

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ  
ESCOLA POLITÉCNICA  
CURSO DE ENGENHARIA DE COMPUTAÇÃO**

**BRUNO SCHEREMETA  
GUILHERME ALVES FERREIRA  
VINICIUS AGUETILDE PELICK**

**PROJETO WORLD CLOCK**

**CURITIBA  
2013**

**BRUNO SCHEREMETA  
GUILHERME ALVES FERREIRA  
VINICIUS AGUETILDE PELICK**

**PROJETO WORLD CLOCK**

Trabalho apresentado ao curso de Engenharia de Computação, da Pontifícia Universidade Católica do Paraná, como requisito parcial de avaliação das disciplinas de Física 3 e Resolução de Problemas em Engenharia 1.

Prof. Afonso Miguel

**CURITIBA  
2013**

## RESUMO

A partir de uma estrutura com uma capacidade de rotação, controlada por um pwm, um arduíno é acoplado a uma base redonda, que controlará sete led's fixados em uma placa que está presa na superfície de um cilindro oco verticalmente posicionado em relação a esta base redonda. Conforme a estrutura roda, os led's acendem periodicamente formando uma ilusão de ótica, a imagem de um relógio digital.

**Palavras-Chave:** Relógio, hélice.

## ABSTRACT

From a structure with an ability of rotation is controlled by a PWM, an Arduino is coupled to a round base, which control seven LEDs fixed on a plate which is trapped on surface of a hollow cylinder vertically positioned in relation to this round base . As the structure rotates, the LEDs light up periodically forming an optical illusion, the image of a digital clock.

**Keywords:** Clock, propeller.

## Sumário

<b>1. INTRODUÇÃO</b> .....	<b>5</b>
<b>2. OBJETIVO GERAL</b> .....	<b>5</b>
2.1.    Objetivos específicos .....	5
<b>3. PROJETO</b> .....	<b>5</b>
3.1.    Materiais utilizados .....	6
3.1.1.    World Clock .....	6
3.1.2.    Placa retificadora .....	6
<b>4. DESCRIÇÃO GERAL</b> .....	<b>7</b>
4.1.    Descrição do projeto .....	7
4.2.    Hardware .....	7
4.2.1.    PWM .....	7
4.2.2.    Placa retificadora .....	9
4.3.    Diagrama de Blocos .....	10
4.4.    Problemas e Soluções.....	11
<b>5. CÓDIGO FONTE</b> .....	<b>12</b>
5.1.    Resultado final .....	22
<b>6. CONCLUSÃO</b> .....	<b>23</b>
<b>7. REFERÊNCIAS</b> .....	<b>24</b>

## **1. INTRODUÇÃO**

O World Clock é uma variação de outros projetos já existentes, cujo nome original é “Propeller Clock”, “Relógio Hélice”. O motivo da escolha desta ideia para servir como base para o projeto, é justamente pelo fato de abranger uma parte do conteúdo que está sendo aplicado à turma do terceiro período de Engenharia de Computação, ou seja: movimento, programação (arduíno, C), circuito (arduíno, placa elétrica e componentes).

O projeto em si, não trás nenhuma novidade para o mundo, muito menos uma utilidade benéfica ao ser humano, porém este objeto poderia ser utilizado como um artefato de decoração, e além de um simples relógio digital o W.C., poderia exibir mensagens, de acordo com o que fosse programado em seu núcleo (arduíno).

## **2. Objetivo geral**

Fazer um relógio digital ser visualizado a partir de um conjunto de sete leds que acendem periodicamente, em uma estrutura cilíndrica posicionada verticalmente, que gira de acordo com a velocidade ajustada pelo pwm.

### **2.1. Objetivos específicos**

Para a conclusão do projeto, é necessário que diversas etapas sejam realizadas, por exemplo, a construção do PWM para controle do motor, o anel mecânico para os leds, confeccionar a placa retificadora e filtros e o receptor infravermelho para receber o sinal do controle.

## **3. PROJETO**

O projeto possui muitas informações, desde componentes, peças, materiais, em si o World Clock, possui toda uma descrição junto a sua história de desenvolvimento, esta parte da documentação irá referir-se exclusivamente a isto.

