

“OFICINA DE ARDUINO LUZES CÓDIGO E AÇÃO: PRIMEIROS PASSO NO ARDUINO”

Paulo Ricardo Mello Ataide de Oliveira¹, Jeohvana Vaz², Maria Eduarda Jardim Tavares³, Sofia Magalhães Pires⁴, Marlipe Garcia Fagundes Neto⁵

¹Estudante de Engenharia Elétrica da Universidade Federal de Goiás, Aparecida de Goiânia, Brasil(paulo.ricardo2@discente.ufg.br)

²Estudante de Engenharia Elétrica da Universidade Federal de Goiás, Porungatu, Brasil(jeohvana@discente.ufg.br)

³Estudante de Engenharia Elétrica da Universidade Federal de Goiás, Aparecida de Goiânia, Brasil(maria_eduarda23456@discente.ufg.br)

⁴Estudante de Engenharia Elétrica da Universidade Federal de Goiás, Aparecida de Goiânia, Brasil(sofiamagalhaes@discente.ufg.br)

⁵Professor Doutor em Engenharia Universidade Federal de Goiás e tutor do PET Engenharias, Goiânia, Brasil(Marlipe.ufg.br)

Resumo: Este trabalho relata a oficina de introdução ao Arduino realizada pelo PET Engenharias da UFG no ENAPET 2025. A atividade, de caráter prático, abordou conceitos de microcontroladores e a execução de três projetos progressivos com hardware físico. O objetivo foi desmistificar a eletrônica e a programação para estudantes de diversas áreas, promovendo o protagonismo petiano e o desenvolvimento de competências tecnológicas transversais.

Palavras-chave: Arduino; PET Engenharias; ENAPET; Microcontroladores; Oficina prática

INTRODUÇÃO

Em um contexto acadêmico e profissional cada vez mais digital, a habilidade de entender e interagir com lógica de programação e eletrônica passou a ser uma competência relevante para estudantes de todas as áreas. A formação de profissionais capazes de desenvolver soluções inovadoras depende do aprimoramento de habilidades como raciocínio lógico, resolução de problemas e capacidade de projetar sistemas, atributos nos quais são desenvolvidos através do pensamento computacional (Wing, 2006).

Nesse contexto, a plataforma Arduino se destaca como uma ferramenta educacional eficaz, sendo amplamente reconhecida por seu custo reduzido, e apresenta uma tecnologia fácil de utilizar para criar projetos baseados em microcontroladores (Monk, 2012).

O Programa de Educação Tutorial (PET), vinculado à Secretaria de Educação Superior do

MEC, tem como premissa o desenvolvimento de atividades de ensino, pesquisa e extensão de forma integrada. Formado por grupos tutoriais de aprendizagem, o PET proporciona aos alunos participantes a realização de atividades extracurriculares com o intuito de complementar a formação acadêmica dos estudantes (UFG, 2026). O PET Engenharias da Universidade Federal de Goiás, formado pelos estudantes de engenharia elétrica, mecânica e computação, atua nessa perspectiva, promovendo ações que articulam o conhecimento técnico à formação cidadã e interdisciplinar dos seus membros e da comunidade ao redor.

O Encontro Nacional dos Grupos PET (ENAPET) é o principal evento que reúne petianos de todo o país, constituindo-se como um espaço de troca de experiências, apresentação de atividades e fortalecimento da identidade petiana. A edição de 2025 foi sediada pela Universidade de Brasília (UnB), nos dias 21, 22 e 23 de novembro, e reuniu grupos PET de diversas instituições federais.

Em consonância com esse contexto, o PET Engenharias da UFG propôs e executou uma oficina de introdução ao Arduino durante o ENAPET 2025, com o objetivo de proporcionar a petianos de diferentes áreas do conhecimento uma experiência prática com microcontroladores. A atividade visou desmistificar a tecnologia, oferecer uma base sólida em eletrônica básica e programação embarcada, e reforçar o caráter interdisciplinar e inovador do Programa de Educação Tutorial.

MATERIAL E MÉTODOS

A oficina de Arduino “**LUZES, CÓDIGO E AÇÃO: PRIMEIROS PASSOS COM ARDUINO**” foi planejada e conduzida por quatro bolsistas do PET Engenharias da UFG e contou com a participação de 9 petianos de diferentes grupos e instituições, nos quais foram divididos em 3 grupos de 3 pessoas, além dos dois petianos da comissão de organização do XXX ENAPET. O evento ocorreu de forma presencial no espaço reservado para atividades durante o ENAPET 2025.

