

Capítulo 3

Sistemas numéricos

Sistemas numéricos são sistemas de notação usados para representar quantidades abstratas denominadas números. Um sistema numérico é definido pela base que utiliza. A base é o número de símbolos diferentes, ou algarismos, necessários para representar um número qualquer, dos infinitos possíveis no sistema.

3.1 Introdução

Ao se programar microcontroladores, e também o Arduino, são usados diferentes sistemas de numeração para facilitar a representação de quantidades, variáveis ou grandezas medidas e/ou controladas. Há situações em que é mais conveniente fazer essa representação em um tipo de sistema do que em outro.

Os sistemas numéricos mais utilizados na prática são descritos a seguir.

3.2 Sistema de numeração decimal

O sistema decimal é um sistema de numeração posicional que utiliza a base dez. Baseia-se em uma numeração de posição, onde os dez algarismos indo-arábicos, 0 1 2 3 4 5 6 7 8 9, servem para contar unidades, dezenas, centenas, etc., da direita para a esquerda.

Contrariamente à numeração romana, o algarismo árabe tem um valor diferente segundo sua posição no número: assim, no número 111, o primeiro algarismo da esquerda para direita significa 100, o segundo algarismo 10 e o terceiro 1, enquanto que em VIII (oito em numeração romana) os três “I” significam todos 1.

Assim:

$$347 = 3 \cdot 100 + 4 \cdot 10 + 7 \cdot 1 = 3 \cdot 10^2 + 4 \cdot 10^1 + 7 \cdot 10^0$$

Da direita para esquerda, o primeiro dígito representa as unidades, o segundo as dezenas, o terceiro as centenas, e assim sucessivamente.

No sistema decimal o símbolo 0 (zero) posicionado à esquerda do número escrito não altera seu valor representativo. Assim: 1, 01, 001, ou 0001 representam a mesma grandeza, neste caso a unidade. O símbolo zero posto à direita implica multiplicar a grandeza pela base, ou seja, por 10 (dez).

3.3 Sistema de numeração binário

O que aconteceria se apenas dois dígitos fossem usados: 0 e 1, por exemplo? Ou se não soubéssemos como determinar se algo é 3 ou 5 vezes maior do que qualquer outra coisa? Ou se estivéssemos restritos a comparar dois tamanhos, ou seja, se só podemos afirmar que algo existe (1) ou não existe (0)?

Gostaríamos de continuar a usar os números da mesma forma como o fazemos agora, mas seria um pouco diferente. Por exemplo, no número 11011010: Quantas páginas de um livro são representadas pelo número 11011010? A fim de aprender isso, temos apenas que seguir a mesma lógica que no exemplo anterior, mas em ordem inversa. Tenha em mente que tudo isso é sobre matemática apenas com dois dígitos, 0 e 1, ou seja, a base do sistema de base 2 (sistema de números binários).

O sistema binário, ou base 2, é um sistema de numeração posicional em que todas as quantidades se representam com base em dois algarismos: zero e um (0 e 1).

Os computadores digitais, e também os microcontroladores trabalham internamente com dois níveis de tensão, visto que o seu sistema de numeração natural é o sistema binário (ligado, desligado). O sistema binário é a base para a álgebra booleana (de George Boole - matemático inglês), que permite fazer operações lógicas e aritméticas usando-se apenas dois dígitos ou dois estados lógicos (sim e não, falso e verdadeiro, tudo ou nada, 1 e 0, ligado e desligado, etc.). Toda a eletrônica digital e computação moderna estão baseadas nesse sistema e na lógica de Boole, que permite representar por circuitos eletrônicos digitais (portas lógicas) os números, caracteres, realizar operações lógicas e aritméticas.

3.4 Sistema de numeração hexadecimal

No início do desenvolvimento dos computadores, foi percebido que as pessoas tinham muitas dificuldades em lidar com números binários. Por esta razão, um novo sistema, com 16 símbolos diferentes foi criado. É chamado sistema de numeração hexadecimal, consistindo dos dez dígitos que já estamos acostumados a usar (0, 1, 2, 3, ... 9) e mais seis letras do alfabeto A, B, C, D, E e F.

Para ilustrar a vantagem de representar um número no sistema hexadecimal, consideremos o seguinte caso: O maior número que pode ser representado por 4 dígitos binários é o número 1111. Ele corresponde ao número 15 no sistema decimal, representado por dois dígitos, enquanto que no sistema hexadecimal é representado por um único dígito, 'F'. É o maior número de 1 dígito do sistema hexadecimal. O maior número escrito com oito dígitos binários é, ao mesmo tempo o maior número de 2 dígitos hexadecimais.

