**Centro Universitário de Anápolis - UniEvangélica**

Ana Caroliny Amancio Veiga

Lucas Galvão Lima

Mauricio Luan

**Relatório de implementação de um display**

Profº Me. Alexandre Tannus

Anápolis Agosto - 2018

# Introdução

Este relatório apresentará uma explicação breve de como foi feito e qual lógica foi usada para desenvolvimento do trabalho com base nos conceitos aprendidos em sala de aula e nas disciplinas de programação, redes de computadores, circuitos digitais e lógica de programação. A princípio, o objetivo deste trabalho é desenvolver o raciocínio lógico e sistêmico tendo como resultado final uma calculadora que faça o cálculos, conversões de números decimais em binários para obter uma saída desejável.

# Materiais usados

Primeiramente foi feito uso da ferramenta online Autodesk Tinkercard® para montagem do circuito por ausência de um Arduino físico.

O projeto usou os seguintes componentes:

|  |  |  |
| --- | --- | --- |
| **Nome** | **Quantidade** | **Componente** |
| U1 | 1 | Arduino Uno R3 |
| U2 | 1 | LCD 16 x 2 |
| Rpot1 | 1 | 250 kOhm, Potenciometro (Potentiometer) |
| R1 | 1 | 220 ohms Resistor |
| R2, R3, R4, R5, R6, R7, R8, R9, R10, R11 | 9 | 1 ohm Resistor |
| SW1, SW2 | 2 | DIP Switch SPST x 4 |
| SW3 | 1 | DIP Switch DPST |
| Breadboardwire | 38 | Fios nas cores: black, White, blue, red, Orange, yellow, green, pink, grey, purple, turquoise. |

# Funcionamento

A princípio, foi desenvolvido o circuito de modo que fosse possível mostrar uma melhor visualização dos componentes. A lógica usada no código é baseado na linguagem C (logo, que a linguagem do Arduino é uma linguagem C-like), usamos criação do *setup* para definir as portas a serem usadas no circuito conforme mostra o código abaixo:

//inclusão das bibliotecas

#include <LiquidCrystal.h>

#include <math.h>

#include <string.h>

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

#define Dip1D 6

#define Dip2D 7

#define Dip3D 8

#define Dip4D 9

#define Dip1E A1

#define Dip2E 1

#define Dip3E 10

#define Dip4E 13

#define Dip0 A0

int Cin = 0;

int Cout = 0;

int Cout4 = 0;

int Cout3 = 0;

int Cout2 = 0;

int Cout1 = 0;

int saida4 = 0;

int saida3 = 0;

int saida2 = 0;

int saida1 = 0;

int saida = 0;

void setup() {

pinMode(Dip1D, INPUT\_PULLUP);

pinMode(Dip2D, INPUT\_PULLUP);

pinMode(Dip3D, INPUT\_PULLUP);

pinMode(Dip4D, INPUT\_PULLUP);

pinMode(Dip1E, INPUT\_PULLUP);

pinMode(Dip2E, INPUT\_PULLUP);

pinMode(Dip3E, INPUT\_PULLUP);

pinMode(Dip4E, INPUT\_PULLUP);

pinMode(Dip0, INPUT\_PULLUP);

// set up the LCD's number of columns and rows:

lcd.begin(16, 2);

}

//Função criada para transformar decimal em binário

int funcaoDecimal(int binario){

int decimal=0,d=1;

do{

decimal = decimal+(binario%10)\*d;

d = d\*2;

binario = binario/10;

}while(binario!=0);

return decimal;

}

void loop() {

int Dip1EStatus = digitalRead(Dip1E);

int Dip2EStatus = digitalRead(Dip2E);

int Dip3EStatus = digitalRead(Dip3E);

int Dip4EStatus = digitalRead(Dip4E);

int Dip0Status = digitalRead(Dip0);

int Dip1DStatus = digitalRead(Dip1D);

int Dip2DStatus = digitalRead(Dip2D);

int Dip3DStatus = digitalRead(Dip3D);

int Dip4DStatus = digitalRead(Dip4D);

