TRABALHO 7

ELEMENTOS DE LÓGICA COMBINACIONAL

Objectivos:

Pretende-se que o aluno tenha um primeiro contacto com o material utilizado nas aulas de electrónica digital, em particular com o módulo de montagem DT-O1 e sua utilização, ao mesmo tempo que procede à montagem de circuitos simples de lógica combinacional.

Material:

- Módulo DT-O1
- Multímetro digital
- Osciloscópio
- Elementos lógicos (SN7400, SN7404, etc),
- fios, resistências, potenciómetros, LEDs, etc.

INTRODUÇÃO AO DT-O1 (DIGITAL TRAINER) (1 hora)

O módulo de montagem DT-O1 foi concebido para simplificar a montagem e o ensaio de circuitos lógicos simples. O módulo inclui uma placa de ensaios *(bread-board),* alimentação TTL compatível (+ 5V), geradores de diversos sinais (impulsos, nível e *clock),* indicadores ligados a LEDs e ainda adaptadores tipo ficha BNC ou banana para ligação a outros instrumentos. Em seguida são sugeridas algumas experiências para que o aluno se familiarize com o DT-O1:

Interruptor principal

Localizado no canto superior esquerdo. Um pouco ao lado está um LED que se acende quando se fornece energia ao módulo.

Nota: Certifique-se sempre que o interruptor está desligado quando montar/desmontar os circuitos.

Placa de ensaios ou *bread board*

Permite estabelecer ou desfazer ligações de forma eficaz. Já é do conhecimento do aluno. Imediatamente acima da placa de ensaios e também à sua direita existem orifícios para ligações eléctricas compatíveis com os fios que se utilizam na placa de ensaios e que servem para ter acesso à alimentação, às funções e às ligações externas.

Alimentação

Aparece em dois sítios, do lado direito da placa de ensaios e, por cima desta, do lado esquerdo. Está identificada com as letras GND e + 5V. Como, tipicamente, em cada montagem é necessário fazer um grande número de ligações à alimentação sugere-se que se usem as faixas de ligação longitudinais da placa de ensaios para GND (terra) e + 5V.

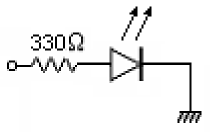
Nota: Embora a alimentação esteja protegida contra curto-circuitos convém evitá-los!

Indicadores

Existem 8 indicadores, numerados de 1 a 8 (LED DISPLAY). Estas ligações funcionam como entradas. Se lhes for aplicado um sinal de nível lógico **0** (> 2.0V) o LED acende e se for de nível lógico **1** (< 0.8V) o LED apaga-se.

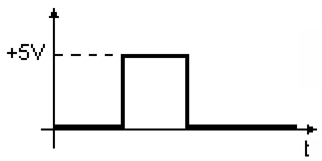
Experiência: Ligue um fio de um LED a + 5V ou a GND. O LED deve acender-se no primeiro caso e manter-se apagado no segundo.

Experiência: Monte o circuito apresentado e ligue a sua entrada alternadamente a + 5V e a GND. Este circuito é equivalente aos do módulo DT-O1. A resistência em série com o LED limita a corrente que passa no LED, evitando que este se queime.



Gerador de impulsos

Existem dois geradores de impulsos, numerados P1 e P2 (PULSE SW). O estado normal da saída é **0** (0V). No momento em que se pressiona o botão a saída correspondente vai a **1** (+ 5V) durante 0.5s (valor aproximado). A duração do impulso gerado é independente do tempo durante o qual se mantém o botão pressionado.



Experiência: Ligue a saída de um gerador de impulsos a um LED. Pressione rapidamente o botão e observe o LED. Em seguida pressione. o botão e mantenha-o pressionado durante alguns segundos. O que se passou com o LED?

Comutadores lógicos

Existem oito comutadores lógicos, numerados de 1 a 8 (LOGIC SW). A saída de cada comutador está a **1** (+ 5V) se o comutador correspondente estiver na posição H *(High)* e a **0** (0V) se o comutador estiver na posição L *(Low).*

Experiência: Ligue um ou mais comutadores ao mesmo número de LEDs e verifique em que condições estes ficam acesos ou apagados.

