Projeto de Graduação

DEE DEPARTAMENTO DE ENGENHARIA ELÉTRICA

23/02/2018

PROJETO DE LABORATÓRIO REMOTO DE CONTROLE DE ELEVADOR DE CARGA

Gabriel Amado Xavier de Oliveira Affonso

PUC RIO

www.ele.puc-rio.br

Projeto de Graduação

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

PROJETO DE LABORATÓRIO REMOTO DE CONTROLE DE ELEVADOR DE CARGA

Aluno(s): Gabriel Amado Xavier de Oliveira Affonso

Orientador(es): William de Souza Barbosa

Trabalho apresentado com requisito parcial à conclusão do curso de Engenharia Elétrica na Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brasil.

Agradecimentos:

Aos meus pais que sempre me apoiaram e ao meu orientador William Barbosa.

Resumo

Este projeto de graduação discute formas de implementação, apresenta uma solução para o controle virtual de um Elevador de Carga e seu manuseio remotamente.

A montagem de um laboratório remoto com ênfase em experiências didáticas se propõe a auxiliar os alunos de engenharia da Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio. O experimento será executado localmente ou remotamente através da Internet usando instrumentos virtuais, Vis, desenvolvidos no LabVIEW (acrônimo para *Laboratory Virtual Instrument Engineering Workbench)* com aquisição de dados e atuação por meio da placa controladora Arduino e poderão futuramente ser compartilhados com outras instituições de ensino. O acesso dos usuários será feito por um servidor web e desenhado para oferecer as vantagens de se estar no laboratório e motivar os estudantes.

O trabalho apresenta todas considerações teóricas fundamentais à implantação e define equipamentos a serem usados na construção do laboratório. Como primeiro Laboratório Remoto a ser construído na Universidade realizará o controle de um Elevador de Carga. O intuito da experiência realizada remotamente é acrescentar e não substituir a presença em laboratórios reais.

Palavras-chave: Laboratório Remoto; Engenharia; LabVIEW; Arduino; e-learning.

Project of a Remote Cargo Lifter Controlled Laboratory

Abstract

This graduation project discusses ways to implement and present a solution to control a Cargo Lifter locally and remotely. An assembly of a remote laboratory with experience in didactic experiments was proposed to help some graduation students of the Pontifical University of Rio de Janeiro, PUC-Rio. The experiment will be executed locally or remotely through the Internet using virtual instruments developed in LabVIEW (acronym for Laboratory Virtual Instrument Engineering Workbench) with the interaction by Arduino and in the future to be shared with other institutions. In addition, user's access will be done by a web server. This article presents all theoretical considerations and definitions to be used on a lab construction. The experiment goal is add more ways to motivate students, the goal is not replace physical lab classes.

Keywords: Remote Lab; Engineering; LabVIEW; Arduino; e-learning.

Sumário

1. Introdução 9

2. Motivação 10

3. Pré-requisitos Teóricos para o Projeto e Montagem do iLab 11

4. Discussão 12

4.1. Controle in loco X Controle Remoto 12

4.2. Microcontrolador 12

4.3. Ambiente de Desenvolvimento Integrado (IDE) 13

4.4. Rede 15

4.5. Servidor 16

5. Descrição do Sistema 17

6. Especificações de Hardware 19

6.5. Arduino (Uno R3) 19

6.6. SparkFun MicroSD Shield 20

6.7. MMA8452Q Micro Acelerômetro 20

6.8. Sensor 21

6.9. Circuito Integrado 7432 22

6.10. Resistores 22

6.11. Reguladores de Tensão 22

6.12. Diodo 22

6.13. Bateria de Níquel - Cádmio 22

6.14. Motor Elétrico 22

6.15. Condutores 22

6.16. Polia 22

6.17. Cabine 22

6.18. Guias 22

6.19. Contrapeso 23

6.20. Cabo 23

6.21. Rolamentos 23

6.22. Estrutura de MDF 23
6.23. Amortecedor de fim de curso (batente) 23
7. Sistema de Controle 24
8. Controle Local X Controle Virtual 25
8.1. Controle Local 25
8.2. Controle Virtual 26
9. Desenho e Descrição da Parte Eletrônica – Hardware 28
10. Arquitetura de Rede – Acesso Remoto 30
11. Web Server– Acesso Remoto 31
12. Conclusão 35
13. Referências Bibliográficas 36

Lista de Figuras

Figura 1 - Placa Microcontroladora 13

Figura 2 - Diagrama de Blocos - LabVIEW 14

Figura 3 - Painel Frontal - LabVIEW 14

Figura 4 - Exemplos de Topologia de Rede 15

Figura 5 - Arquitetura de Rede - Ilab MIT 16

Figura 6 - Foto do Elevador de 4 Níveis - Vista Frontal 17

Figura 7 - Indicador botão físico e botão virtual - Painel Frontal 18

Figura 8 - Arduino UNO R3 19

Figura 9 - Micro SD Shield SparkFun 20

Figura 10 - Acererômetro SparkFun 21

Figura 11 - Fluxo de Potência proposto em [16] 24

Figura 12 - Bateria 25

Figura 13 - Pull Down 26

Figura 14 - Diagrama de Blocos - Controle Virtual 27

Figura 15 - Circuito Elevador Local 28

Figura 16 - Topologia de Rede Estrela 30

Figura 17 - Habilitando servidor 32

Figura 18 - Habilitando o servidor do painel remoto 32

Figura 19 - VIs acessíveis e lista de clientes permitidos. 32

Figura 20 - Web Publishing Tool 33

Figura 21 - Operar painel remoto 33

Figura 22 - Acesso do painel remoto via browser 34

1. Introdução

Idealizado por volta de 1990 em instituições de ensino ao redor do mundo, os iLabs (laboratório remoto) vêm sendo implantados em diversas Universidades ao redor do mundo, aumentando as possibilidades de aprendizado.

O laboratório remoto é definido como um espaço com equipamentos operados à distância pela Internet. Sua única diferença para um laboratório convencional é o controle feito remotamente, não precisando da presença física de alunos e professores no mesmo lugar dos equipamentos.

No projeto proposto e desenvolvido nesse artigo utilizaremos um modelo já desenvolvido de um sistema de um elevador de tração, [16].

Faremos uma **interface virtual** dividida em duas partes: Indicador Físico e Botão Virtual; Implantação da **comunicação via Internet** para controle remoto e **arquitetura de circuito eletro-eletrônico** para atuação e aquisição de dados.

O cliente (usuário) terá a possibilidade de visualizar e controlar as variáveis do sistema por meio de uma interface gráfica através do repositório institucional da PUC-Rio, http://www.maxwell.vrac.puc-rio.br/ ou por meio dos botões físicos.

O iLab implementado é composto por quatro camadas em sua estrutura como sugerido em [3]:

I) A primeira camada ou Rig é a base da estrutura geral, equipamentos de aquisição de dados e o processo físico no iLab. O requerimento e a captura de dados são atribuições dessa camada;

II) A segunda camada comporta o pacote de mídia e de interação. O pacote de mídia contém materiais digitais para a contextualização do Rig e as práticas laboratoriais, o pacote de interação permite o uso do laboratório, somente por meio de agendamento;

III) Objetivos pedagógicos individuais implementados no Rig através do pacote de interação compõem a terceira camada;

IV) A quarta, e ultima camada é a disciplina em si, aulas com diferentes tarefas e prevê o uso de interatividade entre disciplinas e conteúdos acadêmicos.

A construção do Laboratório Remoto envolve uma série de diferentes teorias de Engenharia Elétrica, Computação, Mecânica, Controle e Automação. A arquitetura do sistema de supervisão, controle e manipulação dos dados colhidos exigem conhecimentos de disciplinas dos curso de Elétrica, Controle e Automação principalmente. Já o processo de comunicação usuário – iLab via Internet exige conhecimentos básicos do curso de Computação.

Além do uso para disciplinas em que há o uso das funções do Ilab, a manutenção e o desenvolvimento do laboratório para diferentes aquisições de dados e outras experiências possibilitará tanto alunos mais avançados no curso quanto professores, a aplicação de métodos e teorias futuramente usadas no mercado, ou seja, alunos se formarão com mais uma competência.

2. Motivação

Devido ao crescente desenvolvimento da tecnologia, vemos uma mudança de comportamento dos alunos e de seus anseios. Essas mudanças fizeram com que o interesse por cadeiras de ensino à distância, EAD, ganhassem maior importância.

Para garantir a adequação a essa nova realidade o Departamento de Engenharia Elétrica (DEE) e de Controle e Automação (DECA) vêm desenvolvendo conteúdo permanentemente, disponibilizando-os no Maxwell, repositório institucional da PUC-Rio.

