

# Sistema de Prevenção de Incêndio

Eric Kenzo Taniguchi Onuki - [ericonuki@gmail.com](mailto:ericonuki@gmail.com)

Michel Oleynik - [michel\\_ole@hotmail.com](mailto:michel_ole@hotmail.com)

Yuri Cassio Campbell Borges - [yuri.campbell@gmail.com](mailto:yuri.campbell@gmail.com)

## Professores Orientadores:

Gil Marcos Jess - [gltjess@terra.com.br](mailto:gltjess@terra.com.br)

Afonso Ferreira Miguel - [afmiguel@ccet.pucpr.br](mailto:afmiguel@ccet.pucpr.br)

Edson José Pacheco - [edsonpacheco@gmail.com](mailto:edsonpacheco@gmail.com)



## 1 Abstract

A system to prevent fire was developed as conclusion work of the third term of Computer Engineering at Pontifical Catholic University of Parana. Based on software and hardware interaction, it can identify fire by a signal sent by smoke detector, in advance. It happens activating a device that pulls out the fire extinguisher's security lock. The software can manage up to five smoke sensors and send a signal up to eight fire extinguishers, based on pre-defined rules of fire proximity. Object-oriented programming guarantees code maintenance and greater systems portability.

## 2 Resumo

Projeto de conclusão do terceiro período do curso de Engenharia de Computação da Pontifícia Universidade Católica do Paraná, propõe o desenvolvimento de um sistema de prevenção de incêndio. Baseado na interação entre software e hardware, é capaz de detectar o início de um incêndio pela incidência de fumaça e acionar um dispositivo responsável pela retirada da trava de segurança de extintores. O programa elaborado monitora a situação de até cinco sensores de fumaça e pode disparar a ação em até oito extintores de incêndio, com base em regras pré-estabelecidas de proximidade do fogo. Uma programação orientada a objetos garante a manutenção do código e a portabilidade para sistemas maiores.

## 3 Objetivos

### 3.1 GERAL

Promover o aprendizado dos programas de aprendizagem em Física III, Sistemas Digitais I, Circuitos Elétricos I e Técnicas Avançadas de Programação por meio do desenvolvimento de uma solução de combate a incêndios baseada em software e hardware.

### 3.2 ESPECÍFICOS

Pesquisar princípios físicos da detecção de incêndios, bem como formas de identificar sua origem.

Estudar a portabilidade entre as linguagens de programação Java e C/C++, no intuito de explorar seus potenciais individuais.

Provocar o rompimento de lacres de extintores de incêndio e efetuar uma comunicação efetiva com o Corpo de Bombeiros.

Tornar o sistema implementado viável economicamente, para que tenha sua dimensão de baixo porte validada.

## 4 Descrição do projeto

### 4.1 ORGANIZAÇÃO FÍSICA

Para efeitos de análise funcional do protótipo criado, foram construídas maquetes representativas da situação real de uso para qual o sistema foi projetado.

Vê-se na figura 12, em segundo plano, o suporte feito em madeira para fixação do detector de fumaça, que se estrutura dessa forma devido à comparação com a situação real na qual ele estaria posicionado no teto do ambiente protegido. Logo acima, os circuitos que possibilitaram a integração ao sistema foram organizados representando uma fixação no forro do ambiente.

Em primeiro plano, observam-se suportes, também em madeira, construídos para proporcionar um melhor manuseio e estética às placas com os circuitos impressos. Cada placa recebeu sua base porque, por protótipo, elas ficam fisicamente separadas.

A figura 13 mostra o objetivo para o qual o sistema fora construído: a retirada

automatizada da trava de segurança do extintor. Por projeto, o motor que executa a tarefa fica fixado à parede; para uma representação prática, porém, foi construído o sustentáculo em madeira. Este foi mais seguramente preso ao extintor por meio de duas faixas de velcros, já que somente o suporte não ofereceu a segurança necessária.

O motor, por sua vez, foi preso à haste de forma invertida, numa tentativa de encapsulamento do funcionamento do sistema. Sua respectiva placa de circuito impresso fora afixada no lado oposto porquanto, por projeto, esta ficaria no interior da parede.

Observa-se, pela figura 14, o funcionamento, numa instância física, do mecanismo que retira a trava de segurança do extintor. Utilizou-se barbante como ligação entre o eixo do motor e a trava vide características como ser de comum uso, suficientemente forte e flexível para a tarefa em questão.

A escolha de epóxi deu-se devido à necessidade de um eixo vazado e resistente o bastante para proporcionar firmeza na transmissão de movimento da rotação do eixo original do motor para a retirada da trava. E, como o epóxi é um material resistente e modelável, este se ajustou às demandas perfeitamente.