Nota: Quando não estiverem em uso estes comutadores devem ser mantidos, de preferência, na posição H. A razão é um menor consumo de corrente e uma menor sobrecarga da fonte de alimentação.

Relógio *(Clock)*

Existe um relógio, identificado por CLOCK, que gera uma onda quadrada, com níveis 0V e + 5V, de frequência ajustável desde alguns hertz até algumas dezenas de kilohertz. O ajuste da frequência é feito com um comutador que selecciona a gama de frequências (baixas, 10-40Hz, ou altas, 1k-20kHz) e um potenciómetro que permite uma variação contínua.

Experiência: Ligue a saída do *clock* a um LED. Seleccione baixas frequências e rode o botão do potenciómetro. O que observa? Repita a operação com o selector na posição altas frequências.

Adaptadores tipo BNC e tipo banana

Existem dois conectores tipo BNC, numerados 1 e 2, e dois tipo banana, numerados 3 e 4, que podem ser ligados à placa de ensaios através dos orifícios de ligação, respectivamente, 1, 2, 3 e 4 (ADAPTOR)

Nota: Cada conector tipo BNC estabelece dois contactos eléctricos. Neste caso o segundo contacto (correspondente à malha) está internamente ligado à terra (GND).

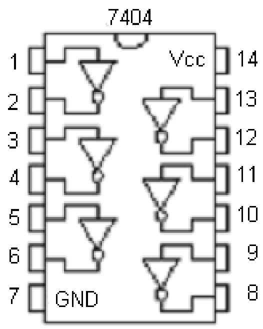
Experiência: Ligue a saída do *clock* a um LED e ao canal A do osciloscópio, neste caso fazendo uso do adaptador tipo BNC e de um cabo de ligação BNC-BNC. Regule o osciloscópio até tornar visível o sinal do *clock*. Meça o valor da tensão quando o sinal está em **0** e quando está em **1** (valores lógicos). Em seguida percorra toda a gama de frequências que o *clock* consegue gerar vizualizando sempre o sinal no osciloscópio.

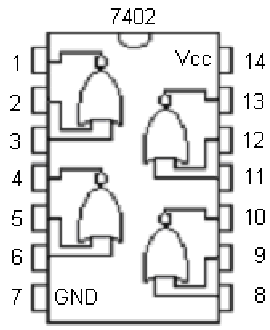
Agora substitua o cabo de ligação BNC-BNC por BNC-banana. Faça as alterações que considerar necessárias até voltar a obter um sinal nítido no osciloscópio. Este era um desafio! Senão conseguiu chame o professor.

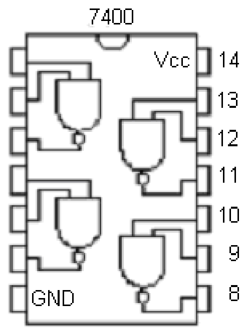
ESTUDO DE ALGUMAS PORTAS LÓGICAS (45 min.)

Vão ser estudadas as portas *(gates* em inglês) NOT (não), NAND (não e) e NOR (não ou). Estes dois últimos tipos de portas lógicas são muito importantes porque é possível projectar e montar um circuito que realize qualquer função de lógica combinacional utilizando só portas NAND ou utilizando só portas NOR. Mais nenhum tipo de portas lógicas tem esta propriedade.

As portas lógicas necessárias estão disponíveis em circuitos integrados com catorze pinos que serão identificados por quatro ou cinco algarismos. Os dois primeiros algarismos são sempre 74 e entre estes e os seguintes poderão aparecer letras. Comum a todos os integrados é a presença das ligações à alimentação, $+V_{cc}$ e GND, que o aluno não deverá esquecer-se nunca de completar. O 7404 inclui seis portas NOT, o 7400 quatro portas NAND de duas entradas e o 7402 quatro portas NOR de duas entradas. Os esquemas com as ligações internas são mostrados a seguir. No laboratório o aluno terá à disposição manuais mais completos.







