



UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO
DEPARTAMENTO DE CIÊNCIAS EXATAS E TECNOLÓGICAS

Wglierber Sousa Freitas

Automação residencial utilizando a plataforma de hardware livre Arduino

Vitória da Conquista
2016

Wglierber Sousa Freitas

Automação residencial utilizando a plataforma de hardware livre Arduino

Trabalho monográfico acadêmico apresentado como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação pela Universidade Estadual do Sudoeste da Bahia – UESB.

Orientador: Roque Mendes Prado Trindade

Vitória da Conquista
2016

Wglierber Sousa Freitas

Automação residencial utilizando a plataforma de hardware livre Arduino

Trabalho monográfico acadêmico apresentado como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação pela Universidade Estadual do Sudoeste da Bahia – UESB.

Vitória da Conquista, 16 de Janeiro de 2016:

Roque Mendes Prado Trindade
Orientador

Hélio Lopes Santos
Convidado 1

Alzira Ferreira da Silva
Convidado 2

Vitória da Conquista
2016

Resumo

Este trabalho trata da criação de um sistema de automação residencial fazendo uso de hardware livre. Muita coisa já existe no âmbito da automação, mas grande parte das tecnologias comumente usadas é pouco acessível. Com o objetivo de demonstrar que é possível utilizar tecnologias abertas e de baixo custo, foi construído um sistema que é capaz de controlar remotamente a iluminação e o portão de uma casa feita em pequena escala por meio do uso de uma placa Arduino, da tecnologia Bluetooth e de um aplicativo para dispositivo móvel. O resultado do trabalho mostrou que, graças às inúmeras possibilidades oferecidas pelo hardware livre, é possível a realização de projetos de formas mais simples e acessíveis.

Palavras-chave: automação residencial, Arduino, hardware livre, Bluetooth, dispositivo móvel.

Abstract

This work is about the creation of a home automation system using open source hardware. There is too much thing about automation, but most of the technology commonly used is not easily accessible. In order to demonstrate that it is possible to use open and low-cost technology, it has built a system that is able to remotely control the lights and the gate of a house built on a small scale by using a Arduino board, Bluetooth and a mobile app. The result of the study showed that, thanks to the numerous possibilities offered by the open source hardware, it's possible to build projects on an easy and accessible ways.

Key-words: home automation, Arduino, open source hardware, Bluetooth, mobile APP.

Lista de figuras

Figura 1 - Automação de sistema de iluminação	10
Figura 2 - Chip microcontrolador	14
Figura 3 - Microcontrolador AVR	15
Figura 4 - Placa Arduino	17
Figura 5 - IDE para o Arduino	18
Figura 6 - Placa clone do Arduino	19
Figura 7 – Rede Piconet	21
Figura 8 – Aparelhos com tecnologia Bluetooth.....	22
Figura 9 – Casa feita em pequena escala	23
Figura 10 - Aba Designer do APP Inventor	24
Figura 11 - Aba Blocks do APP Inventor	25
Figura 12 - Tela inicial do aplicativo	26
Figura 13 - Checkboxes marcados	27
Figura 14 - Mensagem alertando que o Bluetooth está desativado	28
Figura 15 - Correspondência entre ações e caractere enviado	29
Figura 16 - LED de alto brilho	30
Figura 17 - Resistor de 100 ohms	30
Figura 18 - Módulo Bluetooth HC-06	31
Figura 19 - Fios jumpers	31
Figura 20 – Protoboard	32
Figura 21 - Motor com polias e pião	33
Figura 22 - Cremalheira e trilhos	33
Figura 23 - Ponte H	34
Figura 24 - Sensores infravermelhos	35
Figura 25 - Pilhas ligadas em série	36

Lista de abreviaturas e siglas

CPU	Unidade de Processamento Central
PROM	Memória Programável de Somente Leitura
RAM	Memória de Acesso Randômico
I/O	Entrada e Saída
USART	Transmissor/Receptor Universal Síncrono e Assíncrono
USB	Barramento Serial Universal
ROM	Memória de Somente Leitura
IDE	Ambiente de Desenvolvimento Integrado
LED	Diodo Emissor de Luz
GPS	Sistema de Posicionamento Global
Mbps	Megabit por segundo
IBM	International Business Machines
SIG	Special Interest Group
ICM	Industrial Sientific and Medical
Kbps	Kilobyte por segundo
LPM	Link Manager Protocol
SDP	Service Discovery Protocol
PDA	Assistente Pessoal Digital
APP	Aplicativo
GND	Ground
DC	Corrente Contínua
v	Volts

Sumário

1 Introdução	8
1.1 Objetivo Geral	9
1.2 Objetivos Específico.....	9
2 Estado da arte	10
2.1 Automação Residencial	10
2.2 Microcontroladores	13
2.2.1 Microcontroladores AVR.....	15
2.3 O Arduino	16
2.4 A tecnologia Bluetooth	19
3 Desenvolvimento	23
3.1 Sistema Proposto	23
3.2 Ferramentas	23
3.2.1 MIT APP Inventor II	23
3.3 O Aplicativo	25
3.4 Controle de Iluminação	29
3.4.1 Descrição do Hardware.....	29
3.5 Controle do Portão	32
3.5.1 Descrição do Hardware	32
3.6 O Software	36
4 Conclusão	39
Referências	41
Apêndice A Código para o controle do sistema	43

1 Introdução

Assistimos hoje a uma revolução tecnológica sem precedentes. Computadores pessoais, smartphones e um incontável número de dispositivos eletrônicos estão cada dia mais presentes na vida das pessoas. Desde em objetos simples como um relógio, passando por eletrodomésticos, até automóveis, a automação, que pode ser descrita como técnicas aplicadas sobre um processo a fim de torna-lo mais eficiente, se faz presente de uma forma nunca antes vista.

Podemos por exemplo, receber informações até mesmo sobre nosso corpo através de notificações vindas de um aparelho celular com apenas um gesto que antes serviria apenas para olhar as horas. Entramos num automóvel e tudo o que precisamos fazer para ligá-lo é pressionar um botão e logo em seguida a esse ato, o motorista já é capaz de ter acesso a uma gama de informações como, por exemplo, consumo de combustível, temperatura do motor e data da próxima revisão. Juntamos a isso a possibilidade de interação de todos esses dispositivos que, conversando entre si, ampliam suas funcionalidades e sua capacidade de adaptação à novas aplicações. Tudo isso, visando tornar as atividades das pessoas mais acessíveis, mais fáceis e mais eficientes.

Dentro desse contexto, surgiu o termo “domótica” que em outras palavras pode ser descrito como automação residencial. Esse conceito pode ser entendido como o uso de dispositivos e técnicas de informática e eletrônica que são capazes de captar dados, armazenar e trabalhar sobre eles, transformando-os em informação útil. Na prática, significa tornar as atividades corriqueiras dentro de uma residência, mais “robotizadas” e, por consequência, mais eficientes e acessíveis.

Hoje em dia, com a popularização de dispositivos móveis, podemos carregar no bolso uma quantidade imensa de informação que é acessível com apenas alguns toques numa tela. Pensando nisso, foi que surgiu a motivação desse projeto de utilizar essa capacidade de controle da informação para criar o sistema de automação residencial utilizando o hardware Arduino, uma plataforma de hardware livre, que desde sua criação vem possibilitando o desenvolvimento de forma fácil de muitos projetos antes considerados difíceis de sair do papel, seja pelo alto custo ou

pela dificuldade técnica, juntamente com a tecnologia Bluetooth, que é uma das tecnologias de comunicação sem fio mais difundidas atualmente, e um aparelho celular. A combinação desses elementos será responsável por controlar a iluminação e o portão de uma casa feita em pequena escala.

1.1 Objetivo Geral

Desenvolver um sistema utilizando a plataforma de hardware Arduino, a tecnologia Bluetooth e um dispositivo móvel, que seja capaz de controlar o sistema de iluminação e o portão de uma casa feita em pequena escala.

1.2 Objetivos Específicos

Compreender e investigar os conceitos envolvidos nos sistemas de automação residencial, pesquisar como o Arduino pode ser utilizado em sistemas de automação, assimilar e realizar a interligação do hardware livre com a tecnologia Bluetooth e por último, construir um aplicativo para dispositivo móvel que possa se comunicar remotamente, por meio do Bluetooth, com o Arduino.

2 Estado da arte

2.1 Automação Residencial

Também conhecida como domótica, a automação residencial se consiste na capacidade dos equipamentos que compõem uma residência como, por exemplo, os equipamentos de iluminação, funcionarem de maneira que siga instruções pré-estabelecidas por um sistema de controle integrado. Como explicam Muratori e Dal Bó (2008):

(...) o principal fator que define uma instalação residencial automatizada é a integração entre os sistemas aliada à capacidade de executar funções e comandos mediante instruções programáveis. A integração deve abranger todos os sistemas tecnológicos da residência (...)

De acordo com Azevedo e Pizzolato (2004), o sistema que pode ser chamado de domótico é aquele que forma uma rede de comunicação capaz de interconectar uma série de dispositivos, equipamentos e outros sistemas que, em conjunto são capazes de colher informações sobre um ambiente residencial, tomar decisões e realizar ações com o objetivo de gerenciá-lo. O sistema de automação residencial (Figura 1) quando bem integrado com sistemas como o de iluminação, possibilitam a potencialização e otimização de aplicações, gerando mais conforto, comodidade e acessibilidade.

