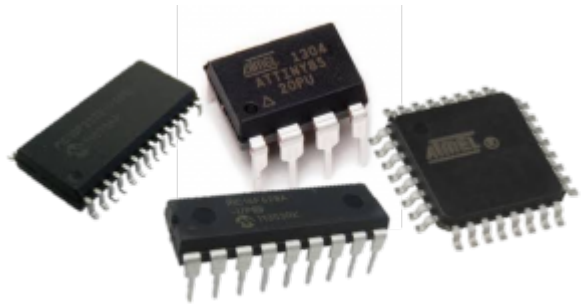


# Microcontroladores



*Vanderlei Alves S. Silva*

A eletrônica é cada vez mais surpreendente e fascinante, é incrível imaginar que podemos ter o domínio de tantas tecnologias em nossas mãos e poder, assim como engenheiros (para aqueles que ainda não são) desenvolver circuitos eletrônicos capazes de fazer coisas fantásticas do tipo acender lâmpadas usando o celular, controlar motores à distância sem o uso de cabos elétricos, criar máquinas que desenham usando canetas esferográficas comum, automatizar uma casa por completo, cuidar de hortaliças e jardins sem a presença constante do homem e também criar robôs com funções específicas para a realização de tarefas. E tudo isso hoje está ao alcance de quem tiver o interesse em aprender.



O interessante nisso tudo é que os circuitos que realizam tais feitos estão ficando cada vez menores em termos físicos e mais baratos, o que torna acessível a muita gente e isso graças ao uso do microcontrolador, também conhecido como MCU, o qual será assunto para este artigo, acompanhe.

Após a criação do primeiro microcontrolador por volta de 1971, pelos engenheiros Michael Cochran e Gary Boone da Texas Instruments, atualmente uma das grandes fabricantes de microcontroladores, os circuitos eletrônicos foram ficando cada vez menores e o número de componentes externos reduzidos fazendo, assim, o custo cair consideravelmente.

Os microcontroladores tomam parte na grande maioria dos circuitos eletrônicos existentes. Podemos encontrá-los em televisores, celulares, automóveis, sistemas embarcados, no comando de máquinas industriais, enfim, em todos os lugares e isto é o que nos dá motivo de querer estudá-los cada vez mais.

### **Mas afinal, o que é um microcontrolador?**

Podemos resumidamente responder a esta pergunta dizendo que é um tipo de circuito integrado (chip) que traz internamente um sistema computacional completo (com exceção do peopeware) e basicamente vem com memória RAM, memória ROM, memória EEPROM, microprocessador, Unidade Lógica e Aritmética (ULA), portas de entrada e saída de dados, conversores AD e/ou DA registradores, clock interno e alguns têm até mesmo comparadores de tensão, controladores de display LCD e temporizadores (TIMER 0, TIMER 1, TIMER 2).

Um microcontrolador pode assumir a função que você desejar - Claro, desde que tenha lógica, não vá esperar que um microcontrolador te teletransporte para uma outra dimensão ou faça você se tornar um super-herói do dia pra noite - Mas, sabendo o que realmente você quer, e repito, dentro da lógica, basta saber programar que ele irá obedecer seus comandos com precisão.



Para a utilização de um MCU (Microcontroller Unit), é preciso conhecer uma linguagem de programação específica para programá-los e é isso que o faz parecer, para muitos, um componente eletrônico complicado de se utilizar. Mas calma! Nada é tão complicado como possa parecer. Atualmente as linguagens usadas no desenvolvimento de códigos para MCU são escritas em inglês, o que foi considerado um avanço, uma vez que antes era preciso usar praticamente a linguagem de máquina para programar um microcontrolador.

Nossa... Você não entende inglês! E agora deve estar se perguntando - “Quer dizer que terei que fazer um curso de inglês para aprender a programar um microcontrolador???” ☐

Bom... um curso de inglês te ajudaria muito mesmo, mas, para programar um MCU não é preciso gastar dinheiro estudando um idioma, as palavras usadas, que na verdade são códigos escritos em inglês, são fáceis de aprender, repetitivas - o que ajuda na memorização - e curtas. Agora você pode está se perguntando - “Mas por que inglês???” - Lembre-se de que o inglês é uma linguagem universal e assim fica fácil para que outras pessoas em outra parte do mundo possa ter acesso a esta tecnologia.

### **Linguagens de programação**

As linguagens de programações mais comuns para microcontroladores são: Assembly, considerada de mais baixo nível de abstração e onde muitos códigos são escritos quase que em linguagem de máquina, sendo portanto uma das mais indicadas; Linguagem C ANSI, a qual é amplamente utilizada pela grande maioria por ser de alto nível e de muito mais fácil compreensão - Particularmente é a que eu mais uso - C/C++, também de alto nível e as linguagens específicas de um determinado grupo de microcontroladores como os AVR's, microcontroladores da plataforma Arduino que usam uma linguagem derivada da C/C++.

