



**UNIVERSIDADE FEDERAL DO CEARÁ  
CAMPUS DE CRATEÚS  
CIRCUITOS DIGITAIS – CRT0384**

**RELATÓRIO – PRÁTICA 07  
UNIDADE LÓGICA ARITMÉTICA – ULA**

**MARLON GONÇALVES DUARTE**

**PROFESSORES: MARCIEL BARROS PEREIRA  
RENNAN FERREIRA DANTAS**

**CRATEÚS – CEARÁ  
OUTUBRO DE 2020**

SUMÁRIO

	Pág.
INTRODUÇÃO	3
OBJETIVOS	7
MATERIAIS E MÉTODOS	8
RESULTADOS E DISCUSSÕES	9
CONCLUSÃO	13
REFERÊNCIAS	14

## INTRODUÇÃO

Em abril de 1970, o engenheiro eletricitista Federico Faggin se juntou à Intel, empresa que já atuava na produção de circuitos de memória, para desenvolver um projeto que foi o estopim para o expandir da computação no mundo. Então, em 1971 era apresentado o primeiro microprocessador que ocuparia as prateleiras dos mercados de computadores no mundo, o Intel 4004. O CI 4004 fazia parte de uma família composta pelo 4001 que é uma memória ROM, pelo 4002 uma memória RAM e 4003 um *shift register*, ou registrador de deslocamento, que atua deslocando os dados conforme sua configuração.

Nas estruturas de um processador como o 4004, existe um circuito que compõe um papel importantíssimo no funcionamento do dispositivo, a **ALU** – *arithmetic logic unit*, em português, unidade lógica e aritmética. Uma ULA de um sistema de 4 bits funciona através de 4 grupos de sinais de entrada compostos por 4 bits cada: Entradas “A” que será manipulada com as entradas “B”, a seleção da forma como serão realizados os cálculos é responsabilidade das entradas “S” e os resultados são exibidos na saída “F”. Uma representação de uma ULA pode ser visto na figura 1.

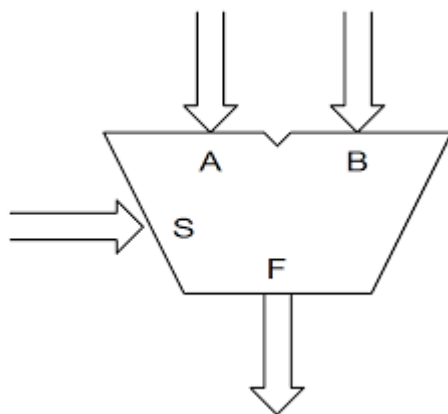


Figura 1: Representação de uma ula com entradas e saídas de dados.

A ULA é um circuito que tem como responsabilidade a execução de cálculos aritméticos e lógicos, em um sistema eletrônico digital. O “seletor” (entradas “S” da ULA) determinará que tipo de dado deve ser exibido na saída através de uma determinada manipulação, aritmética ou lógica. A complexidade da ULA está relacionada à complexidade do sistema, se o sistema for simples a ULA será simples com poucos cálculos, mas, se complexo a ULA terá de realizar inúmeras operações, portanto, será complexa. A ULA pode realizar inúmeras operações, a depender de seu tipo e aplicação, porém sempre terão, pelo menos, funções de soma, subtração, divisão, multiplicação, lógica-OR, lógica-AND, lógica-NOT, lógica-XOR, entre outras.

O conhecimento acerca do sistema que utilizará a ULA é fundamental para avaliar qual tipo deverá ser usado. Tendo consciência do tamanho e necessidade do sistema, é necessário avaliar também a relação entre velocidade e preço. Por exemplo, calculadoras eletrônicas realizam operações complexas de vários tipos, mas não tem uma exigência de velocidade tão alta, assim serão menos dispendiosas. Contudo, os computadores modernos exigem cálculos complexos e extensos a uma velocidade elevada, o que acaba tornando estes muito mais caros.