Figura 1 – Automação de sistema de iluminação



Fonte: Site Kickante

No fim da década de 70, sistemas de ar condicionado, ventilação e aquecimento foram os primeiros sistemas a serem controlados por sistemas eletrônicos. Chips de computadores fizeram com que esses sistemas pudessem ser gerenciados por meio de sensores permitindo respostas mais precisas a alterações climáticas. Porém nessa época ainda não havia integração entre os sistemas. Apenas nos anos 80 é que apareceram sistemas de segurança e iluminação proporcionando a coordenação entre os componentes do mesmo sistema. Nesta década foi que nasceu nos Estados Unidos o conceito de “edifício inteligente”, que naquela época era entendido como aquele que utiliza tecnologia para reduzir custos operacionais, eliminar desperdícios e criar uma estrutura que tendesse a aumentar a produtividade dos usuários. (Neves, 2002).

Segundo Nascimento et Al (2005) as pessoas hoje buscam ter em suas residências, em seu local de trabalho e em outros lugares que frequentam, como por exemplo lojas e supermercados, uma condição de conforto e bem-estar. Para atender essa demanda, é crescente a indústria de materiais de construção, de utilidades domésticas, equipamentos de escritório em geral e de sistemas de informação. No que diz respeito ao conforto, os autores explicam que é uma questão subjetiva e depende de fatores ligados às expectativas de cada indivíduo e que existem vários tipos de conforto como, por exemplo, pode-se citar o conforto térmico que diz respeito à temperatura do ambiente; o luminoso que se relaciona ao equilíbrio entre a iluminação natural e a artificial e o acústico que busca um ambiente com o mínimo de influência de ruído externo. Os sistemas de automação tem ajudado o homem a alcançar esses estados de conforto. Os autores consideram os principais objetivos de um sistema de automação predial como sendo:

- . Centralizar informações inerentes ao funcionamento dos sistemas vitais às operações realizadas dentro de um edifício;
- . Executar o controle lógico para o devido funcionamento automático dos equipamentos;
- . Redução de custos bem como economia de energia, através do uso mais racional de recursos;

- . Comunicação do estado dos sistemas através de interface homem-máquina, antecipando problemas e tomadas de decisão;
- . Fornecer a possibilidade da programação preventiva dos diversos equipamentos.
- . Aumento da segurança das instalações, graças à imediata detecção de situações atípicas e imprevistas, bem como a agilização das providências necessárias para solucioná-las.

Além dos benefícios funcionais citados acima, a integração de entre tecnologia e os serviços realizados dentro de uma residência se faz cada vez mais presente de forma a buscar uma melhor qualidade de vida. Trazendo diversos benefícios para os moradores como conforto, economia, prevenção de acidentes e de falhas nos equipamentos.

Segundo Lipovetsky (2005), a modernidade ganha outro significado, deixando de lado o requinte e se voltando mais para o que o autor denomina “*luxo emocional*”, onde as pessoas passam a valorizar aquilo que lhes oferece prazer. “Vivemos numa sociedade de consumo alimentada pela autonomia do indivíduo. Ela valoriza o bem-estar, as necessidades individuais.”

A habitação é considerada um dos bens de consumo de maior importância para o ser humano por ser capaz de lhe oferecer as necessidades básicas como segurança e bem estar. De acordo com Dias (2004), a domótica, por meio de seu conjunto multidisciplinar de aplicações, bem integrada às residências, é capaz de aumentar a qualidade de vida de quem nelas habita.

Nesse contexto, destaca-se também a questão das necessidades que surgem quando as pessoas se tornam idosas. Além do conforto, estas por sua vez veem a tecnologia como uma auxiliar de grande importância para a realização de tarefas básicas dentro de sua residência. Debert (1999) enfatiza a importância da tecnologia pra que essas pessoas superem as barreiras para se tornarem independentes:

[...] pesquisas recentes têm mostrado que a universalização da Seguridade Social, as melhorias nas condições de saúde e outros avanços tecnológicos, tais como nos meios de comunicação, elevadores, automóveis, entre outros, podem estar sugerindo que viver só, para os idosos, representa mais formas inovadoras e bem

sucedidas de envelhecimento do que de abandono, descaso e/ou solidão.

A automação residencial está conduzindo a questão da habitação á uma verdadeira revolução, remodelando conceitos e fazendo surgir novas idéias. Como diz Wernek (1999):

Depois de o público conhecer uma residência automatizada, não haverá como retroceder, toda a cadeia de concepção da moradia, (a arquitetura construção etc.), evoluirá, e, principalmente, o ocupante do imóvel. Assim, deverão ser necessários vários profissionais que, interagindo, permitirão o real desenvolvimento das técnicas da domótica.

São muitos os benefícios que a automação pode proporcionar. Como diz Nunes (2008), diversas medidas podem ser tomadas com o objetivo de reduzir o consumo e o custo da energia e atingir um balanço positivo entre o gasto e o conforto. Considerando como exemplo a iluminação residencial, ela é um dos fatores que mais interfere na conta de energia, principalmente se as lâmpadas utilizadas não forem de modelos econômicos. Com a automação residencial é possível fazer um monitoramento. Por exemplo, verificar quais lâmpadas estão acesas e apagar aquelas de lugares que estão vazios. O sistema também pode exercer um controle mais elaborado como regular a intensidade de luz em ambientes que precisem de mais ou menos. Utilidade esta que pode funcionar com eficiência em salas de TV ou em ambientes expostos à luz natural.

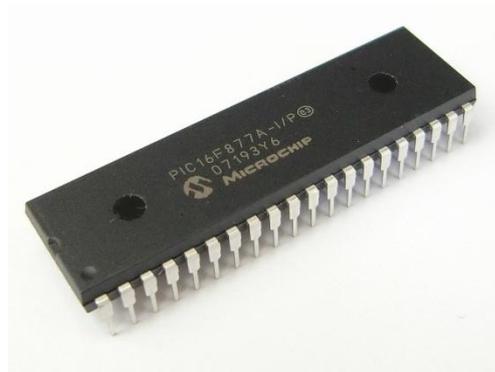
Para que tudo isso seja possível, é necessário o uso de equipamentos eletrônicos e que haja interação entre eles. Eles variam desde um pequeno microcontrolador, passando por dispositivos Bluetooth, até telefones celulares e computadores pessoais.

2.2 Microcontroladores

Podemos definir um microcontrolador, mostrado na Figura 2, como sendo um computador confeccionado dentro de um circuito integrado, possuindo unidade central de processamento, memória e esquema de entrada e saída. De acordo com Silveira (2011), o termo microcontrolador é usado para descrever um sistema

mínimo que inclui uma CPU, memória e circuitos de entrada e saída, tudo montando num único circuito integrado, que pode ter de 8 a até mais de 100 pinos. Alguns microcontroladores podem vir com contadores decimais internos, conversores analógico-digitais, comparadores de tensão e circuitos de comunicação serial, tudo embutido no mesmo encapsulamento.

Figura 2 – Chip microcontrolador



Fonte: Site Soluções Industriais

Como explicam Penido e Trindade (2013), um microcontrolador pode ser descrito como um computador em um único chip que, por sua vez, contém um processador - a Unidade Lógica Aritmética, memória, periféricos de entrada e saída, dispositivos de comunicação serial e temporizadores. Eles surgiram como uma evolução natural dos circuitos digitais quando esses começaram a aumentar a sua complexidade. Chegou a um ponto que se concluiu que seria mais barato e mais simples substituir a lógica das portas digitais por um conjunto formado por processador e software. Os dispositivos presentes em um microcontrolador são:

1. CPU (Unidade de Processamento Central), que tem como finalidade interpretar as instruções do programa.
2. Memória PROM (Memória Programável Somente de Leitura), onde são gravadas as instruções do programa.
3. Memória RAM (Memória de Acesso Randômico), que armazena as variáveis utilizadas pelo programa.
4. Conjunto de I/O (Entrada e Saída), que serve pra controlar dispositivos externos e receber impulsos de sensores.

5. Dispositivos de funcionamento auxiliares, que são o gerador de clock, contadores e USART (Transmissor/Receptor Universal Síncrono e Assíncrono) para comunicação.

2.2.1 Microcontroladores AVR

Os microcontroladores AVR, mostrado na Figura 3, que são fabricados pela Microchip Technology são populares no mercado graças a seu baixo custo, grande base de usuários, grande disponibilidade, grande número de ferramentas disponíveis - e gratuitas – para o desenvolvimento de aplicações e capacidade de programação serial. Os microcontroladores da linha AVR podem ser usados para a realização de um vasto número de tarefas como controle de linha de produção e são encontrados em grande parte dos dispositivos eletrônicos existentes, desde celulares até sistemas de alarme.

Figura 3: Microcontrolador AVR



Fonte: Site Solda Fria

Como diz Lima (2009), os microcontroladores AVR apresentam ótima eficiência de processamento e núcleo compacto. Com uma estrutura avançada, apresentam mais de uma centena de instruções e arquitetura voltada à programação C, o que permite produzir códigos compactos. Ainda segundo o autor, algumas das principais características desse tipo de microcontrolador que o tornam bastante popular são:

- . Eles são capazes de executar poderosas instruções em um único ciclo de clock.

