



Substituindo Arduino por MCU

Relatório de Atividade

ARTHUR HENRIQUE DIAS NUNES, MARIA LUIZA DE ANDRADE ALVES

Petiano: Arthur Nunes

Resumo

As placas Arduino são úteis para prototipagem de projetos e são amplamente utilizadas no grupo. No entanto, quando passada a fase de prototipagem e projeto estiver em sua fase final não é recomendado manter o uso dessas placas. Isso se deve pelo tamanho delas, pela maior possibilidade de mal contato, pelo preço e principalmente pelo fato de gastar uma placa por projeto - é quase uma inutilização da mesma.

Como engenheiros e bons projetistas, devemos ser capazes de substituir a placa por um circuito eletrônico menor, mais compacto, mais barato e com menor gasto energético. Substituir o Arduino por uma placa de circuito impresso com um MCU é o primeiro passo para isso.

Sendo assim, neste relatório descrevemos o circuito mínimo necessário para usar um microcontrolador e partes que podem ser adicionadas, como um circuito de gravação de código ou de gravação de bootloader, suficiente até para criar a própria placa com um MCU. Além disso, também descobrimos como se programar um Arduino usando outro, conhecimento que pode ser útil caso alguma placa não possa ser conectada com o computador.

I. INTRODUÇÃO

Esse projeto é fruto de uma necessidade de um outro: compreender como usar um MCU no lugar de um Arduino para simplificar, reduzir e economizar o circuito. Nesse caso, o MCU é o Microcontrolador Atmega328p. Ele é o mesmo de um Arduino Uno, com a mesma quantidade e tipos de pinos. Esse MCU pode ser encontrado na versão PU ou na AU. Na PU o componente é tipo DIP e é encontrado acoplado na placa Uno de modelos mais comuns, como 28 pinos. Já o AU possui 32 pinos e é tipo SMD e, por isso, consideravelmente menor - cabe na ponta do seu dedo. Este último é encontrado em placas Uno de modelos mais novos.

Neste relatório é usado como referência o Atmega328p, como descrito acima. Dessa forma, as montagens, pinagens, prototipagens e o produto final são específicos desse microcontrolador. Para a utilização de outros, como o Atmega2560 (o mesmo de um Arduino Mega) podem ser necessárias adaptações.

Para o PETEE este projeto representa a capacidade de substituir todos os Arduinos usados em outros projetos que estão na fase final. Assim, as placas podem ser usadas nas Oficinas e

para o desenvolvimento de novos projetos sem precisar desmontar os já finalizados, como já aconteceu. Este projeto representa também um upgrade nos desenvolvimentos do PETEE, é um ótimo conteúdo e será uma ótima prática para o grupo finalizar seus trabalhos da forma aqui proposta.

Para o grupo, é viável também substituir os projetos por uma placa Arduino Nano ou Mini, que possuem ótimos preços e tamanhos mínimos. Ainda assim, construir sua própria placa com um microcontrolador possui duas vantagens: é possível compreender melhor o funcionamento do software e hardware das placas e o consumo de energia é mínimo.

Curiosidades: DIP ou DIL são acrônimos para *dual in-line package*, dispositivos eletrônicos compostos por um retângulo e duas linhas de pinos. Eles são geralmente montados em uma placa de circuito impresso (PCI) por *through-hole technology*, através de furos como o nome sugere. Já SMD é um acrônimo para *surface-mount device*, dispositivos que são soldados em PCIs por *surface-mount technology* (SMT). Estes não necessitam de furos, são consideravelmente menores e necessitam de mais cuidado no processo solda.

II. TEORIA

Microcontroladores

Os MCUs (*MicroController Units*) podem ser resumidos a blocos que contém CPU (*Central Processing Unit*), memória e periféricos. Eles são muito usados em aplicações embarcadas, como é o caso de vários projetos do PETEE. Eles também facilitam o desenvolvimento de sistemas com poucos componentes, possuem baixo custo e são de fácil gravação e regravação de programas, (Ricardo de Oliveira Duarte,).