Exemplo de um pequeno código de programação para microcontrolador escrito em C ANSI:

```

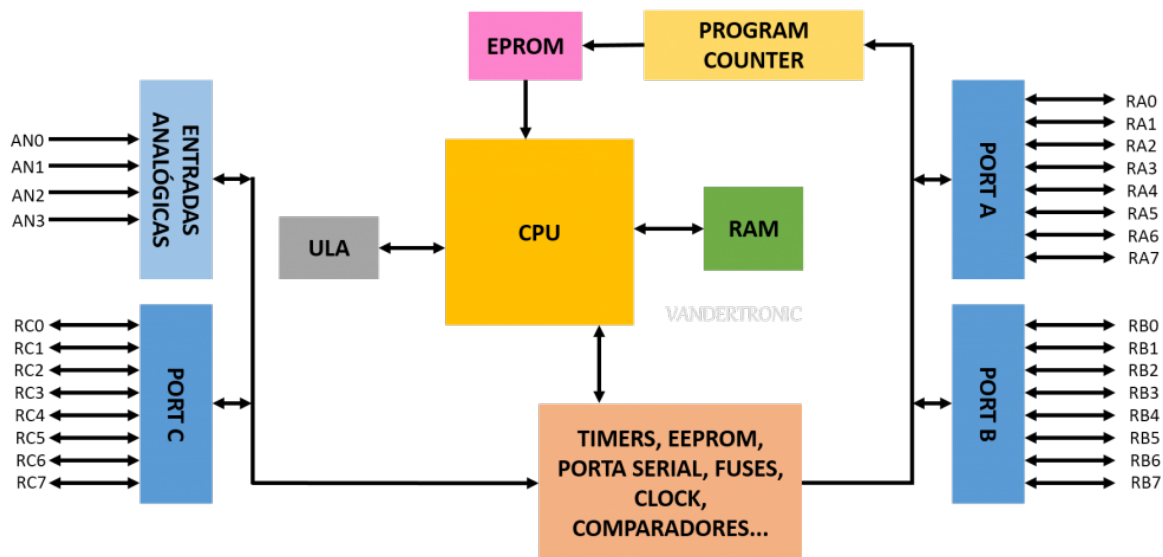
.
. //Este código faz um LED ligado no pino RB4 de MCU piscar infinitamente
.
- void main() { //Inicia a configuração dos pinos
.
.     TRISB = 0x00; //Faz com que o PortB seja saída
.     PORTB = 0x00; //Faz o PortB iniciar em zero
.     ANSEL = 0x00; //Desabilita o conversor A/D
10
.
.     //Fim da configuração dos pinos
12
.     while(1) //Inicia o loop infinito
.     {
-     RB4_bit = 0x01; //Faz o pino RB4 ir ao nível alto
.     delay_ms(500); //Mantem o nível alto em RB4 por 500 milisegundos
.     RB4_bit = 0x00; //Faz o pino RB4 ir ao nível baixo
.     delay_ms(500); //Mantem o nível baixo em RB4 por 500 milisegundos
.     }
20
.
. }
.

```

*Figura 1 - Exemplo de código em C para microcontrolador da família PIC.*

O código acima faz um LED instalado em uma das portas de um MCU piscar durante o tempo em que o circuito estiver sendo alimentado por corrente elétrica. As frases na cor verde escritas logo após // são apenas comentários e não significam nada para o microcontrolador. O código em si são as palavras escritas na cor preta juntamente com seus valores e sinais gráficos obrigatórios com ponto-e-vírgula, igualdade, underline parenteses e chaves.

Mas, vamos deixar para falar sobre linguagem de programação de microcontroladores em um outro post. Vamos analisar a estrutura interna do MCU, ou seja, seu hardware e para isto veja a figura abaixo:



*Figura 2 - Arquitetura interna básica de um microcontrolador*

De acordo com a figura 2 vamos descrever a função de cada um dos blocos que formam o microcontrolador:

**CPU** - É a **Unidade Central de Processamento**, assim como nos computadores, podemos dizer que é o cérebro da máquina. É neste local onde são tomadas as decisões e o processamento dos dados.

**ULA** - Chamada de **Unidade Lógica e Aritmética** é responsável por realizar os cálculos matemáticos e de lógica. Geralmente faz parte interna da CPU.