Manuseamento dos integrados

Ao tirar os circuitos integrados das gavetas de arrumação certifique-se que são mesmo os que pretende, conferindo a inscrição nos integrados. Não confira o aspecto geral da inscrição (que indica apenas que o fabricante deve ser o mesmo), mas o respectivo número.

Manuseie os integrados sempre com o cuidado de não dobrar os pinos.

Para encaixar um integrado na placa de ensaios comece por verificar o estado dos pinos, endireite cuidadosamente com o alicate os que estiverem deformados, pouse o integrado sobre a placa na posição pretendida, verifique que todos os pinos estejam em posição de encaixe e só então pressione o integrado. Certifique-se que a operação foi bem sucedida.

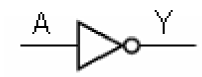
Para desencaixar um integrado da placa de ensaios use uma chave de fendas ou uma pinça como alavanca. Os pinos devem sair todos ao mesmo tempo e não primeiro uns e depois outros. Nunca retire os integrados com os dedos.

Ao repor os integrados (não avariados!) nas gavetas de arrumação tenha o cuidado de não os baralhar.

À menor dúvida consulte o professor.

A porta NOT

Experiência: Seleccione um 7404, encaixe-o na placa e estabeleça a ligação das alimentações. Ligue a saída de um dos comutadores lógicos à entrada de um dos NOTs e a um LED. Ligue a saída desse NOT a outro LED. (Ficaram cinco portas NOT por utilizar.)

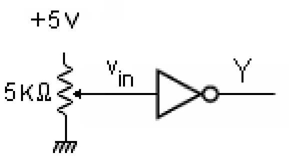


Complete a tabela ao lado, correspondendo **0** ao LED apagado e **1** ao LED aceso.

Os valores da tensão devem ser medidos com um multímetro.

A	(Volt)	Y	(Volt)
0			
1			

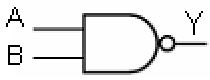
Experiência: O estado lógico **0** não corresponde exactamente a 0V nem o estado lógico **1** a + 5V. Existe uma margem de ruído. Para o confirmar monte o circuito representado no esquema, ligando a saída Y a um LED e lendo v_{in} com o multímetro.



Variando o potenciómetro verifique que:
v_{in} < 0.8V => **LED** aceso
v_{in} > 2.0V => **LED** apagado
(O LED muda de condição para 0.8V<v_{in}<2.0V)

Nota: Nas portas TTL, uma entrada entre 0V e 0.8V é reconhecida como **0** e uma entrada entre 2.0V e 5.0V é reconhecida como **1**. Os valores intermédios (entre 0.8V e 2.0V) não são permitidos.

A porta NAND

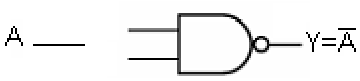


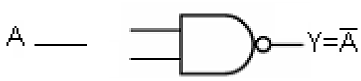
A	B	Y
0	**0**	
0	**1**	
1	**0**	
1	**1**	

Experiência: De forma idêntica ao que fez para a porta NOT, monte o circuito do esquema da porta NAND apresentado à direita e preencha a tabela.

Ligue as entradas e ou saídas que achar necessárias a LEDs de forma a saber o seu estado lógico.

Se pretender pode ler o valor da tensão com o multímetro. Em todo o caso, de agora em diante apenas interessa o valor lógico.
A porta lógica NAND pode ser usada em substituição de uma porta NOT. Para esse fim existem duas maneiras diferentes de fazer as ligações. Tente descobri-las por si, se não as conhece já, e desenhe-as completando os esquemas abaixo:





Experiência: Uma vez que já sabe implementar uma porta NOT usando uma porta NAND, torna-se fácil implementar uma porta AND só com portas NAND: $AB = \overline{\overline{AB}}$. Tirando partido das leis de DeMorgan torna-se muito fácil implementar uma porta OR com portas NAND: $A + B = \overline{\overline{A} \cdot \overline{B}}$. Complete o quadro seguinte com os esquemas apropriados, usando apenas portas NAND:

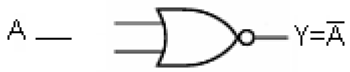
$Y = A \cdot B$ função AND	$Y = A + B$ função OR
$Y = A \oplus B$ função XOR (ou exclusivo)	$Y = ABC + AD$ função arbitrária

A porta NOR

A	B	Y
0	**0**	
0	**1**	
1	**0**	
1	**1**	

Experiência: Preencha a tabela ao lado e verifique que está de acordo com a experiência ou, alternativamente, preencha a tabela a partir de resultados experimentais.

Encontre duas maneiras de implementar a porta NOT com uma porta NOR. Represente-as nos espaços abaixo reservados.



Encontre também uma maneira de implementar uma porta NAND só com portas NOR. Desenhe o seu esquema e verifique experimentalmente.

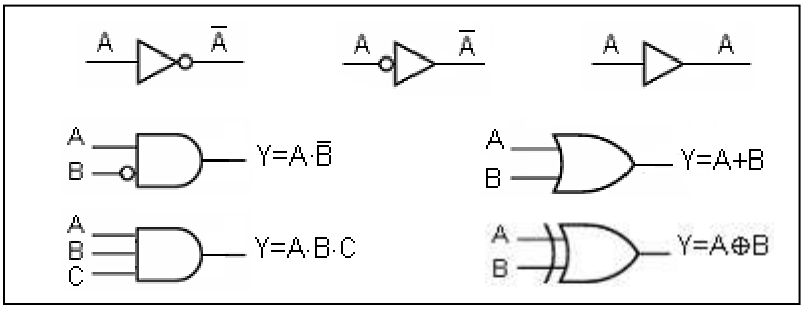
$Y = \overline{A \cdot B}$ função NAND

Tópicos sobre símbolos lógicos

Nos símbolos lógicos que têm vindo a ser utilizados neste guia, uma circunferência representa uma negação (da mesma maneira que um traço sobre uma variável representa uma negação). Essa circunferência tanto pode aparecer nas saídas como nas entradas.

As funções lógicas OR e AND gozam das propriedades comutativa e associativa, pelo que a ordem e associação dos factores é arbitrária. Assim sendo, é indiferente escrever $AB = (A \cdot B) \cdot C$ ou $AB = A \cdot B \cdot C$. Por esse motivo é habitual nos símbolos das funções OR e AND representar um número de entradas superior a dois. O mesmo se passa com as funções NOR e NAND. A função XOR tem um símbolo próprio (ver esquema abaixo).

Os exemplos que se apresentam em seguida servem de exemplificação e explicam-se por si mesmos:



PROJECTO DE CIRCUITOS LÓGICOS

Até agora o aluno teve contacto com as principais portas lógicas, estudou-as isoladamente ou integradas em circuitos muito simples como quando implementou uma porta NAND a partir de portas NOR. Com esta base o aluno está apto a abordar e resolver problemas mais complicados. A principal dificuldade é saber como começar e qual a maneira mais eficaz de chegar a uma solução, de preferência à melhor solução.

No exemplo guiado que se segue são introduzidas algumas ferramentas que, embora elementares, permitem obter bons resultados se os circuitos não forem muito complicados com a grande vantagem de serem de fácil aprendizagem e aplicação.

É muito importante chamar a atenção para o facto de os métodos descritos neste guia se aplicarem a circuitos sem memória, isto é, circuitos em que o estado lógico das saídas depende apenas do estado lógico das entradas nesse instante.

Exemplo guiado (45 min.)

Para este exemplo vamos projectar um circuito de três entradas, A, B e C e duas saídas G e P. As três entradas representam a votação de um júri de três elementos e estão em **1** se o elemento do júri correspondente votar favoravelmente e em **0** no caso contrário. A saída G (ganha) deve ser **1** se e só se dois ou mais elementos do júri votarem favoravelmente. A saída P (perde) deve ser **0** se e só se no máximo um elemento do júri votar favoravelmente.