Geralmente, para o seu funcionamento, são necessários apenas um oscilador e uma fonte de alimentação externos. No caso do Atmega328p não é diferente. No entanto, o oscilador externo pode ser substituído por um oscilador interno, o que será explicado mais adiante e pode-se adicionar um botão push para fazer reiniciar a programação que estiver gravada.

A pinagem do Atmega328p está na Fig. 1.

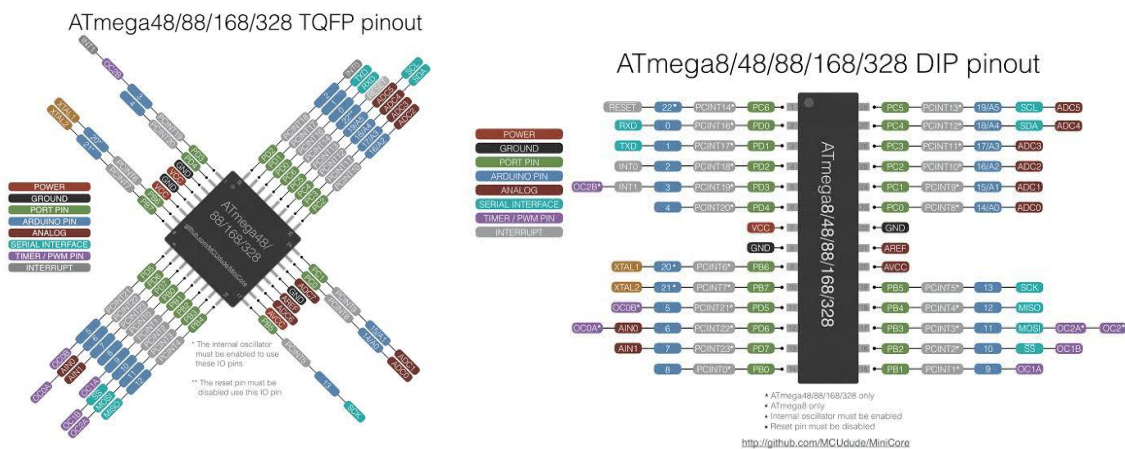


Figura 1: Pinagem Atmega328p. À esquerda, na versão AU e à direita na versão PU.

III PRÁTICA

Bootloader

Bootloader, carregador de inicialização em tradução livre, é um software intermediário entre o hardware e a BIOS de um computador ou smartphone que auxilia o carregamento do sistema operacional. No caso do MCUs, o bootloader é necessário para que ele seja programável e execute os códigos desenvolvidos pelos projetistas. Alguns microcontroladores, como os que vêm nas placas Arduino, já vêm com o bootloader ao comprá-lo. Em outros, é necessário fazer o processo de gravação, chamado de *burn the bootloader*. O processo é simples e rápido. Ele também permite algumas configurações avançadas, como escolher a frequência do oscilador externo que será utilizado ou selecionar o oscilador interno.

É possível realizar o bootloader com uma placa Arduino e com a sua própria IDE. No caso da placa UNO devem ser usados os pinos digitais 10, 11, 12 e 13. O pino 10 será ligado ao RESET do microcontrolador, o 11 ao MOSI, 12 ao MISO e o 13 ao SCK.

A placa deve ser conectada ao computador e na Arduino IDE deve-se selecionar **Tools -> Board -> Arduino Duemilanove or Diecimila**. Em seguida em **Files -> Examples -> ArduinoISP**. Irá aparecer um código na tela e em deve-se realizar o procedimento de gravação na placa. Ainda nesta etapa não são necessárias as conexões entre o Atmega e a placa Arduino. Voltando a IDE, agora com as conexões entre o MCU e a placa feitas, deve-se selecionar a aba **Tools -> Programmer -> Arduino as ISP**. Verifique se a opção foi realmente selecionada e em seguida selecione a aba **Tools -> Burn BootLoader**. Para saber que o processo foi concluído irá aparecer a mensagem *Done burning bootloader*.