Um exemplo de uma ULA é o circuito integrado 74LS181, segundo a folha de dados do componente: “O SN54 / 74LS181 é uma Unidade Lógica Aritmética (ALU) de 4 bits que pode realizar possivelmente 16 operações lógicas, operações em duas variáveis e uma variedade de operações aritméticas.” (*The SN54/74LS181 is a 4-bit Arithmetic Logic Unit (ALU) which can perform all the possible 16 logic, operations on two variables and a variety of arithmetic operations*). Na figura 2 é apresentado um diagrama simplificado do dispositivo. Na saída T, entenda-se por P como determina a folha de dados do dispositivo.

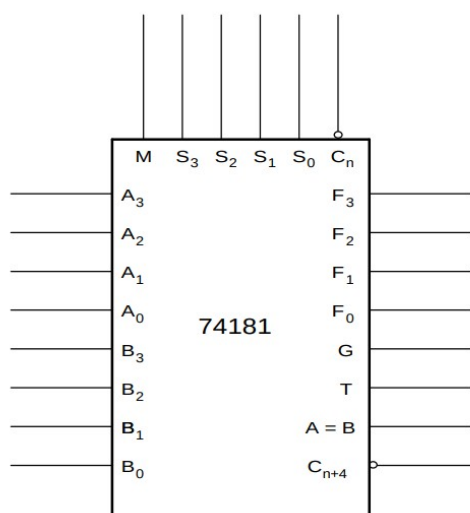


Figura 2: Diagrama simplificado do CI 74LS181.

Os terminais  $A_0 - A_3$  estão para recebimento de um nibble, numeração binária de 4 bits assim como os terminais  $B_0 - B_3$ . Já os de nomenclatura  $F_0 - F_3$ , são saídas de palavras de 4 bits resultas das operações controladas pelos terminais  $S_0 - S_3$ . Como as entradas de controle recebem 4 bits, já se imagina que esse dispositivo possa realizar 16 operações distintas, e sim, ele pode. Contudo o terminal M funciona como um controlador determinando a operação que será realizada, aritmética ou lógica, desta forma, o circuito oferece 32 operações diferentes pois o M seria um bit mais significativo de uma palavra de 5 bits e  $2^5=32$ . A seguir é exibida a tabela 1 com as funções das entradas do dispositivo.

PINAGEM	TIPOS	DESCRIÇÃO
$A_0$ a $A_3$ e $B_0$ a $B_3$	Entradas	Dados de entrada
$C_n$	Entradas	Bit para o chamado “vem um” (Carry in)
$S_0$ a $S_3$	Entradas	Seletor de operações
<b>M</b>	Entradas	Modo de operação: M=0 para operações aritméticas M=1 para operações lógicas
$F_0$ a $F_3$	Saídas	Saída dos resultados
$C_{n+4}$	Saídas	Bit para o chamado “vai um” (Carry out)
<b>G e P (T no diagrama)</b>	Saídas	Expansão por lookahead
<b>A=B</b>	Saídas	Testa a igualdade das entradas

Tabela 1: Tabela de funções dos terminais do CI 74LS181.