Para a execução das atividades práticas, foram utilizados os seguintes materiais: 5 placas Arduino Uno, 4 protoboards, LEDs, resistores, botões (push buttons), transistores MOSFET, jumpers e motores dc 130 de 6v e baterias 18650 de 3,7v. A programação dos arduinos foi realizada por meio do software “Arduino IDE”, e foram utilizados os notebooks dos ministrantes e dos participantes para gravar os códigos nos arduinos.

A metodologia adotou uma abordagem progressiva, estruturada em três projetos de complexidade crescente, precedidos por uma apresentação teórica introdutória. O conteúdo teórico abordou o conceito de microcontrolador, o funcionamento da protoboard e do transistor tipo MOSFET e uma introdução básica ao Arduino IDE.

O primeiro projeto consistiu na montagem do circuito base com um LED externo, introduzindo os participantes à protoboard, ao uso de resistores para proteção do LED e à estrutura básica de um código Arduino com as funções `setup()` e `loop()`. Nessa etapa, os participantes realizaram o projeto Blink, fazendo o LED piscar em intervalos regulares de um segundo.

Como mostrado na figura 1, o objetivo deste circuito é ligar e desligar o LED em loop, e para isso o terminal negativo do LED foi conectado ao pino

GND do Arduino UNO através de fios e a placa protoboard, e o terminal positivo do LED deve ser ligado ao pino 13 do Arduino UNO, pino no qual será habilitado para liberar uma tensão de 5v para acender o LED. Entretanto, o LED não pode ser conectado diretamente ao Arduino, pois a corrente e tensão fornecidas pela placa queimariam o LED, para resolver esse problema é adicionado um resistor no terminal positivo do LED para limitar a corrente no LED.

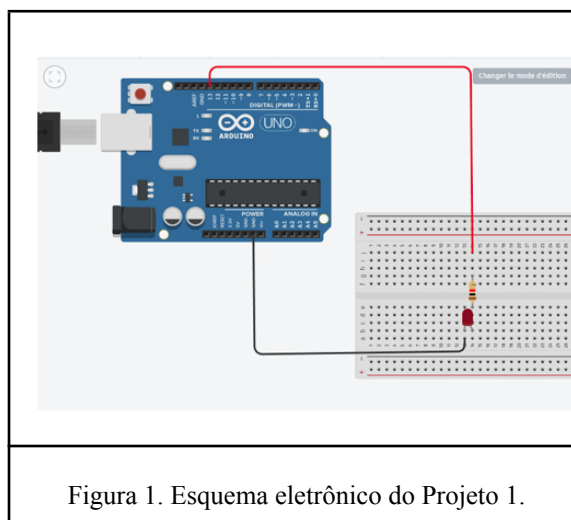


Figura 1. Esquema eletrônico do Projeto 1.

A figura 2 apresenta o código utilizado no Projeto 1. Este código foi escrito na linguagem do Arduino baseada em C/C++ utilizando a IDE Arduino IDE. Em códigos Arduino, a configuração dos pinos é realizada na função “`setup()`”, que serve para configurar os pinos de entrada e de saída do Arduino. Os pinos de entrada, são aqueles nos quais o Arduino recebe um sinal externo, e a partir deste sinal ele executa uma ação, e estes pinos são configurados como:

“`pinMode(pino, INPUT);`”

Enquanto que os pinos de saída do Arduino são configurados para enviar um sinal do microcontrolador para algum componente externo, como no acionamento de um LED. Os pinos de saída do Arduino são configurados como:

“`pinMode(pino, OUTPUT);`”