- . Possuem alta integração e grande número de periféricos além de efetiva compatibilidade com toda a família AVR.
- . Possuem módulos de redução do consumo de energia e demais características para sistemas críticos.
- . Possui conjunto completo e gratuito de softwares para desenvolvimento.
- . Seu preço é acessível.

2.3 O Arduino

O Arduino (Figura 4) é uma placa criada na Itália utilizada como plataforma de prototipagem eletrônica que torna o mundo da robótica mais acessível às pessoas. Iniciado em 2005, a ideia base era que ele fosse utilizado como apoio educacional em aplicações escolares. Ele é constituído basicamente por um microcontrolador Atmel AVR de 8 bits, pinos analógicos e digitais de entrada e saída e entrada USB (para comunicação com computador). A placa possui uma alimentação externa de no mínimo 7 e no máximo 35 v com amperagem mínima de 300 mA. Embutido no Arduino existe um firmware, que mescla memória ROM para leitura e um programa gravado nesse tipo de memória, carregado na memória da placa e compatível com os sistemas operacionais Windows, Linux e Mac OS X.

O software pode ser desenvolvido em linguagem C/C++ com a utilização de uma interface gráfica (IDE) fornecida gratuitamente no site oficial. O Arduino possui ainda uma grande variedade de placas periféricas (os chamados Shields) que são vendidos separadamente e conectados através dos pinos digitais e analógicos. Cada um possui uma finalidade específica, tudo depende do que queira se desenvolver. A combinação dessas Shields, que vão desde sensores de temperatura a interface de rede, é que capacita o usuário a criar uma imensa variedade de projetos. (SOARES; KARLA, 2013).

Figura 4: Placa Arduino



Fonte: Site Wikipedia

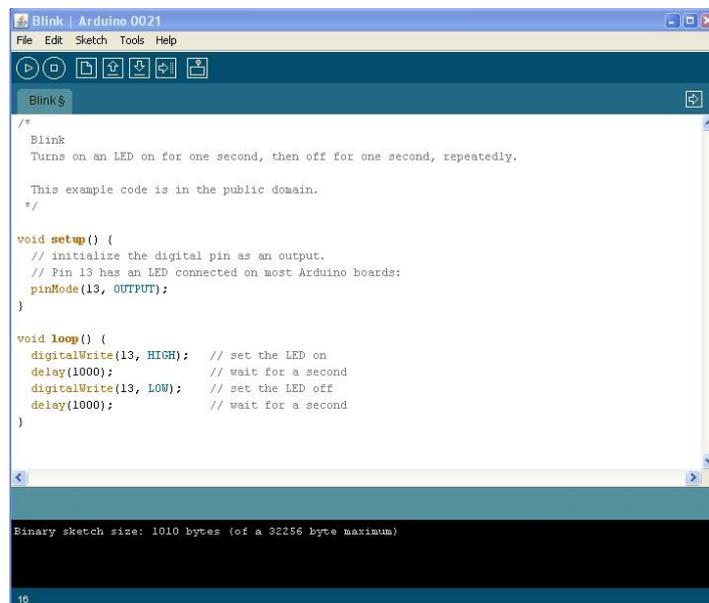
De outra maneira WARREN et al. (2011). descrevem o Arduino como sendo uma pequena central de comandos que permanece aguardando ordens. Com poucas linhas de código você pode fazê-lo ligar ou desligar uma lâmpada, ler um valor através de um sensor e mostrar esse valor através de uma tela. Por causa da sua versatilidade e do grande suporte disponível nas comunidades online, ele tem atraído pessoas que nunca nem mesmo tiveram contato com um microcontrolador. Segundo o autor, a ideia básica é a criação de uma atmosfera onde qualquer pessoa que esteja interessada em desenvolver projetos possa contribuir nem que seja de forma mínima.

Como explica McRoberts (2010), o Arduino pode ser usado no desenvolvimento de objetos interativos independentes, pode ser conectado a um computador, a uma rede e até mesmo à internet para receber e enviar dados, bem como atuar sobre eles. Como exemplo, ele pode receber dados através de sensores e enviá-los para um site na forma de um gráfico. Além disso, ele pode ser conectado a LEDs, displays (mostradores) de matriz de pontos, botões, interruptores, motores, sensores de temperatura, sensores de pressão, sensores de distância, receptores GPS, módulos Ethernet e vários outros dispositivos que emitam dados e possam ser controlados. Desde o projeto Arduino foi iniciado no ano de 2005, mais de 150.000 placas já foram vendidas em todo mundo. Sem falar nas placas não oficiais que certamente superam esse número. A sua popularidade não para de crescer, pois cada vez mais pessoas descobrem o incrível potencial dessa maravilhosa iniciativa de fonte aberta para a realização de projetos de maneira rápida e fácil. Segundo o

autor, a maior vantagem do Arduino sobre as demais plataformas de desenvolvimento baseadas em microcontroladores, é justamente a facilidade da sua utilização. Pessoas que nem mesmo são da área técnica podem, de maneira rápida, aprender o básico e criar seus próprios projetos. Além disso, justamente pelo fato de ele estar se tornando cada vez mais popular, existe uma enorme quantidade de pessoas compartilhando códigos e diagramas para que outras pessoas os usem livremente.

Para programar o Arduino, se faz necessário o uso de uma IDE, como mostrada na Figura 5, que é o software onde se escreve os códigos na linguagem “entendida” pela placa, que por sua vez é baseada na linguagem C++. Tal IDE permite que se escreva a sequência de instruções passo a passo e se faça o upload para o Arduino que executará tais instruções interagindo com o que estiver conectado a ele. Os programas são comumente chamados de sketches.

Figura 5: IDE para o Arduino

The image shows a screenshot of the Arduino IDE interface. The window title is "Blink | Arduino 0021". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for running, saving, and other functions. The main text area contains the following code:

```
/*  
Blink  
Turns on an LED on for one second, then off for one second, repeatedly.  
  
This example code is in the public domain.  
*/  
  
void setup() {  
  // initialize the digital pin as an output.  
  // Pin 13 has an LED connected on most Arduino boards:  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH); // set the LED on  
  delay(1000);           // wait for a second  
  digitalWrite(13, LOW); // set the LED off  
  delay(1000);          // wait for a second  
}
```

At the bottom of the window, a status bar indicates "Binary sketch size: 1010 bytes (of a 32256 byte maximum)" and the page number "16".

Fonte: Site Hobbytronics

De forma mais técnica, RENNA et al. (2013) define o Arduino com sendo uma plataforma de computação física. Um sistema digital ligado a sensores e atuadores que permitem construir sistemas que percebam a realidade e respondam com ações físicas. Feito sobre uma placa microcontroladora com acesso de entrada e saída I/O,

sobre a qual foram criadas bibliotecas e funções no intuito de simplificar a sua programação por meio de uma sintaxe similar às linguagens C e C++. Em suma, ele pode ser visto como uma unidade de processamento capaz de mensurar variáveis do ambiente externo e transformá-las em um sinal elétrico correspondente através dos sensores ligados aos seus terminais de entrada. De posse dessa informação, ele a processa computacionalmente, ainda podendo atuar no controle ou acionamento de algum outro elemento eletro-eletrônico conectado aos terminais de saída.

Tanto o hardware quanto o software do Arduino são de fonte aberta. O que quer dizer que o código, os esquemas e os projetos em geral podem ser utilizados livremente por qualquer pessoa e com qualquer propósito. Conseqüentemente surgiram várias placas-clone e outras baseadas no projeto original. Um exemplo de uma dessas clones feita pela empresa brasileira Robocore é mostrado na Figura 6.

Figura 6: Placa clone do Arduino



Fonte: Site Robocore

2.4 A Tecnologia Bluetooth

O Bluetooth é uma tecnologia capaz de transmitir dados e voz em half-duplex (quando no sistema temos um transmissor e um receptor e ambos podem tanto transmitir quanto receber dados) numa taxa de transmissão que chega a 1 Mbps

sem a necessidade do uso de fios entre dispositivos eletrônicos. Com o uso do Bluetooth num sistema de automação residencial, a implementação se torna mais simples e com custo reduzido. (LEE; CHOI, 2003).

Seu início ocorreu em 1994 quando a empresa Ericsson buscava alternativas para diminuir o número de fios entre celulares e acessórios. Ela precisava de uma tecnologia que fosse, sobretudo, barata, para que o preço de seus dispositivos não precisasse sofrer um aumento considerável e que consumisse pouca energia, visando não diminuir a durabilidade da bateria dos mesmos. No ano de 1998, ela se juntou com a IBM, Intel, Nokia e Toshiba para formar o Bluetooth SIG (Bluetooth Special Interest Group), criando um padrão universal e aberto para a comunicação sem fio entre dispositivos. Justamente por causa da participação de várias empresas na criação desse padrão, o Bluetooth passou a ter uma função muito importante que foi a de integrar dispositivos de diversas marcas. Foi aí que começou a ideia de dispositivos móveis (celulares, laptops, dispositivos de áudio) de diversas companhias se comunicarem e se integrarem de maneira rápida e fácil. (LACERRA, 2008).