Enunciado (ou concebido) o problema, o passo seguinte é convertê-lo em linguagem algébrica. Há diversas formas de o fazer e uma das mais simples consiste em construir a tabela de verdade.

A tabela de verdade, apresentada à direita, consiste na descrição do estado das saídas para todas as combinações possíveis de estados das entradas.

A partir da tabela de verdade pode obter-se a expressão algébrica de uma saída com um raciocínio do tipo «G é verdadeiro na "quarta linha" ou G é verdadeiro na "sexta linha" ou ...». A expressão "quarta linha" pode ser traduzida por «A é falso e B é verdadeiro e C é verdadeiro». O resultado final é:

A	B	C	G	P
0	**0**	**0**	**0**	**1**
0	**0**	**1**	**0**	**1**
0	**1**	**0**	**0**	**1**
0	**1**	**1**	**1**	**0**
1	**0**	**0**	**0**	**1**
1	**0**	**1**	**1**	**0**
1	**1**	**0**	**1**	**0**
1	**1**	**1**	**1**	**0**

$$G = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

e, para a função P,

$$P = \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C}$$

Uma vez obtida uma expressão algébrica pode montar-se o correspondente circuito. No entanto há vantagem em procurar simplificar a expressão antes de a implementar. Vejamos um caminho possível de simplificação:

$G = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$	
$G = (\overline{A}B + A\overline{B})C + AB(\overline{C} + C)$	distribut. de $\bullet$ em relaç. a +
$G = (\overline{A}B + A\overline{B})C + AB$	$\overline{C} + C = 1$; elemento neutro de $\bullet$
$G = (\overline{A}B + A\overline{B})C + AB(C + 1)$	elemento neutro de +
$G = (\overline{A}B + A\overline{B} + AB)C + AB$	distribut. de $\bullet$. em relaç. a +; associat. de +
$G = (\overline{A}B + A(\overline{B} + B))C + AB$	distribut. de $\bullet$ em relaç. a +
$G = (\overline{A}B + A)C + AB$	$\overline{B} + B = 1$; elemento neutro de $\bullet$
$G = ((\overline{A} + A)(B + A))C + AB$	distribut. de + em relaç. a $\bullet$
$G = (A + B)C + AB$	$\overline{A} + A = 1$; elemento neutro de $\bullet$
$G = AC + BC + AB = \overline{\overline{AC} \cdot \overline{BC} \cdot \overline{AB}}$	

Por um processo análogo obteríamos:

$$P = \overline{A}\,\overline{C} + \overline{B}\,\overline{C} + \overline{A}\,\overline{B} = \overline{A + C} + \overline{B + C} + \overline{A + B}$$

Embora haja regras que ajudam a escolher os passos a dar para simplificar uma expressão, nem sempre é fácil aplicá-las quando se tem pouco treino.

De utilização mais intuitiva são os mapas de Karnaugh. Estes podem ser construídos a partir da tabela de verdade e têm a forma de uma matriz, escrevendo-se em cada célula, que corresponde a uma combinação de estados das entradas, o valor lógico da função. Uma condição fundamental na construção do mapa de Karnaugh é que entre duas colunas ou duas linhas adjacentes apenas uma variável de entrada mude de estado. Nos mapas apresentados a seguir, entre a primeira e a segunda coluna apenas a variável C muda de estado, entre a segunda e a terceira colunas apenas a B, etc.

A \ BC	00	01	11	10
0	0	0	1	0
1	0	1	1	1

Função G

A \ BC	00	01	11	10
0	1	1	0	1
1	1	0	0	0

Função P

Para se entender a utilização dos mapas de Karnaugh para a simplificação de funções consideremos as duas células seleccionadas por uma linha a carregado. Estas células podem ser identificadas por $A\overline{B}C$ e ABC e o conjunto das duas por $A\overline{B}C + ABC = AC$. Esta simplificação, que com um pouco de treino pode ser escrita directamente, verifica-se sempre que se selecciona um conjunto de células em forma de rectângulo de lados 1, 2, 4, etc. (potências inteiras de dois). Para obter a função G juntamos os três rectângulos desenhados: $G = AC + BC + AB$. Eficiente, não é verdade?