### **3.1. Materiais utilizados**

Para cada parte do projeto, um conjunto de materiais foram utilizados, este tópico irá separar os materiais para suas respectivas partes.

#### **3.1.1. Word Clock**

- Quatro rolamentos com 10mm de diâmetro externo e 3mm para o furo interno;
- Base de madeira prensada;
- Astes de alumínio moldados para fixação das demais partes;
- Parafusos auto atarrachantes para fixação das astes de alumínio na base;
- Anel cilíndrico de cano PVC;
- Motor DC 12v/1A;
- Peça de motor;
- Tubo de aço inox 4mm de diâmetro para proteger o eixo;
- Barra roscada de 3mm de diâmetro utilizado como eixo;
- Regulador de tensão 7812 para sincronização da imagem;
- Roscas;
- Arduíno ATmega1280.

#### **3.1.2. Placa retificadora**

- 7 leds auto brilho, 6 azul e 1 verde;
- Reguladores de tensão 7812 e 7805;
- Três baterias 3V de lítio recarregáveis;
- Resistor de 220  $\Omega$ ;
- Ponte retificadora;
- Placa universal para placa retificadora.

## **4. DESCRIÇÃO GERAL**

Todo projeto deve ter um histórico, desde a ideia que lhe deu origem, problemas encontrados juntos de suas resoluções no desenvolvimento, alterações feitas da ideia original do projeto até o resultado final de todo o esforço. Esta parte da documentação será utilizada para demonstrar todo desenvolvimento deste projeto.

### **4.1. História do Projeto**

Inicialmente, o nome World Clock, foi escolhido a partir de um projeto já existente chamado Propeller Clock (Relógio Hélice). O World Clock, tinha como objetivo inicial formar a imagem do globo terrestre, junto ao relógio digital a partir da ilusão de ótica formada pelos leds que piscam de forma organizada (controle realizado pelo arduíno que contém uma programação interna em linguagem C) em uma determinada frequência. Tinha-se em mente a utilização de setenta e dois leds para a formação da imagem digital e da imagem do globo. Após vários brain storms, e com o auxílio dos professores orientadores, foi optado por utilizar apenas sete leds somente para a formação da imagem do relógio, pois existem outros métodos para realizar o controle de tantas saídas ao mesmo tempo com um método mais avançado. Então esta ideia com os demais leds, foi postergada para um futuro projeto que este grupo pode aprimorar o World Clock..

### **4.2. Hardware**

O projeto teve duas partes de hardware que foram confeccionadas e feitas para suprir necessidades, o pwm e a placa retificadora.

#### **4.2.1. PWM**

O PWM – Pulse Width Modulation, ou seja, modulação por largura de pulso – foi uma solução simples para o controle de velocidade do motor. Inicialmente tinha-se a ideia de utilizar simplesmente um potenciômetro para regular as rpm do motor, porém com muita potência iria ser dissipada no potenciômetro, o que não é muito

viável. Sendo assim, surgiu a ideia, junto ao professor orientador, para que um PWM fosse desenvolvido afim de controlar a rotação do motor, a partir das larguras de pulsos que o CI NE555, envia de acordo com o ajuste que é no potenciômetro ligado ao circuito.

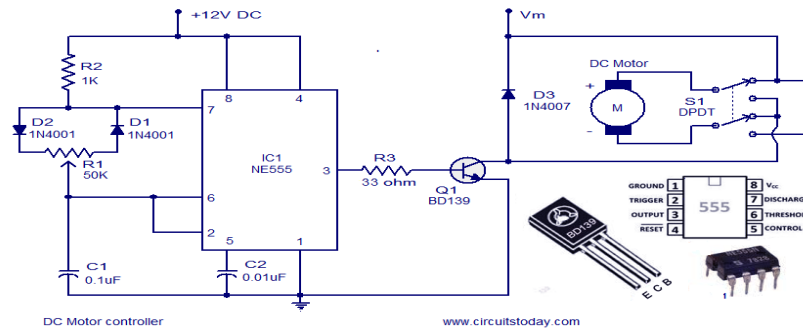


Fig. 1 – Circuito base para desenvolvimento do PWM.

