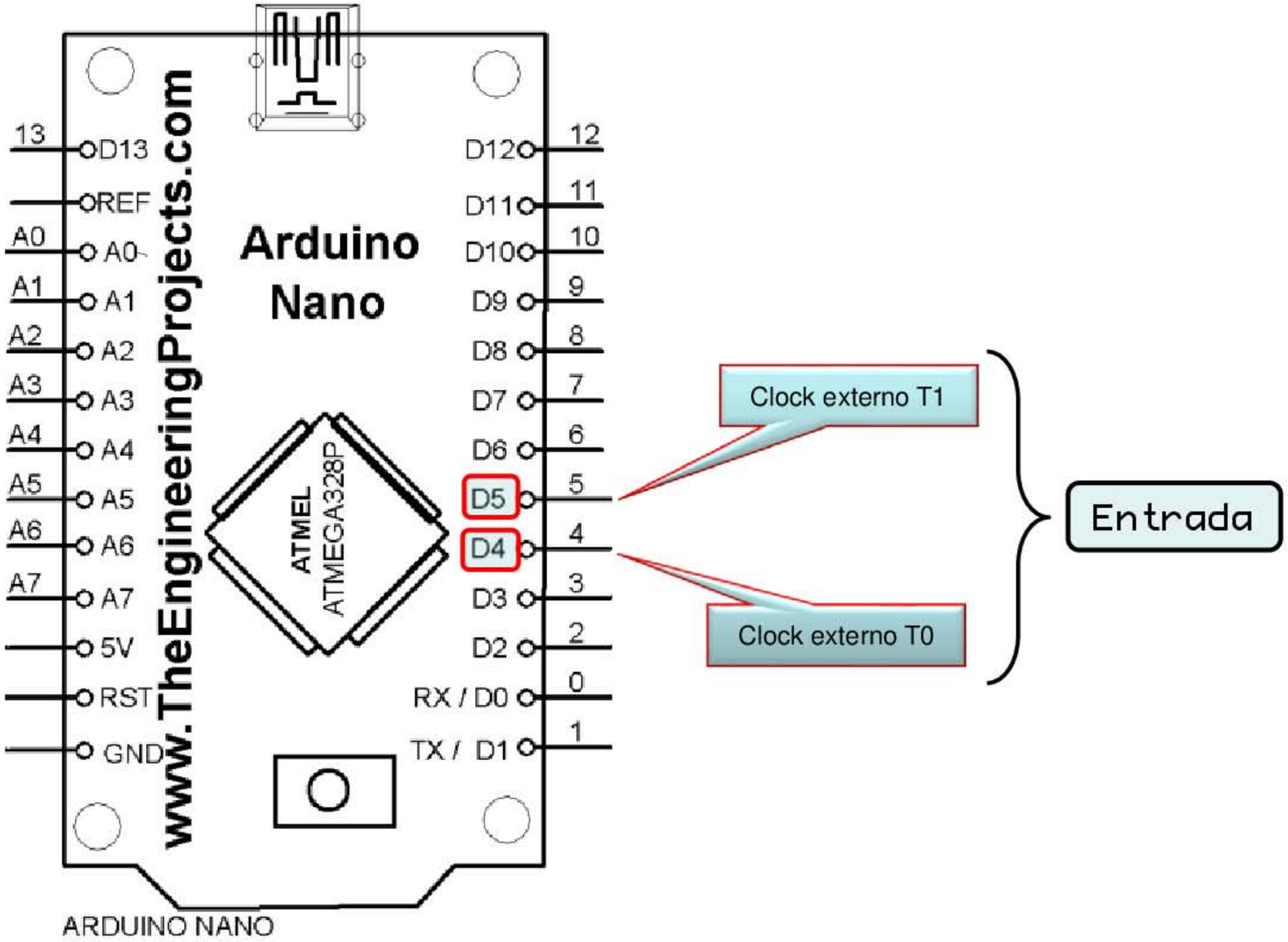
TIMERS

http://setesys.com.br/blog/google-timer/

© Prof. Engº esp Luiz Antonio Vargas Pinto
www.vargasp.com

Revisado em Maio de 2022

Características externas



O tempo no Arduino

- São disponibilizados os seguintes comandos de tempo na IDE:
  - delay()
  - delayMicroseconds()
  - millis()
  - micros()

delay()

- Pausa a execução do programa por um tempo em mS.
- Certas coisas continuam a acontecer enquanto o delay atua, e esta função não desabilita as interrupções.
  - ☒ Comunicação serial no pino RX será gravada no ATMEGA, PWM (analogWrite) serão mantidos.
- Delay() utiliza o Timer 0

delayMicroseconds()