As disciplinas centrais que compõem os cursos do Centro Técnico Científico (CTC) necessitam em sua maioria de carga horária teórica e prática. A parte prática provida por laboratórios presenciais pode ser auxiliada por práticas remotas possibilitando a flexibilidade de horários de estudantes e professores, maior interação pessoal entre alunos e professores do que em grandes turmas, desenvolve habilidades dos alunos com softwares de engenharia e pode se dividir os custos de implantação e manutenção do laboratório com outras instituições de ensino, haja vista que cada universidade pode ter o seu próprio laboratório de experiências remotas focado em um assunto. Porém, os intrutores e/ou professores responsáveis devem se atentar aos alunos que se sentirem isolados pelo fato da distância implícita a natureza do sistema, assim como alunos se sentirem perdidos pela liberdade de realizar os experimentos de acordo com seu agendamento, conexões de internet lentas e computadores antigos também são um desafio para o sucesso do projeto.

A experiência remota possibilita uma série de vantagens tanto para estudantes, professores e instituições de educação. A construção do laboratório remoto na Universidade irá proporcionar uma oportunidade no desenvolvimento das cadeiras oferecidas assim como na criação e melhoramento da qualidade dos cursos à distancia.

3. Pré-requisitos Teóricos para o Projeto e Montagem do iLab

Para começarmos as atividades de pesquisa para construção do laboratório foi necessário o nivelamento do conhecimento em teoria de controle, controle de processos, teoria básica em Instrumentação Industrial e softwares LabVIEW e Arduino. Tais conhecimentos foram utilizados para a especificação do projeto, tal como escolher equipamentos para a montagem do processo remotamente controlado, a fim de se fazer experiências didáticas. Os seguintes assuntos foram estudados:

a) Modelos de Sistemas;

b) Modelos por Variáveis de Estado, Equação Dinâmica;

c) Sistemas Compostos;

d) Propriedades, Controlabilidade, Observabilidade e Estabilidade de Sistemas;

e) Localização de Raízes de Polinômios;

f) Método do Lugar das Raízes;

g) Método de Nyquist;

h) Realimentação de Estado;

i) Mapa de Kaurnaugh

j) Observador de Luenberger;

k) Condicionamento de sinais;

l) Protocolos de Comunicação;

m) Sensores e Atuadores;

n) Sistemas de Comunicação;

o) Circuitos Elétricos e Elétrônicos;

p) LabVIEW;

q) Arduino;

r) Linguagem C/C++;

s) Ferrramenta Digilent LINX;

t) Servidor Web.

4. Discussão

Com tantas opções de equipamentos de aquisição de dados de distintos fabricantes, pacotes de interface de software – hardware de diferentes linguagens de fornecedores diversos, bibliotecas de funcionalidades dedicadas e mais infinitas possibilidades de integração feito por meio de circuitos elétricos com controle feito por instrumentos virtuais, temos que avaliar o custo benefício da arquitetura de nosso sistema.

Nesta seção analisaremos diferentes formas de solução para a o problema delimitado no início: Laboratório Remoto de Controle de Elevador de Carga.

4.1. Controle in loco X Controle Remoto

A segurança em sistemas automatizados é quesito de protagonismo no desenvolvimento de soluções.

A divisão em controle local e controle remoto diminui o risco de comando falho, essa diminuição é proporcional ao grau de modularização do projeto.

Várias opções de compartimentalização são possíveis.

Podemos ter somente um microcontrolador para aquisição de dados e atuação no sistema, com o mesmo circuito elétrico, assim a diferença do controle local versus controle remoto ficaria nas funções chamadas por uma função principal de um instrumento virtual, VI, funcao_local e funcao_remoto *slaves* de uma funcao_master, por exemplo.

Dois microcontroladores, cada um com seu circuito elétrico e seu próprio algoritmo, com dois sistemas operando separadamente pode ser uma solução.

Podemos, também, ter um sistema já pronto somente para controle local e termos que adicionar uma solução remota para incremento do sistema. Ou podemos ter a situação inversa, solução remota pronta e adição da solução local. Assim precisamos entender o que já está feito e adaptar a solução.

Esse primeiro tópico da Discussão, nos servirá de ponto de partida para o entendimento do que precisamos construir considerando fatores como custo benefício.

4.2. Microcontrolador

Como dito na Introdução, temos a opção de diversas placas microcontroladoras.

Uma placa de desenvolvimento de microcontroladores é uma placa de circuito impresso (PCB) com circuitos e hardware projetado para facilitar a experimentação com determinados recursos de placas de microcontrolador. As placas de desenvolvimento são combinadas com um processador, memória, como LCD, teclado, USB, porta serial, ADC, RTC, CIs de driver de motor, slot para cartão SD, Ethernet, com recursos de depuração. Isso nos salvará de mexer com as conexões com fios de jumper e a placa.

Para a solução de nosso problema, podemos combinar placas de desenvolvimento como UNO R3 da Arduino com placas de aquisição de dados do fabricante National Instruments (DAQ – MX1000); podemos combinar diferentes placas Arduino com diferentes *shields* de que irão adicionar funcionalidades extras ao microcontrolador; ainda, podemos utilizar o mesmo microcontrolador conectado em série ou paralelo, dependendo de nossa solução.

Existem placas de fabricantes distintos com funcionalidades semelhantes, porém com material constitutivo e design diferentes que podem ser motivo de escolha em determinados experimentos e não em outros. Fatores como resistência do material, partes de proteção física de componentes e outras características dos materiais não podem passar despercebidas.

Fatores como custo, material disponível, facilidade de implementação e muitos outros fatores devem ser variáveis de maior peso no processo de arquitetura do sistema.

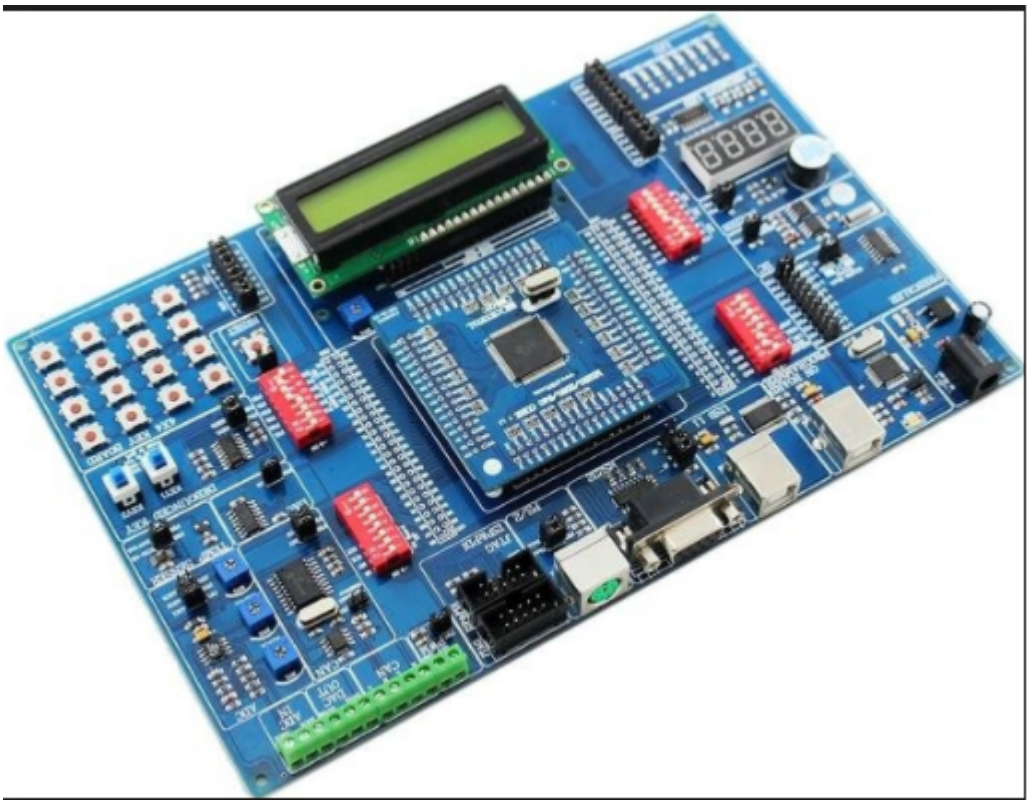

Figura 1 - Placa Microcontroladora

4.3. Ambiente de Desenvolvimento Integrado (IDE)

A diferença entre as possíveis escolhas é bastante clara.

Caso optemos por desenvolver o código por linguagem sequencial, lista de métodos e padrões para serem seguidos e concluídos, devemos optar pela IDE do Arduino.