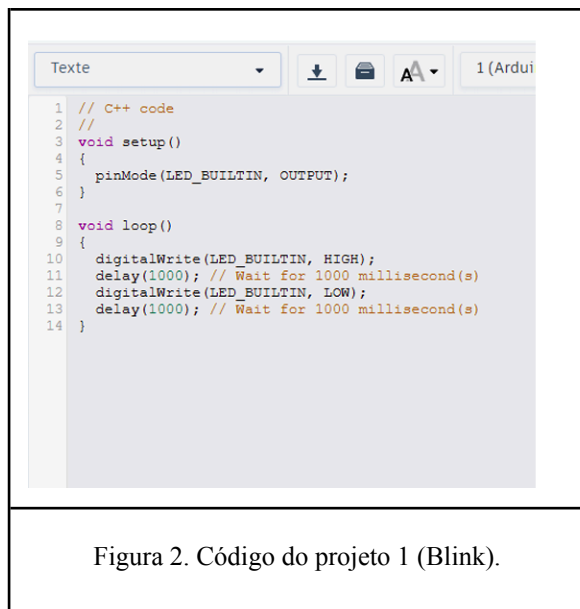
No código apresentado na figura 2, o pino 13 foi nomeado como `LED_BUILTIN`, e foi configurado como saída na função `setup()`. Os

códigos escritos dentro das chaves da função `setup()` são executados apenas uma vez. Após a função `setup()` encontra-se a função `loop()`, onde realmente é escrito as ações que devem ser executadas pelo Arduino, e os códigos dentro de `loop()` são executados em loop.

Para o código da figura 2, utilizou-se o comando “`digitalWrite(LED_BUILTIN, HIGH)`”, este comando serve para o Arduino acionar o pino `LED_BUILTIN` (pino 13) com 5V, ligando o LED. Em seguida, foi implementado o comando “`delay(1000)`”, que servirá para que o Arduino não execute nenhuma ação durante 1 segundo (1000 ms), fazendo com que o LED permaneça aceso durante este tempo.

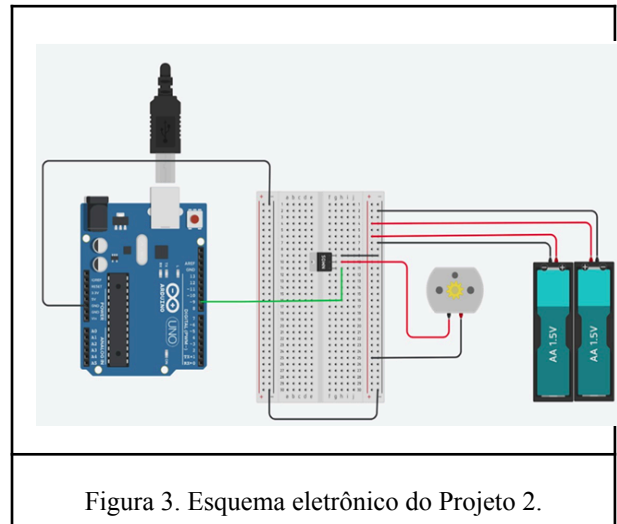
O código apresenta a o comando “`digitalWrite(LED_BUILTIN, LOW)`”, que vai desabilitar os 5v do pino `LED_BUILTIN`, desligando assim o LED.

Na próxima linha de código ocorre mais um delay de 1 segundo, para permanecer o LED desligado, e como esta é a última linha de código da função `loop()`, logo este bloco de código irá se repetir de novo, fazendo com que o LED ligue e apague durante a cada 1 segundo.



O segundo projeto substituiu o LED por um motor DC controlado por um transistor MOSFET, demonstrando o uso do transistor como chave eletrônica e a possibilidade de controlar atuadores de maior potência por meio de uma saída digital do Arduino. O código utilizado foi o mesmo do projeto

1, no qual o motor era acionado de forma autônoma, alternando entre ligado e desligado em intervalos programados de 1 segundo. Entretanto, ao longo da oficina foram feitos testes de intervalos maiores e menores para o acionamento do motor.



Conforme apresentado na figura 3, o circuito é semelhante com o circuito do LED, porém o LED foi substituído por um motor dc, e ocorreu a adição de um Metal-Oxide-Semiconductor Field Effect Transistor, (MOSFET) em português, Transistor de Efeito de Campo de Óxido de Metal. O MOSFET é um componente eletrônico que funciona como amplificador para pequenos sinais, ou uma chave para circuitos de corrente contínua (Sedra e Smith, 2004).

O MOSFET possui 3 terminais chamados de dreno, porta e fonte, em que ao aplicar uma tensão no terminal da porta, o MOSFET funciona como chave fechada, acionando o componente conectado ao transistor, como um motor, mas quando é interrompida a alimentação ao terminal porta do MOSFET, o transistor funciona como uma chave aberta, interrompendo a alimentação do motor.

