

# Capítulo 2

## Introdução ao Arduino

*Amado por muitos estudantes e hobbystas das áreas de eletrônica, computação e robótica, o Arduino vem ganhando cada vez mais adeptos. Mas afinal, o que é o Arduino e o que ele tem de tão especial? Como nasceu o Arduino? Para que serve um Arduino? Quais as vantagens? Como eu começo a programar? Nesse capítulo vamos apresentar um resumo sobre o que é Arduino e como ele pode ser utilizado em projetos microcontrolados.*

### 2.1 O que é Arduino?

Arduino é uma plataforma de prototipagem eletrônica de hardware livre e de placa única, desenvolvida a partir de 2005 por um grupo de 5 pesquisadores (Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis), projetada a partir de um microcontrolador AVR da Atmel com suporte de entrada/saída, uma linguagem de programação padrão, a qual tem sua origem na linguagem *Wiring* e é essencialmente C/C++, além de um ambiente de desenvolvimento integrado (*IDE - Integrated Development Environment*) que você pode instalar no seu computador Windows, Linux ou MAC. O objetivo do projeto é criar ferramentas que são acessíveis, com baixo custo, flexíveis e fáceis de se usar por profissionais das áreas técnica, artística, designers e também por estudantes, hobbystas e projetistas amadores, principalmente para aqueles que não teriam alcance aos controladores mais sofisticados e de ferramentas mais complexas e de preços mais elevados.

O Arduino é uma plataforma formada por dois componentes: A placa, que é o hardware básico usado para construir os projetos e o IDE Arduino, baixado gratuitamente pela internet, que é o software onde é escrito o que se quer que a placa faça. A figura 1 apresenta uma placa e a imagem do IDE do Arduino que será usado para o desenvolvimento e testes de sua programação.

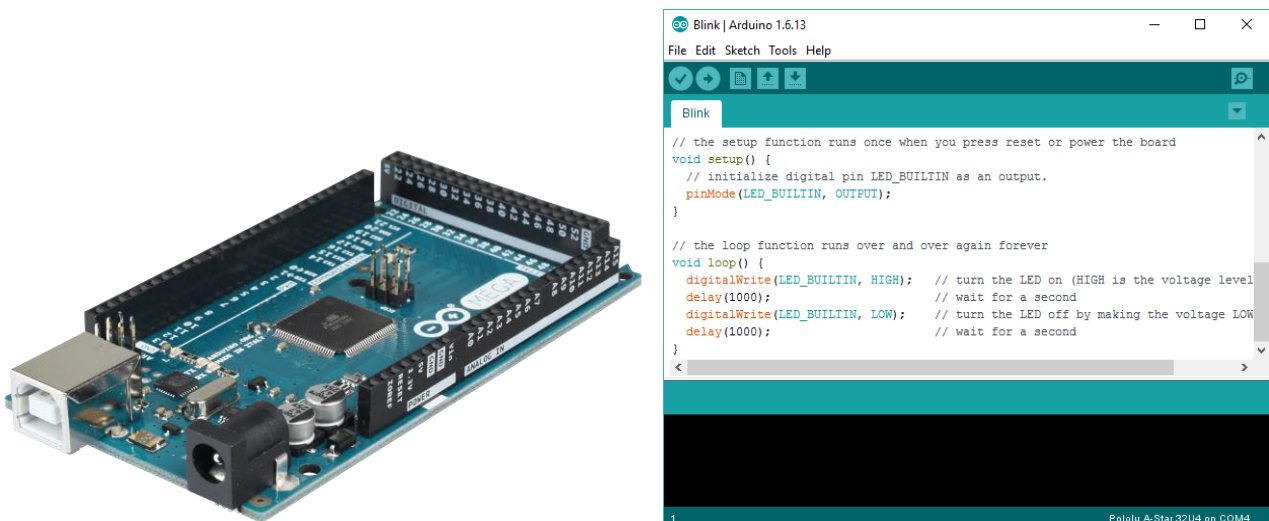


Figura 1 – Placa de prototipagem Arduino e ambiente de desenvolvimento integrado (IDE).