De acordo com KOAYASHI (2004), o Bluetooth é uma especificação industrial para a comunicação em curta distância entre dispositivos sem uso de qualquer tipo de cabo. Tudo isso com baixo custo e alta operabilidade. Cada dispositivo Bluetooth tem embutido nele um chip que é um microrádio. Os sinais transmitidos por esse rádio alcançam distâncias curtas (tipicamente de até 10 metros). O protocolo opera numa faixa de licença livre chamada de ISM (Industrial Scientific, and Medical) em 2,45 gigahertz e pode alcançar velocidades de até 723,1 kbit/s. Para evitar interferência com outros protocolos que utilizam a mesma frequência, o Bluetooth divide a faixa em 79 canais e intercala o uso dos canais até 1600 vezes por segundo. A arquitetura funciona baseada em cinco camadas da seguinte forma:

- . Rádio: especifica detalhes da interface, ou seja, a frequência, o salto de canais, o esquema de modulação e o alcance de transmissão.

- . Baseband: estabelece a conexão dentro de uma rede, promove endereçamento, formato de pacote, sincronização e controle de energia.

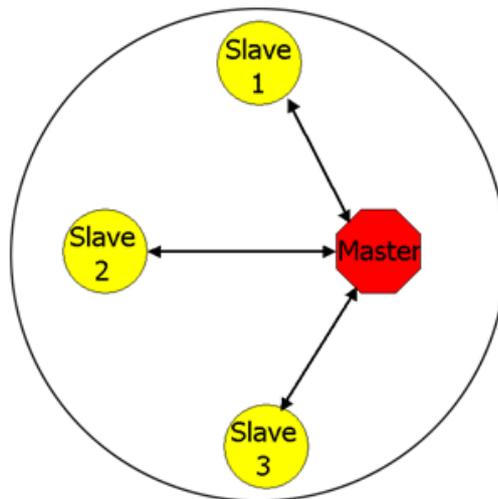
- .Link manager protocol (LPM): faz o setup entre dispositivos e gerencia requisições externas.

.Logical link control and adaptation protocol (L2CAP): Adapta os protocolos da camada LPM à camada baseband.

. Service discovery protocol (SDP): gerencia as informações do dispositivo e fornece serviços para a comunicação entre dois dispositivos Bluetooth.

Os dispositivos Bluetooth conectam-se através de piconets, conforme ilustrado na Figura 8, que são redes formadas por um dispositivo mestre conectado a um ou mais escravos. Eles podem se encontrar em quatro estados de operação: mestre, em que estão no controle do piconet; escravo ativo, onde estão conectados e participando ativamente da rede; escravo passivo, em que continuam participando da rede, mas em modo de baixa prioridade; e em espera, caso que não está conectado a nenhum piconet, está apenas aguardando pedidos ocasionais de outros dispositivos.

Figura 8: Rede Piconet



Fonte: Site EE Times

A tecnologia Bluetooth está se tornando cada dia mais aparente quando se fala em dispositivos eletrônicos de comunicação. De celulares até automóveis, podemos notar o uso dessa tecnologia como uma das principais alternativas no que diz respeito à comunicação sem fio entre dispositivos. Hoje em dia, podemos com alguns toques de tela, trocar arquivos com outras pessoas através do smartphone ou até mesmo atender uma ligação enquanto dirigimos graças à possibilidade de comunicação sem fio proporcionada pelo Bluetooth.

O Bluetooth é flexível o suficiente para ser usado em quase tudo e qualquer pessoa pode criar uma rede pessoal em casa ou na rua com dispositivos compatíveis com essa tecnologia, sejam esses dispositivos teclados, mouses, scanner, PDAs, laptops, fones de ouvido, celulares, como ilustrado na Figura 9. O SIG está trabalhando constantemente em ritmo acelerado para poder prover melhorias e atender ao feedback dos fabricantes. Além de pesquisas relacionadas a aspectos como velocidade, segurança e ruído no sinal, existe um esforço contínuo para o desenvolvimento de novos perfis o que garantirá o aumento das possibilidades de uso e de que o Bluetooth permaneça competitivo frente às outras tecnologias do gênero. (KOAYASHI, 2004).

Figura 9: Aparelhos com tecnologia Bluetooth



Fonte: Site GTA/UFRJ

3 Desenvolvimento

3.1 Sistema Proposto

O sistema a ser desenvolvido se consiste em uma casa em pequena escala, mostrada na Figura 9, em que o sistema de iluminação e o de abertura e fechamento do portão serão automatizados. Em outras palavras, esses sistemas serão controlados remotamente via Bluetooth pelo usuário através de um aplicativo instalado em um telefone celular.

Figura 9: Casa feita em pequena escala



Fonte: O autor

3.2 Ferramentas

3.2.1 MIT APP Inventor II

Para o desenvolvimento da aplicação móvel usada para controlar o sistema, foi utilizado o MIT APP Inventor II, que é uma ferramenta gratuita que oferece uma introdução ao desenvolvimento de aplicativos para a plataforma móvel Android. Com ela, é possível desenvolver tais aplicativos sem a necessidade de conhecimento técnico em linguagem de programação. Ele substitui esse método (visto pela maioria das pessoas como complexo e demorado) por uma simples interface visual onde as tarefas são basicamente arrastar botões e blocos lógicos que são equivalentes aos

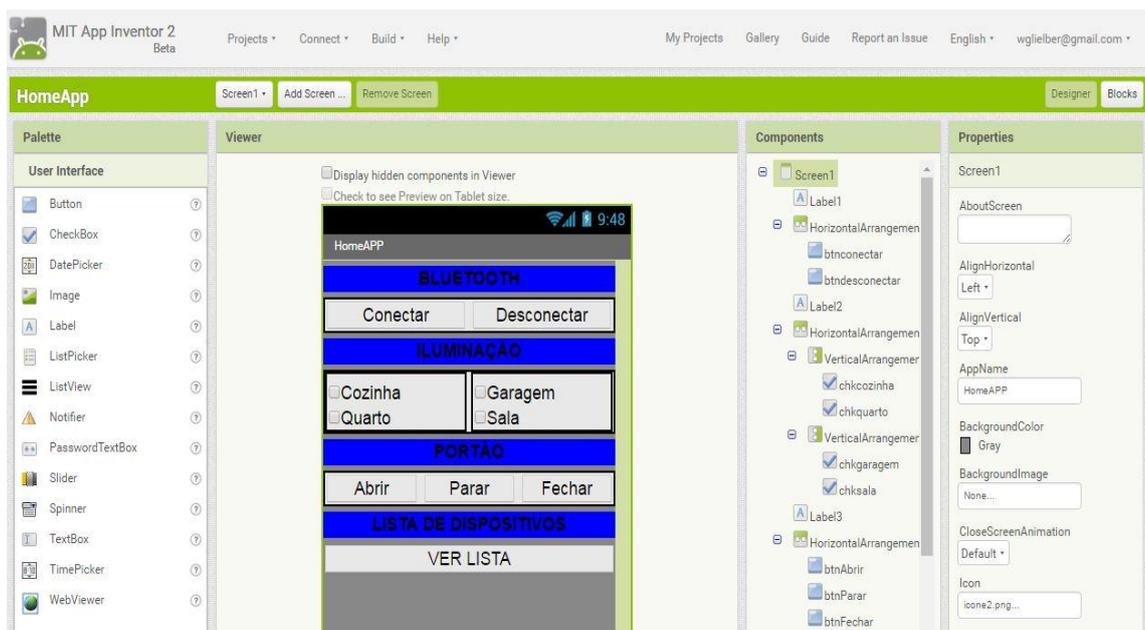
blocos de funções da linguagem de programação. Com isso, um usuário comum que não tenha o conhecimento técnico exigido para a construção de aplicativos utilizando linguagens de programação, pode desenvolver seus projetos apenas dedicando algumas horas de estudo.

Para fazer uso da ferramenta é necessário apenas que se tenha uma conta de e-mail do Google, não é necessário nenhum tipo de download ou de instalação, o que facilita a sua portabilidade. Através dessa conta, se faz o acesso à página do APP Inventor e a partir daí, já se pode criar um novo projeto que ficará salvo na própria conta do usuário.

A ferramenta possui duas áreas principais para o desenvolvimento. São elas a parte “Designer” e a parte “Blocks”. O usuário pode alternar entre elas usando os botões na parte superior direita.

Na parte “Designer”, mostrada na Figura 10, é onde são feitos os componentes visuais do aplicativo a ser desenvolvido. São eles: botões, caixas de texto, imagens de fundo, títulos, entre outros. Também é nessa parte que são definidas as disposições e tamanhos desses componentes na tela. Ou seja, é onde será definido tudo que será apresentado visualmente ao usuário do aplicativo.