Como mostrado no circuito da figura 3, o terminal porta do MOSFET está conectado ao pino 9 e a cada 1 segundo o pino `LED_BUILTIN` será ativado, e enviará 5V para o MOSFET, acionando o motor dc, e após passado 1 segundo, o pino `LED_BUILTIN` será desativado, fazendo com que o eixo do motor pare de girar, e este loop se repet..

O terceiro projeto incorporou ao circuito anterior um botão como elemento de entrada digital. Os participantes aprenderam a utilizar portas digitais como entrada, configurar o resistor de pull-up interno do Arduino e implementar uma lógica condicional para acionar o motor somente quando o botão fosse pressionado.

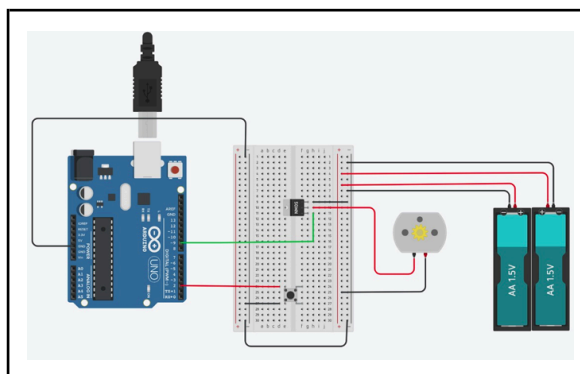


Figura 4. Esquema eletrônico do Projeto 3.

De acordo com o esquema eletrônico da Figura 4, o terceiro projeto incorporou ao circuito anterior um botão como elemento de entrada digital. Os participantes aprenderam a utilizar portas digitais como entrada, configurar o resistor de pull-up interno do Arduino e implementar uma lógica condicional para acionar o motor somente quando o botão fosse pressionado.

```

1 // Declaração dos pinos
2 const int MOTEPin = 9; // LED / motor / dispositivo de saída
3 const int butpin = 2; // Botão com resistor pull-up interno
4 int state;
5
6 void setup() {
7   pinMode(MOTEPin, OUTPUT);
8   pinMode(butpin, INPUT_PULLUP); // Mantém o pino em HIGH quando
9 }
10
11 void loop() {
12   // Leitura do botão
13   state = digitalRead(butpin);
14
15   // Lógica de acionamento:
16   // INPUT_PULLUP inverte a lógica:
17   // HIGH = não pressionado | LOW = pressionado
18   if (state == LOW) {
19     digitalWrite(MOTEPin, HIGH); // Botão pressionado: liga
20   } else {
21     digitalWrite(MOTEPin, LOW); // Botão solto: desliga
22   }
23 }
24
25
26

```

Figura 5. Código do projeto 3.

Conforme apresentado na Figura 4, o circuito mantém a estrutura do projeto anterior com o

MOSFET e o motor DC, mas inclui a adição de um botão (push button) conectado ao pino 2 do Arduino e ao barramento de GND da protoboard. O código utilizado no Projeto 3 (Figura 4) introduz a leitura de sinais externos para a tomada de decisão do microcontrolador. Primeiramente, na função setup(), o pino do motor (MOTEPin) é configurado como OUTPUT, enquanto o pino do botão (butpin) é configurado como INPUT_PULLUP. Essa configuração de pull-up interno é fundamental, pois garante que o pino receba um sinal de 5V (nível lógico alto) quando o botão estiver em repouso, evitando flutuações elétricas que poderiam causar acionamentos indevidos.

Na função loop(), o Arduino realiza a leitura constante do estado do botão através do comando digitalRead(butpin). A lógica condicional if (state == LOW) é implementada para verificar se o botão foi pressionado. Como o botão está conectado ao GND e configurado com pull-up, o pressionamento fecha o circuito e envia um sinal de nível baixo (LOW) ao pino 2. Quando essa condição é verdadeira, o comando digitalWrite(MOTEPin, HIGH) aciona o terminal porta do MOSFET, que atua como uma chave fechada e permite o funcionamento do motor. Caso o botão não esteja pressionado (else), o pino do motor permanece em LOW, mantendo o transistor como uma chave aberta e o motor desligado.