O Arduino IDE é uma aplicação escrita em Java multiplataforma. Foi desenvolvido para introduzir a programação a pessoas não familiarizadas com o desenvolvimento de software e possui estrutura simples. Com duas funções, setup e loop, conseguimos executar muitos programas. A biblioteca wiring permite programar em C/C++. A italiana Smart Projects fabrica o arduino original, porém a americana SparkFun também possui algumas marcas sob a mesma licença.

A possibilidade de usar uma linguagem orientada à objetos, pode ser satisfeita utilizando o LabVIEW IDE. A linguagem Labview, figura 2, conecta fios em blocos de funções e é lido da esquerda para a direita. Através de dezenas de blocos que fazem funções como a soma de valores, leitura de sinal recebido do microcontrolador e até mesmo blocos que fazem o controle Proporcional, Integrativo e Derivativo (PID), diagrama de blocos. Atrelado a esse diagrama de blocos existe o painel frontal, figura 3, onde existem atuadores e indicadores virtuais para funcionamento do sistema escrito em blocos. Também é possível instalar *add-ons*, Arduino – LabVIEW Interface por exemplo, pacotes de ferramentas específicas para realizar determinadas experiências, como pacote de aquisição de dados, possibilitam a integração de vários sistemas. Caso desejemos incorporar código escrito em C ou qualquer outro código, o LabVIEW aceita.

A IDE do LabVIEW ainda possui opções práticas, poucos cliques, para tarefas complexas como configurar um servidor Web e configuração simples de placas de dados de fabricação própria e também de outros fabricantes.

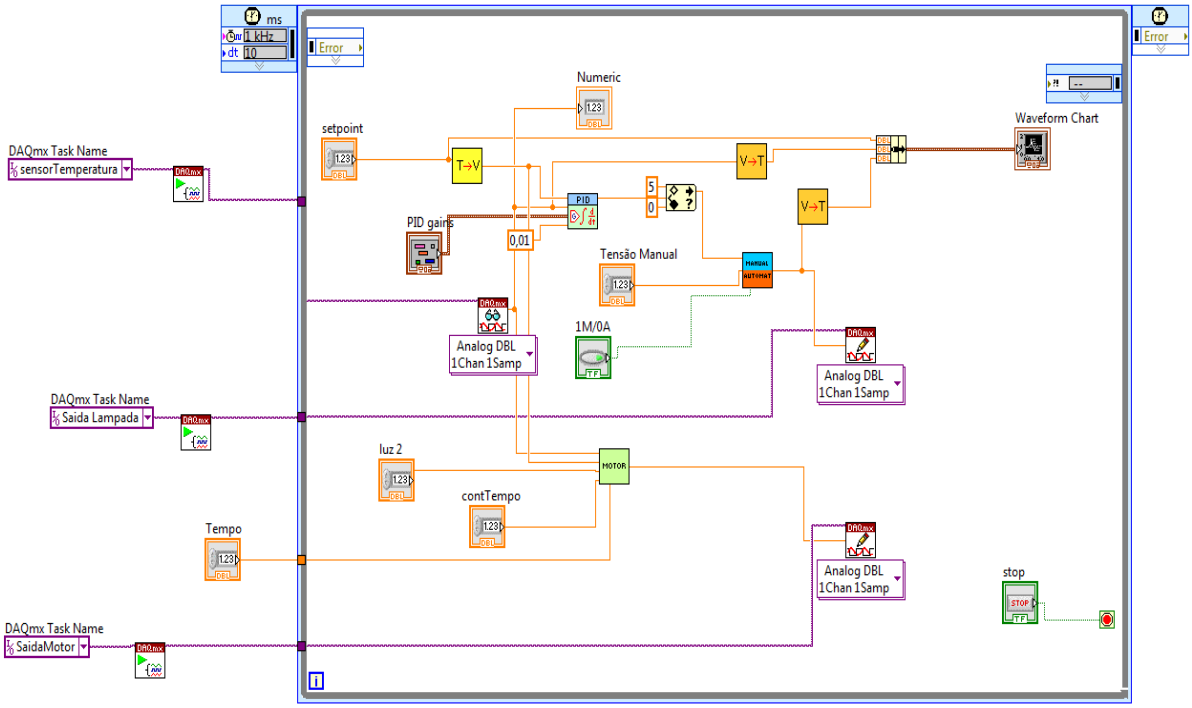

Figura 2 - Diagrama de Blocos - LabVIEW

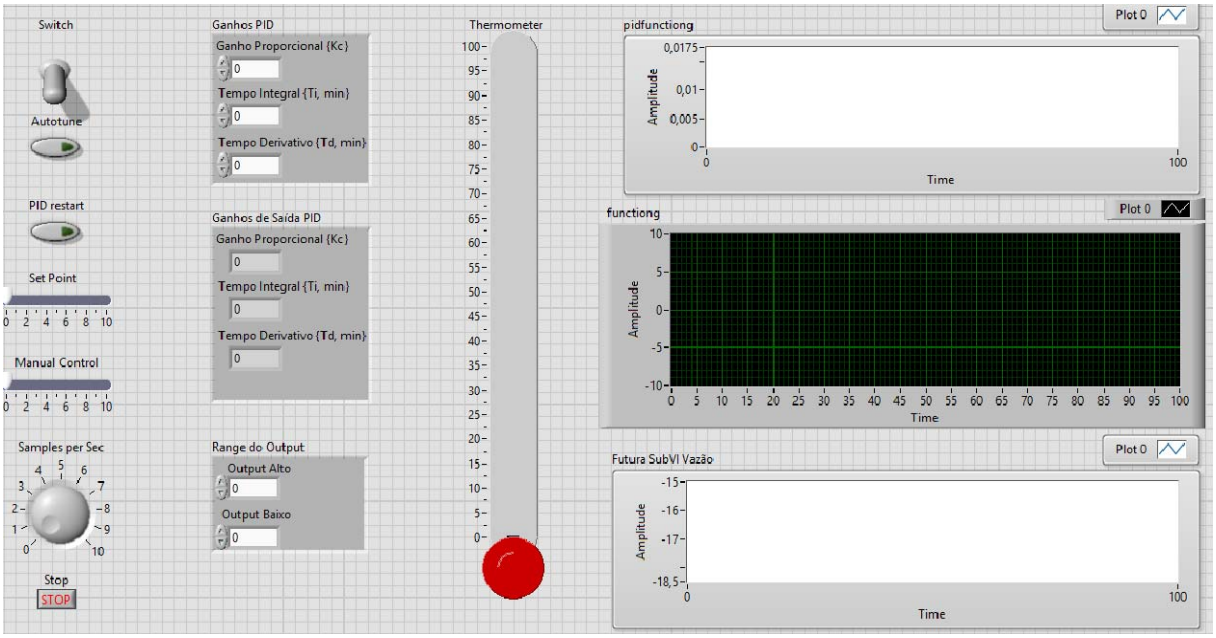

Figura 3 - Painel Frontal – LabVIEW

4.4. Rede

A topologia de rede é a estrutura que está interconectada através de recursos de comunicação, geograficamente distribuídos, permitindo a troca de dados entre estas unidades e podem ser caracterizados de duas formas: Topologia Física e topologia lógica.

A topologia física é a aparência do *layout* da rede, representa a forma como os cabos são conectados e os nós distribuídos. Essa característica influencia pontos críticos como flexibilidade, velocidade e segurança.

Enquanto a topologia lógica descreve o fluxo de dados atráves da rede. Refere-se a maneira como os sinais são transmitidos entre os dispositivos sem contar a interligação física entre eles.

Existem algumas topologias básicas de rede, Figura 4.

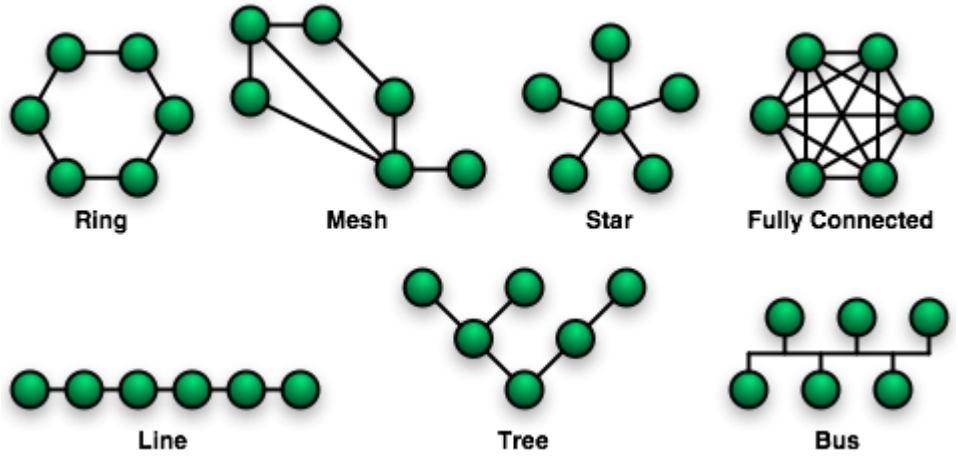

Figura 4 - Exemplos de Topologia de Rede

A estrutura da rede inclui hardware, camadas funcionais, interfaces e os protocolos de comunicação entre os nós.