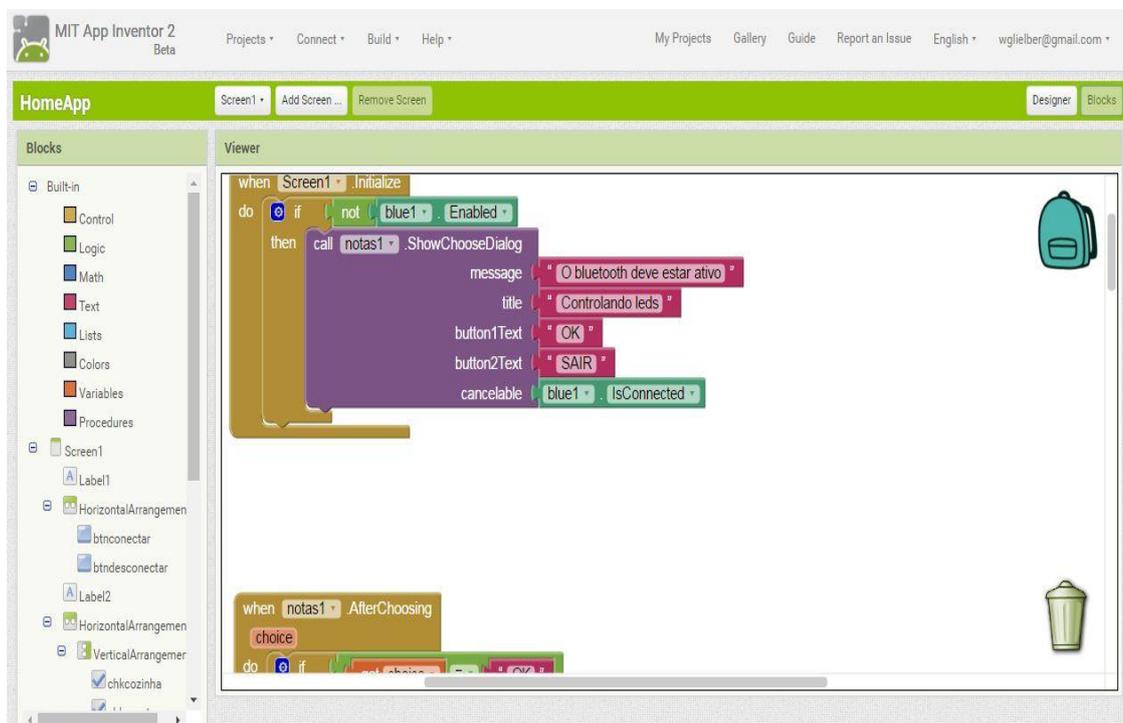
Figura 10: Aba Designer do APP Inventor



Fonte: O autor

Na parte “Blocks”, mostrada na Figura 11, é onde se define a funcionalidade do projeto. É nela que são criados os blocos funcionais que irão gerar um programa que executará as funções desejadas pelo usuário. Para tal, são usados blocos de função que são como peças de um quebra-cabeça que inclui variáveis, texto, estruturas de controle, blocos lógicos, procedimentos, funções matemáticas, listas etc. Tudo isso tendo como base a interface gráfica já criada na parte “Designer”.

Figura 11: Aba Blocks do APP Inventor



Fonte: O autor

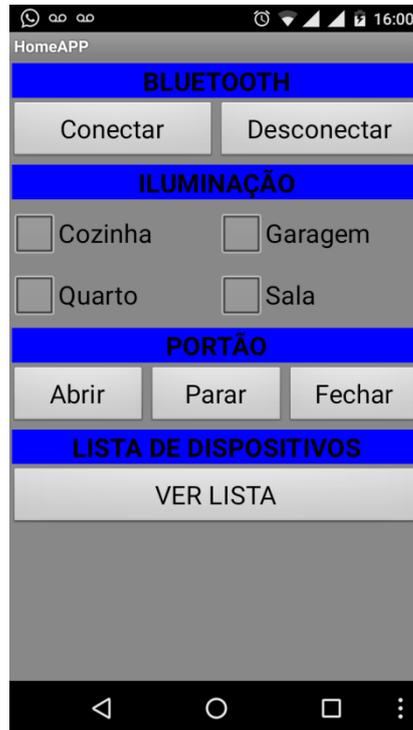
3.3 O Aplicativo

Para que a iluminação da casa e o portão pudessem ser controlados remotamente foi criado um aplicativo para smartphone, nomeado por “HomeAPP”, que é capaz de enviar sinal Bluetooth. Para tal, foi usada a ferramenta APP Inventor II, descrita anteriormente.

Todas as funções do aplicativo estão disponíveis em uma única tela, que é justamente a tela inicial, mostrada na Figura. 12. A tela é subdividida em quatro

blocos: bluetooth, iluminação, portão e lista de dispositivos. A separação foi feita de acordo com a função que cada botão dentro dos blocos desempenha.

Figura 12: Tela inicial do aplicativo



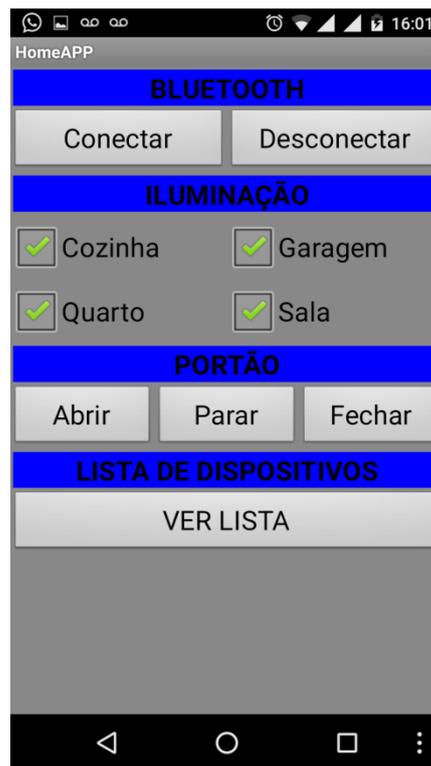
Fonte: O autor

No bloco "Bluetooth", existem dois botões nomeados com "Conectar" e "Desconectar". Quando o usuário aciona o botão "Conectar", o aplicativo irá mostrar a lista de dispositivos Bluetooth que já estão pareados (que já foram previamente conectados pela função nativa do sistema operacional do celular) com o aparelho, então basta escolher um desses dispositivos tocando na tela e a conexão será estabelecida. Por outro lado, quando o usuário acionar o botão "Desconectar" a conexão é imediatamente desfeita com o dispositivo.

O bloco "Iluminação" ao invés de possuir botões, possui quatro checkboxes, que são caixas de marcação que desempenham funções diferentes dependendo se estiverem marcadas ou desmarcadas. Ao todo são quatro checkboxes, cada uma representando um cômodo da casa. São eles: cozinha, quarto, garagem e sala. Quando o usuário marca uma checkbox tocando em cima, ele fica sinalizado em

verde como mostrado na Figura 13, e os leds responsáveis pela iluminação do respectivo cômodo são acesos e quando a checkbox é desmarcada eles são apagados.

Figura 13: Checkboxes marcados



Fonte: O autor

O bloco “Portão” possui três botões, que são “Abrir”, “Parar” e “Fechar”. Quando acionado, o botão “Abrir” fará com que o motor do portão gire em um sentido, permitindo a abertura do mesmo. O botão “Parar” interrompe essa rotação e o botão “Fechar” rotaciona o motor no sentido contrário fazendo com que o portão feche.

Por fim, existe o bloco “Lista de Dispositivos”, que possui apenas o botão “Ver Lista”. Ao ser pressionado pelo usuário, ele mostrará uma tela com os dispositivos que estão conectados no momento.

O aplicativo possui mensagens que auxiliam o usuário no uso de suas funções. Ao abrir o aplicativo, se o Bluetooth do dispositivo não estiver ligado, uma

notificação “O Bluetooth deve estar ativo”, como mostrado na Figura 14, aparecerá na tela e caso ele tente acionar algum botão na tela sem que tenha ativado, a mensagem “Ative o Bluetooth” ficará centralizada até que este seja ativado. Com o Bluetooth ligado, ao acionar o botão “Desconectar”, além de a conexão ser desfeita, será mostrada a mensagem “Dispositivo desconectado”.

Figura 14: Mensagem alertando que o Bluetooth está desativado



Fonte: O autor

Para que seja feita sua interação com o Arduino, é necessário que sejam enviados dados para que o mesmo interprete. No caso desse aplicativo os dados que são enviados são caracteres que se diferenciam de acordo com o botão ou checkbox pressionado, como é descrito na Figura 15.

Figura 15: Correspondência entre ações e caractere enviado

Botão/checkbox	Caractere enviado
“Cozinha” marcado	‘A’
“Cozinha” desmarcado	‘a’
“Quarto” marcado	‘B’
“Quarto” desmarcado	‘b’
“Garagem” marcado	‘C’
“Garagem” desmarcado	‘c’
“Sala” marcado	‘D’
“Sala” desmarcado	‘d’
“Abrir”	‘E’
“Parar”	‘F’
“Fechar”	‘G’

Fonte: O autor

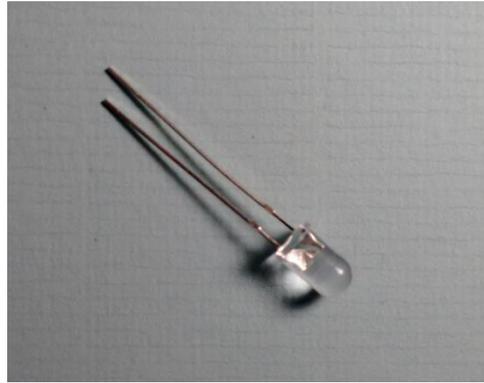
3.4 Controle de Iluminação

O controle da iluminação da casa é feito de forma automatizada em que o usuário poderá acionar (ligar ou desligar) as lâmpadas dos quatro cômodos remotamente através do aplicativo do celular.