## 4.2 CIRCUITO ELÉTRICO CONSTRUÍDO

No intuito de atender aos objetivos de portabilidade descritos anteriormente, todo o circuito elétrico foi planejado de forma distribuída, com o cuidado de produzir sinais fortes o suficiente para que pudessem ser interpretados a média distância. Nesse sentido, o sistema pode ser dividido em três partes principais: sensores, central de controle e executores. A transmissão entre eles dá-se por meio de um fio telefônico, adotado devido ao fácil manuseio e conexão, além da garantia de um fluxo de corrente adequado.

Os sensores, por sua vez, dividem-se em dois tipos: um real, capaz de detectar a presença de fumaça no ambiente, e quatro simulados, responsáveis pelo envio de sinais genéricos. Adotou-se essa estratégia para evitar a aquisição de vários detectores, que, na prática, só acarretariam em um aumento do custo financeiro total do projeto.

O sensor real consiste em um circuito desenvolvido para tratar o sinal emitido por um detector de fumaça fotoelétrico, com o objetivo básico de aumento do período dos pulsos de tensão. Baseado no circuito integrado LM555, o dispositivo recebe como entrada uma onda quadrada de frequência 394,3 Hz e pico de tensão 11,3

V, modificando-a para adquirir um período <sup>[1]</sup> de 2,97 s e altura em torno de cinco volts — características suficientes para um correto estímulo da porta paralela. Os componentes presentes — excetuando-se a porta lógica NOT, utilizada para gerar uma onda positiva na saída — são todos especificados no *datasheet* do CI, como modelo de aplicação.

Em seu tempo, os sensores simulados, são, na verdade, apenas chaves que alternam entre os estados alta impedância e zero volt, enviados à placa central (vide figura 2).

Essa, por sua vez, é responsável pelo recebimento e envio de sinais para e da porta paralela. Com exceção da entrada para o sensor real — que possui bateria

própria para a geração de tensão —, as demais foram estabelecidas com base em circuitos *pull up*, que ora fornece cinco volts à porta, ora zero volt, de acordo com a posição da chave no sensor simulado.

No que diz respeito às saídas da porta paralela, encontra-se dois tipos de circuito: um para o executor real, outro para três executores simulados. O receptor real é um motor de tensão contínua, escolhido devido às características de maior força e menor precisão, adequadas ao conceito do projeto. Alimentado com 13,2 V, tensão superior à fornecida pelo computador, precisa estar conectado a um circuito capaz de fornecer tensão quando devidamente estimulado (a conexão é feita de modo direto, conforme indica a figura 5).

Tal lógica é a implementada na central de controle e visível na figura 1. Quando o sinal proveniente do conector paralelo é de nível lógico alto, há estímulo do transistor, que permite a passagem de corrente pelo relé e o conseqüente fechamento de sua chave interna. Para evitar danos aos componentes, acrescentam-se ainda resistores e diodos, que limitam o fluxo de corrente em um único sentido.

Por sua vez, os executores simulados são, na verdade, apenas LEDs ligados em série a resistores — utilizados para reduzir a corrente a que os LEDs ficam submetidos — e conectados diretamente às saídas da porta paralela. O diagrama de seus circuitos pode ser visualizado na figura 3.

Uma vez detalhado o princípio de funcionamento e visualizados os diagramas dos circuitos elétricos, pode-se, ainda, verificar os diagramas das placas de circuito impresso, presentes no item 7. Com isso, é possível obter uma significativa noção do projeto desenvolvido como um todo, o que, na prática, garante sua reprodutibilidade técnica por qualquer pessoa com conhecimentos mínimos de eletrônica.

## 5 Lista de materiais

<b>Quantidade</b>	<b>Descrição</b>
15 cm	Barbante
02	Bornes
01	Cabo paralelo macho-macho
25 m	Cabo telefônico
01	Capacitor 10 nF
01	Capacitor eletrolítico 100 µF
04	Chaves tipo alavanca
01	Circuito integrado LM555
01	Circuito integrado SN7404
01	Conector DB25 fêmea
18	Conectores RJ11 fêmea
18	Conectores RJ11 macho

- 01 Detector de fumaça fotoelétrico
- 02 Diodos 1N4148
- 01 Fonte de tensão
- 03 LEDs
- 01 Motor DC 13,2V
- 01 Placa para circuito impresso 15x30 cm
- 01 Relé 12 V / 12 A0
- 01 Resistor 1 k $\Omega$
- 01 Resistor 27 k $\Omega$
- 03 Resistores 330  $\Omega$
- 05 Resistores 470  $\Omega$
- 01 Transistor BC548