A arquitetura de rede para laboratórios remotos deve ser a mais básica possível, um servidor hospeda a aplicação e é acessado pelo usuário via Internet. O servidor é conectado, na outra ponta, aos instrumentos do laboratório.

A configuração do servidor pode ser algo muito complicado ou pode-se usar ferramentas prontas. Pode-se também usar uma placa controladora, *shield*, para acesso à Internet.

O acesso ao servidor pode ser feito diretamente ou por meio de *service broker*, figura 5, a definição do acesso é feita levando em consideração se o usuário terá acesso direto ao servidor ou não.

A comunicação da placa ou das placas de controle com o servidor do experimento pode ser feita através dos padrões: USB, serial, ethernet ou qualquer interface de comunicação disponível num computador.

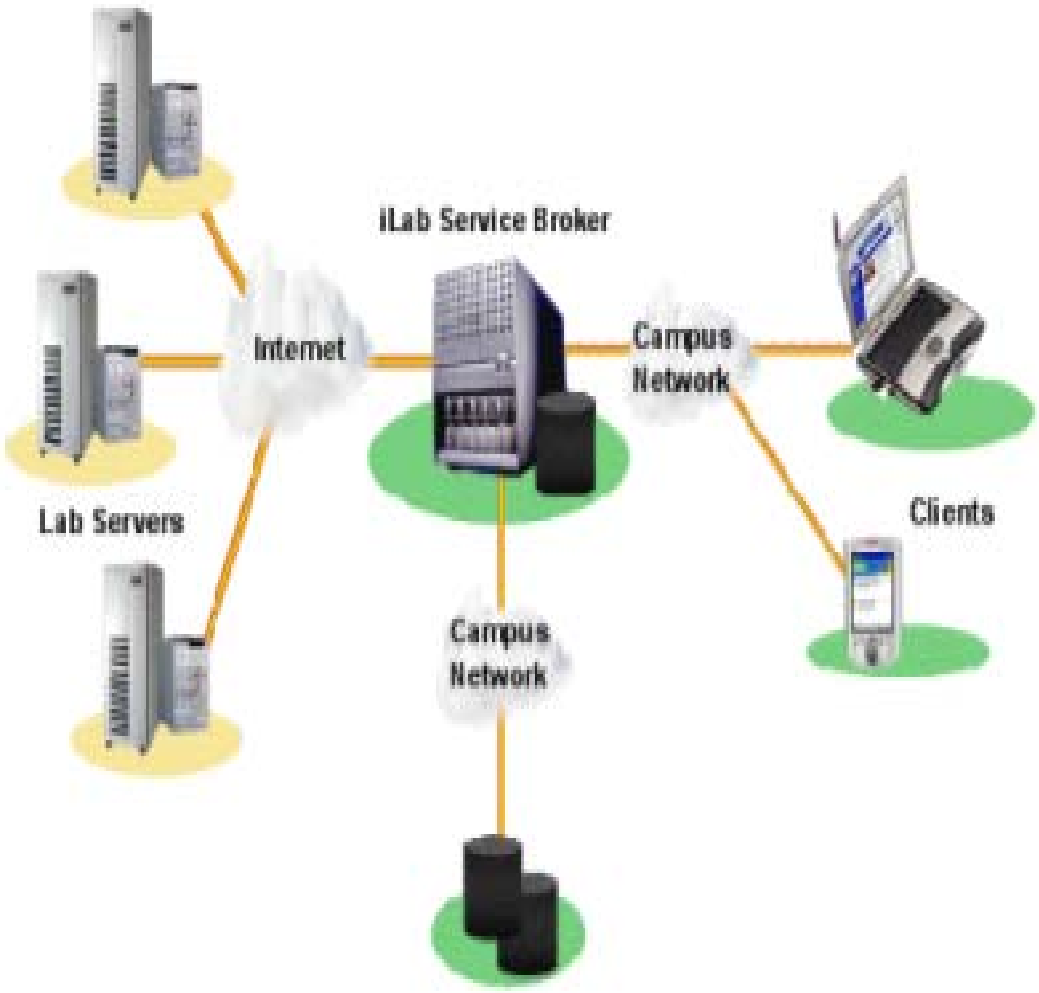

Figura 5 - Arquitetura de Rede - Ilab MIT

4.5. Servidor

Servidor web é o software responsável por aceitar pedidos HTTP de clientes e devolver essas requisições também em HTTP.

O processo começa quando o computador do cliente solicita conexão para o servidor web instalado em outro computador, como não sabemos o momento que essa solictação será feita, o servidor web precisa estar sempre disponível.

Tudo que se enquadre como arquivo pode ser enviado como resultado de um pedido HTTP, programas e scripts podem ser executados nos servidores web.

A origem do conteúdo de resposta do pedido do cliente pode ser estática ou dinâmica, no caso de uma página dinâmica como o laboratório remoto, o pedido é recebido e processado pelo servidor web que vai criar o conteúdo dinamicamente, por meio do labview por exemplo, e enviará ao cliente.

5. Descrição do Sistema

O presente trabalho visa projetar um sistema de controle remoto de um elevador de tração, já existente [16], – Figura 6.

A montagem do sistema é composta de 9 elementos centrais, são eles:

- Elevador de tração de quatro níveis;
- Placas Arduino – UNO R3;
- Placa microSD shield sparkfun;
- Linguagem Arduino (linguagem proveniente do C++);
- Linguagem LabVIEW (linguagem orientada à objeto);
- Circuito Eletro-eletrônico;
- Sensores e atuadores;
- Maxwell;
- *Web Service Request* – pacote de comunicação do LabVIEW.

Figura 6 - Foto do Elevador de 4 Níveis - Vista Frontal

O sistema de funcionamento e interação com o elevador é dividido em três partes principais:

1. A primeira parte compreende a modelagem do elevador funcionando no laboratório sem conexão com a internet (desenvolvido anteriormente). A placa Arduino UNO R3, juntamente com a placa microSD shield sparkfun e o micro acelerômetro, fazem com que o elevador funcione perfeitamente com comandos realizados in loco, apertando os botões físicos;

2. A segunda parte é o instrumento virtual, módulo responsável por fazer o controle virtual e monitorar o controle local através da interface gráfica dividida em dois paíneis principais, indicador virtual e botão físico, entradas digitais, – Figura 7. Abas de números 1, 2, 3 e 4 somente para sinalização de ruídos, entradas analógicas.

3. A terceira parte é o comando do elevador feito remotamente. O sistema conecta duas pontas, usuário – elevador de tração. Porém, para o usuário conseguir manipular o equipamento, é necessária a construção de um sistema confiável, seguro, rápido e de fácil entendimento. Isso é feito através do pacote de comunicação do LabVIEW, *Web Service Request*.

Baseado nessas características, a arquitetura do sistema é composta por um controlador (computador), rede de comunicação (internet), servidor, equipamentos de atuação e aquisicação de dados, condutores, componentes elétricos e eletrônicos para circuito de comando, circuito de alimentação e câmera para acompanhamento visual da experiência.