A imagem acima representa o circuito base utilizado para o desenvolvimento do PWM utilizada no projeto. Algumas alterações foram feitas, para melhor adaptação as necessidades do World Clock. O potenciômetro R1 de 50kΩ foi substituído por um de 100kΩ, pois com este valor era possível manter os pulsos que chegam até o motor, de forma a manter o motor em sua velocidade máxima e também em estado de repouso. Outro componente trocado foi o resistor R3, que foi substituído por um valor de 2kΩ.



Fig. 2 – PWM / Fonte chaveada



Fig. 3 - PWM dentro de sua caixa, já finalizado.



#### 4.2.2. Placa retificadora



Fig. 4 - Placa retificadora

Depois de vários Brains Storms, e pesquisas, para conseguir fazer com que a alimentação chegasse à estrutura superior, um método foi encontrado. Os motores dc, possuem um parte na sua estrutura que está diretamente relacionado com os imãs, esta parte é chamada de parte coletora do motor.



Fig. 5 – Retirando a parte coletora do motor.

Foi recordada a base inferior do motor, para que estas partes essenciais fossem utilizados. Esta base inferior do motor possui duas micro escovas de carvão. O coletor do motor foi posicionado de forma que seus dois polos conforme girem, entre em contato com estas escovas que estão ligados em pequenas chapas de cobre, e possuem um alimentação proveniente de uma fonte chaveada de 18V, por sua vez, conforme a estrutura rotaciona, partes do coletor encostam nestes carvões, porém como esta parte fica em constante rotação é impossível que somente com esta estrutura a tensão de alimentação do arduino, mantenha-se constante sem que o seu sinal seja alternado. Para solucionar isso foi desenvolvido uma placa retificadora, para que o sinal de tensão seja constante, e não haja percas nem ruídos conforme a estrutura roda.



Fig. 5 – Peça de motor + Coletor.

### 4.3. Diagrama de Blocos

Este tópico é exclusivamente, para melhor entendimento das etapas de funcionamento do projeto.

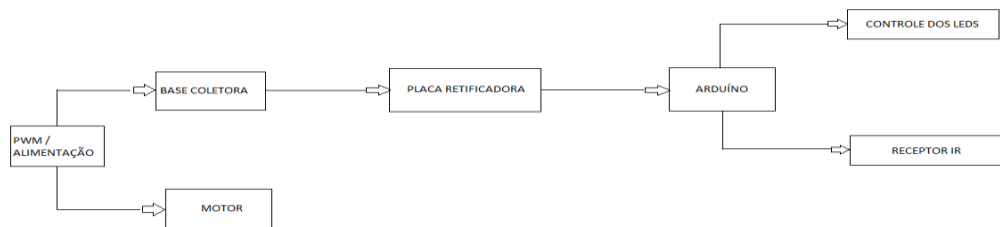


Fig. 4 – Diagrama de Blocos World Clock.

Resumindo, a caixa do PWM, possui junto de si uma fonte chaveada, então, com apenas uma estrutura foi possível realizar a alimentação do motor e a do circuito. Passada esta etapa, a tensão chegará a base coletora, passando o sinal de tensão de forma alternada para a estrutura superior, para que esta tensão não seja alternada, como já informado em tópicos anteriores, a tensão vai passar por uma retificação encontrada no bloco referente a placa retificadora. Feita a retificação do sinal, a alimentação chega ao arduíno que por sua vez irá realizar o controle dos leds a partir de uma programação interna feita em C, e também poderá realizar alterações de horário a partir de um receptor infravermelho que serve para receber o sinal de um controle, para que então as alterações sejam feitas.

#### 4.4. Problemas e soluções

<b>PROBLEMAS</b>
Realizar a alimentação do circuito que gira com a estrutura.
Como retificar a alimentação que alterna de positivo para negativo
Manter a imagem do relógio parada
Balanceamento da Estrutura
Para fazer um relógio, precisa-se de um um loop infinito com delay de 1s, e para a formação da imagem precisa de outro loop infinito. O arduino não tem processamento suficiente para rodar estes dois loops infinitos.