## 6 Diagramas elétricos

FIGURA 1 - CENTRAL DE CONTROLE

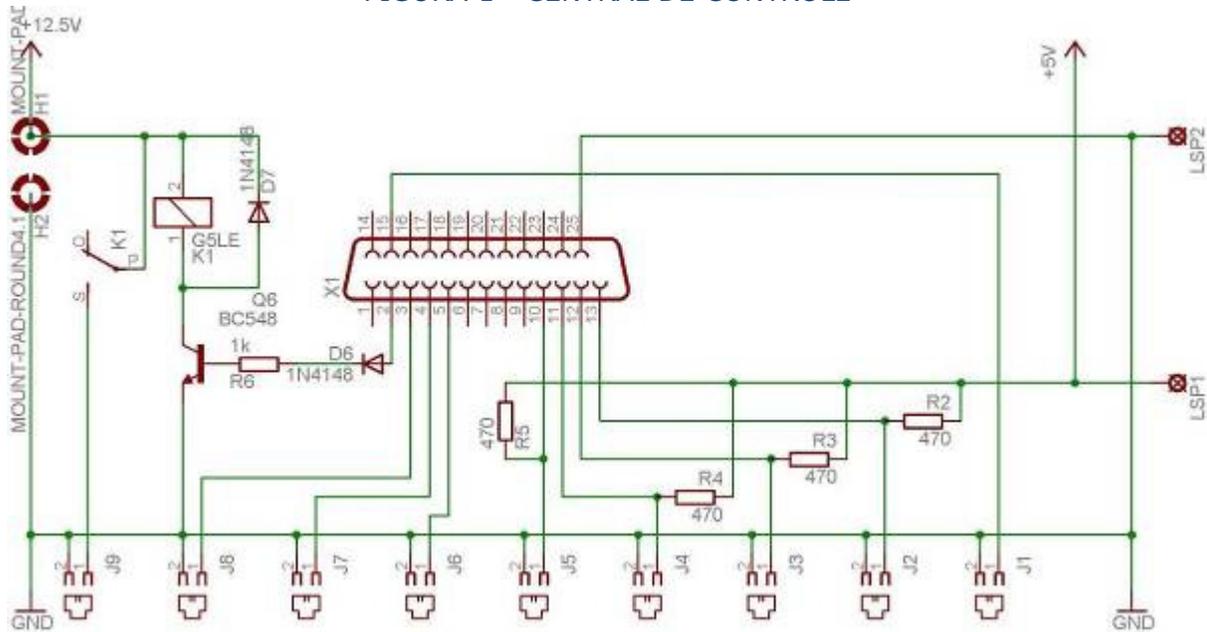


FIGURA 2 - CHAVES SIMULADORAS DE DETECTORES DE FUMAÇA

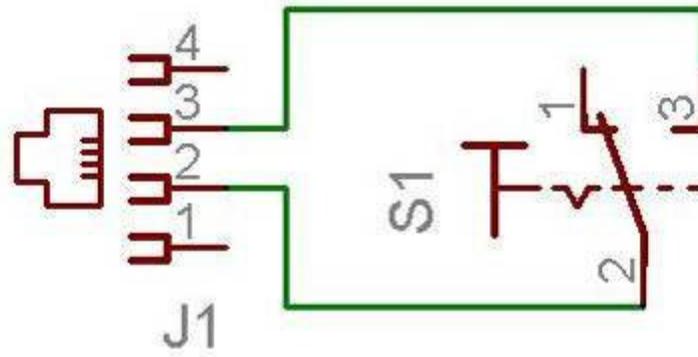