Na figura 1.3 pode ser observado um exemplo da correlação existente entre números no sistema binário e no sistema hexadecimal.

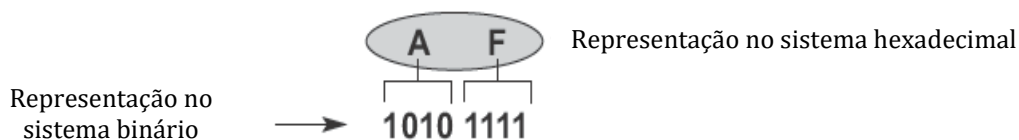


Figura 1 – Mesmo número sendo representado por 8 dígitos binários e 2 dígitos hexadecimais.

3.5 Conversão entre sistemas de numeração

O sistema numérico binário é mais comumente usado no estudo de sistemas digitais, como os microcontroladores. O sistema decimal, por sua vez, é mais compreensível, enquanto o sistema hexadecimal seria algo como um meio termo. Portanto, é muito importante aprender como converter números de um sistema numérico para outro.

Conversão binário-decimal

Dígitos em um número binário têm valores diferentes dependendo da posição que eles ocupam no número. Além disso, cada posição pode conter 1 ou 0 e seu valor pode ser facilmente determinado através da contagem de sua posição da direita para esquerda.

Para fazer a conversão de um número binário para decimal é necessário multiplicar os valores dos dígitos correspondentes (0 ou 1) pelo seu respectivo peso na base binária e adicionar todos os resultados. Veja o exemplo a seguir:

Número binário	Mesmo número no sistema decimal
110	$1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 6$

Note-se que, por exemplo, para representar números decimais de 0 a 3, são necessários apenas dois dígitos binários. Para números maiores, dígitos binários extras devem ser usados. Assim, a fim de representar números decimais 0-7, são necessários três dígitos binários, conforme o exemplo acima. Para os números 0-15 são necessários quatro dígitos, etc.

Com n dígitos binários é possível representar 2^n valores decimais. Para saber qual o maior número decimal que pode ser representado com n dígitos binários, simplesmente calculamos o valor de $2^n - 1$. Assim, por exemplo, se $n = 4$:

$$2^4 - 1 = 16 - 1 = 15.$$

Assim, usando 4 dígitos binários, é possível representar números decimais de 0 a 15, o que equivale a 16 diferentes valores no total.

Conversão hexadecimal-decimal

A fim de fazer a conversão de um número hexadecimal em decimal, cada dígito hexadecimal deve ser multiplicado pelo seu respectivo peso na base hexadecimal e adicionar todos os resultados, semelhante ao que foi feito no sistema binário. Por exemplo:

A37E	Número hexadecimal	
		$14 \cdot 16^0 = 14 \cdot 1 = 14$
		$7 \cdot 16^1 = 7 \cdot 16 = 112$
		$3 \cdot 16^2 = 3 \cdot 256 = 768$
		$10 \cdot 16^3 = 10 \cdot 4096 = 40960$
		<u>41854</u>
	Mesmo número no sistema decimal	

Conversão hexadecimal-binário

Não é necessário realizar nenhum cálculo a fim de converter números de hexadecimal para binário. Os dígitos hexadecimais são simplesmente substituídos por dígitos binários equivalentes. Desde que o máximo valor representado por um dígito hexadecimal é equivalente ao número decimal 15, nós precisamos usar quatro dígitos binários para representar um dígito hexadecimal. Por exemplo:

E4	=	11100100
		└──┴──┘ └──┴──┘
		E 4

O quadro comparativo mostrado na figura 2, na página seguinte, contém os valores dos números que vão de 0 a 255 nos três diferentes sistemas numéricos apresentados. Esta é a maneira mais fácil de entender a lógica comum aplicada a todos os sistemas. Observe que para representar os valores 0-255 nesses sistemas são necessários:

- * 3 dígitos no sistema decimal;
- * 8 dígitos no sistema binário;
- * 2 dígitos no sistema hexadecimal.

O sistema numérico decimal juntamente com os sistemas binário e hexadecimal são considerados os sistemas numéricos mais importantes para as aplicações com microcontroladores.

