

Quarto Projeto: Resolução de Problemas Reais com Arduino Shield Multifunction

Cronômetro de Produção Industrial

1. Lucas Gontijo Riani, Pedro Henrique Lopes De Melo, Bernardo Barbosa Heronville

Código do Projeto:

```
#include <TimerOne.h>

#include <MultiFuncShield.h>

// Variáveis de controle

bool rodando = false;      // Indica se o cronômetro está em execução

bool pausado = false;      // Indica se o cronômetro está pausado

int Cronometro = 0;        // Contador do tempo (em segundos)

unsigned long atualizado = 0; // Marca o último momento de atualização

const int tempo_maximo = 40; // Tempo limite do cronômetro (em segundos)

void setup() {

    Serial.begin(9600);      // Inicia comunicação serial para debug

    Timer1.initialize();     // Inicializa o Timer1

    MFS.initialize(&Timer1); // Inicializa o MultiFunc Shield com o Timer1

    MFS.write(Cronometro);    // Exibe o tempo inicial (0) no display

}
```

```
void loop() {

    byte btn = MFS.getButton();    // Lê o botão pressionado no Shield

    if (btn) {

        byte buttonNumber = btn & B00111111;    // Extrai o número do botão (1 a 3)

        byte buttonAction = btn & B11000000;    // Extrai o tipo de ação (clique curto, longo, etc.)

        if (buttonAction == BUTTON_SHORT_RELEASE_IND) { // Se foi um clique curto

            switch (buttonNumber) {

                case 1: // Botão 1: Iniciar o cronômetro

                    rodando = true;

                    pausado = false;

                    Cronometro = 0;

                    MFS.writeLeds(LED_ALL, OFF);    // Apaga todos os LEDs

                    break;

                case 2: // Botão 2: Pausar ou continuar

                    if (rodando) {

                        pausado = !pausado;        // Alterna entre pausado e rodando

                    }

                    break;

                case 3: // Botão 3: Parar e resetar

                    rodando = false;

                    pausado = false;

                    Cronometro = 0;

            }

        }

    }

}
```

```

    MFS.writeLeds(LED_ALL, OFF);    // Apaga os LEDs

    MFS.write(0);                    // Reseta o display

    break;

}

}

}

// Se o cronômetro estiver rodando e não estiver pausado

if (rodando && !pausado) {

    unsigned long atual = millis(); // Tempo atual desde que o Arduino ligou

    if (atual - atualizado >= 1000) { // Verifica se passou 1 segundo

        atualizado = atual;           // Atualiza o tempo da última contagem

        Cronometro++;               // Incrementa o tempo

        Serial.print("Tempo: ");

        Serial.println(Cronometro);

        MFS.write(Cronometro);      // Atualiza o display com o novo tempo

        // Atualiza os LEDs com base no tempo

        if (Cronometro <= 40) {

            byte leds = 0b0000;

            if (Cronometro >= 10) leds = 0b0001;

            if (Cronometro >= 20) leds = 0b0011;

            if (Cronometro >= 30) leds = 0b0111;

```

```

    if (Cronometro >= 40) leds = 0b1111;

    MFS.writeLeds(LED_ALL, OFF); // Limpa LEDs antes de atualizar

    MFS.writeLeds(leds, ON);    // Liga os LEDs correspondentes
}

// Quando atinge o tempo máximo
if (Cronometro == tempo_maximo) {

    piscarLeds(3); // Pisca os LEDs 3 vezes

    som(5);        // Emite 5 bipes

    pausado = true; // Pausa o cronômetro
}

}

}

}

// Função para piscar os LEDs
void piscarLeds(int times) {

    for (int i = 0; i < times; i++) {

        MFS.writeLeds(LED_ALL, ON);

        delay(200);

        MFS.writeLeds(LED_ALL, OFF);

        delay(200);

    }

}

```

```
// Função para emitir bipes sonoros
```

```
void som(int times) {
```

```
    for (int i = 0; i < times; i++) {
```

```
        MFS.beep();
```

```
        delay(100);
```

