

**Uni Evangélica- Centro Universitário de Anápolis**

**Prof.: Alexandre Tannus**  
**Disciplina: Circuitos Digitais**

## **RELATÓRIO**

### **Etapa 3 – “ Somador + Subtractor ”**

**Acadêmicos: Charlley Junior Jabbar**  
**Mateus Correia Bezzan**

## Sumário

1. INTRODUÇÃO .....	3
2. FUNDAMENTAÇÃO TEÓRICA .....	4
3. Desenvolvimento do Projeto .....	7
CONSIDERAÇÕES FINAIS .....	12
REFERENCIAS.....	13

## 1. INTRODUÇÃO

Os circuitos digitais através dos diversos avanços tecnológicos atingiram o seu auge durante a “**era eletrônica**”, onde todas as soluções eram atingidas por meio de sistemas analógicos, também conhecidos como sistemas lineares. (IDOETA & CAPUANO).

Tudo começou através dos estudos da obra intitulada *An Investigation of the Laws of Thought*, criada pelo matemático inglês **George Boole** ( 1815-1864 ) , obra esta que apresentava um sistema matemático de análise lógica conhecido como **Álgebra de Boole**. (IDOETA & CAPUANO)

Apenas em 1938, o engenheiro americano **Claude Elwood Shannon** utilizou as teorias da **álgebra de boole** para a solução de problemas de circuitos de telefonia com relés, tendo publicado um trabalho denominado *Symbolic Analysis of Relay and Switching* (IDOETA & CAPUANO), pragmaticamente incluindo na área da tecnologia o campo da eletrônica digital para o mundo.

Os estudos de Boole foram passados por diversos outros estudiosos que no final os postulados de Boole deram origem às principais **funções lógicas** sendo as variáveis e as expressões envolvidas uma derivação da **álgebra de Boole** denominada como **Booleanas**, afirma Idoeta como as funções lógicas e, ou, não , ne e nou acabaram vindo para a realidade da sociedade. Funções lógicas estas que são encontradas apenas em 2 estados distintos:

➔ **O estado 0 (zero) e**

➔ **O estado 1 (um)**

O estado **0** representará o **não, falso**, chave desligada, interruptor desligado, ausência de tensão, aparelho desligado, em outras palavras significa a ausência ou a negação de algo.

O estado **1** representa **sim, verdadeiro**, chave ligada, presença de tensão, chave ligada, de forma sucinta o **1** é o contrário do **0** , onde tudo está presente e se resume à um mero **sim ou verdadeiro**.

O fator crucial para a identificação dos objetos no mundo real é a imagem, ou seja o que reflete a luz e é reproduzida no nossos cérebros afim da criação de um julgamento arbitrário do resultado.

## 2. FUNDAMENTAÇÃO TEÓRICA

### a) Álgebra de Boole

As variáveis de Boole podem ser representadas por letras, estas que estão sujeitas à assumir valores entre 0 e 1.

De acordo com Idoeta e Capuano (pág. 89) , a expressão de Boole consiste na representação de uma sentença matemática composta por termos que assumem variáveis booleanas e os resultados podem ser entre 0 e 1.

### b) Teorema de Morgan

O teorema de Morgan representa um fator que ocorre com as correlações entre os postulados onde o complemento do produto é igual à soma dos complementos:

$$(\overline{A \cdot B}) = \overline{A} + \overline{B}$$

Consequentemente temos então que da mesma forma que o produto dos complementos reagem a soma dos complementos resultará no produto de cada complemento:

$$(\overline{A + B}) = \overline{A} \cdot \overline{B}$$

### c) Simplificação de circuitos lógicos variáveis e Expressões na Álgebra de Boole

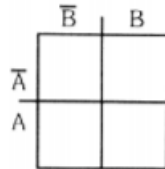
Com os conceitos dos postulados de **Boole** e as suas expressões booleanas, somos capazes de realizar simplificações das expressões consequentemente os circuitos digitais são reduzidos uma vez que cada redução reduz a quantidade de componentes necessários para realizar a mesma função.