O Arduino pode ser usado para desenvolver projetos interativos, admitindo entradas de uma série de sensores e controlando uma variedade de atuadores como luzes, motores ou outras saídas físicas. Projetos feitos com o Arduino podem ser independentes, ou podem se comunicar com outros projetos e também com softwares rodando em outros sistemas computacionais, como desktops, notebooks, tablets, smartphones, etc.

Os campos de atuação do Arduino são vários, podendo ter aplicações na área de impressão 3D, robótica, engenharias, musical, ou até mesmo em ambiente escolar para o desenvolvimento de experimentos, otimizando assim o aprendizado dos alunos em áreas como física, química e matemática.

É possível implementar a automação de casas, automóveis, escritórios, criar um novo brinquedo, um novo equipamento ou melhorar um já existente. Tudo vai depender da criatividade. Por exemplo, um uso simples de um Arduino seria para acender uma luz por certo intervalo de tempo, digamos, 30 segundos, depois que um botão fosse pressionado. Nesse exemplo, o Arduino teria uma lâmpada e um botão conectados a ele. O circuito aguardaria pacientemente até que o botão fosse pressionado; uma vez pressionado o botão, ele acenderia a lâmpada e iniciaria a contagem. Depois de 30 segundos, apagaria a lâmpada e aguardaria um novo estímulo do botão. Essa configuração poderia ser utilizada para controlar uma lâmpada em um closet, por exemplo. Esse conceito poderia ser estendido pela conexão de um sensor, como um sensor de movimento, para acender a lâmpada quando uma pessoa entrasse no closet. Outros inúmeros projetos podem ser feitos com o Arduino, como um painel solar que se move de acordo com a incidência do sol, um jardim que manda uma mensagem no *Twitter* quando precisar ser regado, uma caixa de brinquedos que seria aberta por leitura de impressão digital, um robô espião sem fio, entre muitos outros.

## 2.2 Histórico

Anteriormente, para se confeccionar um circuito que interagisse com o ambiente ao seu redor, era necessário desenvolver projetos a partir do zero que servissem apenas a uma aplicação específica. Para se fazer pequenas alterações nas funcionalidades do circuito era necessário um estudo crítico, além de bastante trabalho.

Com o advento dos microcontroladores, foi possível que problemas que eram tratados com o desenvolvimento de hardware fossem tratados usando ferramentas de software. Dessa forma, um mesmo circuito poderia executar funções totalmente diferentes, bastando apenas reprogramá-lo e alterar alguns parâmetros do programa. Mesmo assim, trabalhar com microcontroladores não era algo tão trivial. Desta forma, surgiu o Arduino com a promessa de tornar o seu uso simples e acessível a qualquer um.

O projeto se iniciou na cidade de Ivrea, Itália, em 2005, com o intuito de interagir em projetos escolares de forma a ter um orçamento menor que outros sistemas de prototipagem disponíveis naquela época. Seu sucesso foi sinalizado com o recebimento de uma menção honrosa na categoria Comunidades Digitais em 2006, pela Prix Ars Electronica, além da marca de mais de 50.000 placas vendidas até outubro de 2008.

Atualmente, seu hardware é construído baseado em um microcontrolador Atmel AVR, sendo que este não é um requisito formal e pode ser estendido se tanto ele quanto a ferramenta alternativa suportarem a linguagem Arduino e forem aceitas por seu projeto. Considerando esta característica, muitos projetos paralelos se inspiram em cópias modificadas com placas de expansão, e acabam recebendo seus próprios nomes.

Apesar do sistema poder ser montado pelo próprio usuário, os mantenedores possuem um serviço de venda do produto pré-montado, através deles próprios e também por distribuidores oficiais com pontos de venda mundiais.