//Saída

saida4 = ((Dip4EStatus && !Dip4DStatus && !Cin) + (!Dip4EStatus && !Dip4DStatus && Cin) + (Dip4EStatus && Dip4DStatus && Cin) + (!Dip4EStatus && Dip4DStatus && !Cin));

Cout4 = ((Dip4DStatus && Cin)||(!Dip0Status && Dip4EStatus && Dip4DStatus)||(!Dip0Status && Dip4EStatus && Cin)||(Dip0Status && !Dip4EStatus && Dip4DStatus)||(Dip0Status && !Dip4EStatus && Cin));

//Saída

saida3 = ((Dip3EStatus && !Dip3DStatus && !Cout4) + (!Dip3EStatus && !Dip3DStatus && Cout4) + (Dip3EStatus && Dip3DStatus && Cout4) + (!Dip3EStatus && Dip3DStatus && !Cout4));

Cout3 = ((Dip3DStatus && Cout4)||(!Dip0Status && Dip3EStatus && Dip3DStatus)||(!Dip0Status && Dip3EStatus && Cout4)||(Dip0Status && !Dip3EStatus && Dip3DStatus)||(Dip0Status && !Dip3EStatus && Cout4));

//Saída

saida2 = ((Dip2EStatus && !Dip2DStatus && !Cout3) + (!Dip2EStatus && !Dip2DStatus && Cout3) + (Dip2EStatus && Dip2DStatus && Cout3) + (!Dip2EStatus && Dip2DStatus && !Cout3));

Cout2 = ((Dip2DStatus && Cout3)||(!Dip0Status && Dip2EStatus && Dip2DStatus)||(!Dip0Status && Dip2EStatus && Cout3)||(Dip0Status && !Dip2EStatus && Dip2DStatus)||(Dip0Status && !Dip2EStatus && Cout3));

//Saída

saida1 = ((Dip1EStatus && !Dip1DStatus && !Cout2) + (!Dip1EStatus && !Dip1DStatus && Cout2) + (Dip1EStatus && Dip1DStatus && Cout2) + (!Dip1EStatus && Dip1DStatus && !Cout2));

Cout1 = ((Dip1DStatus && Cout2)||(!Dip0Status && Dip1EStatus && Dip1DStatus)||(!Dip0Status && Dip1EStatus && Cout2)||(Dip0Status && !Dip1EStatus && Dip1DStatus)||(Dip0Status && !Dip1EStatus && Cout2));

//Saída

saida = ((!Dip1EStatus && Dip1DStatus && !Cout1 && !Dip0Status) + (Dip1EStatus && !Dip1DStatus && !Cout1 && !Dip0Status) + (!Dip1EStatus && !Dip1DStatus && !Cout1 && Dip0Status) + (Dip1EStatus && Dip1DStatus && !Cout1 && Dip0Status) + (!Dip1EStatus && Dip1DStatus && Cout1 && Dip0Status) + (Dip1EStatus && !Dip1DStatus && Cout1 && Dip0Status) + (Dip1EStatus && Dip1DStatus && Cout1 && !Dip0Status) + (!Dip1EStatus && !Dip1DStatus && Cout1 && !Dip0Status));

Cout = ((Dip1DStatus && Cout1)||(!Dip0Status && Dip1EStatus && Dip1DStatus)||(!Dip0Status && Dip1EStatus && Cout1)||(Dip0Status && !Dip1EStatus && Dip1DStatus)||(Dip0Status && !Dip1EStatus && Cout1));

//Executa a conversão de dip para um char

char Dip1EStatusS[5]; sprintf(Dip1EStatusS, "%i",Dip1EStatus);

char Dip2EStatusS[5]; sprintf(Dip2EStatusS, "%i",Dip2EStatus);

char Dip3EStatusS[5]; sprintf(Dip3EStatusS, "%i",Dip3EStatus);

char Dip4EStatusS[5]; sprintf(Dip4EStatusS, "%i",Dip4EStatus);