De acordo com Idoeta e Capuano , ambos afirmam que há 2 meios para a realização da simplificação uma onde a pessoa realiza o uso da Álgebra de **Boole**, o segundo meio é por meio de uma técnica “ gráfica” que são os mapas de **Veitch- Karnaugh**, como será representado no tópico de desenvolvimento do projeto de um contador hexadecimal .

#### d) Mapas de Veitch – Karnaugh – Simplificando expressões lógicas.

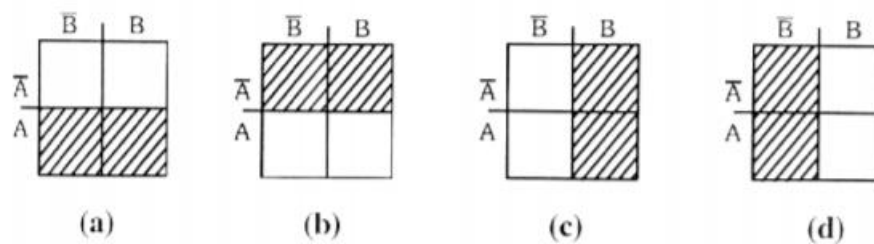
O mapa de Karnaugh é representado de acordo com a necessidade das entradas, o mesmo é viável até atingir o valor de 4 variáveis de entradas a partir disso o mesmo passa a ser muito complexo e de difícil compreensão.

Observa-se a figura 1.0 , representando 2 variáveis:



**Figura 1.0 Mapa de Karnaugh de duas variáveis.**

na figura 1.4 representamos todas as regiões que são possíveis ser observadas no mapa, as mesmas varrem as diversas possibilidades de combinações entre as duas variáveis..



**Figura 1.1 Regiões do Mapa de Karnaugh**

(a) Região onde  $A = 1$ ;

(b) Região onde  $A = 0$  ( $\bar{A} = 1$ );

(c) Região onde  $B = 1$ ;

(d) Região onde  $B = 0$  ( $\bar{B} = 1$ ).

Portanto com 2 variáveis somos capazes de obter 4 combinações, estas que são as influenciadores das futuras expressões extraídas:

A	B
0	0
0	1
1	0
1	1

**Tabela 1.0 Regiões cobertas pelo mapa de duas entradas**

Porém a versão que foi usada para o desenvolvimento foi uma adaptação deste mapa diretamente com o código binário onde facilita a análise dos mapas resultantes e preenchimento por meio de uma tabela verdade conforme a figura de um mapa criado por meio de 4 entradas :

A	CD	00	01	11	10
AB					
00					
01					
11					
10					

**Figura 1.2 Mapa Adaptado de Karnaugh**

O mapa adaptado segue o funcionamento de acordo com as saídas resultantes da tabela verdade , no caso da figura 1.5 os dados que iriam preencher as linhas e colunas do mesmo são as saídas de “A” .

#### e) Circuitos Aritméticos

De acordo com Idoeta & Capuano os circuitos Aritméticos estão dentro de um conjunto de circuitos combinacionais aplicados para a finalidade específica nos sistemas digitais, destacam-se os circuitos aritméticos.

Apesar de serem os principais a para o desenvolvimento de construções de unidades lógicas aritméticas ULA, para os microprocessadores e, ainda, encontrados disponíveis em circuitos integrados comerciais (IDOETA & CAPUANO, 2009)

## 1. Somador

O somador possibilita efetuar a soma de números binários com 1 algarismo. Para se fazer a soma de números binários de mais algarismos, esse circuito torna-se insuficiente, pois não possibilita a introdução do transporte de entrada proveniente da coluna anterior relata Capuano.