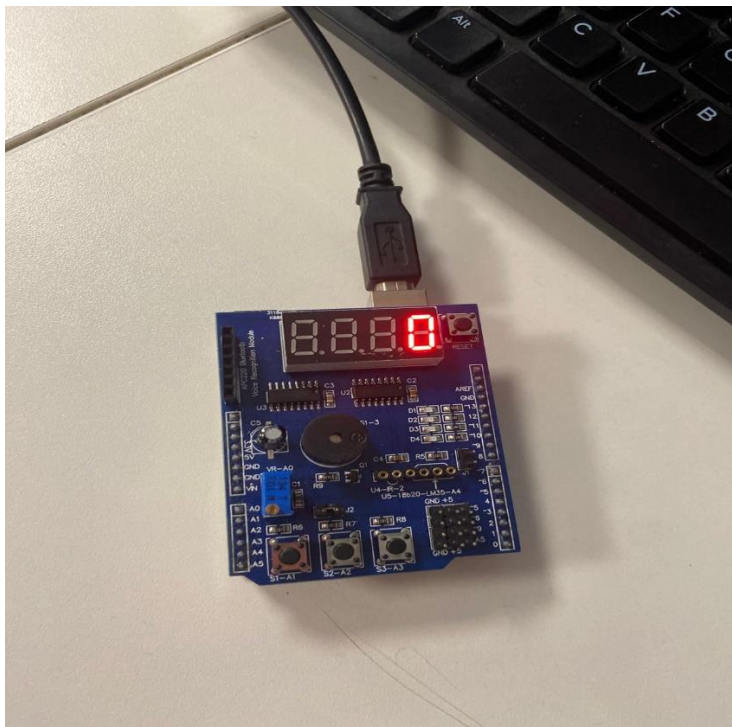
```
    }
```

```
}
```

Vídeo do Projeto:

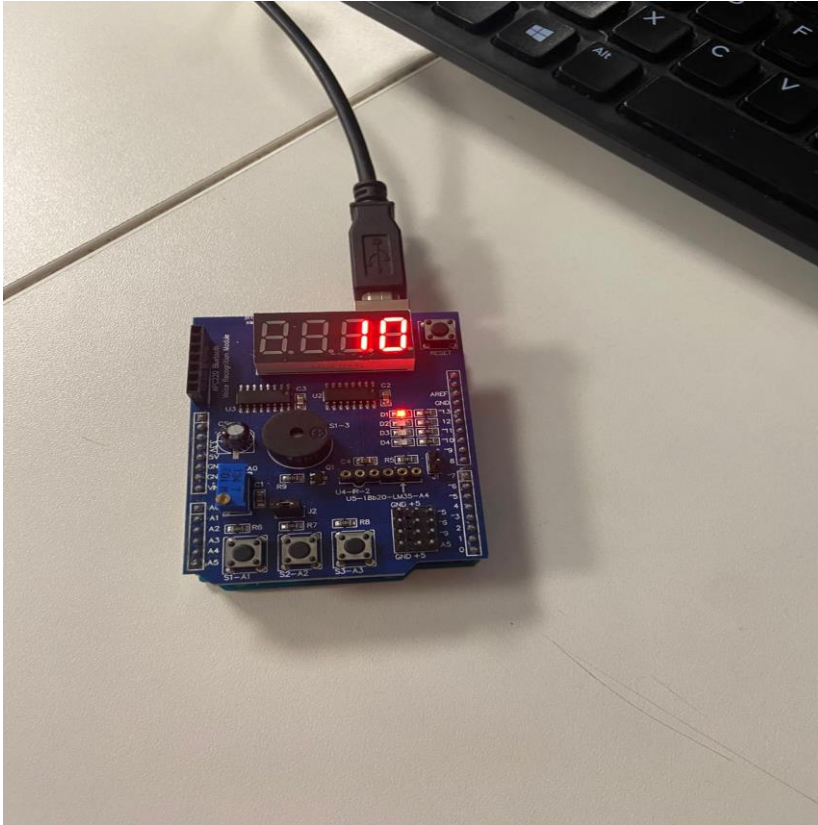
<https://drive.google.com/file/d/11kLykGfhYeUhYwA81SKOenOEeHTYozDL/view?usp=sharing>

Fotos explicando o Projeto:

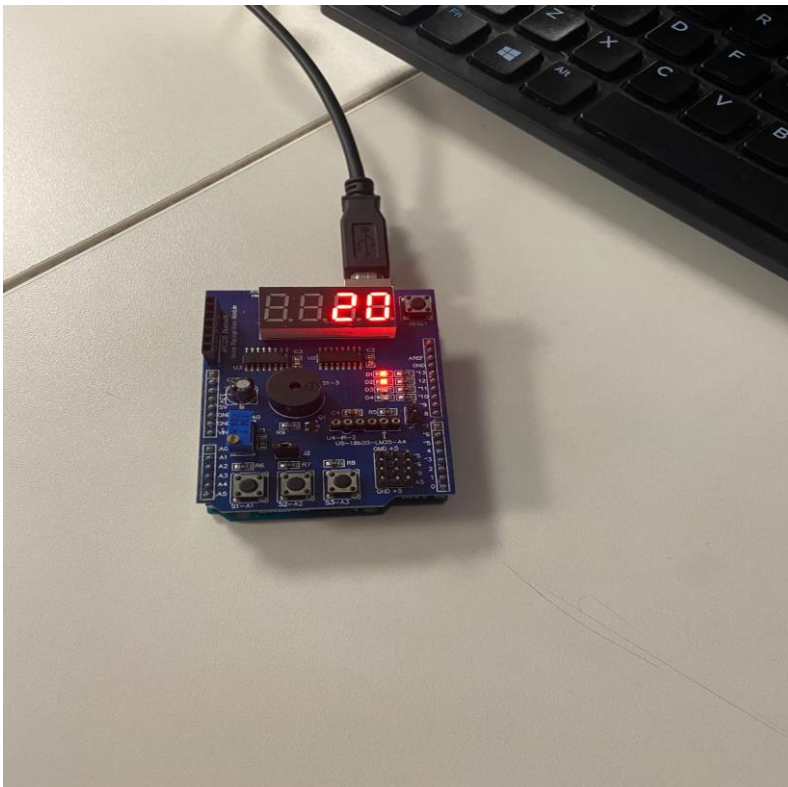


Cronômetro iniciado pelo botão da esquerda. O LEDs acesos devido ao tempo.

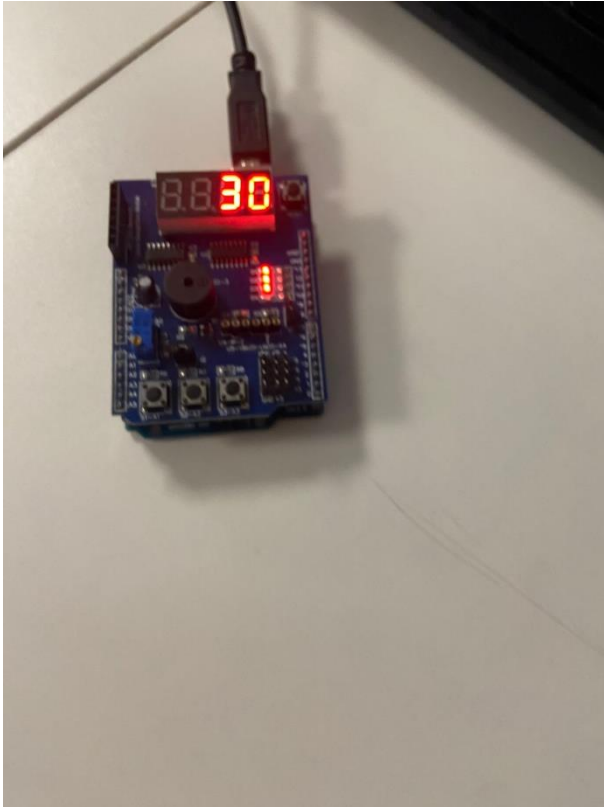
Ou cronômetro foi reiniciado pelo botão da direita e voltou ao número zero.