<b>SOLUÇÕES</b>
Utilizou-se um pedaço de motor que contém a parte receptora e os ímãs que mandam e alimentam o circuito.
Confeccionar uma placa retificadora para manter o sinal de alimentação constante.
Utilizou-se um transistor 7812 posicionado estrategicamente, que toda vez que recebe um pulso de alimentação, ele envia um sinal para uma entrada do arduino, esta entrada é utilizada no programa, para ativar o programa somente quando estiver em HIGH.
Fixar o motor o motor estrategicamente, e uma base redonda com a placa retificadora e o arduino posicionados próximo ao eixo central.
A partir do clock do arduino, o tempo que o arduino demora para realizar uma tarefa inteira, foi recolhido e foi dividido por 1000, com este valor obtido, foi possível realizar um contador de segundos para formar o relógio.

## 5. CÓDIGO FONTE ( C )

```
#include <IRremote.h>
int aux=0;
int aux2=0;
int RECV_PIN = 2;
IRrecv irrecv(RECV_PIN);
decode_results results;

void setup() {
  for(int i=7; i<=12;i++){
    pinMode(i, OUTPUT);
  }
  irrecv.enableIRIn();
  pinMode(53,INPUT);
  Serial.begin(9600);
}

void loop() {
  if (irrecv.decode(&results)){

    long int decCode = results.value;

    switch (decCode){

      case 4105841032:

        while(aux<=60){
          if(digitalRead(53)==HIGH){
            escreve();

            aux ++;
          }
        }
        break;
      case 1595074756:

        while(aux2<=60){
          if(digitalRead(53)==HIGH){
            escreve2();

            aux2 ++;
          }
        }
        break;
    }
    irrecv.resume();
  }
}

void escreve() {

  delay(20);
  letraP();
}
```

```

        delay(2);
        letraU();
        delay(2);
        letraC();
        delay(3);
        letraP();
        delay(2);
        letraR();
        delay(80);
    }
    void escreve2() {

        delay(20);
        numeroUm();
        delay(2);
        numeroDois();
        delay(2);
        numeroTres();
        delay(3);
        numeroQuatro();
        delay(80);

    }
    void numeroUm() {
        digitalWrite(13, HIGH);
        digitalWrite(9, HIGH);
        delay(1);

        digitalWrite(9, LOW);

        digitalWrite(8, HIGH);
        delay(1);
        digitalWrite(8, LOW);

        for(int i=7; i<=12; i++){
            digitalWrite(i, HIGH);
        }
        delay(1);
        for(int i=7; i<=12; i++){
            digitalWrite(i, LOW);
        }

    }
    void numeroDois() {
        digitalWrite(13, HIGH);

        /*    ****
            *
            ****
            *
            ****
        */
        //Primeira parte
        digitalWrite(7, HIGH);

```

```

digitalWrite(10,HIGH);
digitalWrite(11,HIGH);
digitalWrite(12,HIGH);
delay(1);
digitalWrite(7,LOW);
digitalWrite(10,LOW);
digitalWrite(11,LOW);
digitalWrite(12,LOW);

//Segunda parte
digitalWrite(7,HIGH);
digitalWrite(10,HIGH);
digitalWrite(12,HIGH);
delay(1);
digitalWrite(7,LOW);
digitalWrite(10,LOW);
digitalWrite(12,LOW);

//TERCEIRA PARTE
digitalWrite(7,HIGH);
digitalWrite(10,HIGH);
digitalWrite(12,HIGH);
delay(1);
digitalWrite(7,LOW);
digitalWrite(10,LOW);
digitalWrite(12,LOW);

//QUARTA PARTE
digitalWrite(7,HIGH);
digitalWrite(8,HIGH);
digitalWrite(9,HIGH);
digitalWrite(10,HIGH);
digitalWrite(12,HIGH);

delay(1);
digitalWrite(7,LOW);
digitalWrite(8,LOW);
digitalWrite(9,LOW);
digitalWrite(10,LOW);
digitalWrite(12,LOW);
}

void numeroTres(){
digitalWrite(7, HIGH);
digitalWrite(9, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(9, LOW);
digitalWrite(12, LOW);

digitalWrite(7, HIGH);
digitalWrite(9, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(9, LOW);
}