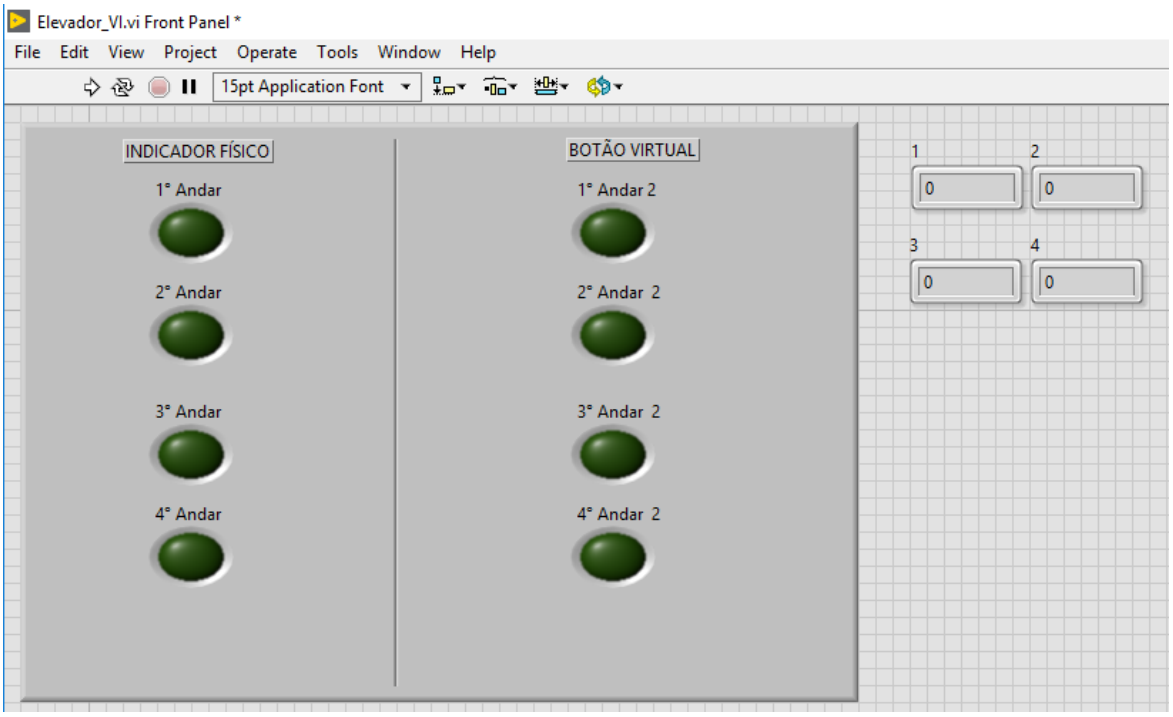

Figura 7 - Indicador botão físico e botão virtual - Painel Frontal

6. Especificações de Hardware

A simplificação de nosso elevador consiste numa estrutura sólida de MDF com quatro níveis, uma cabine, quatro botões de chamada instalados em cada pavimento, quatro sensores, uma bateria para alimentação do sensor da cabine, duas placas de controle (arduíno), uma placa microcontroladora (micro SD shield sparkfun), uma placa com acelerômetro, uma *breadboard* com circuito eletro-eletrônico, um servidor para hospedagem do software de controle e conexão com a internet, um motor elétrico, alimentação das placas arduíno e alimentação do circuito.

6.1. Arduino (Uno R3)

Uma típica placa Arduino Uno Rev3 é uma placa microcontroladora que têm 14 pinos de entrada/saída digitais, 6 entradas analógicas, um cristal *quartz* de 16MHz, uma conexão USB, uma entrada para alimentação, entrada ICSP e um botão de reset. Ela em si não possui qualquer recurso de rede e possui pouca capacidade de memória por isso é comum combinar uma ou mais placas para objetivos distintos. A interface é simples, podendo ser escrita em várias linguagens. A mais popular é a Processing.

Responsável por controlar os movimentos de subida e descida do protótipo.

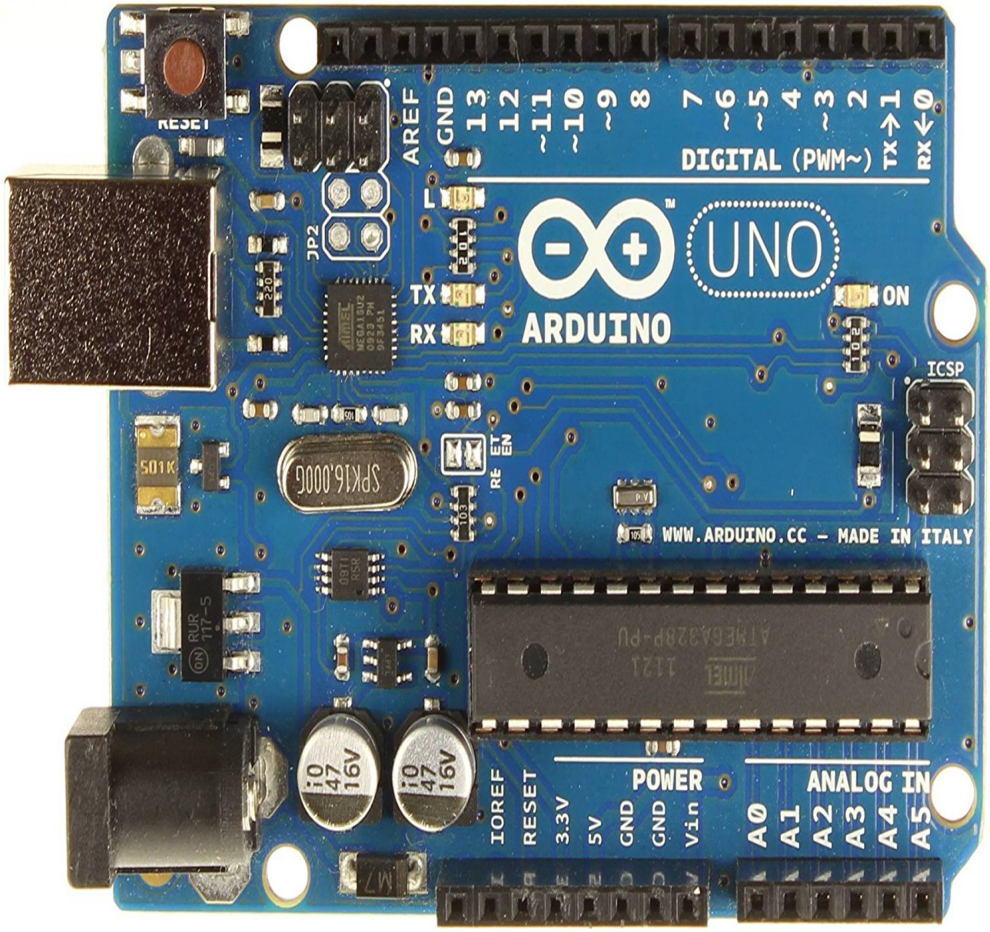

Figura 8 - Arduino UNO R3

6.2. SparkFun MicroSD Shield

Essa placa (*shield*) dá suporte para a placa Arduino processar a aplicação sem a necessidade de um computador, nessa memória fica hospedada a programação para execução do controle do elevador .

Os *shields* Arduino são placas de circuitos modulares que se encaixam no Arduino para incutir funcionalidades extras. Se desejar conectar o Arduino à Internet e postar no Twitter, existe um *shield* para isso. Existem dezenas (centenas) de *shields* por aí, todos os quais tornam o Arduino mais do que apenas uma placa de desenvolvimento.

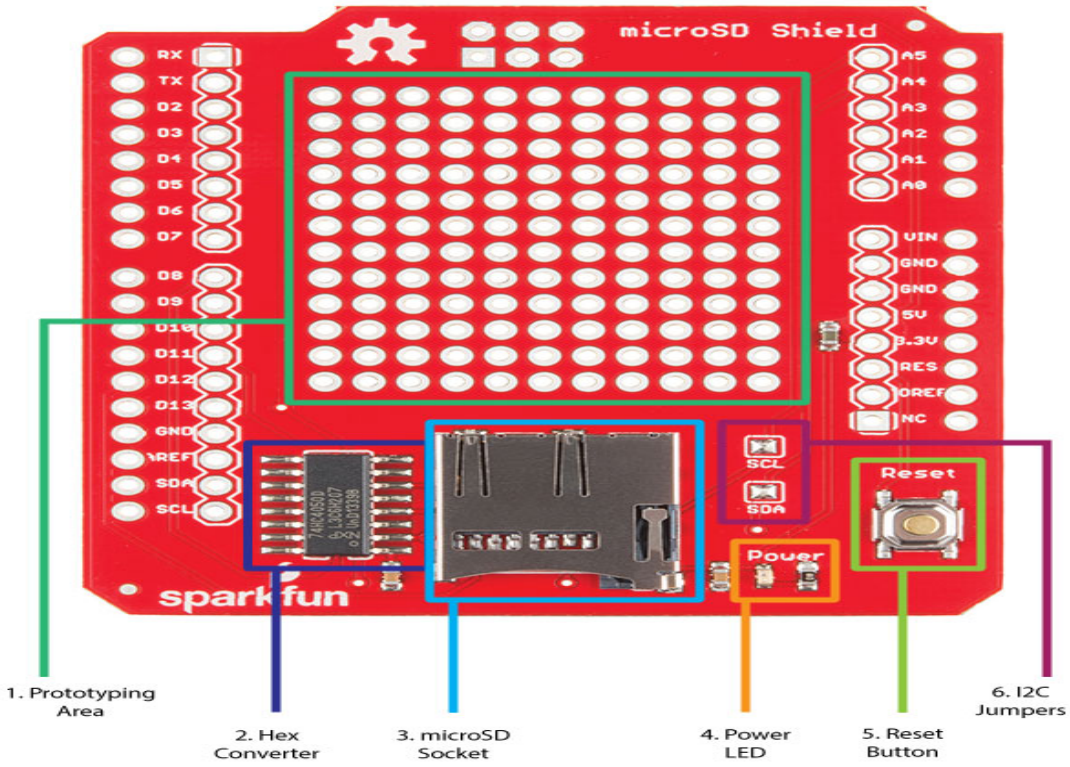

Figura 9 - Micro SD Shield SparkFun

6.3. MMA8452Q Micro Acelerômetro

Placa de fuga com acelerômetro micro usinado capacitivo de baixa potência com 12 bits de resolução, utilizado em projetos que precisem sentir orientação ou movimento.