- Pausa a execução do programa por alguns µS.
- Atualmente, o maior valor de parâmetro deste comando é 16383; valores maiores podem produzir um delay extremamente curto.
- Range de 3 µS a 16383 µS.
- Valores menores não são muito precisos
- A partir da versão Arduino 0018, delayMicroseconds() não desativa interrupções.
- Sintaxe:

```
delayMicroseconds(valor_em_uS);
```

  - Tipo unsigned int.

millis()

- Retorna o número de mS desde que o Arduino começou a executar o programa corrente.
- Este valor irá estourar, voltando a zero após ±50 dias.
- Sintaxe:

```
variavel = millis();
```

  - Tipo unsigned long.
- millis() utiliza o Timer 0

micros()

- Retorna o número de µS desde que o Arduino começou a executar o programa corrente.
- Este valor irá estourar - voltar a 0 - após ±70 minutos.
- No board Portenta esta função tem uma resolução de 1 µS em todos os núcleos.
- Nos boards de 16 MHz, caso do Duemilanove e Nano, esta função tem uma resolução de 4 µS isto é, o valor retornado é sempre múltiplo de 4

Detalhes do Arduino Portenta PH7

- A CPU é dual core
  - Um core ARM Cortex® M7 rodando a 480 MHz e;
  - um core ARM Cortex® M4 rodando a 240 MHz.
- Os núcleos permitem chamadas de funções entre eles

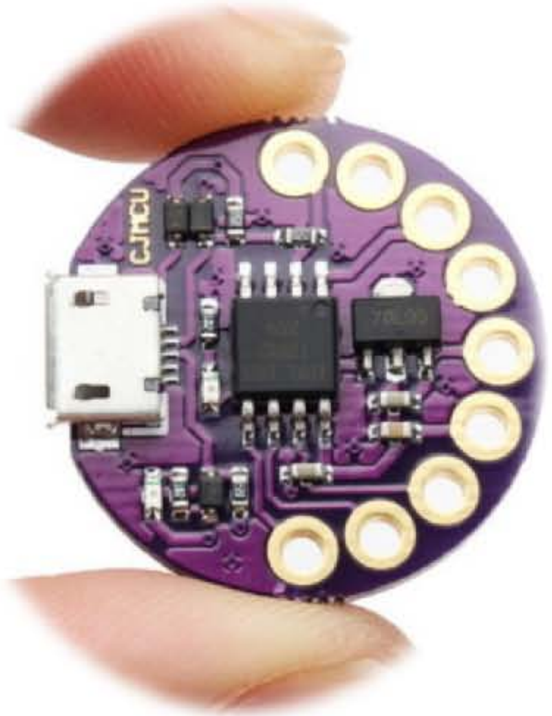
- Em boards de 8 MHz, caso do LilyPad, esta função tem uma resolução de 8 µS isto é, o valor retornado é sempre múltiplo de 8.

- Sintaxe:

  variavel = micros();

  - Tipo unsigned long.

- micros() utiliza o Timer 0



Operando Manualmente

Considerações Gerais

- Devido ao tamanho da apresentação, os detalhamentos das operações dos timers dos Arduinos UNO, Nano, Mini, Micro e Lillipad estão na apresentação de Timers do ATMEGA 328p

O Timer 0

Timer0

- Devido algumas exigências da IDE do Arduino, o Timer0 não pode ser utilizado em aplicações diretamente sobre o ATMEGA328p.
- O que não significa que não se possa configurá-lo, mas que o sistema irá impedir a compilação.

O Timer 1

Timer1

- No UNO é o timer utilizado pela biblioteca de controle de servos pois usa PWM.
  - Se o projeto em uso não estiver utilizando essa biblioteca pode-se usar esse timer para outra finalidade.
- No Mega esse timer só será utilizado para controlar os servos se o projeto estiver empregando mais de 12 servos.

O Timer 2

Timer2

- Esse timer é utilizado pela função tone().
- Então se ela não for necessária esse timer estará livre para outras aplicações.

Os Timers 3 e 4

Timer3 e Timer4

- Esses timers só estão presentes no Mega
- E só serão utilizados pela biblioteca caso o projeto use mais de 24 servos.
- De 25 a 36 servos o timer3 será utilizado
- De 37 a 48 servos o timer4 será utilizado.

O Timer 5

Timer5

- No Mega esse é o timer padrão para o controle de servos, ou seja, se você estiver utilizando de 1 a 12 servos, apenas esse timer será utilizado pela biblioteca.