III. PRÁTICA

Funcionamento

Para o circuito de funcionamento do microcontrolador são necessários os seguintes componentes:

- 1 x Cristal Oscilador de 16MHz;
- 2 x Capacitores de 22pF;
- 1 x Capacitor de 100nF (opcional, mas recomendado);
- 1 x Resistor de 10k Ω ;
- 1 x Botão tipo push (opcional);
- 1 x Fonte de alimentação de 5VDC.

Um terminal do oscilador deve estar ligado ao pino XTAL1 e o outro ao XTAL2. Cada um desses terminais também deve estar ligado a um capacitor de 22pF e os capacitores ligados ao GND.

Em seguida, o pino de reset deve ser ligado ao resistor, que por sua vez deve ser ligado ao VCC. Isso porque o pino de reset é ativado em nível lógico baixo, desta forma evita que ele seja acionado. Sendo assim, esse resistor desempenha a função de pull-up. Pode-se

adicionar um botão para dar o reset quando pressionado. Para isso o botão deve ter um terminal ligado ao pino e outro ligado ao GND.

Finalmente, a fonte de alimentação deve ser adicionada entre o VCC e o GND. É recomendado que adicionar um capacitor de desacoplamento para eliminar ruídos vindos da fonte, isto é, um filtro passa-baixa. Uma montagem comum é um capacitor de 100nF entre VCC e GND.

O esquemático desta etapa está na Fig. 2. A montagem na protoboard pode ser vista na Fig. 7.

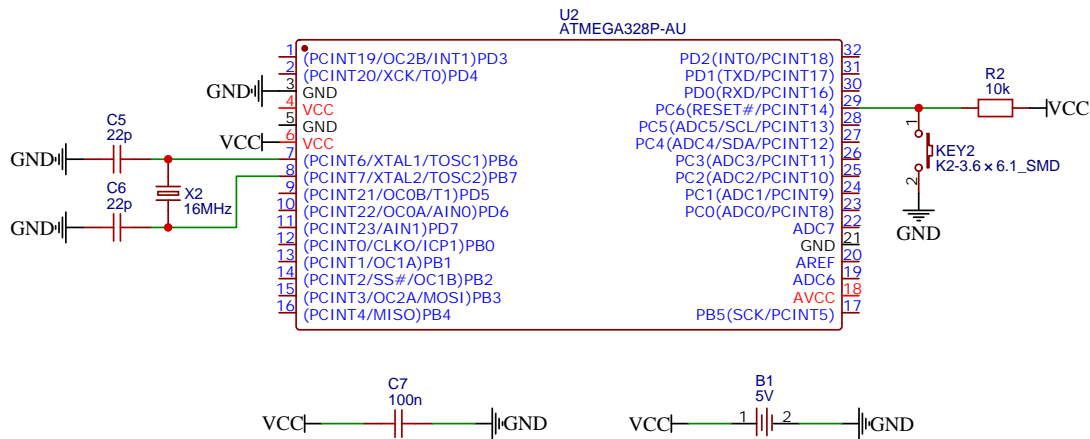


Figura 2: Esquemático do circuito de funcionamento.

Bootloader

A montagem desta etapa está na Fig. 3. Ela deve ser acrescida da montagem anterior para funcionar. Nesse caso, a fonte externa não será necessária se o bootloader alimentar o sistema. (WR Kits, b).

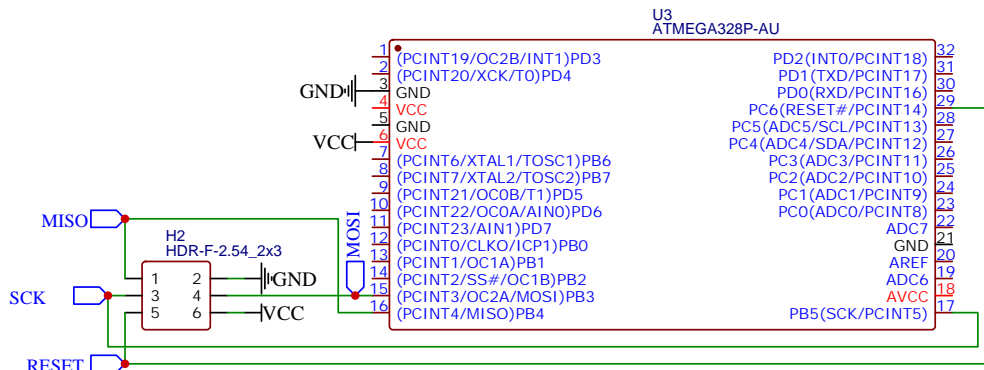


Figura 3: Esquemático do circuito de bootloader.