```

```

digitalWrite(12, LOW);

digitalWrite(7, HIGH);
digitalWrite(9, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(9, LOW);
digitalWrite(12, LOW);

digitalWrite(7, HIGH);
digitalWrite(9, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(9, LOW);
digitalWrite(12, LOW);

for(int i=7; i<=12;i++){
    digitalWrite(i, HIGH);
}
delay(1);
for(int i=7; i<=12;i++){
    digitalWrite(i, LOW);
}

}

void numeroQuatro(){
    for(int i=7; i<=9; i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=9; i++){
        digitalWrite(i, LOW);
    }

    int aux=0;
    while(aux<=1){
        digitalWrite(9, HIGH);
        aux++;
        delay(1);
    }
    digitalWrite(9, LOW);

    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }

}

void numeroCinco(){
    for(int i=7; i<=9; i++){

```

```

    digitalWrite(i, HIGH);
    delay(1);
}
for(int i=7; i<=9; i++){
    digitalWrite(i, HIGH);
}

int aux=0;
while(aux<=1){
    digitalWrite(7, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(12, HIGH);
    aux++;
    delay(1);
}
    digitalWrite(7, LOW);
    digitalWrite(9, LOW);
    digitalWrite(12, LOW);

for(int i=9; i<=12; i++){
    digitalWrite(7, HIGH);
    digitalWrite(i, HIGH);
}
delay(1);
for(int i=9; i<=12; i++){
    digitalWrite(7, LOW);
    digitalWrite(i, LOW);
}

}

void numeroSeis(){

}

void letraM(){
    digitalWrite(13, HIGH);

    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }

    digitalWrite(8, HIGH);
    delay(1);
    digitalWrite(8, LOW);

    digitalWrite(9, HIGH);
    delay(1);
    digitalWrite(9, LOW);

    digitalWrite(10, HIGH);
    delay(1);
    digitalWrite(10, LOW);
}

```



```

digitalWrite(9, HIGH);
delay(1);
digitalWrite(9, LOW);

digitalWrite(8, HIGH);
delay(1);
digitalWrite(8, LOW);

for(int i=7; i<=12;i++){
    digitalWrite(i, HIGH);
}
delay(1);
for(int i=7; i<=12;i++){
    digitalWrite(i, LOW);
}

}

void letraA(){
    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }

    digitalWrite(7, HIGH);
    digitalWrite(10, HIGH);
    delay(1);
    digitalWrite(7, LOW);
    digitalWrite(10, LOW);

    digitalWrite(7, HIGH);
    digitalWrite(10, HIGH);
    delay(1);
    digitalWrite(7, LOW);
    digitalWrite(10, LOW);

    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }

}

void letraE(){

    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }
}

```

```

digitalWrite(7, HIGH);
digitalWrite(10, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(10, LOW);
digitalWrite(12, LOW);

digitalWrite(7, HIGH);
digitalWrite(10, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(10, LOW);
digitalWrite(12, LOW);

digitalWrite(7, HIGH);
digitalWrite(10, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(10, LOW);
digitalWrite(12, LOW);
}
void letraJ(){
digitalWrite(7, HIGH);
digitalWrite(10, HIGH);
digitalWrite(11, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(10, LOW);
digitalWrite(11, LOW);

int aux=0;
while(aux<=2){
digitalWrite(7, HIGH);
digitalWrite(12, HIGH);
aux++;
delay(1);
}
digitalWrite(7, LOW);
digitalWrite(12, LOW);

for(int i=7; i<=11; i++){
digitalWrite(i, HIGH);
}
delay(1);
for(int i=7; i<=11; i++){
digitalWrite(i, LOW);
}
}

void letraS(){