3.4.1 Descrição do Hardware

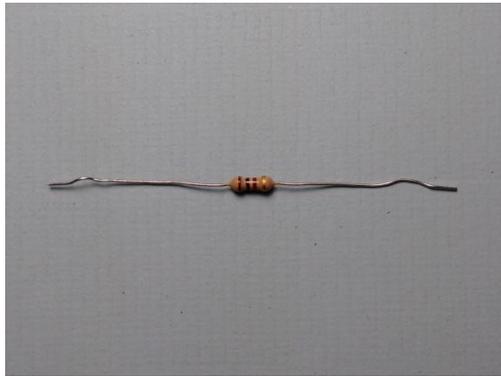
Para a iluminação dos cômodos foram utilizadas 12 lâmpadas de LED de alto brilho, como a mostrada na Figura 16, que foram divididas em quatro para cada ambiente (garagem, sala, cozinha e quarto). Cada quadra de LEDs estão ligadas a um resistor de 100 ohms (Figura 17). O resistor é um componente eletrônico que, no caso desse projeto, oferece resistência à passagem de corrente elétrica dentro de um circuito, evitando dessa forma que componente (no caso os LEDs) sejam danificados devido à alta corrente que sai diretamente da fonte.

Figura 16: LED de alto brilho



Fonte: O autor

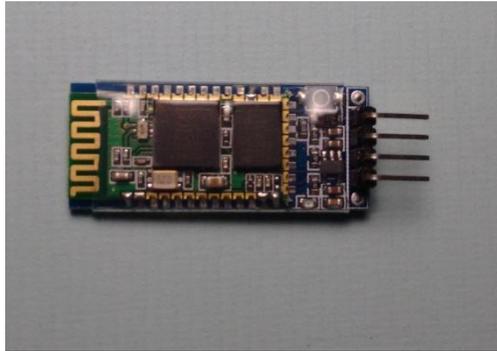
Figura 17: Resistor de 100 ohms



Fonte: O autor

Para o recebimento dos dados vindos do APP, foi utilizado uma placa Arduino Uno e um módulo HC-06, mostrado na Figura 18. O HC-06 é um módulo bluetooth que pode ser conectado ao Arduino para receber dados de um dispositivo que também utilize a tecnologia bluetooth.

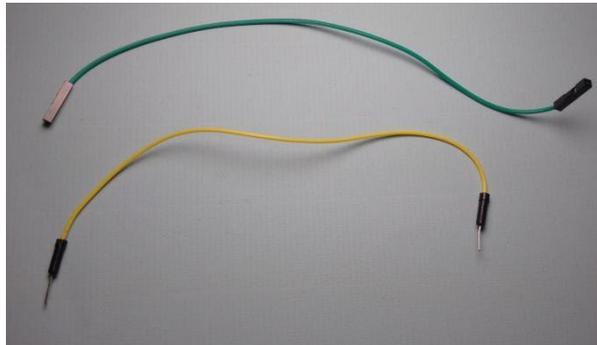
Figura 18: Módulo Bluetooth HC-06



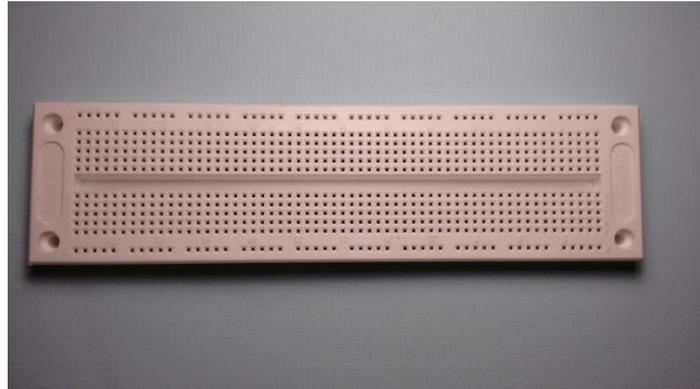
Fonte: O autor

A conexão entre esses componentes principais é feita utilizando fios jumpers de cobre, como os da Figura 19, e uma protoboard (Figura. 20), que é uma placa repleta de furos e conexões condutoras utilizada em montagem de circuitos elétricos experimentais. Sua maior vantagem é a facilidade de inserção de componentes já que não é necessário que seja feito qualquer tipo de solda.

Figura 19: Fios jumpers



Fonte: O autor

Figura 20: Protoboard**Fonte: O autor**

A montagem dos componentes foi feita da seguinte forma: cada quadra de LEDs é ligada por um polo aos pinos digitais 7, 10, 11 e 12 do Arduino e o outro polo, ao passar pelo resistor é ligado ao GND (terra). O módulo Bluetooth conectado à saída de 5 v e ao GND do Arduino e às portas digitais RX e TX. Estas portas são responsáveis pela a comunicação serial do Arduino, ou seja, elas traduzem o sinal para comunicação com o USB do computador. Através dessas portas também, é que o módulo bluetooth irá se comunicar com o Arduino, enviando para ele os dados recebidos do celular. Todos os componentes dessa etapa são alimentados pela mesma fonte de energia de 5 v que é responsável por ligar o Arduino.

3.5 Controle do Portão

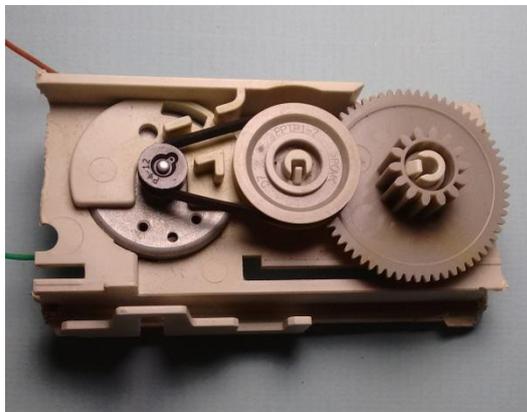
O controle do portão da casa é feito de forma automática onde o usuário pode executar as funções de abrir, parar e fechar através dos três respectivos botões na tela principal do aplicativo do celular.

3.5.1 Descrição do Hardware

Para que o portão seja controlado, foi utilizado o mesmo Arduino, a mesma protoboard e o mesmo módulo Bluetooth descritos na seção 3.4.1, bem como fios jumpers semelhantes para a interligação dos componentes. Para simular o sistema de portão eletrônico de uma casa real, foram utilizados componentes de um drive de DVD de desktop, mais especificamente os componentes presentes no mecanismo

de abertura e fechamento da tampa onde se coloca a mídia. São eles: um motor DC (corrente contínua) - que faz parte de uma classe de motores que transforma diretamente uma corrente elétrica em energia mecânica (rotação) - e o sistema que vem acoplado sobre ele, composto por uma correia entre duas polias que conectam o motor ao pião de rotação, como se pode ver na Figura 21. Esse conjunto terá a função de movimentar o portão.

Figura 21: Motor com polias e pião



Fonte: O autor

Também foram retirados do drive uma cremalheira, que serve para conectar o portão ao pião do motor e os trilhos sobre os quais ele desliza, todos mostrados na Figura 22. Para simular o portão foi utilizada uma folha de papel cartão por ser ao mesmo tempo leve (o que facilita sua tração pelo motor) e rígida.

Figura 22: Cremalheira e trilhos

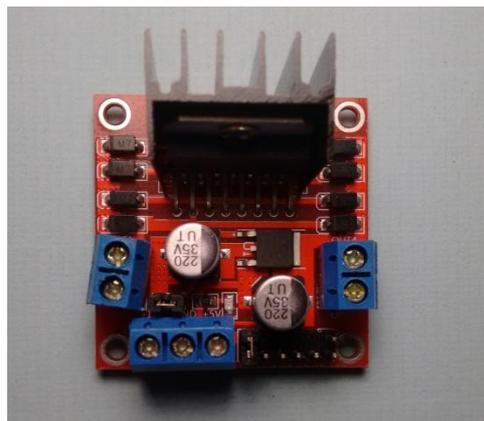


Fonte: O autor

Para que se estabeleça uma comunicação entre o Arduino e o motor utilizado para o portão foi utilizada uma ponte H, mostrada na Figura 23, que é uma placa contendo um circuito simplificado capaz de controlar motores de corrente contínua diretamente a partir de sinais elétricos, ou seja, é possível controlar tanto o sentido de rotação do motor, quanto a sua velocidade e potência. A ponte H possui duas entradas de energia, uma para cargas de até 5 v e outra para até 12 v; possui quatro saídas de corrente (OUT1, OUT2, OUT3 e OUT4), que são capazes de controlar até dois motores e possui seis entradas (FNA, FNB, IN1, IN2, IN3, IN4). As entradas FNA e FNB recebem dados do Arduino para que seja controlada a velocidade de rotação dos motores. Já as outras entradas, atuando aos pares, recebem informação para que seja controlado o sentido de rotação do motor.