FIGURA 3 - LEDS INDICATIVOS DE ROMPIMENTO DOS LACRES

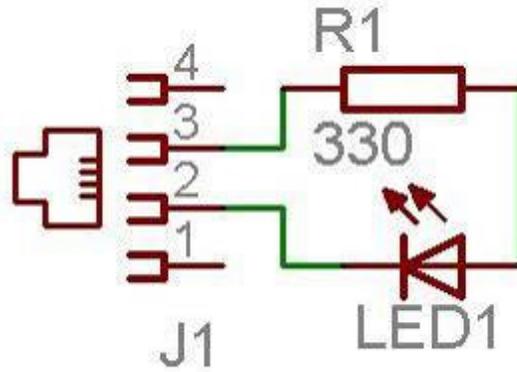


FIGURA 4 - CIRCUITO DE TIMER PARA O DETECTOR DE FUMAÇA

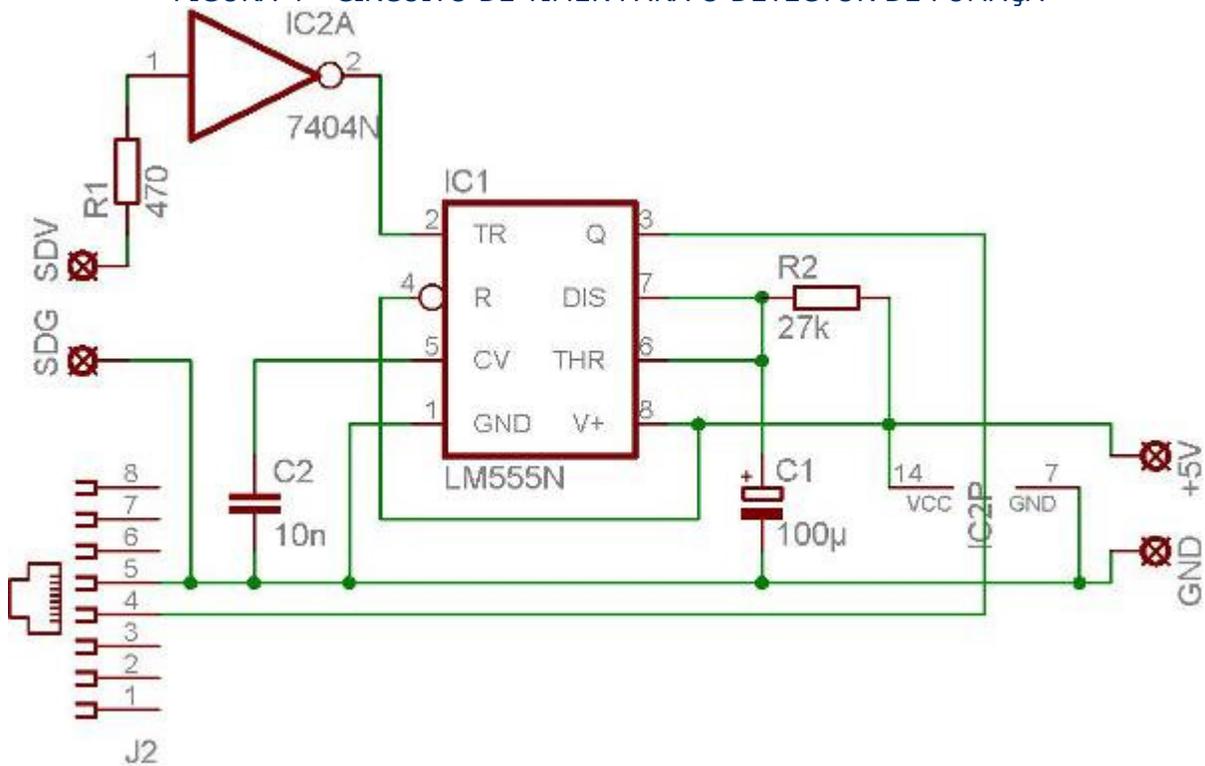
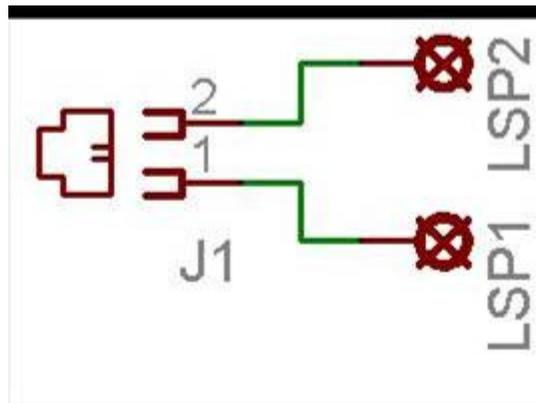