Um acelerômetro capacitivo funciona de modo que a aceleração no dispositivo desloca uma placa móvel de um capacitor em relação a placas fixas no dispositivo. Desta, forma altera-se a capacitância de cada capacitor.

O acelerômetro usado na placa mede vibração em três eixos: X, Y e Z. Quando o objeto de estudo é acelerado, a medida será maior que 0; quando atinge velocidade constante, a medida será 0; e quando o objeto desacelera, o contrário acontece, a medida será menor que 0. Ruídos podem acontecer, assim cabe a correta determinação de valores aceitáveis para uso.

A placa contém seis pinos como podemos ver na Figura 10.

- Uma entrada para o terra, 0V;
- Uma entrada para a alimentação da placa, 1,95 – 3,60V;
- Uma entrada/saída, i/o, bi-direcional serial, I²C sinal de dados (SDA);
- Uma entrada/saída, i/o, bi-direcional serial, I²C sinal de *clock* (SCA);
- Duas saídas, I1 e I2, interrupção programável que pode indicar dados prontos, mudança de orientação, toque.

Figura 10 - Acererômetro SparkFun

6.4. Sensor

Utlizamos sensores fotoelétricos com um emissor e quatro receptores. O emissor fica localizado na lateral esquerda da cabine, alimentado por bateria independente localizada no topo da cabine. Os quatro receptores ficam localizados um em cada pavimento na parte de

dentro da estrutura, tem três fios (botão de chamada, comunicação e alimentação), alimentados pela alimentação geral.

São responsáveis por comunicar para o controlador (arduino) qual o andar do elevador e auxilia a regulagem de velocidade do motor.

6.5. Circuito Integrado 7432

O circuito integrado 7432 contém catorze pinos e quatro portas OU.

6.6. Resistores

Auxiliam na parte eletrônica doze resistores de 1kΩ e dois resistores de xxxΩ.

6.7. Reguladores de Tensão

Auxiliam na parte eletrônica quatro reguladores de tensão de xxxV.

6.8. Diodo

Auxiliam na parte eletrônica quatro diodos.

6.9. Bateria de Níquel - Cádmio

Bateria independente de 4,8V (DC) – 600mAh localizada no topo da cabine, alimenta o emissor localizado na parte lateral da cabine.

6.10. Motor Elétrico

Motor (motor module VB-1) de xxV e xxxW para movimentação vertical do elevador. Alimentação feita por dois fios, localizado no topo da estrutura, conectado a polia para realização do torque .

6.11. Condutores

Na lateral esquerda na parte da frente fica localizado o caminho (eletroduto) que os condutores fazem para conectar os sensores e botões de cada andar, o motor elétrico à *breadboard* e ao controlador.

6.12. Polia

A polia é uma peça mecânica muito comum a diversas máquinas, utilizada para transferir força e energia cinética. A polia é constituída por uma roda de material rígido. Acionada pelo cabo a polia gira em um eixo, transferindo movimento e energia a outro objeto.

A polia tem inércia não desprezível e está apoiada em mancais com dissipação.

6.13. Cabine

Cabine de MDF sustentada pelos cabos e atrelada ao contra peso, massa não desprezível.

6.14. Guias

Servem para guiar o movimento do elevador, são trilhos para a correta translação. São três guias de cada lado da estrutura (vista frontal), seis no total, quatro para guiar a cabine e duas para guiar o contra peso.

6.15. Contrapeso

Esse artifício é usado para equilibrar a carga, serve como resistência ao movimento.

6.16. Cabo

Cabo ligado em duas pontas, cabine e contra peso. Possue flexibilidade, não são rígidos mas não deslizam em relação a polia e não podem ser modelados como massa desprezível.

6.17. Rolamentos

São dez rolamentos no total, quatro de cada lado da cabine para o deslizamento nas guias e dois no contra peso.

6.18. Estrutura de MDF

Armação de sustentação do sistema feita com MDF.

6.19. Amortecedor de fim de curso (batente)

Só atua em caso de emergência, representado por um conjunto mola-amortecedor.

7. Sistema de Controle

O controle do sistema é realizado por quatro sensores de presença; quatro receptores, um em cada andar e um emissor na cabine; um atuador, motor elétrico localizado no topo do elevador; e um acelerômetro (sensor digital). O sistema de controle tem a mesma lógica tanto para processo realizado localmente, tanto para o processo feito remotamente. Após o sistema estar completamente alimentado e ligado, o controle do sistema do elevador de tração de quatro pavimentos, consiste na seguinte lógica:

- O botão de chamada é pressionado localmente ou virtualmente em um determinado andar. Essa ação dispara um sinal (voltagem), através do condutor (fio), para o controlador (arduino). O controlador por sua vez realiza uma série de conferências para entender três coisas: Qual andar chamou o elevador, aonde se encontra a cabine nesse momento e qual vai ser a ordem de comando para o motor girar (sentido horário ou sentido anti-horário);

- Caso a cabine se encontre no andar acima da onde foi chamado, o motor gira num sentido. Caso a cabine se encontre no andar abaixo da onde foi chamado, o motor gira no sentido contrário;

- Com a ordem para atuação do motor, a cabine se movimenta, o emissor passa pelos receptores do sensor fotoelétrico, a cabine para quando o receptor atrelado ao botão de chamada receber emissão de luz do emissor.

- O acelerômetro mede se a cabine está acelerando ou desacelerando, dependendo da amplitude do sinal, e se vai se deslocar para cima ou para baxo, dependendo da polaridade do sinal, maior que zero ou menor que zero.

- O sensor fotoelétrico e o acelerômetro combinados determinam o sentido que a cabine deve se deslocar e controlam potência do motor.

- O sistema funciona de forma simples, o primeiro pavimento a ter o botão de chamada pressionado tem prioridade para a chegada da cabine.

Quando o botão de chamada é apertado novamente, o processo recomeça.

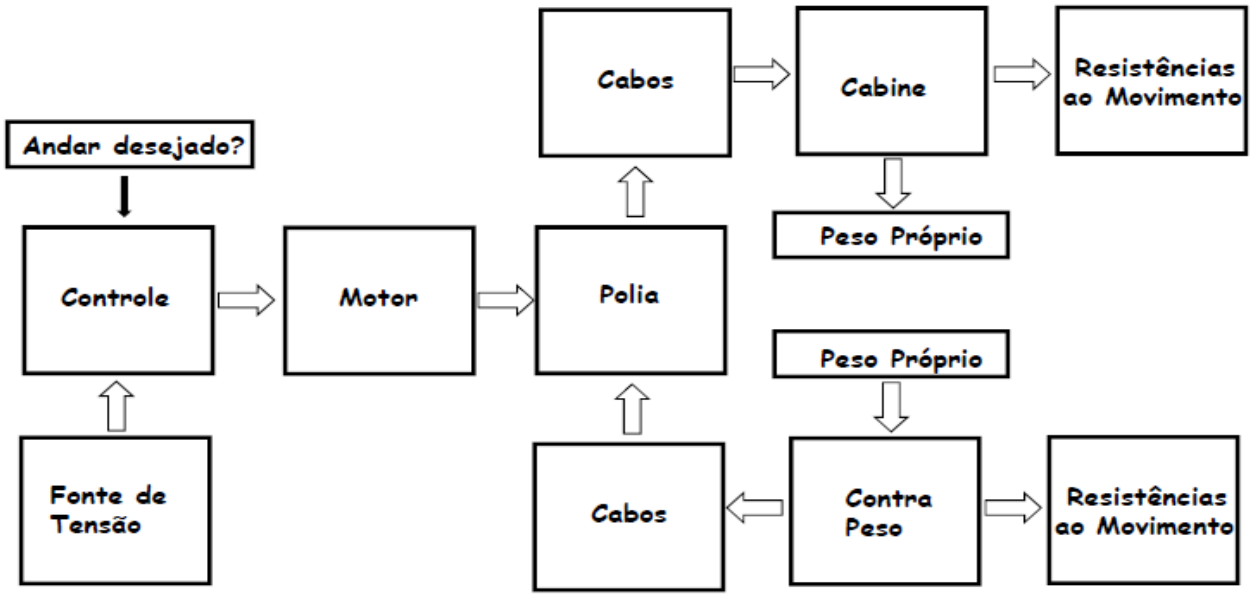

Figura 11 - Fluxo de Potência proposto em [16]

8. Controle Local X Controle Virtual

Como o projeto do elevador já se encontrava pronto com controle local, a decisão foi pela integração com a estrutura para controle virtual.