De acordo com a informação passada através dessas entradas, a passagem de corrente nas saídas “OUT” pode ser alternada entre dois sentidos ou ser nula, o que dá controle sobre o motor. O uso dessa ponte H é fundamental nesse projeto, pois é necessário que o portão se movimente em sentidos diferentes dependendo da necessidade (abrir ou fechar) e com velocidade pré-definida.

Figura 23: Ponte H

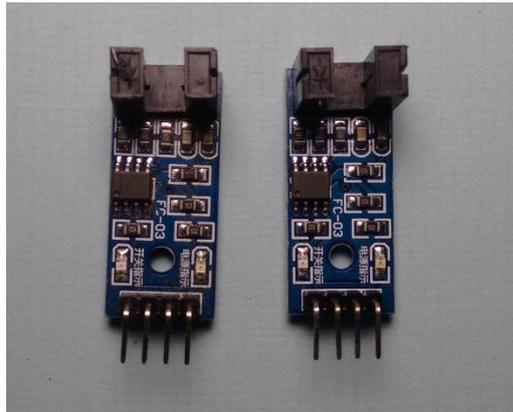


Fonte: O autor

Os últimos componentes utilizados foram dois módulos sensores infravermelhos para Arduino, como se pode ver na Figura 24, que são módulos compostos por um emissor e um receptor de luz infravermelha e são capazes de enviar a informação da interrupção da passagem dessa luz devido à presença de um

objeto ao Arduino. Esses sensores são usados nesse projeto como uma condição de parada para o portão. Quando este estiver aberto ou fechado por completo, o sensor é acionado e a informação é passada para que seja possível interromper automaticamente a rotação do motor.

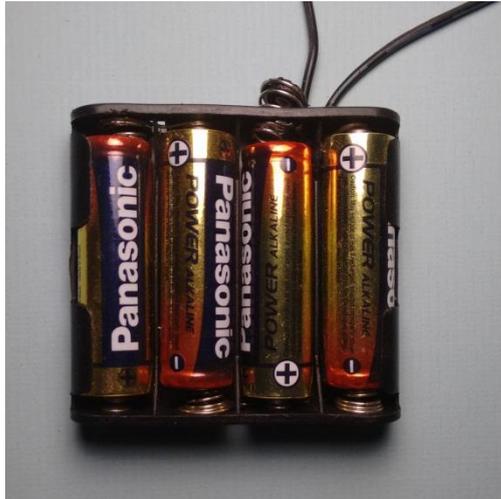
Figura 24: Sensores infravermelhos



Fonte: O autor

A ligação de todos os componentes é feita da seguinte forma: as saídas OUT3 e OUT4 da ponte H, que são as saídas de corrente, são ligadas aos dois polos do motor (positivo e negativo). Os pinos digitais 4, 5 e 9 do Arduino são conectados, respectivamente, às entradas IN3, IN4 e FNB da ponte H. Os pinos digitais 2 e 3 são ligados às conexões digitais dos sensores ópticos (um em cada). Com relação à alimentação, os sensores conectados à saída de 5 v do Arduino e o GND (terra) deste também é ligado aos terminais GND dos sensores. Já a corrente que alimenta a ponte H que conseqüentemente alimentará o motor é feita por 4 pilhas AA de 1.5 v cada (Figura 25). As pilhas são ligadas em série. Tal ligação permite que se possa somar as correntes de cada pilha resultando em uma corrente final de 6 v que é ligada à entrada +12 v da ponte H. O polo negativo das pilhas, bem como o GND da ponte, é ligado ao GND do Arduino.

Figura 25: Pilhas ligadas em série



Fonte: O autor

3.6 O Software

O software responsável por controlar a iluminação e o portão da casa foi implementado na linguagem própria do Arduino que, por sua vez, é baseada na linguagem C++, sobre a qual são feitas algumas modificações. Modificações estas que se traduzem no uso de funções criadas e adaptadas para o uso específico e facilitado de controle de hardware. O código é escrito numa IDE chamada Arduino 1.0.5-r2. Esta IDE permite que o programa seja facilmente transmitido para a placa apenas conectando-se o USB e utilizando o botão “upload” do menu.

Para construir um programa para o Arduino é necessário que existam duas funções principais: a função `setup()`, que é utilizada no início do programa para inicializar variáveis e parâmetros e a função `loop()`, que contém os procedimentos que permanecem em um laço infinito de repetição enquanto o Arduino estiver ligado.

O código desse trabalho, que pode ser encontrado no Apêndice A, funciona da seguinte forma:

Inicialmente é feita a declaração das variáveis que servem para controle do estado do portão (fechado, fechou, abriu e abrindo). Tais variáveis são booleanas (assumem o valor de verdadeiro ou falso) serão usadas da função que controla os sensores ópticos descrita mais adiante. Dentro da função `setup()`, é feita a configuração dos pinos do Arduino, onde todos foram definidos como saída

(OUTPUT) com exceção dos pinos 2 e 3 que serão usados pelos sensores e, neste caso, são definidos como pinos de interrupção.

Em seguida, tem-se a função `Serial.begin()` que é usada quando se faz uso da comunicação serial do Arduino e que, nesse caso, é utilizada na comunicação com o módulo Bluetooth. Já dentro da função `loop()`, a primeira coisa feita é a leitura da porta serial através da função `Serial.read()`. O que é lido da porta serial é armazenado dentro de uma variável chamada “z” e o valor dessa variável é que será usado para a tomada de decisão das funções que executam as funções de movimentar o motor e acender os leds. Cada ação feita pelo usuário no aplicativo enviará um caractere diferente de acordo com a fig. 15 mostrada na seção 3.3. A estrutura condicional que se segue (if-else) é responsável por avaliar o valor da variável z e direcionar para determinada função. Quando são recebidos os caracteres ‘A’, ‘B’, ‘C’ ou ‘D’ é chamada a função `digitalWrite()` que irá setar (definir), respectivamente, os pinos 7, 11, 10 e 12 do Arduino como “HIGH”, ou seja, permitirá passagem de corrente, o que fará acender os LEDs determinados. Por outro lado, quando forem recebidos os caracteres ‘a’, ‘b’, ‘c’ ou ‘d’, a função irá setar, respectivamente, os pinos 7, 11, 10 e 12 como “LOW”, bloqueando a passagem de corrente e, conseqüentemente, apagando os LEDs. Ao receber o caractere ‘E’ e a variável “abriu” estiver definida como “false”, a função `digitalWrite()` seta o pino 4 como “LOW” e o 5 como “HIGH” que estão conectados, respectivamente às entradas IN3 e IN4 da ponte H. Tal combinação gera a informação para que o motor gire em sentido anti-horário que, faz o portão abrir. Já ao receber o caractere ‘F’ e a variável “fechou” estiver definida como “false”, ocorre o inverso. O pino 4 é setado como “HIGH” e o 5 como “LOW”, fazendo a informação gerada pela ponte H girar o motor em sentido horário, fechando o portão. O ultimo caso, ocorre quando é recebido o caractere ‘G’, onde `digitalWrite()` irá, neste caso, definir ambos os pinos 4 e 5 como “LOW”. Dessa forma não haverá passagem de corrente para as entradas IN3 e IN4, conseqüentemente, a informação será para interromper a rotação do motor.

Por fim, existem as funções que são responsáveis pelo controle dos sensores opticos, que são: `contatoAbrir()` e `contatoFechar()` que são chamadas quando ocorre uma leitura em um dos sensores. Essa leitura pode ocorrer de duas formas:

.Quando o portão abrir por completo e “tocar” o sensor que fica em uma das extremidades. Dessa forma, o sensor passará a informação para o pino de interrupção 2 do Arduino e a função `attachInterrupt`, que é a função pré-definida responsável por gerenciar os pinos de interrupção, irá chamar a função `contatoAbrir()` que usará a `digitalWrite()` para definir os pinos 4 e 5 como “LOW” e, exatamente como ocorre na descrição anterior quando se recebe o caractere ‘G’, a rotação do motor é interrompida.

.Quando o portão fechar por completo e “tocar” o sensor que fica na extremidade oposta. Esse sensor passa a informação de leitura dessa vez para o pino 3 do Arduino e a função `attachInterrupt` agora passa o controle para a função `contatoFechar()` que também usa a `digitalWrite()` para definir os pinos 4 e 5 como “LOW”, também interrompendo a rotação do motor.

A diferença entre as duas funções `contatoAbrir()` e `contatoFechar()` está no fato de elas mudarem os valores das variáveis de estado de forma diferente. Tais variáveis são checadas na estrutura `if-else` juntamente com o recebimento do caractere. Quando `contatoAbrir()` é chamada significa que o portão está completamente aberto, a variável “abriu” então é definida como “true” e a variável “fechou” como “false”. Assim, nesse exemplo, quando é enviado o caractere ‘E’ para abrir o portão, a outra condição que é o estado da variável “abriu” ser “false” não será obedecida e o portão não será movimentado. Em outras palavras, o motor não poderá girar no sentido de abrir o portão, já que o mesmo está completamente aberto. Já a função `contatoFechar()` define as duas variáveis de forma inversa impedindo de forma semelhante que se gire o motor no sentido de fechar o portão quando este já esteja completamente fechado.