FIGURA 5 - CONECTOR PARA O MOTOR



## 7 Diagramas das placas de circuito impresso

FIGURA 6 - CENTRAL DE CONTROLE

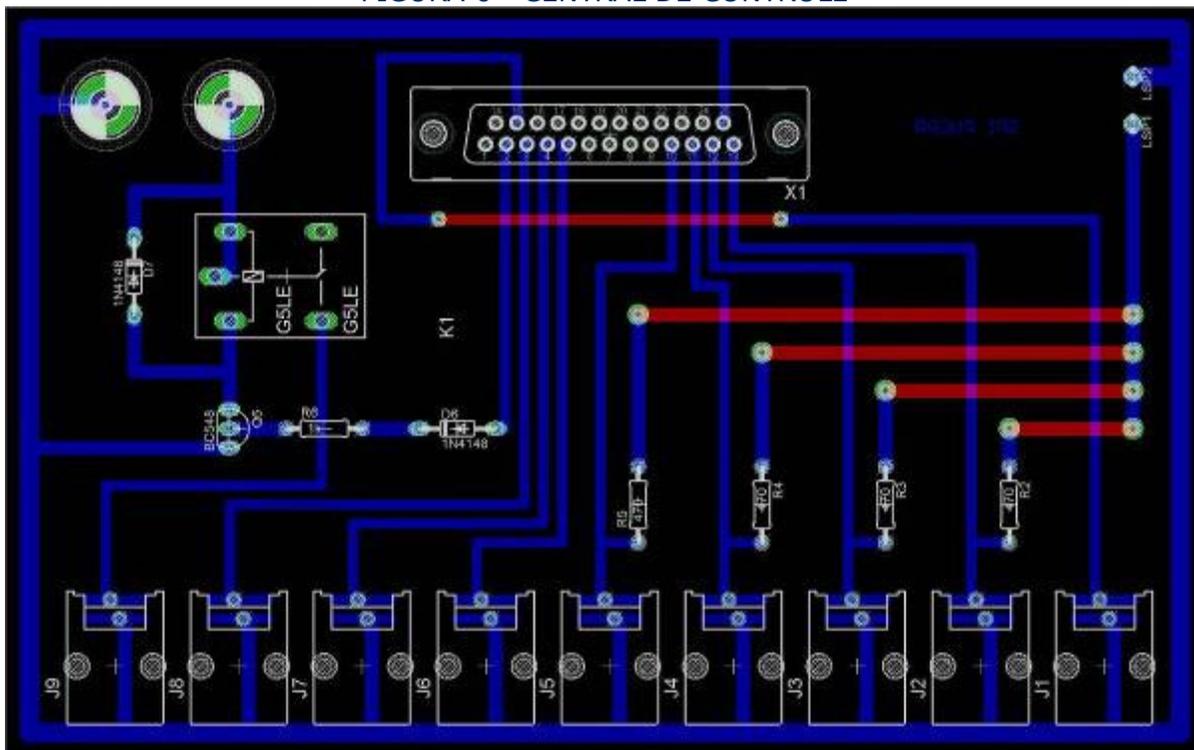


FIGURA 7 - CHAVES SIMULADORAS DE DETECTORES DE FUMAÇA

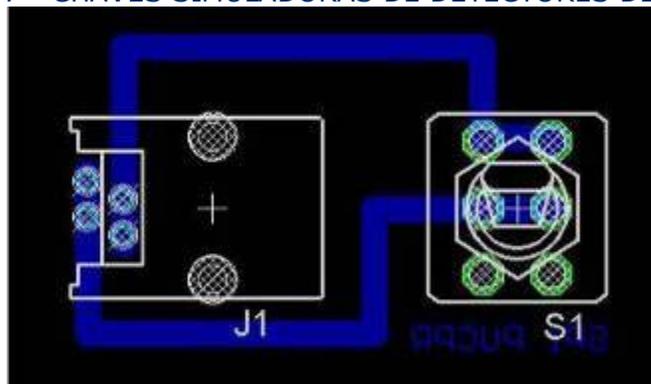


FIGURA 8 - LEDS INDICATIVOS DE ROMPIMENTO DOS LACRES

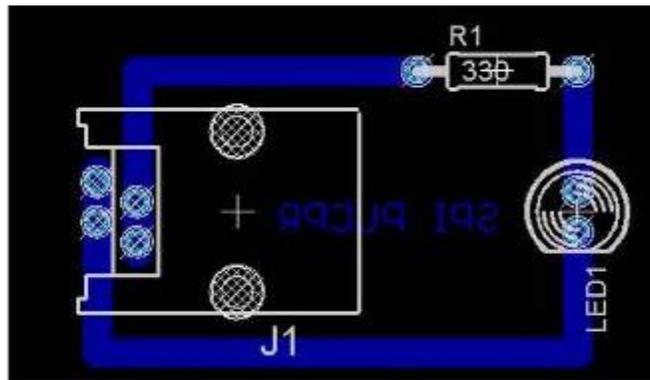


FIGURA 9 - CIRCUITO DE TIMER PARA O DETECTOR DE FUMAÇA

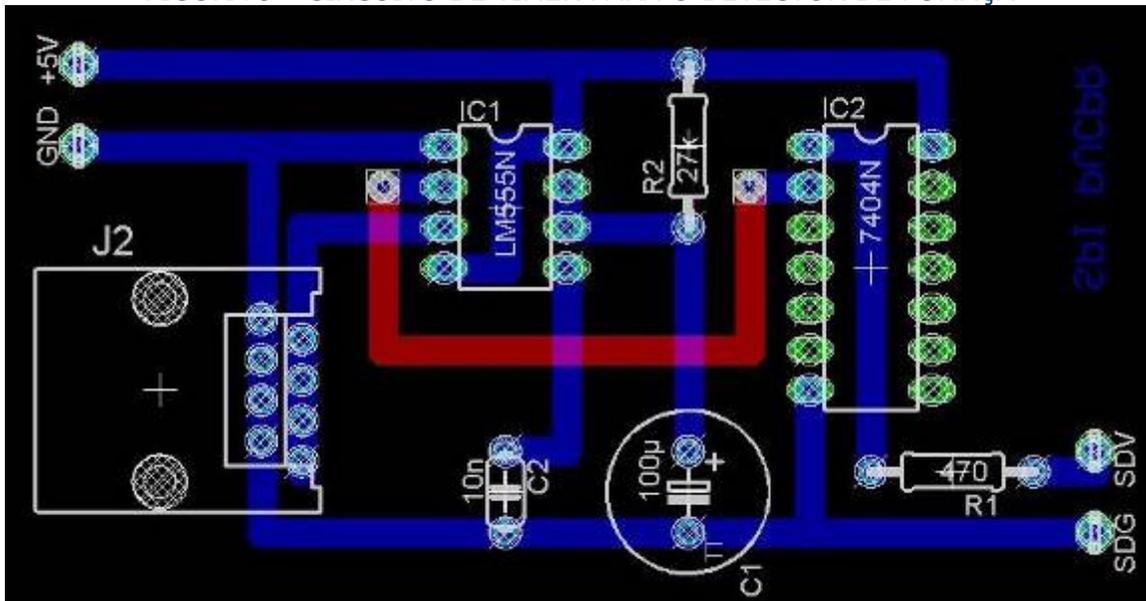


FIGURA 10 - CONECTOR PARA O MOTOR



## 8 Software desenvolvido

O software deste projeto foi desenvolvido partindo-se da premissa de que o programa deveria ser o mais genérico possível, de modo a atender, com poucas modificações, todo e qualquer ambiente em que o sistema viesse a ser implantado.