A maior vantagem dessa plataforma de desenvolvimento sobre as demais é a sua facilidade de utilização: pessoas que não são da área técnica podem aprender o básico e criar seus próprios projetos em um intervalo de tempo relativamente curto. Falando em termos práticos: O Arduino é hoje um pequeno computador que você pode programar para processar entradas e comandar saídas entre o dispositivo e os componentes externos conectados a ele, interagindo com o ambiente por meio de hardware e software.

## 2.3 A família Arduino

A família Arduino é composta de uma série de simpáticas plaquinhas, com diversos recursos de hardware, nascidas com a intenção de tornar mais fácil, barata e econômica a criação de dispositivos inteligentes capazes de interagir com o ambiente por meio de sensores e atuadores. Além disso, é adotado o conceito de hardware livre, o que significa que qualquer um pode montar, modificar, melhorar e personalizar o Arduino, partindo do mesmo hardware básico.

Cada placa dispõe de um microcontrolador Atmel AVR com componentes complementares para facilitar a programação e conexão com outros circuitos. Um importante aspecto é a disposição padrão dos conectores, permitindo a interligação com outros módulos de expansão, conhecidos como shields. Os Arduinos originais utilizam a série de chips megaAVR, especialmente os ATmega8, ATmega168, ATmega328, ATmega1280 e ATmega2560. A grande maioria das placas inclui um regulador linear de 5 volts e um oscilador de cristal de 16 MHz. As placas são pré-programadas com um software conhecido como bootloader, que simplifica o carregamento de programas para a memória flash embutida, em comparação com outras plataformas de desenvolvimento que geralmente demandam um chip ou placa programadora externa.

A figura 2 mostra as principais placas da família Arduino disponíveis no mercado.