## 2. Subtractor

O subtrator possibilita-nos efetuar a subtração de números binários de 1 algarismo. Para se fazer uma subtração com números binários de 1 algarismo. Para se fazer uma subtração com números de mais algarismos, este circuito torna-se insuficiente, pois não possibilita a entrada do transporte (Te), proveniente da coluna anterior.

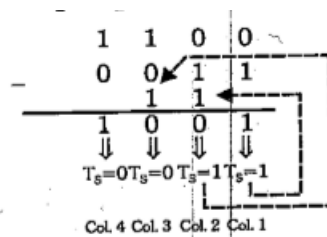


Figura 1.3 Subtração Binária

## 3. Desenvolvimento do Projeto

O projeto consiste em recriar uma calculadora onde a mesma realizará cálculos de Soma e Subtração, apresentando os números selecionados para calcular de 0 à 9 além de apresentar o valor resultante no final a qual as mesmas deveriam estar implementadas no **Arduino** (hardware) que aceita a linguagem de programação C para micro controladores. Circuito este implementado que se chama sketch que apresentará obrigatoriamente duas funções: **setup()** e **loop()**.

O projeto está dividido em 3 etapas onde serão apresentadas os desafios a serem trilhados na implementação do microcontrolador fazendo uso tanto do conhecimento em *hardware* quanto de *firmware*.

O projeto foi desenvolvido após a realização das tabelas verdade e os mapas de *Veitch-Karnaugh* onde foi desenvolvidas as saídas das entradas fornecidas por um interruptor binário de 4 *switchs* que futuramente serão influenciados por um

interruptor(DIP DPST) que ditará a operação realizada entre “Soma” ou “Subtração”,

Tabela Verdade produzida pelo projeto :

M1	A	B	Te	S	Ts	
0	0	0	0	0	0	Soma Completa (M1 = 0)
0	0	0	1	1	0	
0	0	1	0	1	0	
0	0	1	1	0	1	
0	1	0	0	1	0	
0	1	0	1	0	1	
0	1	1	0	0	1	
0	1	1	1	1	1	
1	0	0	0	0	0	Subtractor Completo (M1 = 1)
1	0	0	1	1	1	
1	0	1	0	1	1	
1	0	1	1	0	1	
1	1	0	0	1	0	
1	1	0	1	0	0	
1	1	1	0	0	0	
1	1	1	1	1	1	

Tabela 1.1 Tabela Verdade do Projeto do Somador/Subtractor de números

Após o desenvolvimento da tabela verdade conforme, o data sheet do display de led a seguir , teremos então a representação das saídas da tabela verdade no mapa de Karnaugh:

(saída) S	00	01	11	10
0	0	1	0	1
1	1	0	1	0

Tabela 1.2 Mapa de Karnaugh da saída “S” de Soma

Após a análise do mapa de Karnaugh temos então uma expressão matemática que será justamente a saída lógica do código, teremos um caso especial.

Expressão Matemática :

$$S = A \oplus B \oplus Te \text{ ( Para a saída do caso de Soma)}$$

(saída) S	00	01	11	10
0	0	1	0	1
1	1	0	1	0

Tabela 1.3 Mapa de Karnaugh da saída “S” de Subtração



Expressão Matemática :

$$S = A \oplus B \oplus Te \text{ ( Para a saída do caso de Soma)}$$

(saída) Ts	00	01	11	10
00	0	0	1	0
01	0	1	1	1
11	0	0	1	0
10	0	1	1	1

Tabela 1.4 Mapa de Karnaugh da saída “Ts” de Soma

Expressão Matemática :

$$Ts = BTe + \overline{M}1Ate + \overline{M}1AB + M1\overline{A}te + M1\overline{A}B$$

Foi possível observar que das **2 saídas S e Ts** , ambas segue um mesmo padrão tanto nas tabelas de Soma completa quanto nas tabelas de Subtração completa, sendo resultante das 2 tabelas temos a expressão algébrica da simplificação das mesmas :

$$\text{Para as saídas S : } S = (A \oplus B \oplus Te) + (A \oplus B \oplus Te)$$