```

```

digitalWrite(7, HIGH);
digitalWrite(8, HIGH);
digitalWrite(9, HIGH);
digitalWrite(12, HIGH);
delay(1);
digitalWrite(7, LOW);
digitalWrite(8, LOW);
digitalWrite(9, LOW);
digitalWrite(12, LOW);

int aux=0;
while(aux<=1){
    digitalWrite(7, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(12, HIGH);
    aux++;

}
digitalWrite(7, LOW);
digitalWrite(9, LOW);
digitalWrite(12, LOW);

for(int i=9; i<=12; i++){
    digitalWrite(7, HIGH);
    digitalWrite(i, HIGH);
}
delay(1);
for(int i=9; i<=12; i++){
    digitalWrite(7, LOW);
    digitalWrite(i, LOW);
}
}

void letraI(){
    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }
}

void letraC(){
    for(int i=8; i<=11;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=8; i<=11;i++){
        digitalWrite(i, LOW);
    }

int aux=0;
while(aux<=1){
    digitalWrite(7, HIGH);
    digitalWrite(12, HIGH);
    aux++;
    delay(1);
}

```

```

    }
    digitalWrite(7, LOW);
    digitalWrite(12, LOW);

    digitalWrite(8, HIGH);
    digitalWrite(11, HIGH);
    delay(1);
    digitalWrite(8, LOW);
    digitalWrite(11, LOW);
}

void letraP(){
    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }

    int aux=0;
    while(aux<=1){
        digitalWrite(7, HIGH);
        digitalWrite(9, HIGH);
        aux++;
        delay(1);
    }
    digitalWrite(7, LOW);
    digitalWrite(9, LOW);

    for(int i=7; i<=9;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=9;i++){
        digitalWrite(i, LOW);
    }
}

void letraU(){
    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }

    int aux=0;
    while(aux<=1){
        digitalWrite(12, HIGH);
        aux++;
        delay(1);
    }
    digitalWrite(12, LOW);

    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
}

```

```

    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }
}

void letraR(){
    for(int i=7; i<=12;i++){
        digitalWrite(i, HIGH);
    }
    delay(1);
    for(int i=7; i<=12;i++){
        digitalWrite(i, LOW);
    }

    digitalWrite(7, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(10, HIGH);
    delay(1);
    digitalWrite(7, LOW);
    digitalWrite(9, LOW);
    digitalWrite(10, LOW);

    digitalWrite(7, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(11, HIGH);
    delay(1);
    digitalWrite(7, LOW);
    digitalWrite(9, LOW);
    digitalWrite(11, LOW);

    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(12, HIGH);
    delay(1);
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    digitalWrite(12, LOW);
}

```

## 5.1. Resultado da programação

Após toda programação e estrutura ter sido completada, o efeito desejado finalmente foi alcançado, logo abaixo o resultado final do projeto.



**Fig. 4 – Resultado Final**

## 6. CONCLUSÃO

Utilizando a ideia de movimento e a parte de hardware, é possível criar coisas incríveis com criatividade e pesquisa. Existem métodos muito mais simples de se fazer este projeto, porém o conhecimento até então obtido não era suficiente para tal, tanto que de setenta e dois leds, o projeto final ficou só com sete, e mesmo assim possui um efeito impressionante.

Diversos problemas foram encontrados, e correr atrás de soluções, implementar funcionalidades, todos estes itens citados foram importantes para o aprimoramento do conhecimento do grupo.

O arduíno, apesar de ser uma ferramenta muito útil, possui suas limitações, limitações estas que foram citadas no tópico de problemas e soluções. A forma para qual este problema se encaminhou, mostra como o pensamento em grupo pode fazer com que mesmo com pouco recurso, um pouco de criatividade e boa vontade faça deste pouco recurso, uma ferramenta mais do que necessária para resolver qualquer empecilho.

## **7. REFERÊNCIAS**

- Auxílio em programação e funcionamento do Arduino

<http://www.arduino.cc/>

- Montagem do PWM

<http://www.circuitstoday.com/dc-motor-controller>