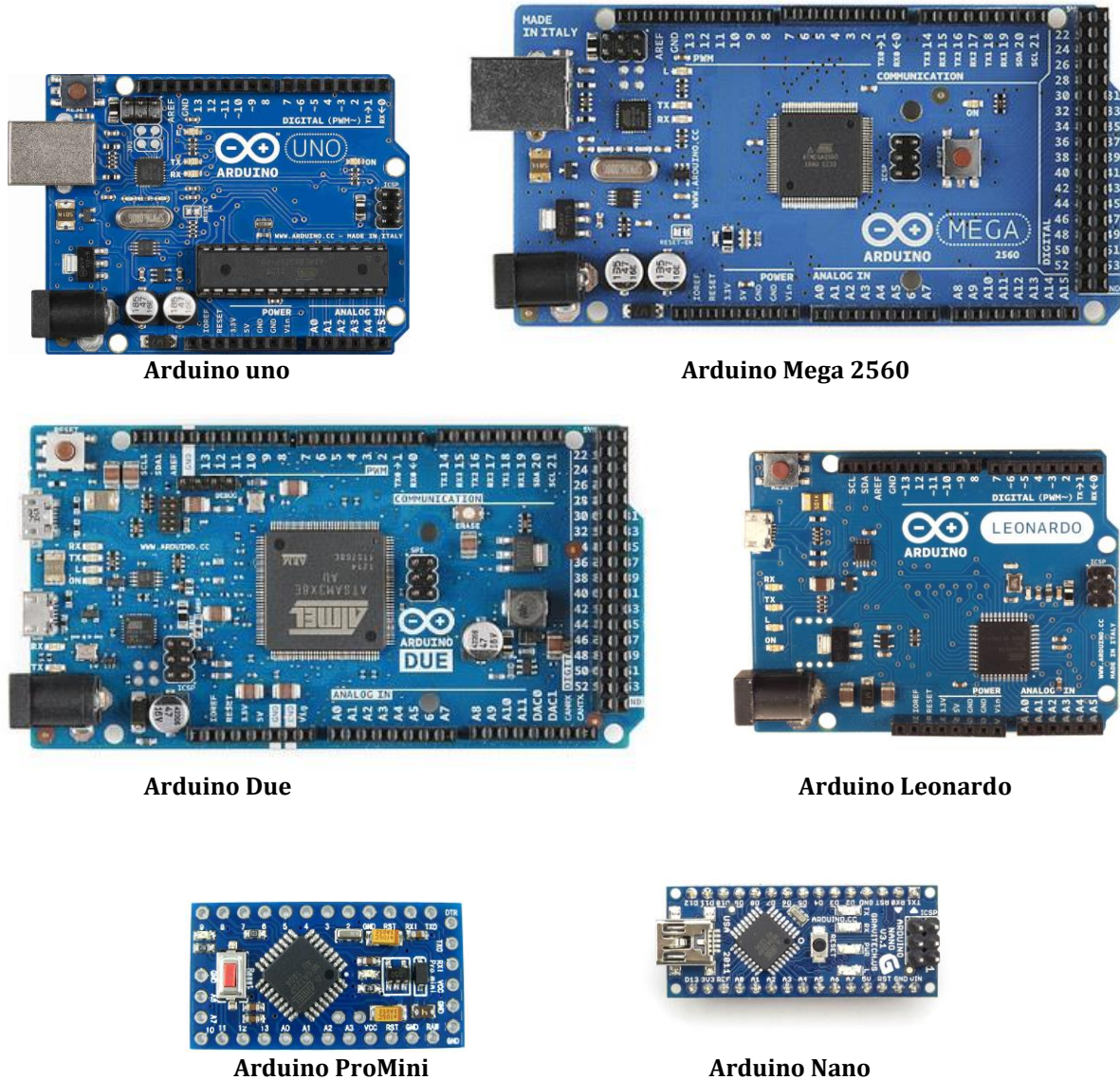


Figura 2 – Principais placas da família Arduino.

Para explicar as partes básicas de uma placa Arduino, utilizaremos uma das mais populares placas como exemplo, a Arduino Uno, representada na figura 3.

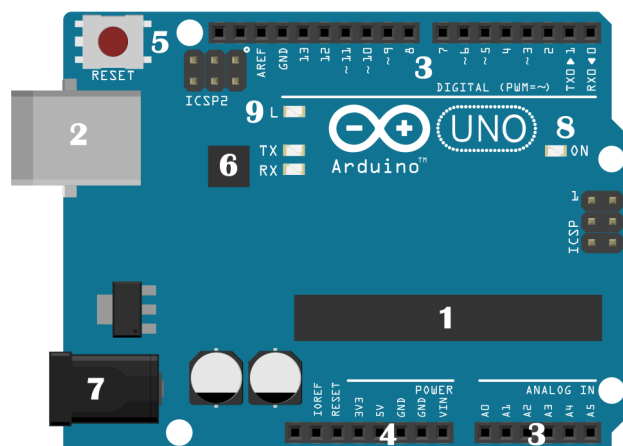


Figura 3 - Representação da placa Arduino Uno.

Na figura 3, as partes básicas da placa são:

- 1. Microcontrolador:** É o cérebro do Arduino. Este é o dispositivo programável que roda o código que é enviado à placa. No Arduino Uno, o microcontrolador ATmega328 é utilizado, dispondo de 32 kB de memória flash e 2 kB de memória SRAM (falaremos com mais detalhes sobre tipos de memória nos próximos capítulos). De maneira simplificada, a memória flash é o local no qual o programa será salvo, já a SRAM é a memória na qual os dados e variáveis serão salvas. A diferença básica entre esses dois tipos de memória é que a flash não perde seus dados caso o Arduino seja desligado ou reiniciado, o que não ocorre com a SRAM;
- 2. Conector USB:** Conecta o Arduino ao computador, além de alimentar a placa;
- 3. Pinos de Entrada e Saída:** Conjunto de pinos enumerados que podem ser individualmente programados para funcionarem como entradas ou saídas, fazendo com que o Arduino interaja com o meio externo;
- 4. Pinos de Alimentação:** Fornecem diversos valores de tensão que podem ser utilizados para energizar os componentes externos do seu projeto;
- 5. Botão de Reset:** Botão que reinicia o dispositivo;
- 6. Conversor Serial-USB e LEDs TX/RX:** O conversor Serial-USB é um chip que permite que o microcontrolador e o computador se comuniquem, visto que o microcontrolador ATmega328 não possui uma porta USB própria. No Arduino Uno, o microcontrolador Atmega16U2 é programado para agir como conversor. Os LEDs TX e RX acendem quando o Arduino está transmitindo e recebendo dados pela porta USB.
- 7. Conector de Alimentação:** Permite com que uma fonte alimente a placa. Caso o Arduino esteja sendo alimentado pela porta USB e por uma fonte externa, o hardware seletor escolherá automaticamente a fonte.
- 8. LED de Alimentação:** Indica quando a placa está energizada.
- 9. LED Interno:** LED conectado ao pino 13 que pode ser programado para sinalizar alguma operação do Arduino.