A escolha da linguagem de programação C++ teve como principal motivo a maior facilidade de gerenciamento de memória, necessária para uma manipulação direta dos bits da porta paralela e inexistente em linguagens modernas. De modo análogo, a opção por um código orientado a objetos visa ao preenchimento dos

requisitos de portabilidade definidos anteriormente como premissas de trabalho.

Isso posto, a estrutura do software partiu de necessidades surgidas no âmbito eletrônico, tal como a comunicação e integração entre software e hardware através da porta paralela. Devido à impossibilidade de se utilizar rotinas pré-definidas de acesso direto à memória nos sistemas operacionais modernos — característica atribuída a eles no intuito de aumento de segurança —, houve necessidade da utilização de uma DLL (biblioteca de enlace dinâmico, na sigla em inglês) externa. Para tal, optou-se pelo uso da biblioteca *inpout32.dll*, que permite a comunicação com a porta paralela através de comandos de *HyperTerminal*<sup>[2]</sup>.

O primeiro passo efetuado foi, então, a criação de uma classe que pudesse incorporar os comandos de comunicação entre o computador e a porta paralela, denotada *CComunicacao* [sic]<sup>[3]</sup> — classe genérica que, tecnicamente, comunica-se com o dispositivo através de qualquer meio. A classe *CComunicacaoParalela*, por sua vez, herda de *CComunicacao* a característica de comunicar-se com o dispositivo, porém especifica sua realização através da porta paralela. Na prática, possui métodos capazes de enviar ou receber dados por meio dela. Uma vez implementada, foi possível realizar testes nos circuitos eletrônicos outrora construídos pelo grupo.

O programa, em seu tempo, foi organizado com base em uma instanciação de *CComunicacaoParalela*. Assim sendo, a classe *CData* — genérica para qualquer tipo de dados com um tamanho de  $n$  bits — é uma interface do programa para com ela, utilizada ali de forma instanciada para a saída de dados e, nos detectores, para verificação dos bits de entrada.

De forma semelhante, a classe *CDispositivo* é genérica para qualquer dispositivo de prevenção de incêndio, podendo ser especificada a critério do desenvolvedor. No caso presente, foi derivada para controle exclusivo de extintores de incêndio e assumiu o nome de *CExtintor*. Essa classe possui uma vinculação direta com a instância de *CComunicacaoParalela*, de forma que cada objeto criado possa acionar seu respectivo dispositivo. Como os lacres de segurança são únicos, uma vez retirados, o sistema do extintor não pode ser acionado novamente.