Ou seja:

$$S = A \oplus B \oplus Te$$

Para as saídas Ts:

$$Ts = BTe + \overline{M}1Ate + \overline{M}1AB + M1\overline{A}te + M1\overline{A}B$$

O processo de simplificação da mesma é :

$$Ts = BTe + \overline{M}1Ate + \overline{M}1AB + M1\overline{A}te + M1\overline{A}B$$




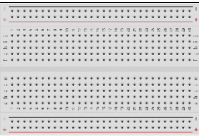
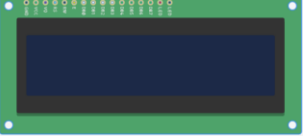


$$Ts = BTe + Te(\overline{M}1A + M1\overline{A}) + B(\overline{M}1A + M1\overline{A})$$

$$Ts = BTe + [Te(M \oplus A)] + [B(M \oplus A)]$$

$$Ts = BTe + (M \oplus A) \cdot (B + Te)$$

Dessa forma somos capazes identificar as duas saídas e como o circuito deverá agir logicamente.

Os materiais utilizados para o desenvolvimento do sketch – etapa 3 foram de acordo com a **Tabela 1.6** :

Material	Imagem
<b>1 x Arduino</b>	
<b>2 x Interruptor</b>	
<b>10 x Resistores</b>	
<b>1 x Placa de ensaio <u>pequena</u></b>	
<b>1 x LCD Display 16x2</b>	
<b>1 x Potenciômetro</b>	
<b>1 x Interruptor DIP DPST</b>	

**Tabela 1.5 Materiais para construção do projeto.**

#### a) Bibliotecas

Afim de programar o Display LCD 16x2, será utilizada uma biblioteca pré instalada no Arduino. Conhecida como *LiquidCrystal*, que é responsável em apresentar diversas funções que serão usada para a apresentação dos valores de sofreram a soma e os valores que realizará a subtração.

Segue uma tabela com as funções utilizadas da biblioteca e suas respectivas ações:

Begin (int coluna, int linha)	Define a dimensão da tela
-------------------------------	---------------------------

Clear()	Limpa a tela
SetCursor (int coluna, int linha)	Define a posição do cursor para a próxima escrita
print(data)	Imprime um dado no display. O dado pode ser texto, número ou caractere criado.

**Tabela 1.6 Funções da Biblioteca LiquidCrystal.h**

**b) Funções / procedimentos utilizados no Código fonte para a criação das portas lógicas.**

**Void setup()** // momento no qual são definidas as portas de entrada e saída de dados.

**Void loop()** // serve para realizar as repetições das variáveis contadoras e as variáveis de entrada para manter o sistema em funcionamento.

**Processo void bool (Com lógicas)”** // sendo responsáveis em realizar os procedimentos lógicos resultantes dos mapas de Karnaugh.

**c) Funções específicas do Arduino:**

**digitalWrite(segA,);** // Enviar dado de saída para a porta 0 do microcontrolador acendendo a LCD como resultado de envio.

**digitalRead(botA);** // 1 Realiza a leitura da entrada no interruptor de valor 1 , aderindo valores de 0 ou 1 para o mesmo.