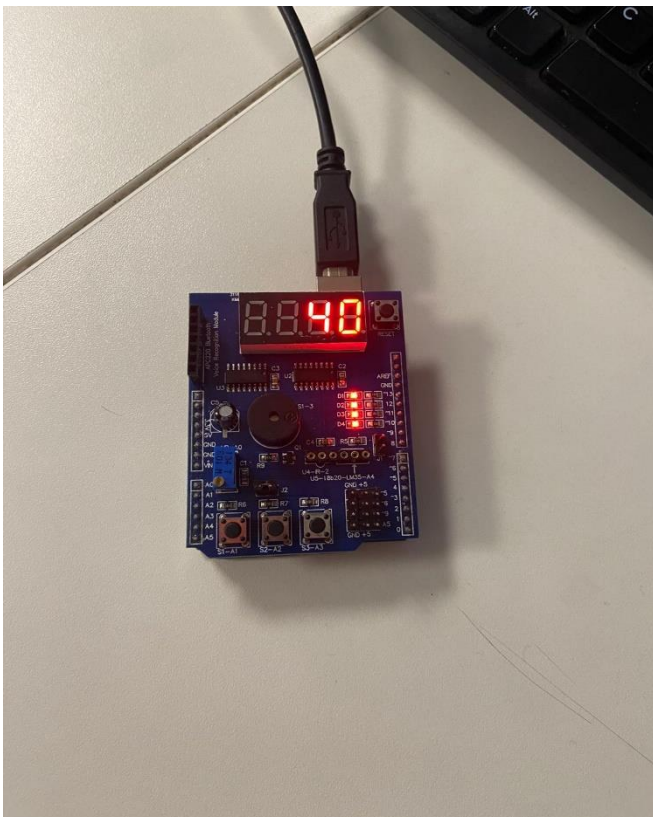
Cronômetro aos 10 segundos, logo a 1ª LED foi acesa.



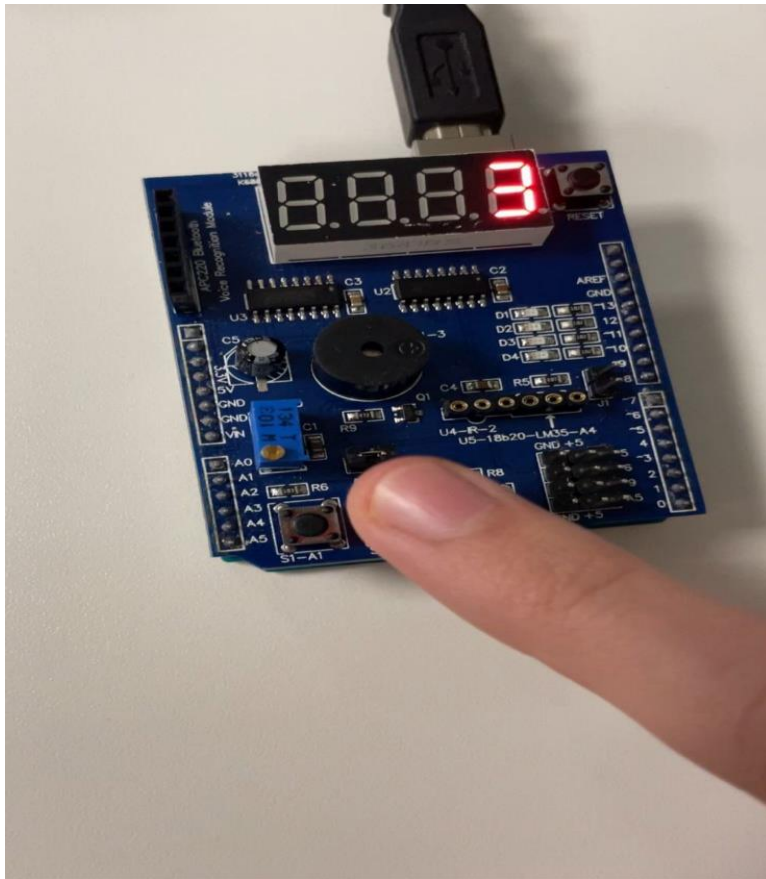
Cronômetro aos 20 segundos, logo duas LEDs estão acesas.



Cronômetro aos 30 segundos, logo três LEDs estão acesas.



Cronômetro aos 40 segundos, logo as quatro LEDs estão acesas e o buzzer foi ativado.



Neste momento,o botão do meio está sendo pressionado,o que faz o cronômetro pausar e também pode ser utilizado para continuar a contagem(despausando).