//Faz uma conversão para uma string auxiliar

char auxiliarDipA[256];

sprintf(auxiliarDipA, "%s%s%s%s", Dip1EStatusS, Dip2EStatusS, Dip3EStatusS,Dip4EStatusS);

//Faz a conversão para inteiro

int DipA = atoi(auxiliarDipA);

//Converte dipB para um char

char Dip1DStatusS[5]; sprintf(Dip1DStatusS, "%i",Dip1DStatus);

char Dip2DStatusS[5]; sprintf(Dip2DStatusS, "%i",Dip2DStatus);

char Dip3DStatusS[5]; sprintf(Dip3DStatusS, "%i",Dip3DStatus);

char Dip4DStatusS[5]; sprintf(Dip4DStatusS, "%i",Dip4DStatus);

//Conversão da entrada dipB para uma string auxiliarDipB

char auxiliarDipB[256];

sprintf(auxiliarDipB, "%s%s%s%s", Dip1DStatusS, Dip2DStatusS, Dip3DStatusS,Dip4DStatusS);

//converte um string para inteiro

int DipB = atoi(auxiliarDipB);

//Conversão das saídas para char

char CoutS[5]; sprintf(CoutS, "%i",Cout);

char saida1S[5]; sprintf(saida1S, "%i",saida1);

char saida2S[5]; sprintf(saida2S, "%i",saida2);

char saida3S[5]; sprintf(saida3S, "%i",saida3);

char saida4S[5]; sprintf(saida4S, "%i",saida4);

//Conversão das saídas para char

char auxiliar[256];

sprintf(auxiliar, "%s%s%s%s%s", CoutS, saida1S, saida2S,saida3S,saida4S);

//Faz a conversão de um string para inteiro

int resultado = atoi(auxiliar);

//Imprime no display

lcd.setCursor(0, 0);

lcd.print(funcaoDecimal(DipA));

//muda o estado da chave. Se for - vira mais e vice versa

if(Dip0Status == 0)

{

lcd.print(" + ");

} else if(Dip0Status == 1){

lcd.print(" - ");

}

lcd.print(funcaoDecimal(DipB));

lcd.setCursor(0, 1);

//clear no display

lcd.print(funcaoDecimal(resultado));

delay(3000);

lcd.clear();

delay(500);

}

**O circuito lógico**

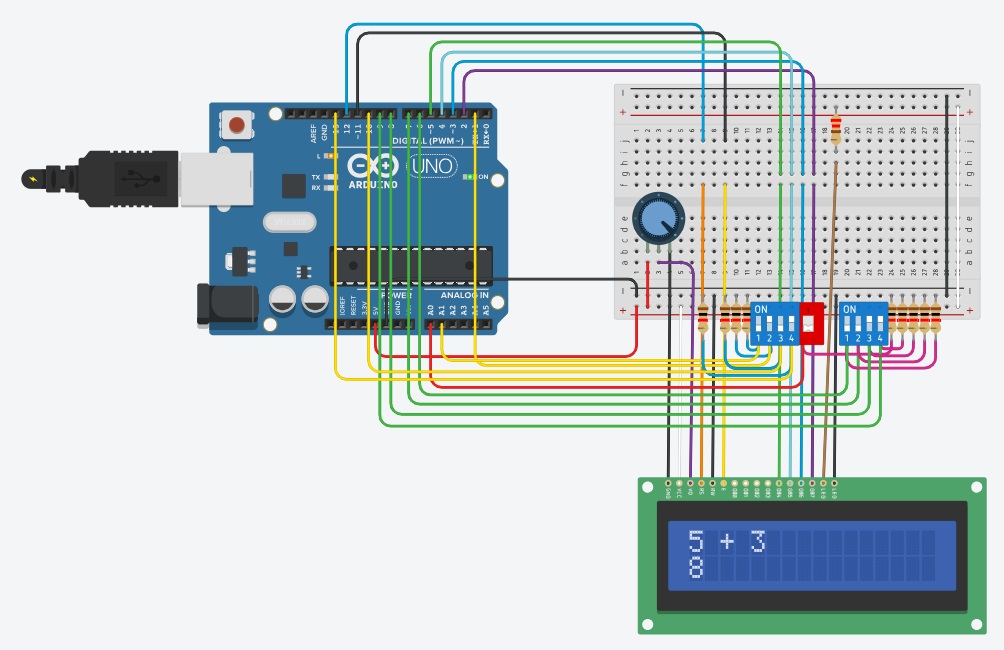


Figura 1: circuito lógico executando a lógica de implementação de adição.