Todo o material de apoio utilizado na oficina foi organizado em uma apostila desenvolvida pelos próprios membros do PET Engenharias, disponibilizada aos participantes durante a atividade.

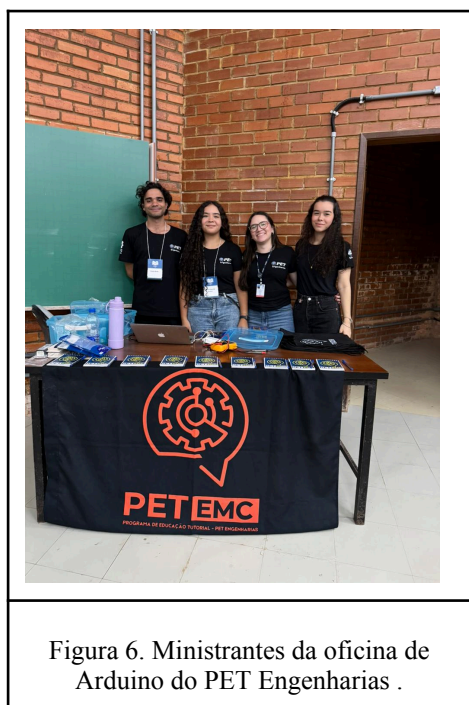


Figura 6. Ministrantes da oficina de Arduino do PET Engenharias .

que conteúdos ligados à eletrônica e microcontroladores atraem estudantes de outras áreas do conhecimento além da elétrica e computação, assim como podem ser importantes para estas outras áreas também.

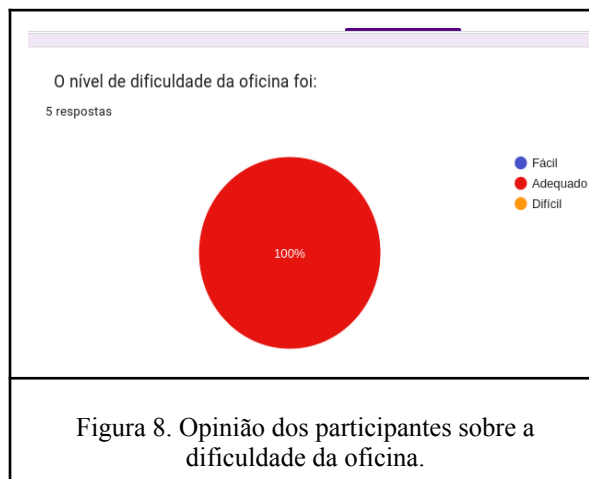


Figura 8. Opinião dos participantes sobre a dificuldade da oficina.

RESULTADOS E DISCUSSÃO

A oficina do PET Engenharias sobre introdução ao arduino contou com participantes de diversos cursos, e instituições diferentes, como mostrado no formulário de feedback abaixo.

Como mostrado na figura 8, 100% dos participantes consideraram o nível da oficina como “adequado”. Desta forma pode-se concluir que os conteúdos apresentados na oficina foram ideais para iniciantes em arduino, de forma que o conteúdo não foi nem extremamente básico, porém não foi avançado.

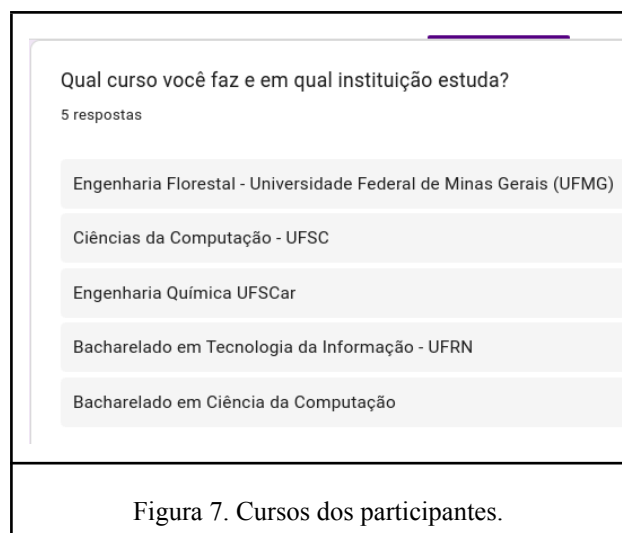


Figura 7. Cursos dos participantes.