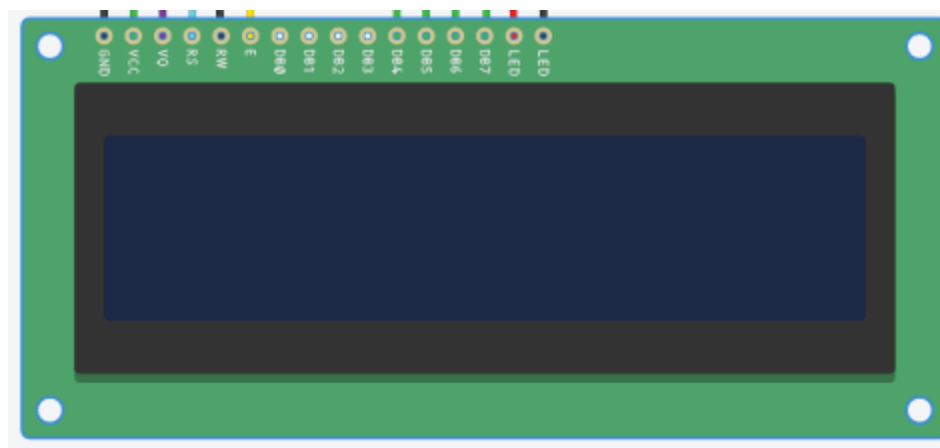
**d) Display LCD 16x2 Arduino**

Este foi o *template* utilizado para configurar o *display* de *lcd* , segue de acordo com a tabela 1.3 , a representação de como funciona as funções e onde ligar as partes do mesmo:

Pino	Nome	Função	Conexão no circuito
1	<b>Vss</b>	Terra	GND
2	<b>Vdd</b>	Alimentação positiva (5V)	5V
3	<b>Vo</b>	Contraste do LCD	Pino central do potenciômetro
4	<b>Rs</b>	Register Select – Define se o dado enviado ao LCD é um comando (nível lógico 0) ou um caractere (nível lógico 1)	Pino 6

5	<b>R/W</b>	Read/Write – Define se será realizada leitura (nível lógico 1) ou escrita (nível lógico 0) de dados.	<b>GND</b>
6	<b>E</b>	Enable – Habilita recepção de sinal do LCD	Pino 4
7-10	<b>D0-D3</b>	Dados – não utilizado em modo 4 bits	Desconectado
11-14	<b>D4-D7</b>	Dados	Pinos 3,2, 1, 0
15	<b>A (LED + )</b>	Anodo da backlight. Deve ser ligado à alimentação positiva com um resistor em série	5V – com resistor 220 $\Omega$ em série
16	<b>K (LED - )</b>	Catodo da backlight. Deve ser ligado ao terra	<b>GND</b>

**Tabela 1.7** Posições do Display de LCD no Arduino e onde conectado foi os mesmos.



**Figura 1.4** *Display Sheet* Arduino.

## CONSIDERAÇÕES FINAIS

O recurso tem por objetivo, através do arduino esclarecer os contextos desejados. Sua presença tem desenvolvido na melhoria e automatização de processos , fazendo com que a tecnologia seja usada ao favor da humanidade. O Arduino em junção com as funções lógicas no mundo real é a representação da ascensão do progresso onde o mesmo é capaz de atingir a verossimilhança dos processos que realizamos manualmente em nossa sociedade. Em suma, o Arduino e as portas lógicas nos microcontroladores é apropriado para realizações de trabalhos simples que são realizados manualmente porém de forma automatizada e com maior confiabilidade.

### **Futuros trabalhos**

Com o decorrer do projeto foi possível observar a amplitude na qual o Arduíno é capaz de atender a sociedade humana. Tanto no meio físico quanto virtual, o projeto foi finalizado com o desenvolvimento de um circuito capaz de calcular operações como soma e subtração, em outros contextos também é viável a realização de cálculos como multiplicação e divisão. O uso do display de lcd para futuros projetos como atendimento em consultórios ou até mesmo para controle de filas. O mesmo pode ser implementado para atribuir em métodos ágeis por outro lado com um uso em específico para complementar a eficácia dos métodos ou até mesmo auxiliar.

### **REFERENCIAS**

Idoeta, I.V. & Capuano, F.G.; **Elementos de Eletrônica Digital**, 12ª. edição, Érica, 1987.

Mendelson, E. ; **Álgebra booleana e circuitos de chaveamento**, McGraw-Hill, 1977.

Karnaugh, M. "**The Map Method for Synthesis of Combinational Logic Circuits**". **Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics**. 1953.