É fácil fazer a conversão entre as bases. No entanto, essas conversões podem causar confusão. Por exemplo, o que significa a frase “É preciso contar 110 produtos na linha de montagem”? Dependendo se tratar de um valor binário, decimal ou hexadecimal, a quantidade de produtos a serem contados poderia ser 6, 110 ou 272, respectivamente! Assim, a fim de evitar mal-entendidos, diferentes prefixos e sufixos são adicionados diretamente aos números para distinguir cada uma das representações.

O prefixo \$ ou 0x, bem como o sufixo h marcam os números no sistema hexadecimal. Por exemplo, o número hexadecimal 10AF pode ser escrito como \$10AF, ou 0x10AF, ou ainda 10AFh. Da mesma forma, os números binários geralmente usam os prefixos % ou 0b. Se um número não tem nem o prefixo nem sufixo é considerado decimal. Infelizmente, esta maneira de marcar os números não é padronizada e depende da concreta aplicação.

DEC.	BINARY								HEX.
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0	2
3	0	0	0	0	0	0	1	1	3
4	0	0	0	0	0	1	0	0	4
5	0	0	0	0	0	1	0	1	5
6	0	0	0	0	0	1	1	0	6
7	0	0	0	0	0	1	1	1	7
8	0	0	0	0	1	0	0	0	8
9	0	0	0	0	1	0	0	1	9
10	0	0	0	0	1	0	1	0	A
11	0	0	0	0	1	0	1	1	B
12	0	0	0	0	1	1	0	0	C
13	0	0	0	0	1	1	0	1	D
14	0	0	0	0	1	1	1	0	E
15	0	0	0	0	1	1	1	1	F
16	0	0	0	1	0	0	0	0	10
17	0	0	0	1	0	0	0	1	11
.....									
.....									
.....									
253	1	1	1	1	1	1	0	1	FD
254	1	1	1	1	1	1	1	0	FE
255	1	1	1	1	1	1	1	1	FF

Figura 2 - Tabela comparativa mostrando os valores de 0 a 255 escritos em três diferentes sistemas numéricos: decimal, binário e hexadecimal.

3.6 Conceito de *bit*

Um bit (*binary digit*) nada mais é que um dígito binário. Semelhante ao sistema de numeração decimal, em que os dígitos de um número não têm o mesmo valor (por exemplo, os dígitos do número decimal 444 são os mesmos, mas têm valores diferentes), a significância de um bit depende de sua posição no número binário. Como não há sentido falar de unidades, dezenas, etc, em números binários, seus dígitos são referidos como o bit zero (bit mais à direita), primeiro bit (o segundo da direita), segundo bit, terceiro bit e assim sucessivamente. Além disso, uma vez que o sistema binário utiliza dois dígitos apenas (0 e 1), o valor de um bit pode ser 0 ou 1.

3.7 Conceito de *byte* e *nibble*

Um byte é composto por oito bits agrupados. Se um bit é um dígito, é lógico dizer que bytes representam números. Todas as operações matemáticas podem ser realizadas sobre os bytes, igualmente aos números decimais.

Semelhante aos dígitos de um número qualquer, os dígitos de um byte não têm o mesmo significado. O bit mais à esquerda tem o maior valor, e é chamado de bit mais significativo (*MSB – Most Significant Bit*). O bit mais à direita tem o menor valor e é chamado de bit menos significativo (*LSB – Least Significant bit*).

Os 8 bits de um byte podem assumir valores 0 ou 1, independentemente. Assim, esses 8 bits podem ser combinados em $2^8 = 256$ formas diferentes. O maior número decimal que pode ser representado por um byte é, por sua vez, $2^8 - 1 = 255$.

Um *nibble* é referido como metade de um byte (4 bits). Dependendo de que metade do byte estivermos falando (esquerda ou direita), nos referimos ao “*nibble* alto” (*high nibble*) ou “*nibble* baixo” (*low nibble*), respectivamente.

Os conceitos de bit, byte e nibble podem ser entendidos observando a figura 3. ■

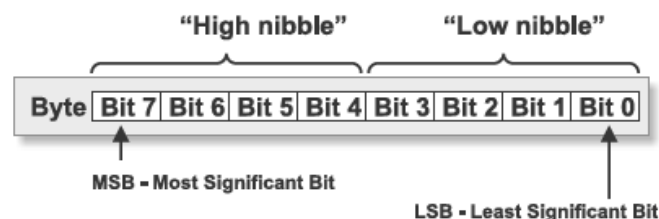


Figura 3 – Representação dos conceitos de bit, byte e nibble.