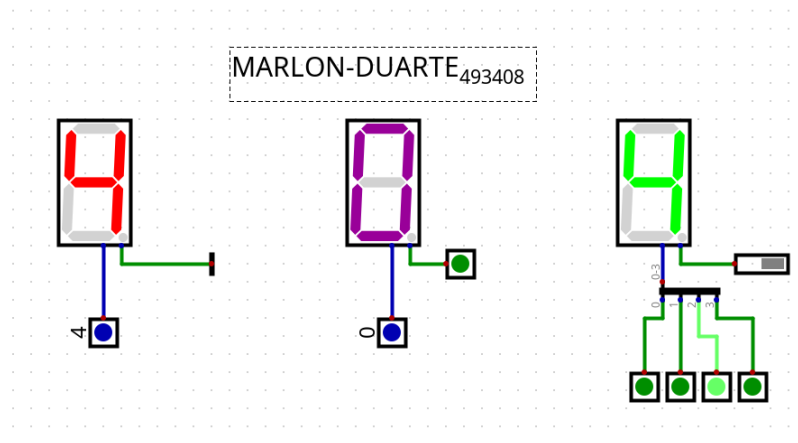
As entradas G e P (no diagrama, P está como T), têm a função de “gerador de vai um”, no que diz respeito ao bit mais significativo. A letra “G” está relacionada com a palavra “Generate” (em português *gerar*). O *Generate* atua quando as entradas  $A_0 - A_3$  e  $B_0 - B_3$  do somador são iguais a 1, independente do “vem um” ( $C_n$ ), a propagação do “vai um” existirá, ou seja, foi “*gerado*”. Assim o *carry in* (vem um) da sequência receberá o valor 1. A letra “P” está relacionada com a palavra “*Propagate*” (em português *propagar*). Quando as entradas  $A_0 - A_3$  e  $B_0 - B_3$  forem iguais a 1 e 0 ou a 0 e 1 respectivamente, o *carry in* sairá no *carry out*, portanto, será “propagado”. De uma forma mais direta, a compreensão destas informações pode ser visualizada na tabela 2.

	A – A	B – B	$C_{out}$
P	0	1	$C_{in}$
P	1	0	$C_{in}$
G	1	1	1

Tabela 2: Tabela verdade das portas P e G para carry out no somador

Existe também um pino com nomenclatura A=B, este pino terá saída alta quando todas as quatro saídas F estiverem em alta pois é constituída de um AND entre as saídas  $F_i$ . Desta forma, poderá ser utilizada para indicar a equivalência lógica em 4 bits quando a unidade está no modo de subtração. A saída A=B é aberta em relação ao coletor do transistor e pode ser usada através de *wired-AND* (método que consiste em compor uma porta lógica usando apenas componentes eletrônicos básicos como: fios, diodos, resistores, etc.) com outras saídas A=B de outro CI para fornecer uma comparação para mais de 4 bits. O sinal A=B também pode ser usado com o sinal  $C_{n+4}$  para indicar  $A > B$  e  $A < B$ .

Na prática 7, são iniciadas as atividades com unidades lógico e aritméticas – ULA, que são componentes fundamentais em todo sistema computacional, haja vista a função destes dispositivos que é a realização de diversos cálculos necessários tanto na obtenção de resultados propriamente matemáticos, como para abstrações utilizadas pela computação. O circuito integrado utilizado nessa prática é o 74LS181 que já foi comentado anteriormente. O CI foi montado em um sistema com 4 displays hexadecimais. Na figura 3 são mostradas possibilidades de ligação deste display.



*Figura 3: Representação de um display de 7 segmentos hexadecimal em que se pode perceber a presença de apenas dois pinos de entrada.*

O display hexadecimal disponível na biblioteca do aplicativo Digital possui uma entrada composta – é uma entrada que, no diagrama, apresenta apenas um pino mas que este pino comporta 4 bits de fluxo. Além do pino composto esse display tem mais um pino, porém esse pino é comum e serve apenas para controlar o brilho do ponto decimal.

## **OBJETIVOS**

- Interpretar o funcionamento de uma ULA como a que integra o intel 4004;
- Preencher uma tabela verdade alterando os valores das entradas, do seletor e do modo;
- Apresentar as funções dos pinos  $P$ ,  $G$  e  $A=B$ ;
- Implementar uma circuito capaz de fazer operações em 8 bits utilizando duas ULAs;

## MATERIAIS E MÉTODOS

Para a realização desta prática foram utilizadas algumas ferramentas bastante comuns a essas práticas, dentre elas estão: *Datasheet* (folha de informações) dos componentes; simulador de circuitos lógicos “Digital”, um projeto base com o CI 74LS181 em extensão .dig fornecidos pelo professor. Durante a busca para as soluções dos problemas, foram priorizados alguns passos como uma forma de organizar o trabalho, bem como minimizar as falhas que viessem a ser sofridas, obtendo a seguinte sequência:

1. Fazer uma leitura reflexiva das tarefas;
2. Estudar as folhas de dados (*datasheet*) do circuito integrado 74181;
3. Refazer as ligações dos projetos pré implementados pelo material de apoio no intuito de compreender as possibilidades de configurações;
4. Responder às tabelas verdade solicitadas pelo material para as quais foi necessária extrema atenção vide a grande quantidade de dados inseridos;
5. Implementar o circuito para manipulação de 8 bit utilizando dois CI 74181 e agora 6 displays hexadecimais;



## RESULTADOS E DISCUSSÕES

A questão inicial desta prática foi respondida já na introdução deste relatório, no instante em que se é falado sobre o funcionamento de uma ULA, onde na oportunidade também foi realizado um comentário a respeito do primeiro processador comercial desenvolvido pela intel, cujo qual possui em sua estrutura interna, dentre outras coisas, de uma ULA conforme o seu diagrama apresentado na figura 4.

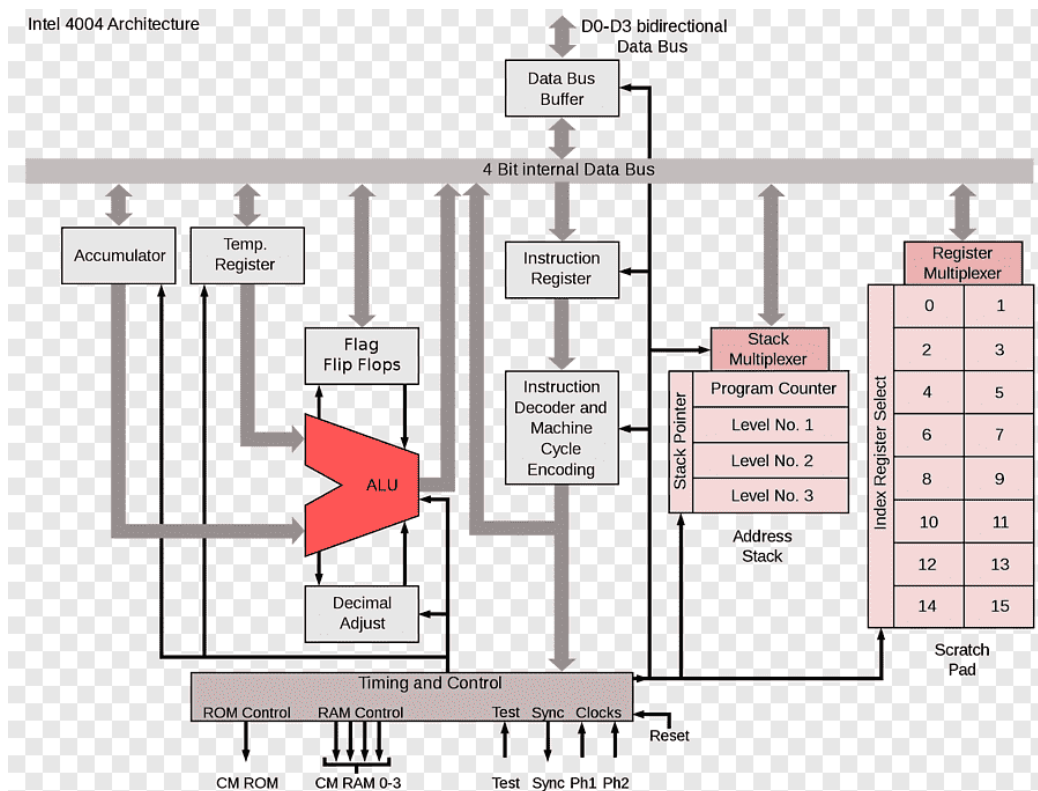


Figura 4: Diagrama em blocos do circuito integrado 4004 desenvolvido pela Intel.