Para entendermos a escolha que foi feita para a construção do controle virtual, precisamos antes ver a estrutura previamente montada para o controle local.

8.1. Controle Local

O circuito para controle local do elevador de carga foi montado tendo que satisfazer algumas características.

Como o circuito funciona de modo autonômo, sem conexão com um computador, conectamos um adaptador DC (5V, 2A) numa tomada de 127V para entregar a tensão de alimentação do Arduino, 5V.

A alimentação do acelerômetro, micro SD *shield*, sensores e motor foi construida com ajuda de quatro reguladores de tensão, resistores e capacitores. O motivo pelo qual foram usados quatros reguladores de tensão se dá pelo fato de que cada componente possui uma tensão de alimentação e corrente de entrada distintas.

Já a alimentação do emissor do sensor fotoelétrico, localizado na lateral da cabine, foi feita por uma bateria instalada no teto da cabine, figura 12.

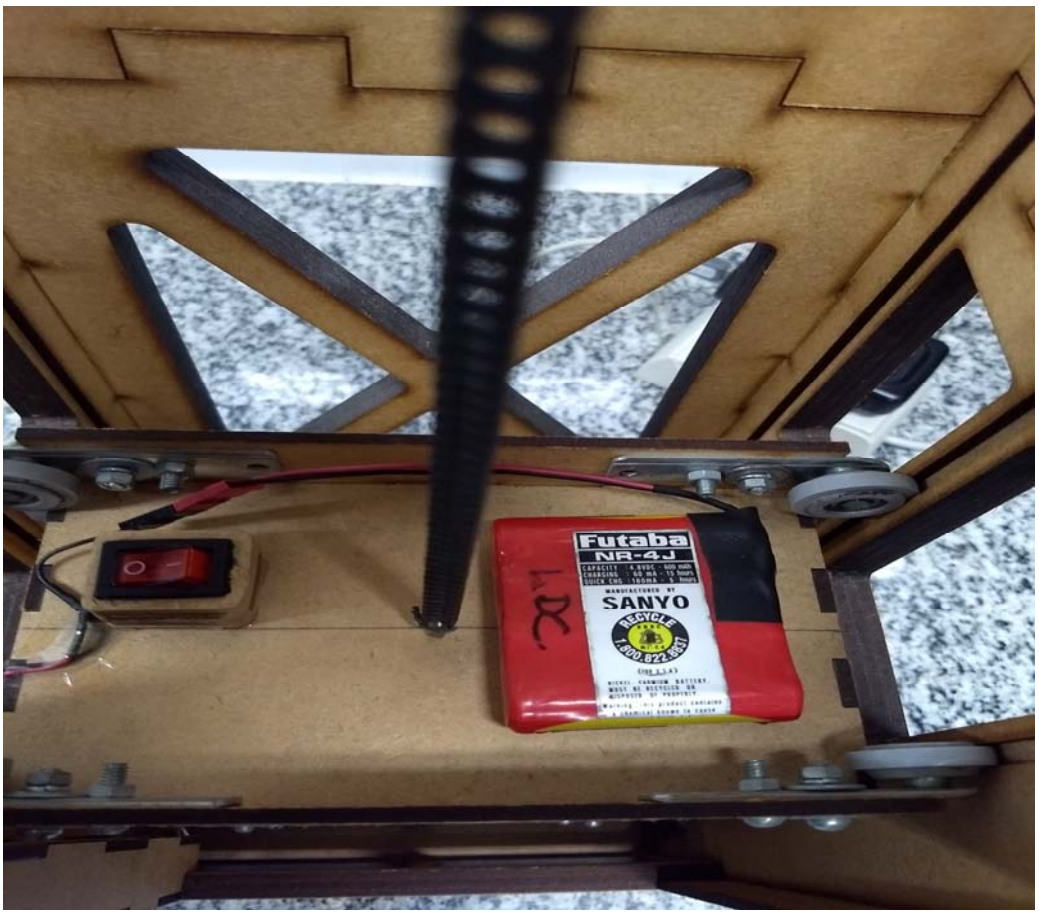

Figura 12 – Bateria

Para a conexão dos botões de chamada de cada nível do elevador na placa Arduino, foi conectado um resistor ligado no terra para a eliminação dos ruídos na aquisição do sinal de comando do elevador, garantindo que o nível lógico das portas estejam em 0 ou em 1. Essa técnica amplamente usada na eletrônica, se chama pull down, figura 13.

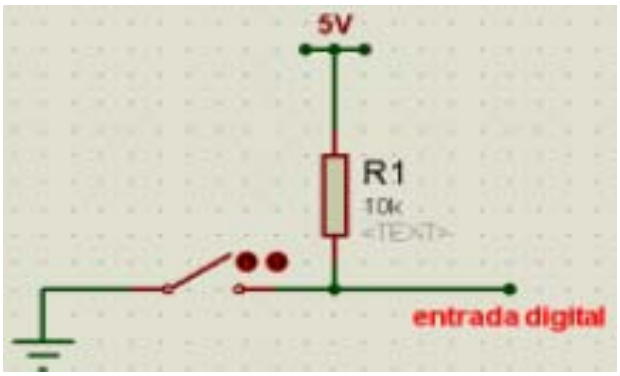

Figura 13 - Pull Down

8.2. Controle Virtual

A escolha foi por outra placa Arduino UNO R3 para aquisição e atuação no circuito que adicionamos ao circuito existente.

Como dito anteriormente em Descrição do Sistema, uma das partes principais é o intrumento virtual, VI, que monitora os botões físicos do elevador e possui botões virtuais também para chamada do elevador.

Conectamos os condutores referentes aos botões de chamada físicos na nova placa Arduino para termos o indicador físico e conectamos 4 saídas digitais para fazerem o papel dos botões virtuais.

Tanto condutores que representam os botões físicos quanto os que representam os botões virtuais foram conectados em uma circuito integrado, 7432, porta lógica OU, e a saída dessa porta foi conectada aonde os condutores dos botões físicos ficavam conectados no controle local. Essa conexão leva o sinal de saída da porta OU, sinal de chamada do elevador, para a outra placa Arduino que já tomava a decisão de chamar o elevador.

Para o perfeito funcionamento do circuito integrado, foi usado novamente o pull down na saída do botão virtual.

Apesar do uso da placa Arduino, foi usado o programa LabVIEW para desenvolvimento do programa de aquisição de dados e atuação. Isso foi possível devido a ferramenta LIFA.

A Interface LIFA (LabVIEW Interface For Arduino) está disponível como um pacote de Instrumentos Virtuais através da Rede de ferramentas LabVIEW. Para o LabVIEW, o Arduino aparece como um dispositivo de instrumento serial. Para se comunicar com instrumentos de série no LabVIEW, precisamos ter a versão mais recente do driver NI-VISA.

O diagrama de blocos do controle virtual, figura 14, é dividido em 3 blocos. O primeiro bloco é para aquisição de sinais do botão físico, entradas digitais. O segundo bloco é para atuação no sistema, botões virtuais de chamada do elevador, saídas digitais. O terceiro bloco é dispensável, serve somente para conferir o nível lógico dessa sáida, foi usado para o uso do pull down.