De acordo com a figura 7, os cursos dos participantes na oficina foram: engenharia florestal, ciências da computação, engenharia química e tecnologia da informação. Mostrando desta forma

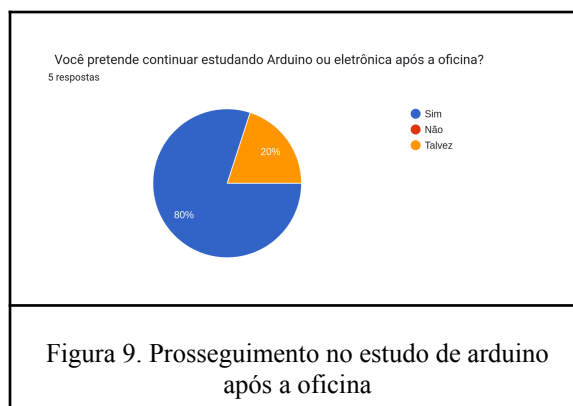


Figura 9. Prosseguimento no estudo de arduino após a oficina

De acordo com a figura 9, 80% dos participantes que responderam o formulário de feedback afirmaram continuar os estudos de arduino após a oficina, e 20% dos participantes afirmaram que talvez continuariam. Estes dados afirmam outra vez como conteúdos relacionados a microcontroladores e eletrônica básica são capazes de atrair pessoas de outras áreas do conhecimento, fora da elétrica e da computação. Além disso, é

possível inferir que tais conhecimentos podem ser aplicados de forma útil em diversas áreas diferentes.

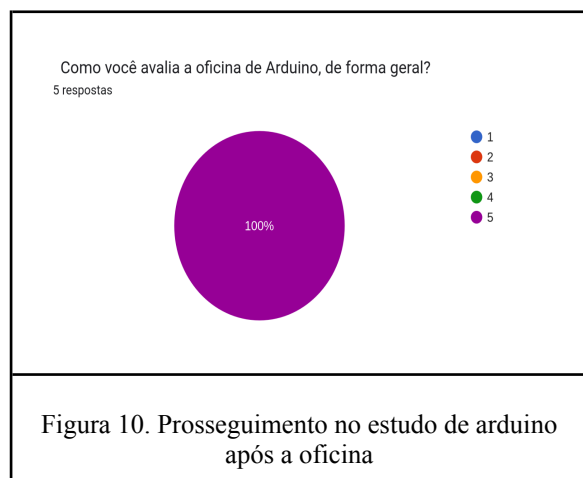


Figura 10. Prosseguimento no estudo de arduino após a oficina

Por fim, em uma escala de 1 a 5, 100% dos participantes que preencheram o formulário de feedback da oficina de arduino, avaliaram a oficina com a nota 5, mostrando uma percepção extremamente positiva do trabalho realizado pelos membros do PET Engenharias.

CONCLUSÃO

Para concluir, como mostrado pelas respostas do formulário de feedback dos participantes, o nível de dificuldade da oficina foi adequado, a maior parte dos participantes pretendem continuar os estudos em arduino e 100% dos que responderam o formulário de feedback, atribuíram a nota máxima à oficina. Desta forma pode-se concluir que o curso de arduino básico, atendeu os seus objetivos, disseminando assim conhecimentos de eletrônica e microcontroladores para estudantes de diversas áreas do conhecimento diferentes.

AGRADECIMENTOS

Agradecimentos ao MEC e ao Fundo Nacional de Desenvolvimento da Educação (FNDE), que graças ao custeio anual dos grupos PET, o PET Engenharias foi capaz de arcar com os custos da viagem e dos componentes necessários para a realização da oficina de Arduino para o XX ENAPET.

E agradecimentos ao PET Engenharias, por permitir a realização da oficina.

REFERÊNCIAS

- Monk, S. Programming Arduino: Getting Started with Sketches. McGraw-Hill, New York, 2012.
- Sedra, A. S.; Smith, K. C. Microelectronic Circuits. 5. ed. Oxford University Press, New York, 2004.
- UFG. PET Engenharias UFG. Universidade Federal de Goiás, Goiânia, 2026.
- Wing, J. M. Computational Thinking. Communications of the ACM, v. 49, n. 3, p. 33-35, 2006.