**RAM** - É a Memória de Acesso Randômico, que assim como nos computadores, serve para realizar o armazenamento de informações momentâneas durante um cálculo. Pode-se considerar como sendo uma memória de rascunho, exemplo: Quando o microcontrolador faz a leitura de um sensor, as informações vindas do sensor são armazenadas na RAM para que sejam mais tarde processadas e após o processamento a RAM é apagada. Ela não armazena nada de modo definitivo e toda vez que o microcontrolador é desligado ou reiniciado, ela é completamente apagada.

**PORT A, B e C** - São os terminais chamados de portas do microcontrolador, também conhecidas por portas digitais e são por elas que as informações digitais entram ou saem. A maioria das portas são

bidirecionais, o que significa que servem tanto como saída ou entrada de dados. Algumas podem ser apenas de entrada e outras somente de saída. Quando uma porta é bidirecional representamos essa condição com uma seta de duas pontas ao lado da referida porta, quando é unidirecional representamos com uma seta de uma ponta apontando para dentro do micro se for de entrada e para fora se for de saída. É por meio das portas que o microcontrolador interage com o meio externo.

**ENTRADAS ANALÓGICAS** - São portas apenas de entrada que servem para capturar sinais analógicos do meio externo. Esses sinais podem ser os mais variados como som, temperatura, luminosidade, umidade entre outros que possam variar em uma senoide. Com a captura desses sinais é possível realizar a automação de diversos processos, por exemplo, na indústria essas portas podem capturar o nível de um reservatório e através das portas digitais poderá ligar ou desligar uma bomba obedecendo a uma lógica de programação, ou então, em casa, podem capturar a medida de peso de um botijão de gás e fazer as portas digitais acionarem um sistema de aviso de que o gás está acabando.

**EPROM** - É a memória usada para armazenar firmware, que é o programa digitado pelo projetista, e pode ser escrita e apagada entre 10.000 vezes a 100.000 vezes. Esse programa é quem vai definir as funcionalidades e modo de trabalho do microcontrolador.

**EEPROM** - Memória interna que pode ser usada para o armazenamento de informação. Ela é capaz de guardar os dados e mantê-los mesmo se o microcontrolador for desenergizado. Dependendo do microcontrolador os dados poderão permanecer retidos por aproximadamente 40 anos, após esse tempo não se garante que as informações permaneceram intactas. Cada célula dessa memória possui um limite para a quantidade de escritas, ou seja, podemos escrever e apagar uma informação dentro de um certo número de vezes que pode variar entre 100.000 vezes como no ATmega328P, usado na plataforma Arduino, a 1.000.000 de vezes como no caso dos microcontroladores da família PIC, exemplo PIC16F628A.

**PROGRAM COUNTER** - Serve para definir ao processador interno do

microcontrolador a sequência de execução do código e no caso de uma rotina de interrupção ele armazena a linha seguinte para quando o processador retornar da interrupção e assim poder continuar com a execução sequencial do código.

Além desses itens que acabamos de definir, um microcontrolador pode conter diversos como Timers, Porta serial para comunicação com computadores, comparadores de tensão entre outros e isto vai depender do tipo de microcontrolador usado.

**Dica:** Para você começar a entender melhor e se aprofundar nos conhecimentos sobre microcontroladores poderá iniciar com a mais famosa das plataformas de estudos que é o Arduino. Em breve começaremos com uma série de experiências e explicações sobre esta placa de prototipagem incrível. Aguardem!



*Figura 3 - Plataforma Arduino*

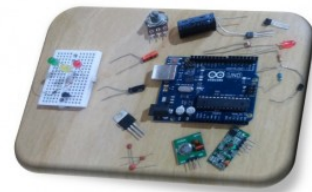
## **Conclusão**

Os microcontroladores estão sendo usados praticamente em todos os equipamentos que nos rodeiam e começar a conhecer sobre este circuito integrado realmente é algo necessário para quem é ou pretende ser um excelente profissional em eletrônica. Aprender a programar um MCU fará você ter um diferencial diante de muitos outros profissionais da área e abrirá um leque enorme de possibilidades.



Então é isso pessoal! Espero poder ter feito vocês entenderem um pouco

sobre esses fantásticos componentes eletrônicos e que, particularmente, sempre prefiro usar para desenvolver os projetos que a mim são solicitados. Usando microcontroladores eu ganho tempo, precisão e qualidade nos meus trabalhos.



E se você estiver interessado em fazer um curso completo sobre microcontroladores ou Arduino e assim conquistar um certificado, deixe um comentário e preencha os campos com seu nome e e-mail que logo entraremos em contato. Dependendo de sua localidade o curso poderá ser online através da nossa Sala de Aula ou presencial em nossa instituição.

***Gostou deste conteúdo? Quer muito mais assuntos interessantes e úteis? Então faça uma doação ou torne-se um colaborador e apoie esta obra.***

Doar



**APOIA.se**