A questão 2 da prática 7, solicita a procura pelos resultados de saída nas portas  $F_{0-3}$  do CI através de uma série de nibble disponibilizados em uma lista inseridos em suas entradas  $A_{0-3}$ ,  $B_{0-3}$ ,  $M$  e  $S_{0-3}$ . Essa parte foi cansativa pela grande quantidade de operações que foram necessárias realizar para se obter as tabelas. Nessa questão as entradas de  $F$  deveriam estar setadas para 0110, 1001, 1011 e 1110 com o  $M$  em baixa. Dessa forma seriam manipuladas as entradas em  $A$  minus  $B$  minus 1, que é uma nomenclatura para complemento de 2 segundo o datasheet. Com 1001 seria a soma de  $A$  e  $B$ . Com 1011 seria  $A + B$  que é um OR bitwise. E, finalmente, 1110 seria  $A \neg B$  somado a  $A$ . O resultado das operações pode ser observado na tabela 3.

Input A	Input B	M=0/S=0110	M=0/S=1001	M=0/S=1011	M=0/S=1110
0101	1100	0x8/0b1000	0x11/0b10001	0xd/0b1101	0x6/0b0110
1111	0011	0x1b/0b11011	0x12/0b10010	0xf/0b1111	0x16/0b10110
1011	1011	0xf/0b1111	0x16/0b10110	0xb/0b1011	0xb/0b1011
0100	0101	0xe/0b1110	0x9/0b1001	0x5/0b0101	0x4/0b0100
1000	0111	0x10/0b10000	0xf/0b1111	0xf/0b1111	0x10/0b10000
1010	0010	0x17/0b10111	0xc/0b1100	0xa/0b1010	0x12/0b10010
1100	1110	0xd/0b1101	0x1a/0b1010	0xe/0b1110	0xc/0b1100
0001	1011	0x5/0b101	0xc/0b1100	0xb/0b1011	0x1/0b0001
1001	1111	0x9/0b1001	0x18/0b11000	0xf/0b1111	0x9/0b1001
1011	0000	0x1a/0b11010	0xb/0b1011	0xb/0b1011	0x16/0b10110
0101	0010	0x12/0b10010	0x7/0b0111	0x7/0b0111	0xa/0b1010
0100	1010	0x9/0b1001	0xe/0b1110	0xe/0b1110	0x8/0b1000
0011	0010	0x10/0b10000	0x5/0b0101	0x3/0b0011	0x4/0b0100
0010	1010	0x7/0b111	0xc/0b1100	0xa/0b1010	0x2/0b0010
1011	1110	0xc/0b1100	0x19/0b11001	0xf/0b1111	0xc/0b1100
0101	0101	0xf/0b1111	0x9/0b1001	0x5/0b0101	0x5/0b0101

*Tabela 3: Tabela correspondente à 2ª questão da prática 7.*

Na questão 3 era solicitado algo bastante parecido com a 2, porém com um diferencial em relação a entrada M que agora deveria ser alta. Assim, mudaram as operações que seriam realizadas entre os resultados de entrada. Com 0110 em F, o CI faz operações tipo negação de XOR. Já no caso de 1001 é apenas um XOR. Com 1011 obtém-se um OR entre A e B. Por fim, com 1110 a operação será de um AND entre A e B. O resultado pode ser observado na tabela 4.