III PRÁTICA

Gravação

Nesta etapa, será necessário adicionar:

- 1 x Capacitor 100nF;
- 1 x Módulo conversor USB-Serial FTDI, Fig. 9. Este pode ser substituído por uma placa Arduino Uno sem o microcontrolador.

A montagem desta etapa está na Fig. 4. Ela deve ser acrescida da montagem anterior para funcionar. Nesse caso, a fonte externa não será necessária se o gravador alimentar o sistema. (WR Kits, a), (educ8s.tv,).

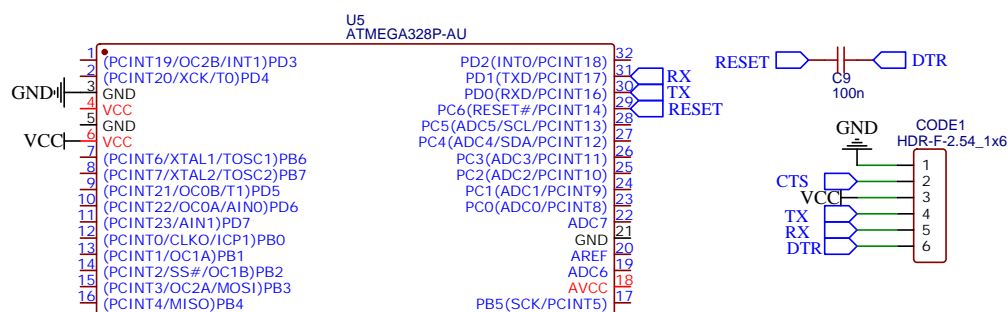


Figura 4: Esquemático do circuito de gravação.

DIY Placa Arduino

Com as descrições feitas até aqui, é possível criar a sua própria DIY placa "Arduino". Um esquemático desta etapa está na Fig. 5. Depois disso, foram adicionados conectores para que seja possível usar quaisquer pinos desejados.