4 Conclusão

Tomando como base a revolução tecnológica que a humanidade vivencia hoje e a vasta gama de possibilidades que ela pode oferecer, esse trabalho trouxe um exemplo de como o seu uso adequado a certas situações, pode oferecer alternativas para a realização de atividades cotidianas, como o acendimento de lâmpadas em uma casa e o controle de portões, pode ser feito de forma mais fácil e cômoda para os seus moradores.

Foi realizado um levantamento das tecnologias utilizadas, incluindo o histórico de seu surgimento e de sua constante evolução, bem como a utilização no cotidiano das pessoas. Com essa análise, foi possível perceber o quanto sistemas computacionais estão se tornando pervasivos, ou seja, estão ficando cada vez mais intrínsecos para aqueles que os utilizam. Isso se traduz em sistemas automação e controle, comunicação entre dispositivos eletrônicos diversos. É formado um verdadeiro ambiente digital que se mescla ao cotidiano tornando-se cada vez mais imperceptível e comum.

A utilização do hardware livre mostrou-se muito satisfatória uma vez que todos os requisitos para o funcionamento do sistema foram atendidos de forma fácil e relativamente rápida. Tudo isso graças ao Arduino ser uma plataforma de hardware livre, o que torna o conhecimento sobre ele aberto, passível de ser continuamente expandido e compartilhado. Também à enorme comunidade que existe em torno dele, o que fez as pesquisas, o desenvolvimento do código e a utilização dos recursos do hardware serem bem explorados. Milhares de pessoas compartilham nos fóruns temáticos de discussão seus projetos e ideias, ajudando a propalar ainda mais esse universo. E esse projeto, que por um lado se baseou muito dessa informação disponível livremente, acabou também por contribuir para sua expansão.

Durante o processo de construção desse sistema, novas ideias foram surgindo a todo o momento e muito disso se deve justamente ao grande leque de possibilidades que o hardware livre pode oferecer. Ideias que não foram colocadas em prática tanto por limitações de tempo tanto para que o projeto não se tornasse

promíscuo. Portanto, deixa-se como trabalhos futuros a criação de um sistema web para que o controle seja exercido remotamente de qualquer lugar que tenha acesso à internet e implementar um sistema inteligente para que a iluminação varie seu comportamento a depender da luminosidade externa.

Diante disso, pôde-se chegar a ideia de que esse trabalho foi de alguma forma capaz de contribuir para a sociedade em geral, tanto por ser um sistema de automação que teve como objetivo inicial proporcionar mais conforto e comodidade para as pessoas que dele fizeram uso, quanto por, conseqüentemente, acabar servindo como uma espécie de legado que servirá de base e incentivo, encorajando novas ideias e projetos.

Referências

- MURATORI, J. R.; DAL BÓ, P. H. **Automação Residencial: histórico, definições e conceitos**. S.D. Disponível em: <http://www.osetoreletrico.com.br/web/documentos/fasciculos/Ed62_fasc_automacao_capl.pdf>. Acesso em 05 jun.2015.
- LIPOVESTSKY, G. **O luxo muda de Cara**. Época, Rio de Janeiro, v.15, n.1, p. 42-47,2005.
- DIAS, C. L. A. **Domótica: aplicabilidade às edificações residenciais**. 2004.96f. Tese (Mestrado em Engenharia Civil)- Universidade Federal Fluminense. Niterói, 2004.
- DEBERT, G. G. **A reinvenção da velhice: socialização e processos de reprivatização do envelhecimento**. São Paulo: Editora da Universidade de São Paulo, Fapesp, 1999. 253p.
- WERNECK, S. B. F. **Domótica: União de arquitetura e tecnologia da informação na edificação residencial urbana**. 1999. 105f. Tese (Mestrado em Arquitetura)- Universidade Federal do Rio de Janeiro. Rio de Janeiro, 1999.
- NUNES, R. J. C. **Home Automation: A Step Towards Better Energy Management**. Technical University of Lisbon: Lisbon, 2008.
- SILVEIRA, J. A. **Experimentos com o Arduino**. São Paulo: Ensino Profissional, 2011. 280 p.
- PENIDO, E. C. C.; TINDADE, R. S. **Microcontroladores**. Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais. Ouro Preto, 2013.
- LIMA, C. B. **Os Poderosos μ Controladores AVR**. Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina. Florianópolis, 2009.
- SOARES, K. **O que é um Arduino e o que pode ser feito com ele**. Disponível em: <<http://www.techtudo.com.br/noticias/noticia/2013/10/o-que-e-um-arduino-e-o-que-pode-ser-feito-com-ele.html>>. Acesso em: 10 set.2015.
- WARREN, J.; ADAMS, J.; MOLLE, H.. **Arduino Robotics**. EUA: Apress, 2011. 628 p.
- McROBERTS, M.. **Arduino Básico**. São Paulo: Novatec, 2011. 456 p.
- AZEVEDO, L. C.; PIZZOLATO, N. D.. **DOMÓTICA: Aplicabilidade e Sistemas de Automação Residencial**. Vértices, São Paulo, v. 6, n. 3, p. 30-54, 2004.

NEVES, R. P. A. A.. **Espaços arquitetônicos de alta tecnologia: Os edifícios inteligentes**. 2002. 106f. Tese (Mestrado em Arquitetura)- Escola de Engenharia de São Carlos. São Carlos, 2002.

NACIMENTO, M.; JOTA, F. G.; ASSIS, E. S.. **A automação como ferramenta para eficiência energética em edificações**. Anais do Congresso Brasileiro de Eficiência Energética. Belo Horizonte, 2005.

RENNA, R.; BRASIL, R.; CUNHA, T.; BEPPU, M.; FONSECA, E.. **Introdução ao kit de desenvolvimento Arduino**. Universidade Federal Fluminense. Niterói, 2013.

LEE, K.; CHOI, J.. **Remote-Controlled Home Automation System via Bluetooth Home Network**. Fukui University. Fukui, 2003.

LACERRA, I.. **Bluetooth**. Universidade de São Paulo. São Paulo, 2008.

APÊNDICE A - Código para controle do sistema

```

//declaração das variáveis de estado
bool fechando;
bool abrindo;
bool abriu;
bool fechou;

void setup()
{
  //inicializa todas as variáveis de estado
  fechando = false;
  abrindo = false;
  abriu = false;
  fechou = false;
  //attachInterrupt configura a interrupção
  attachInterrupt(0, contatoAbrir, RISING);
  attachInterrupt(1, contatoFechar, RISING);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(12, OUTPUT);

  //configura o uso da porta serial onde será feita a comunicação com o
  módulo bluetooth
  Serial.begin(9600);
}

void loop() {

  if (Serial.available()) {
    char z;
    z = Serial.read();

    //----- MOTOR -----

    //Gira no sentido anti-horário "Abrir"
    if ((z == 'E') && (!abriu))
    {
      fechando = false;
      abrindo = true;
      abriu = false;
      fechou = false;
      digitalWrite(5, HIGH);
      digitalWrite(4, LOW);
      analogWrite(9, 200);
      delay(50);
    }

    //Para o motor "Parar"
    else if (z == 'F')
    {
      fechando = false;
      abrindo = false;
    }
  }
}

```

```

    abriu = false;
    fechou = false;
    digitalWrite(5, LOW);
    digitalWrite(4, LOW);
    analogWrite(9, 200);
    delay(50);
}

//Gira no sentido anti-horário horário "Fechar"

else if ((z == 'G') && (!fechou))
{
    fechando = true;
    abrindo = false;
    abriu = false;
    fechou = false;
    digitalWrite(5, LOW);
    digitalWrite(4, HIGH);
    analogWrite(9, 200);
    delay(50);
}
//----- LEDS -----

else if (z == 'A')
{
    digitalWrite(7, HIGH);
    delay(50);
}
else if (z == 'a')
{
    digitalWrite(7, LOW);
    delay(50);
}
else if (z == 'B')
{
    digitalWrite(11, HIGH);
    delay(50);
}
else if (z == 'b')
{
    digitalWrite(11, LOW);
    delay(50);
}
else if (z == 'C')
{
    digitalWrite(10, HIGH);
    delay(50);
}
else if (z == 'c')
{
    digitalWrite(10, LOW);
    delay(50);
}
else if (z == 'D')
{
    digitalWrite(12, HIGH);
    delay(50);
}
else if (z == 'd')
{
    digitalWrite(12, LOW);
}

```

```
        delay(50);
    }
}

//para o motor quando o portao toca o sensor ao abrir
void contatoAbrir()
{
    if (!fechando)
    {
        //muda estado para abriu
        fechando = true;
        abrindo = false;
        abriu = true;
        fechou = false;

        digitalWrite(5, LOW);
        digitalWrite(4, LOW);
        analogWrite(9, 200);
        delay(50);
    }
}

//para o motor quando o portao toca o sensor ao fechar
void contatoFechar()
{
    if (!abrindo)
    {
        fechando = false;
        abrindo = true;
        abriu = false;
        fechou = true;
        digitalWrite(5, LOW);
        digitalWrite(4, LOW);
        analogWrite(9, 200);
        delay(50);
    }
}
```