Input A	Input B	M=1/S=0110	M=1/S=1001	M=1/S=1011	M=1/S=1110
0101	1100	0x6/0b0110	0x19/0b11001	0xd/0b1101	0x4/0b0100
1111	0011	0x13/0b11001	0x1c/0b11100	0xf/0b1111	0x13/0b10011
1011	1011	0xf/0b1111	0x10/0b10000	0xb/0b1011	0x6/0b0110
0100	0101	0xe/0b1110	0x1/0b0001	0x5/0b0101	0x4/0b0100
1000	0111	0x10/0b10000	0xa/0b1010	0xf/0b1111	0x10/0b10000
1010	0010	0x17/0b10001	0x8/0b1000	0xa/0b1010	0x12/0b10010
1100	1110	0xd/0b1101	0x12/0b10010	0xe/0b1110	0xc/0b1101
0001	1011	0x5/0b0101	0xa/0b1010	0xb/0b1011	0x1/0b0001
1001	1111	0x9/0b1001	0x16/0b10110	0xf/0b1111	0x9/0b1001
1011	0000	0x14/0b10100	0xb/0b1011	0xb/0b1011	0x10/0b10000
0101	0010	0x18/0b11000	0x7/0b0111	0x7/0b0111	0x0/0b0000
0100	1010	0x1/0b0001	0xe/0b1110	0xe/0b1110	0x0/0b0000
0011	0010	0x1e/0b11110	0x1/0b0001	0x3/0b0011	0x2/0b0010
0010	1010	0x7/0b0111	0x8/0b1000	0xa/0b1010	0x2/0b0010
1011	1110	0xa/0b1010	0x15/0b1111	0xe/0b1110	0xa/0b1010
0101	0101	0xf/0b1111	0x0/0b0000	0x5/0b0101	0x5/0b0101

Tabela 4: Tabela resultante da resolução da 3ª questão.

A questão 4 da prática 7 indaga sobre as funções dos pinos P, G e A=B. Porém, assim como a questão 1, esse assunto já foi abordado no início deste relatório. Cabendo aqui fazer apenas um breve comentário relacionado a essas portas: Nas pesquisas que foram realizadas na internet, há algumas explicações sobre essas portas mas não foi encontrado nenhum projeto que utiliza estas portas, o que acabou trazendo uma insegurança que foi contornada com uma leitura cuidadosa da folha de dados do componente. Como mencionado no início do relatório, P é um *propagate*, G um *generate* e A=B apenas tem relação com os bits de  $F_{0-3}$  ou em conjunto o  $C_{n+4}$  serve como um comparador.

A última questão desta prática solicita a implementação de um circuito que realiza as mesmas operações do circuito disponibilizado pelo professor, porém que, agora, trabalhe não mais com 4 bits e sim com 8 bits. Para obter esse resultado foi necessário somar dois CIs 74181 e relacioná-los através das portas  $C_{n+4}$  e  $C_n$ , carry out e carry in respectivamente. O resultado pode ser observado na figura 5.