## 2.4 Shields

As placas Arduino não possuem recursos de hardware além daqueles necessários para operação básica do microcontrolador. Entretanto, elas possuem conectores que podem ser usados para combinar a capacidade de processamento do microcontrolador com outras placas que trazem consigo outros elementos de circuito, como sensores, controladores, módulos de comunicação, relés, displays, botões, teclados, entre outros, expandindo a capacidade criativa dos projetistas. Tais expansões são chamadas de *shields*, pois muitas dessas placas são conectadas sobre a placa básica do Arduino, formando uma espécie de “escudo” (tradução direta de shield), como mostrado na figura 4.

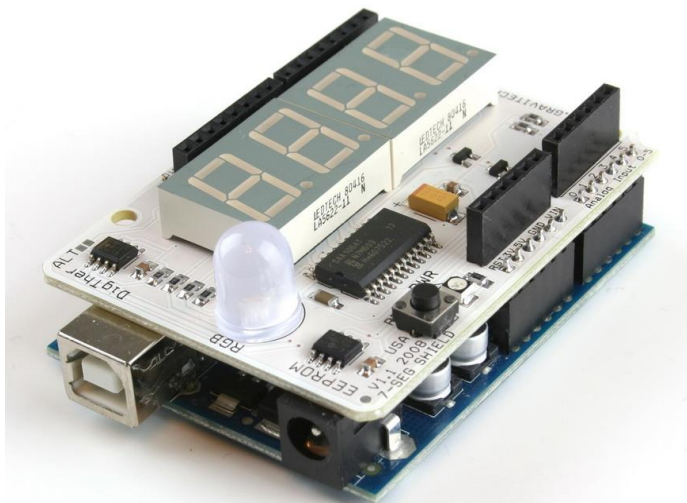


Figura 4 – Arduino com um shield de displays de 7 segmentos instalado sobre a placa.



Os shields podem ainda ser empilhados ou interligados com a placa do Arduino por meio de fios, chamados *jumpers*, como mostrado na figura 5, aumentando ainda mais as funcionalidades do Arduino.