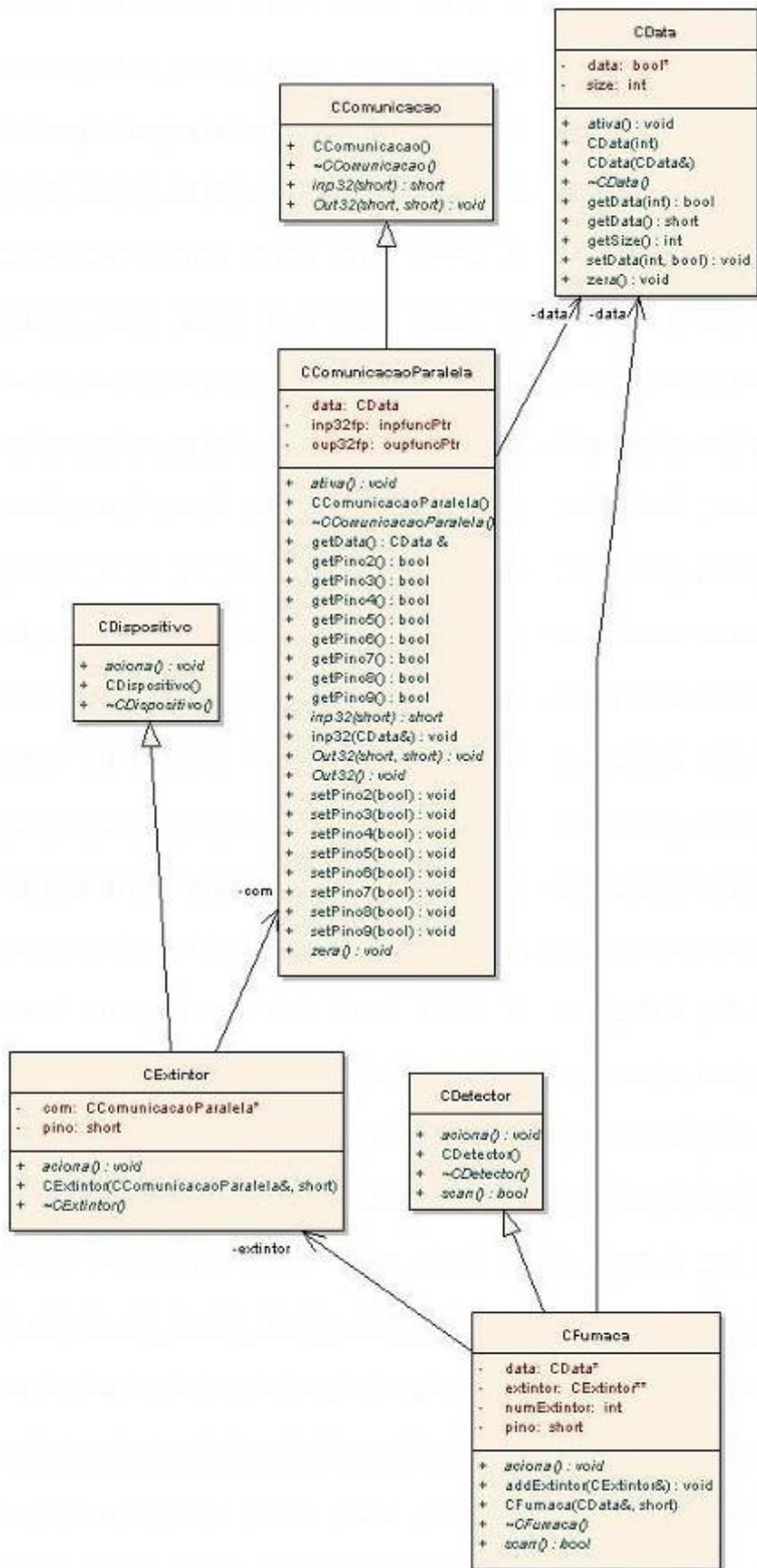
Por sua vez, a classe *CDetector* é genérica para os dispositivos detectores: o desenvolvedor pode especificá-la para qualquer tipo de sensor desejado, ampliando e garantindo a portabilidade do sistema. Derivada para detectores de fumaça, a classe *CFumaca* garante o funcionamento específico. Cada instância recebe um vínculo de *CData* para entrada de dados, conforme descrito anteriormente, de modo com que possa verificar a detecção de fumaça com base na análise do *byte* recebido.

Cada objeto do tipo *CFumaca* pode ser vinculado a um ou mais objetos do tipo *CExtintor*. É interessante notar que um extintor também pode estar vinculado a diversos detectores de fumaça. Dessa forma, quando um sensor detecta a presença de incêndio, pode acionar todas as instâncias de *CExtintor* a ele vinculadas. Tal procedimento é importante para que nenhum extintor localizado nas proximidades do fogo deixe de ser utilizado em seu combate.

Para uma melhor visualização de toda a lógica do *software*, seu diagrama de

classes foi construído seguindo o padrão UML (linguagem de modelagem unificada, na sigla em inglês) e encontra-se na figura 11 logo abaixo.

FIGURA 11 - DIAGRAMA UML DO PROGRAMA DESENVOLVIDO



FONTE: ELABORADO PELOS AUTORES

## 9 Conclusão

O desenvolvimento de projetos de baixo porte, embora não se revele a um primeiro olhar, é uma tarefa detentora de relativa complexidade, principalmente quando seus executores não possuem todo o conhecimento técnico necessário para tal. Por outro lado, a experiência é rica em aprendizado, o que, por si só, garante a justificativa de execução do trabalho. Nesse sentido, se fosse possível estabelecer um referencial comparativo, a qualidade do crescimento cognitivo superaria e, muito, a quantidade de produção material efetiva.

Com base nisso, faz-se importante referenciar alguns conhecimentos obtidos. No que tange à eletrônica, foi analisado em profundidade o funcionamento de relés, transistores — estudo focado na diferenciação entre os tipos NPN e PNP —, diodos e circuitos integrados — com ênfase no modelo LM 555, utilizado como temporizador para o sinal proveniente do detector de fumaça. Para isso, foi realizada também uma revisão bibliográfica sobre sensores de incêndio, com destaque para o princípio de funcionamento de detectores iônicos e fotoelétricos.

No que diz respeito à comunicação entre software e hardware, estudou-se o princípio de funcionamento da porta paralela, adotada como interface para o sistema. Após relativo período de testes e ensaios, descobriu-se que — ao contrário do que afirmava a documentação existente — a porta interpreta, em suas entradas, os níveis lógicos baixo e alto quando submetida às tensões de 0 e 5 volts, respectivamente.

Ainda no campo teórico, pesquisou-se a aplicabilidade de motores de passo e tensão contínua, de forma com que este pudesse ter sido escolhido com segurança e fundamentação técnica.