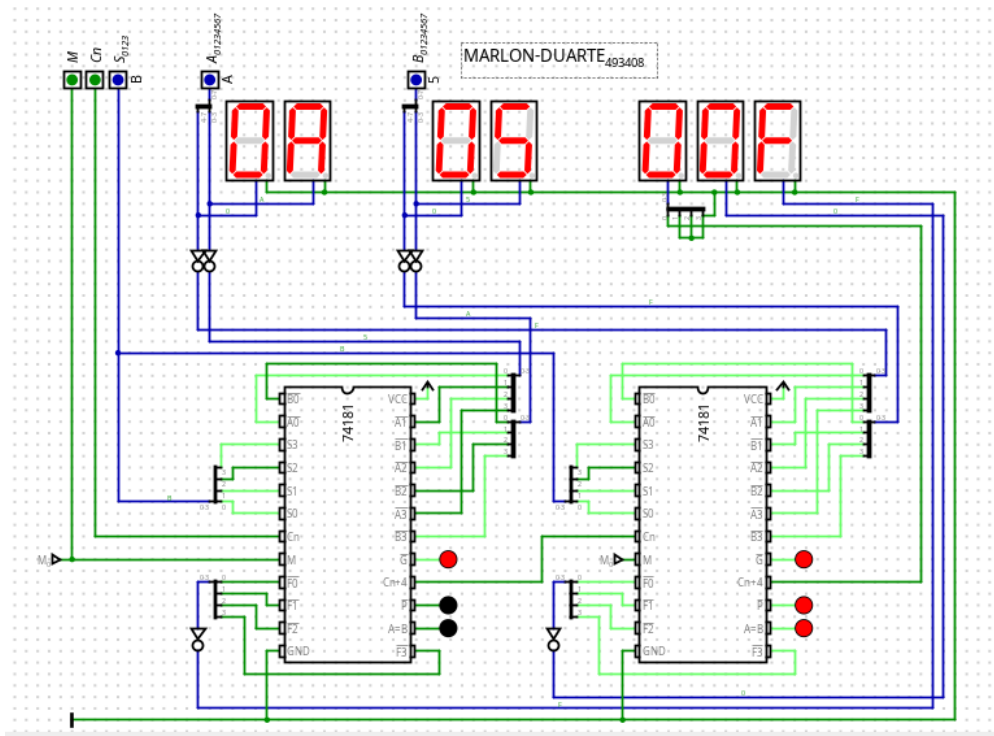


Figura 5: Circuito resultante da união de duas ULAs para trabalhar com 8 bits.

Para a montagem desse circuito, decidiu-se começar do zero, de forma que fosse necessário fazer cada uma das ligações, evitando apenas copiar e colar o projeto disponibilizado pelo professor. Com isso, foi possível perceber melhor como funciona o circuito além de permitir a prática com a montagem do circuito mesmo que em ambiente virtualizado como é o oferecido pelo Digital.

## CONCLUSÃO

Na prática 7, foi possível ter uma noção de como é complexo um sistema interno de um processador apesar de a ULA ser apenas uma das partes que compõe a estrutura de um processador moderno. O processador necessita receber dos programas ou do usuário uma grande quantidade de números, alguns para serem manipulados e outros para informar como devem ser manipulados os números anteriores.

Apesar de ter ficado muito claro o funcionamento da ULA apresentada pela prática 7. É possível perceber que a troca do display hexadecimal por leds comuns, facilitaria a compreensão em um primeiro instante pois algumas operações de uma ULA, por exemplo as lógicas, não são tão claras quando apresentadas de imediato a um novato na eletrônica digital como é o caso dos alunos do curso de ciência da computação. Sendo utilizados leds como saídas, a visualização dos resultados se daria em forma binária, o que é uma maneira mais simples de representação. Quando solicitado uma operação de AND e se insere 3 decimal em A e 2 decimal em B, a saída exibirá 2 decimal pelo fato da operação ser bitwise. Mas o leigo aguardaria na saída um 0 ou 1, falso e verdadeiro, por estar bitolado nas operações lógicas comuns.

Contudo, há de se concordar que esta prática foi de suma importância para a inserção em um universo que fica cada vez mais complexo e cheio de possibilidades. É difícil, no começo da carreira de cientista da computação, imaginar como alguns componentes coloridos soldados em uma placa de fenolite exibem na tela um jogo que pode ser manipulado e lhe prender por horas. Mas a medida em que se tem contato com a maneira como os dados são trabalhados dentro da eletrônica digital, fica possível imaginar como se dão as coisas nesse universo vasto e, praticamente, infinito que é a computação.

## REFERÊNCIAS

DE LIMA KASTENSMIDT, Fernanda Gusmão. Disciplina: Sistemas Digitais para Computação.

INSTRUMENTS, Texas. **The TTL data book for design engineers**. Texas Instruments, 1984.

INTEL 4004. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2020.

Disponível em: <[https://pt.wikipedia.org/w/index.php?title=Intel\\_4004&oldid=59583408](https://pt.wikipedia.org/w/index.php?title=Intel_4004&oldid=59583408)>. Acesso em: 13 out. 2020.

TANENBAUM, Andrew S.; ZUCCHI, Wagner Luiz. **Organização estruturada de computadores**.

Pearson Prentice Hall, 2009.

TOCCI, Ronald J.; WIDMER, Neal S.; MOSS, Gregory L. **Sistemas digitais**. Pearson Educación, 2010.