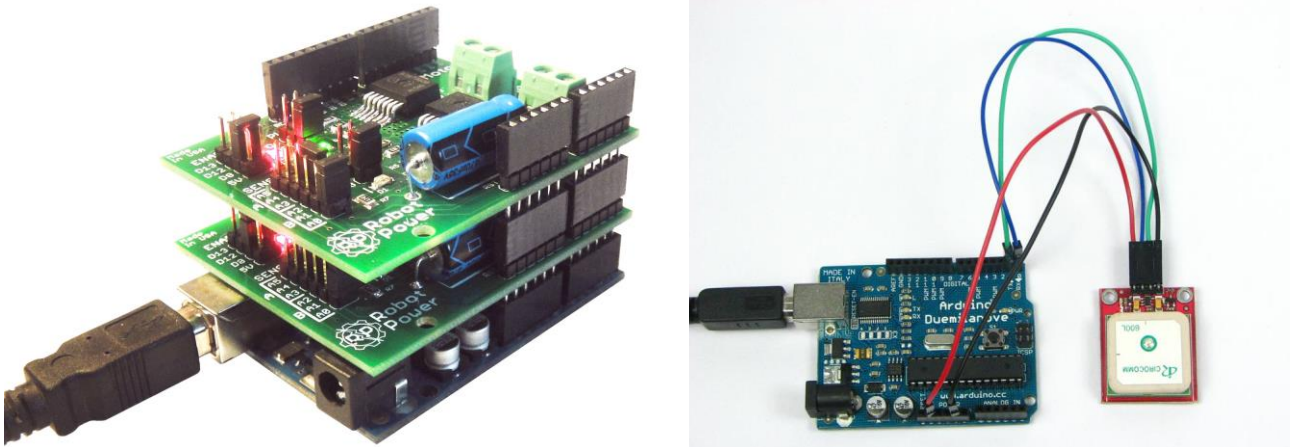


Figura 5 – Arduino com shields empilhados.

A figura 6 mostra um shield Ethernet acoplado ao Arduino Mega 2560. Ao mesmo tempo que permite o acesso do Arduino à uma rede ou até mesmo à internet, esse shield mantém os demais pinos disponíveis para utilização, assim é possível, por exemplo, utilizar os pinos para receber dados de temperatura e umidade de um ambiente, e consultar esses dados de qualquer lugar do planeta com acesso à internet.

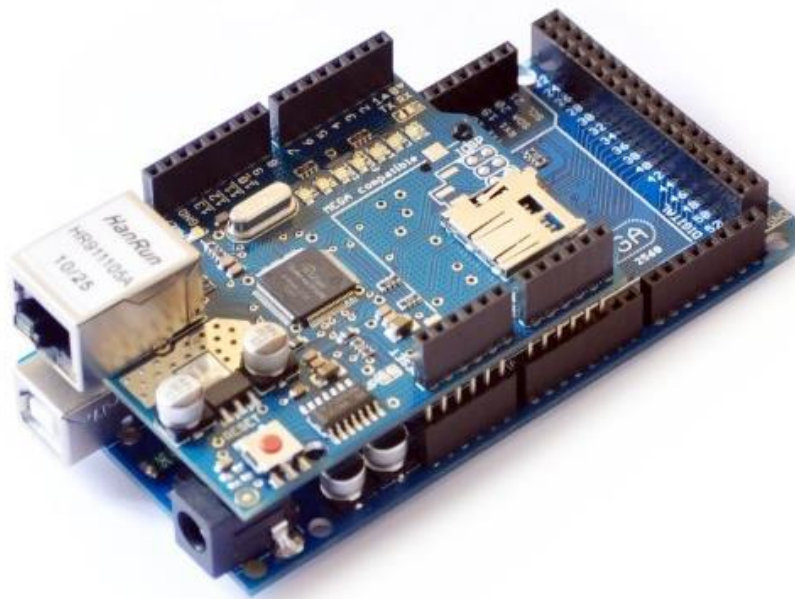


Figura 6 – Shield ethernet montado sobre o Arduino Mega 2560.

Um dos fatores determinantes para a enorme versatilidade e popularidade da plataforma Arduino são os shields. Mesmo com a enorme gama de shields já disponíveis no mercado, caso algum projeto precise de algo diferente, pode-se criar um shield que atenda as necessidades específicas desse projeto.

## 2.5 Programação do Arduino

Escrever um programa em Arduino é muito simples, bastando apenas conectar a placa Arduino ao computador por meio de um cabo USB e utilizar o ambiente de desenvolvimento integrado (IDE), onde o programa é digitado, submetido a testes para encontrar eventuais erros e transferido para o microcontrolador presente na placa. O download do IDE pode ser feito gratuitamente no site oficial do Arduino, acessado por ([www.arduino.cc](http://www.arduino.cc)).