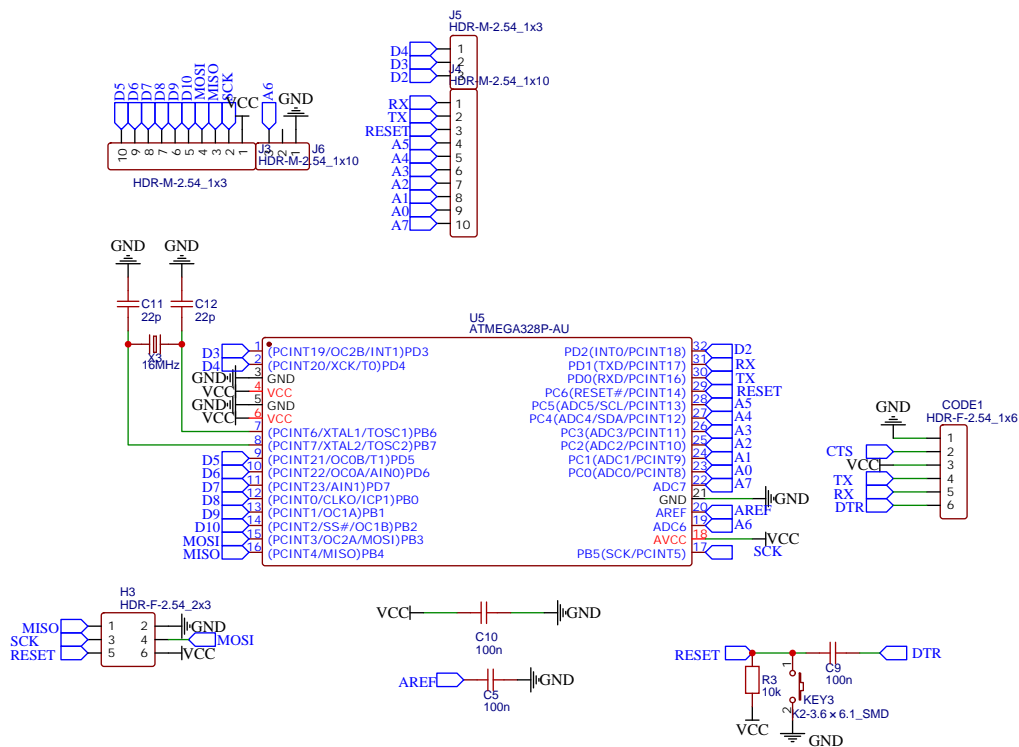


Figura 5: Esquemático do circuito da PCI.

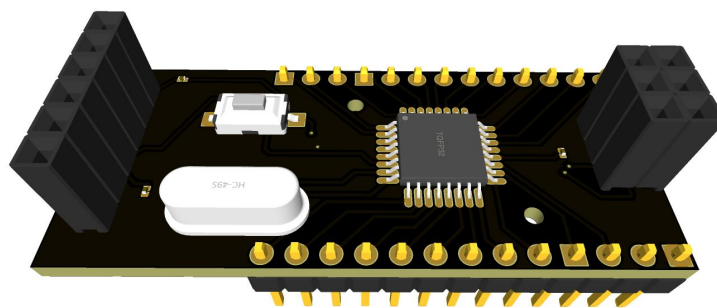


Figura 6: Visualização do modelo em 3D.

IV. EXPERIMENTOS E RESULTADOS

Experimento 1

O primeiro experimento consistiu em um circuito mínimo para a utilização do MCU Atmega328p. Ele foi montado em uma protoboard, e por isso foi usada a versão PU do MCU. Nele foi montada a instrução *Hello World* do Arduino: piscar um LED. A montagem foi pequena e com o seu sucesso já é possível concluir que pode-se substituir alguns Arduinos de projetos que estão em fases finais do PETEE por apenas o MCU e alguns componentes simples descritos na Seção III. A montagem deste protótipo pode ser vista na Fig. 7. A programação do MCU nesse caso foi feita diretamente pela placa Uno, encaixando o Atmega328p-PU na placa.

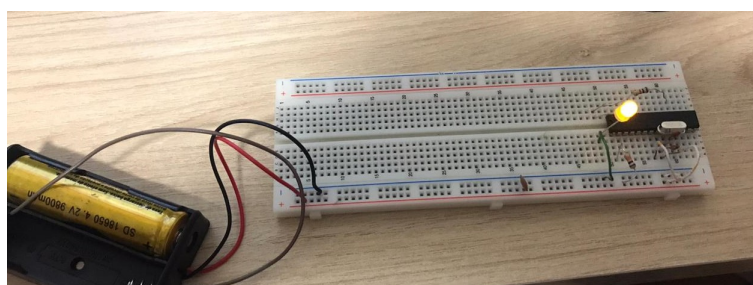


Figura 7: *Atmega328p-PU na protoboard.*

Vale a pena reproduzi-lo em uma PCI, quando possível, nos projetos do grupo: será mais barato, não ocuparemos uma placa Arduino desnecessariamente e o consumo de energia e o tamanho do circuito ficarão menores.

Experimento 2

Este também consistiu na programação de um MCU que estava montado em uma placa, como no anterior. No entanto, a novidade neste experimento é que a programação do controlador não foi feita diretamente pela placa Uno. Neste caso, a placa foi usada como um conversor FTDI, (Instructables, a).

Isso possibilitou que o MCU permanecesse na protoboard, ou em um circuito impresso que pode ser o caso das aplicações nos projetos do PETEE. As ligações foram simples:

- 5V da placa no VCC da protoboard;
- GND da placa no GND da protoboard;
- RESET da placa no RESET do MCU;
- TX da placa no TX do MCU;
- RX da placa no RX do MCU.

O resultado está ilustrado na Fig. 8.