Convém comparar este método com o anterior. A aparente vantagem do mapa de Karnaugh advém da geometria nos indicar o caminho a seguir: o mesmo caminho pode ser seguido algebricamente! Neste caso o «truque» é incluir três vezes a célula ABC. Exemplifiquemos:

$$G = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

$$G = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC + ABC + ABC \qquad \text{idempotência}$$

$$G = (\overline{A}BC + ABC) + (A\overline{B}C + ABC) + (AB\overline{C} + ABC)$$

$$G = BC + AC + AB$$

Agora foi fácil!

Notas: O recurso ao mapa de Karnaugh para simplificar funções é prático desde de que o número de variáveis independentes não ultrapasse quatro (matriz de quatro por quatro). A partir disso as regras são mais complicadas.

Se uma determinada combinação de estados lógicos das variáveis de entrada não poder ocorrer na prática, o valor lógico da função para essa combinação pode ser tomado como **0** ou como **1** consoante der mais jeito para maximizar os "rectângulos".

No caso de se pretender gerar duas ou mais funções a partir das mesmas variáveis convém reparar se há "partes comuns" para não as repetir. No caso da função P acima referida, por exemplo, embora ela possa *ser* gerada directamente a partir das variáveis A, B e C, é vantajoso gerá-la a partir de G: $P = \overline{G}$.

Desafio (1 hora)

Apresentam-se em seguida várias propostas de projectos de circuitos que o aluno pode realizar com os conhecimentos adquiridos. O aluno deverá escolher um dos projectos e realizá-lo no decorrer da aula.

"Majority voter" de 4 entradas

À semelhança do exemplo guiado, projecte e monte um circuito que, a partir de quatro funções de entrada, *A*, B, C e D, gere as seguintes três funções de saída:

G - **1** se e só se houver mais entradas em **1** do que em **0**

P - **1** se e só se houver mais entradas em **0** do que em **1**

E - **1** se e só se o número de entradas em **1** for igual ao de entradas em **0**

Circuito anti-roubo com código de acesso

Projecte e monte um circuito de oito entradas e uma saída. Quatro entradas funcionarão como parâmetros e permitirão guardar e alterar o "segredo". As outras quatro entradas são as variáveis que o utilizador terá de fornecer para ter acesso. A variável de saída é **1** quando as quatro variáveis (código introduzido) forem iguais aos quatro parâmetros correspondentes (código previamente memorizado). Ligue a saída a um LED: ele acenderá sempre que se introduzir o código de acesso correcto.

Gerador de bit de paridade

Projecte e monte um circuito de quatro entradas e uma saída. A saída deve ser **1** se houver um número ímpar de entradas em **1** e deve ser **0** no caso contrário. Este circuito chama-se gerador de paridade par (há sempre um número par de estados **1** entre as cinco variáveis: as quatro entradas mais a saída).

Nota: Os bits de paridade são usados em muitas circunstâncias (p.ex. na RAM dos computadores IBM compatíveis) para detectar a ocorrência de erros.

Contador de entradas em 1

Projecte e monte um circuito de três entradas e duas saídas. As duas saídas devem representar em binário o número de entradas em **1**.

Controlador de semáforos

Projecte e monte um circuito de duas entradas, representando a situação dos semáforos para os automóveis num cruzamento simples (**1**-verde; **0**-vermeJho), e de três saídas, duas delas controlam a passagem dos peões e a terceira acende um LED se se verificar uma situação de incompatibilidade no estado dos semáforos (por exemplo semáforo verde para os automóveis vindos de todas as direcções).

Projecto livre

Projecte e monte um circuito proposto por si. Não inicie o trabalho sem o conhecimento do professor, que avaliará a exequibilidade do projecto.