Fonte: os autores.

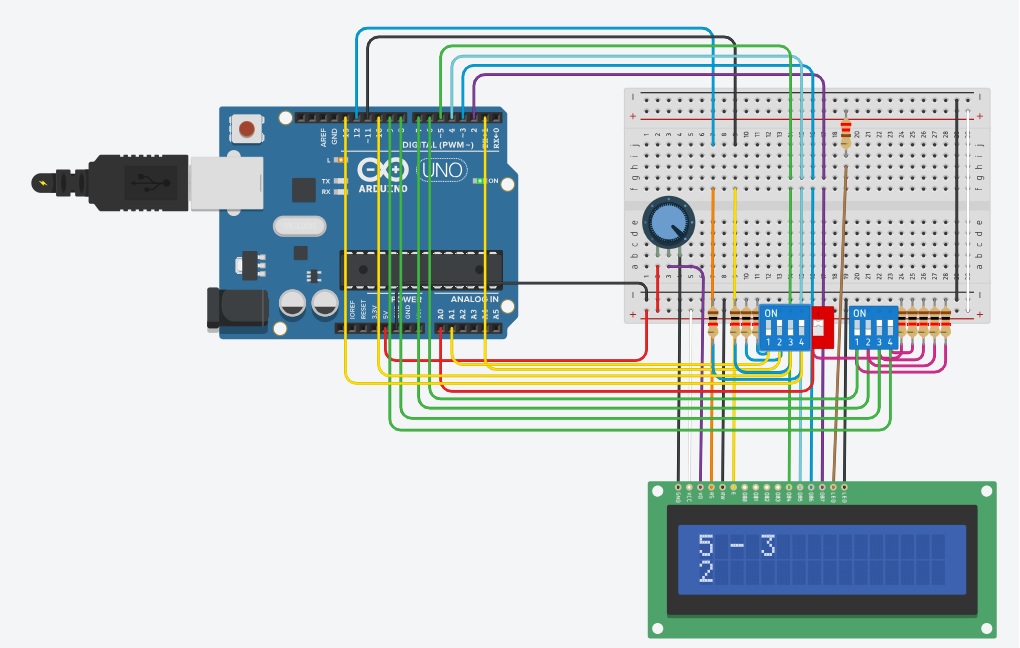


Figura s: Circuito lógico executando a lógica de implementação de subtração.

Fonte: os autores.

**Conclusão**

Neste trabalho, concluímos que a conversação de hardware e software é algo que demanda tempo, pesquisa e paciência. Aprendemos a melhorar a questão de ser autodidata, saber trabalhar em equipe foi crucial para que o trabalho fosse executado com êxito. Além de toda a transição no que tange o aprendizado de homem máquina este trabalho despertou-nos o interesse para o estudo profundo de Arduino e conversação dos seus componentes.

**Referências bibliográficas**

Disponível em: <https://www.arduino.cc/>. Acesso em: 15 de outubro de 2018.

Disponível em: <http://playground.arduino.cc/portugues/learning>. Acesso em: 27 de outubro de 2018.

Disponível em: <https://www.circuitar.com.br/tutoriais/programacao-para-arduino-primeiros-passos/>. Acesso em: 30 de outubro de2018.

Disponível em: <https://multilogica-shop.com/Referencia>. Acesso em: 2 de novembro de 2018.

Disponível em: <https://www.tinkercad.com>. Acesso em: 26 de novembro de 2018.

Disponível em: <https://www.filipeflop.com/forum/>. Acesso em: 10 de novembro de 2018.

Disponível em: <http://labdegaragem.com/profiles/blogs/tutorial-como-utilizar-lcd-shield>. Acesso em: 15 de novembro 2018.