Na figura 7 é mostrado o IDE, apresentando um programa básico já carregado.

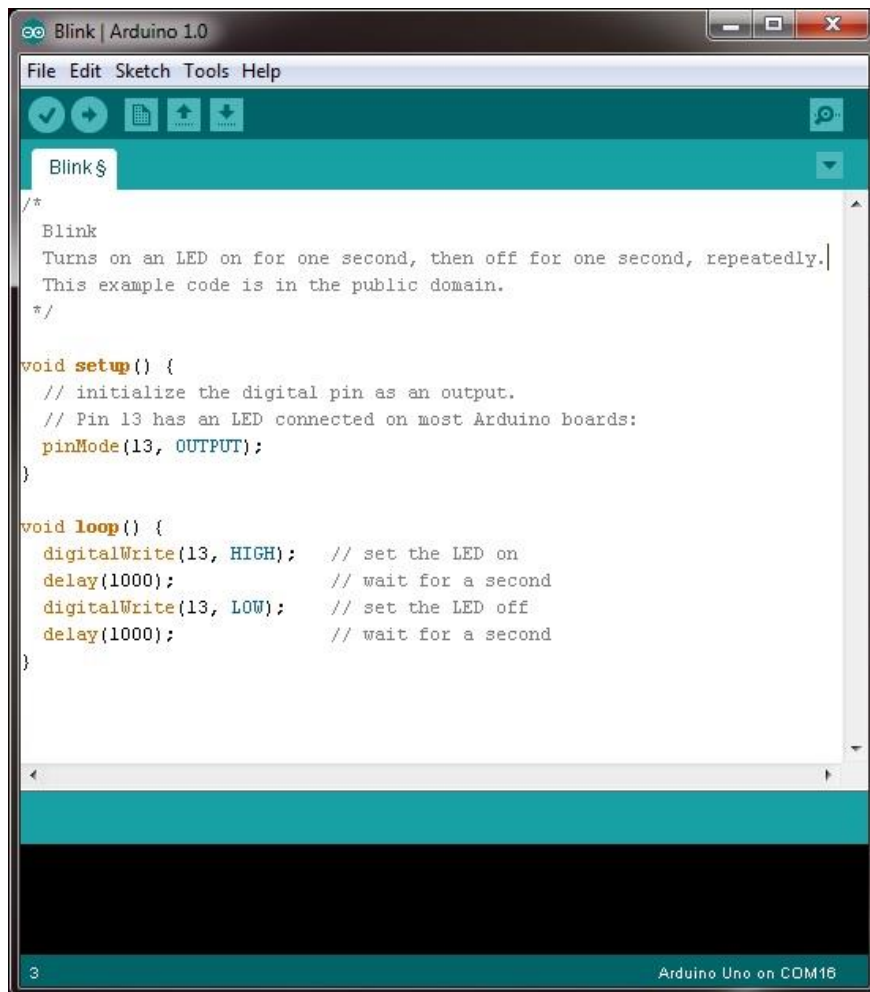


Figura 7 – IDE do Arduino com um programa carregado.

O IDE do Arduino é uma aplicação multiplataforma escrita em Java. É pensado para introduzir a programação a artistas e a pessoas não familiarizadas com o desenvolvimento de software. Inclui um editor de código com recursos de realce de sintaxe, parênteses correspondentes e indentação automática, sendo capaz de compilar e carregar programas para a placa com um único clique. Ele possui a capacidade de programar em C/C++, o que permite criar com facilidade muitas operações de entrada e saída, sendo necessário definir apenas duas funções para fazer um programa funcional:

**setup( )** – Inserida no início do programa, a qual pode ser usada para inicializar configurações, iniciar variáveis, informar se um pino será usado como entrada ou saída, etc.

**loop( )** – Chamada para repetir continuamente um bloco de comandos ou esperar até que a placa seja desligada.

Os detalhes da linguagem de programação Arduino serão explorados e detalhados no decorrer dos capítulos desse curso. ■