Além disso, a experiência proporcionou uma relativa gama de conhecimentos práticos, necessários para uma produção mais eficaz. Percebeu-se, a título de exemplo, o papel fundamental que a orientação a objetos determina na redução de tempo em manutenção de código. A programação estruturada em classes, como também é chamada, permitiu fácil organização e separação dos algoritmos responsáveis pela comunicação, interpretação e tratamento dos dados recebidos e enviados.

De modo análogo, a utilização de um programa específico para a diagramação das placas de circuito impresso se traduziu em maior facilidade e menor desperdício de material no momento de sua perfuração. Todavia, é importante ressaltar que, infelizmente, muito dos componentes encontrados no Brasil não possuem dimensões corretamente traduzidas em software. Na prática, fazem-se necessárias algumas verificações e ajustes anteriores a qualquer atitude decisiva. De qualquer modo, é interessante sempre manter certa quantidade de componentes em sobra, no intuito de evitar maiores preocupações e transtornos durante a aquisição de material.

Merecem destaque também algumas conclusões a respeito do que não pôde ser realizado. Em primeiro lugar, descobriu-se que o Corpo de Bombeiros local não dispõe de um método automatizado e público para o atendimento a situações de emergência. Se fosse necessária uma comunicação com a corporação, provavelmente o software desenvolvido teria de englobar os recentes estudos sobre síntese artificial da voz humana.

De modo análogo, uma breve pesquisa a respeito da JNI (Interface Java para

código Nativo, na sigla em inglês) revelou a sua ineficiência para gerenciar grandes porções de algoritmo escrito em outra linguagem de programação, principalmente quando estruturado em classes. Na prática, isso significou a impossibilidade de se desenvolver uma interface gráfica em Java, haja vista que a comunicação paralela exige uma estrutura de acesso direto à memória, proporcionada pelo C.

Por último, descobriu-se experimentalmente que o rompimento de um lacre, tal como os existentes em extintores, exige altos níveis de corrente e/ou tensão — seja para promover seu derretimento, seja para provocar sua destruição. Com isso, não foi possível desenvolver um circuito capaz de prover tais características

com segurança e baixo custo — condições *sine qua non* <sup>[4]</sup> para a realização do trabalho.

De modo geral, foi possível adquirir uma expressiva experiência na realização de projetos práticos, traduzida em maior facilidade desde a concepção de idéias até a utilização de ferramentas mecânicas e componentes eletrônicos. Enfim, um projeto que surpreendeu por sua simplicidade, mas quem conteve em essência um conceito original e cumprido em toda a execução.

## 10. Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6135**: chuveiros automáticos para extinção de incêndio – especificação. Rio de Janeiro, 1992.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6125**: chuveiros automáticos para extinção de incêndio – métodos de ensaio. Rio de Janeiro, 1992.

BERENDSOHN, Roy. **Trial by fire**: how to fight a fire, and when not. Disponível em: <[http://www.popularmechanics.com/home\\_improvement/home\\_improvement/1314857.html](http://www.popularmechanics.com/home_improvement/home_improvement/1314857.html)> Acesso em: 13 mar. 2006.

BRAIN, Marshall. **How smoke detectors work**. Disponível em <<http://home.howstuffworks.com/smoke.htm>> Acesso em: 05 abr. 2006.

OLIVEIRA, André Luiz Araújo de. **Smoke detectors**. Disponível em: <<http://www.stratacafe.com/image.asp?imageID=7761>> Acesso em: 19 abr. 2006.

\_\_\_\_\_. **New view**. Disponível em: <<http://www.stratacafe.com/image.asp?imageID=10916>> Acesso em: 05 abr. 2006.

USBERCO, João; SALVADOR, Edgard. **Química 2**: físico-química. São Paulo: Saraiva, 2001, p. 506-507.

## 11. Galeria de fotos

FIGURA 12 - SUPORTES CONSTRUÍDOS



FONTE: ELABORADO PELOS AUTORES

FIGURA 13 - SUPORTE CONSTRUÍDO PARA O EXTINTOR



FONTE: ELABORADO PELOS AUTORES

FIGURA 14 - DETALHE DA RETIRADA DA TRAVA DE SEGURANÇA



FONTE: ELABORADO PELOS AUTORES

[1] Conforme indica sua folha de especificação, o período da onda de saída do CI LM555 pode ser obtido matematicamente pela relação  $T = 11 \cdot R_2 \cdot C_1$ , em que os componentes especificados são aqueles presentes na figura 4 do item 6.

[2] *HyperTerminal* é um programa utilizado para conectar-se a outros computadores, sites de *Telnet*, *BBSs*, serviços on-line e computadores host, usando um modem ou um cabo de modem nulo.

[3] Na linguagem de programação C++, é comum denotar nomes de classe prefixados pela letra C, indicativa de classe.

[4] Termo em latim que significa "sem a qual não", utilizado para descrever uma condição ou ação essencial.