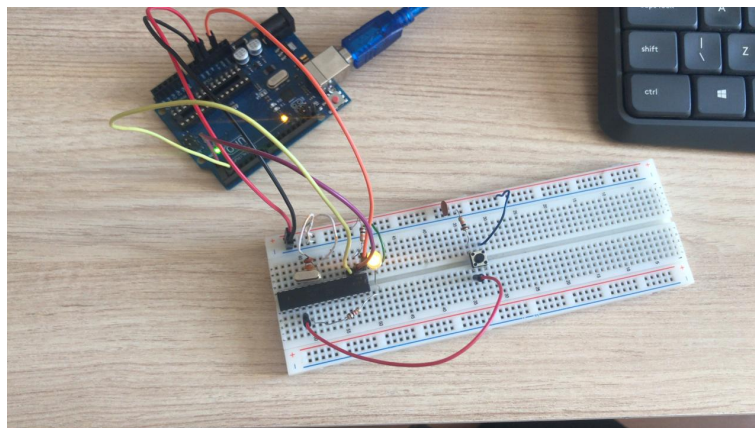


Figura 8: Usando uma placa Uno como um conversor FTDI.

Esse método precede o método mais geral de programação de um microcontrolador em uma placa: usar um conversor FTDI, (educ8s.tv,). Esse conversor é um módulo USB-Serial próprio para essa aplicação. Esse módulo está representado na Fig. 9.

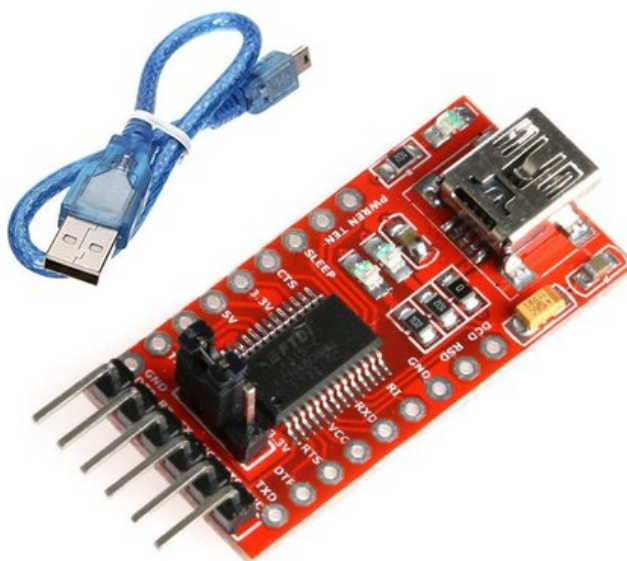


Figura 9: Módulo FTDI.

Experimento 3

Para o terceiro experimento foi usada uma placa Arduino Nano. A novidade dele é a programação da placa: ela não foi feita diretamente usando um cabo de dados porque não havia disponibilidade de um cabo com entrada micro USB no momento. Para isso, foi usada uma placa Uno para realizar a interface. Existem algumas formas de realizar esse procedimento:

IV EXPERIMENTOS E RESULTADOS

- Usando a placa Uno como um ICSP, (Arduino Addiction,);
- Usando a placa Uno como um FDTI, o mesmo usado no Experimento 2, (Instructables, a);
- Usando a placa Uno como um ICSP mas sem fazer modificações nas configurações da IDE. Essa foi a maneira utilizada na Fig 10, (Instructables, b).

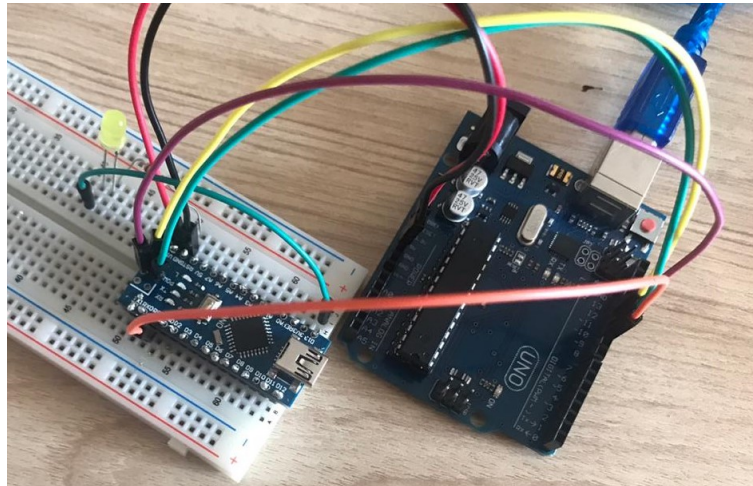


Figura 10: Programação de um Arduino Nano

Experimento 4

Finalmente, o último experimento foi feito utilizando um módulo FTDI para a programação de um MCU, Fig. 11, (educ8s.tv,).

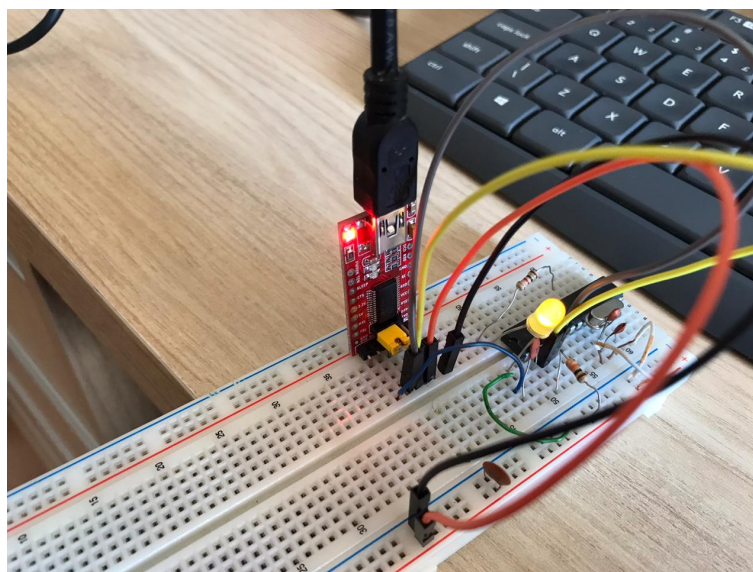


Figura 11: Programação de um MCU usando FTDI

V. DISCUSSÃO

O conhecimento descrito neste relatório é aplicável nos projetos do PETEE. Futuros projetos podem ser iniciados, dando continuidade a esse trabalho, como: (i) aplicação desse conhecimento em protótipos do PETEE; (ii) estudo de outros microcontroladores; (iii) estudo de outras maneiras de se substituir placas Arduino em projetos finais; (iv) circuitos adicionais que podem ser inseridos em conjunto com os descritos aqui, como proteção de alimentação, conversor USB-Serial, dentre outros.

Recomendamos a substituição nos projetos do PETEE e o uso deste conhecimento na criação de novos. Assim, os protótipos finais ficarão mais baratos, mais robustos, com menor gasto energético e mais sofisticados.

REFERÊNCIAS

Arduino Addiction. *Program Arduino Nano via Uno with ICSP*. arduinoaddiction.blogspot.com/2016/02/program-arduino-nano-via-uno-with-icsp.html. Acessado em 01 Junho 2020. 9

educ8s.tv. *Arduino Uno (ATMEGA328P) on a breadboard Tutorial DIY project*. youtu.be/npc3uzEVvc0. Acessado em 07 Maio 2020. 5, 8, 9

Instructables. *Arduino Examples 2 Use an Arduino As a FTDI Programmer*. instructables.com/id/Arduino-Examples-2-Use-an-Arduino-as-a-FTDI-Progr/. Acessado em 01 Junho 2020. 7, 9

Instructables. *Programming Arduino Nano Using UNO*. instructables.com/id/Programming-Arduino-Nano-Using-UNO/. Acessado em 01 Junho 2020. 9

Ricardo de Oliveira Duarte. *Arquitetura e Organização de Computadores*. 2

WR Kits. *Faça uma PCI Arduino Standalone*. youtu.be/RvBLmwF8JIE. Acessado em 05 Maio 2020. 5

WR Kits. *Gravando o Bootloader no Atmega328p*. youtu.be/5u8OGITwsAI. Acessado em 05 Maio 2020. 4