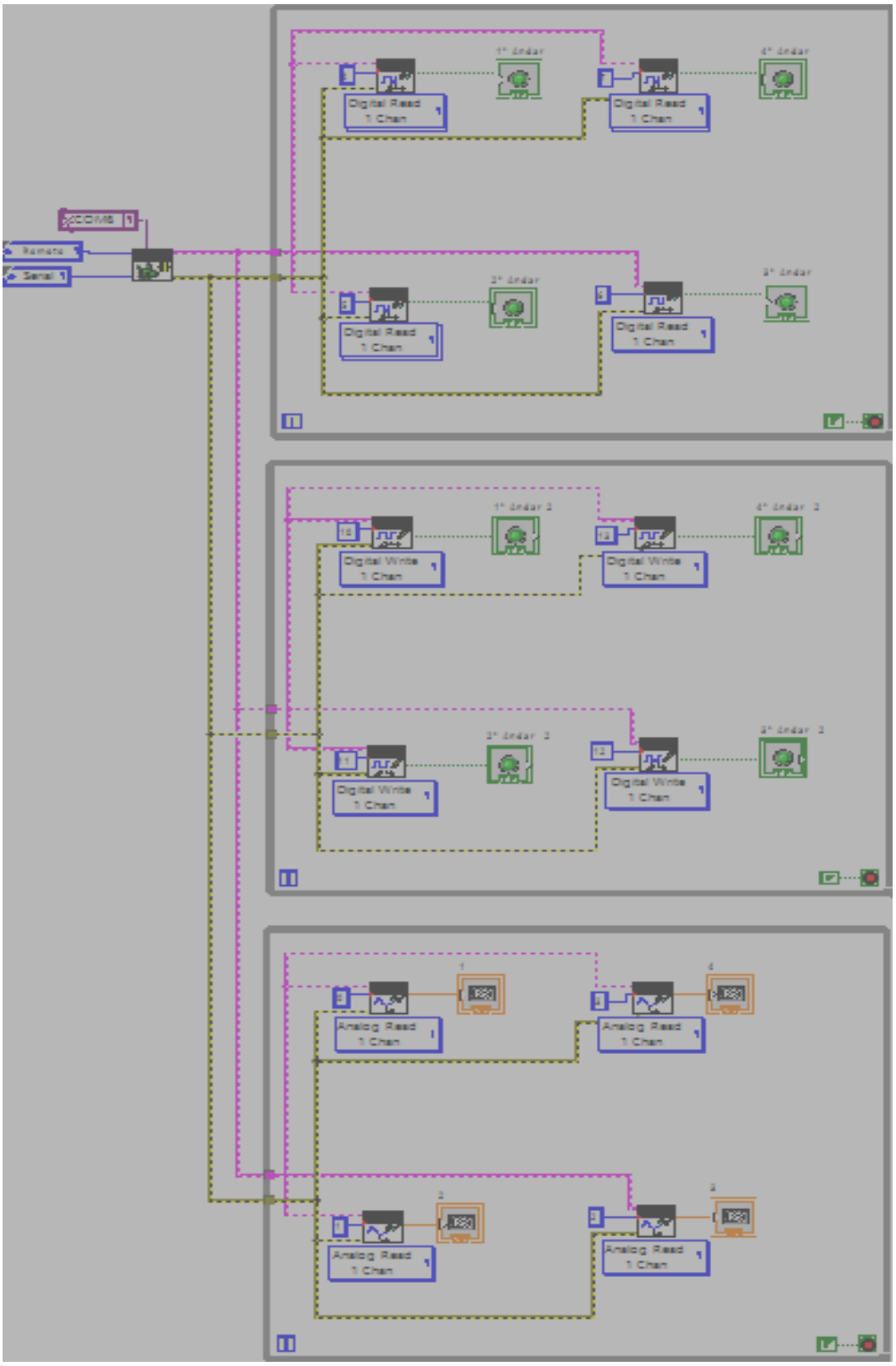

Figura 14 - Diagrama de Blocos - Controle Virtual

9. Desenho e Descrição da Parte Eletrônica – Hardware

O circuito autônomo de controle do elevador, figura 15, se divide em algumas partes.

Os nós dos botões 1, 2, 3 e 4 estão representados duas vezes para melhor visualização do desenho, eles representam os mesmos botões, quatro no total, botões de chamada dos andares. Todos os condutores vindo dos botões de chamada estão em pull down para somente ter dois estados, 0 e 1.

Os terminais de alimentação das placas não foram colocados para simplificação do desenho.

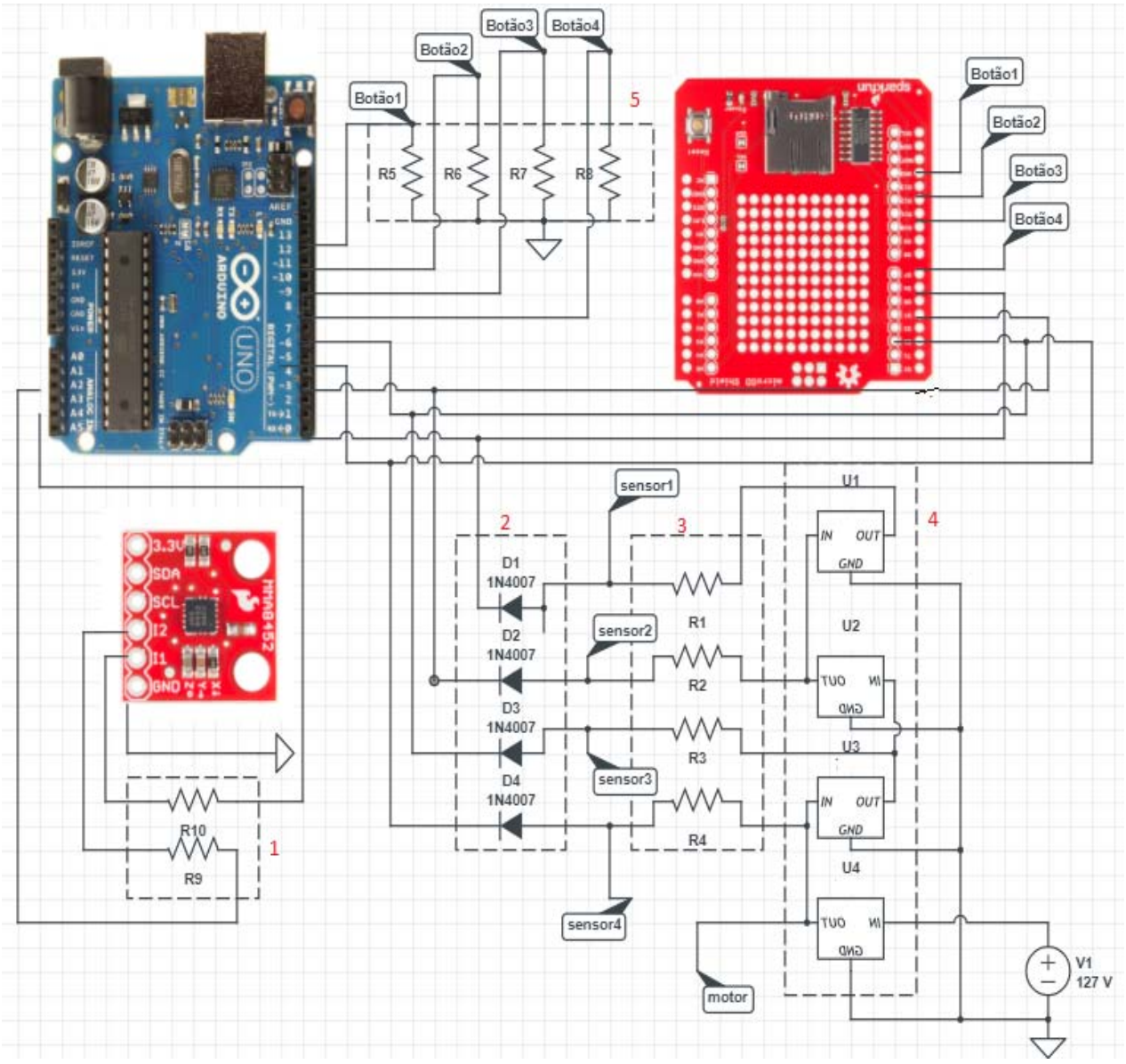

Figura 15 - Circuito Elevador Local

O desenho do circuito para a possibilidade de controle virtual e por conseguinte acesso remoto foi desenvolvido com uma outra placa Arduíno, alguns resistores para pull down e um chip integrado com 4 portas OU.

A integração entre os dois sistemas se dá pelos botões de chamada do elevador, que ao invés de irem para a placa Arduino do controle local, vão direto para a placa Arduino do controle virtual.

Esses condutores fornecem a informação se algum botão físico foi apertado sendo conectados nas portas digitais da placa, quatro condutores dando input digital, indicador físico. Os outputs dados pela placa, botões virtuais, saem por meio das saídas digitais e são controlados no painel frontal.

As portas OU servem para transmitir o sinal do botão físico ou do botão virtual. Os sinais de saída são conduzidos para onde anteriormente ficava conectado o condutor do botão físico.

Os condutores dos botões virtuais devem sofrer pull down, assim como os botões físicos, para somente ter estado 0 ou 1.

10. Arquitetura de Rede – Acesso Remoto

A arquitetura de rede que utilizaremos será a mais básica. Um servidor localizado no laboratório é conectado aos equipamentos do laboratório, hospeda o instrumento virtual do elevador e é acessado pelo usuário. O repositório institucional, Maxwell, é usado como intermediário entre o usuário e o servidor do laboratório.

A topologia em estrela, figura 15, é a mais adequada e de fácil constatação devido ao fluxo de dados característicos a um laboratório remoto. Na rede em estrela, toda informação passa obrigatoriamente por um nó (servidor), que distribui dados devidamente entre os outros nós. A confiabilidade da rede depende do nó central, o número de estações é limitado pelo nó central.

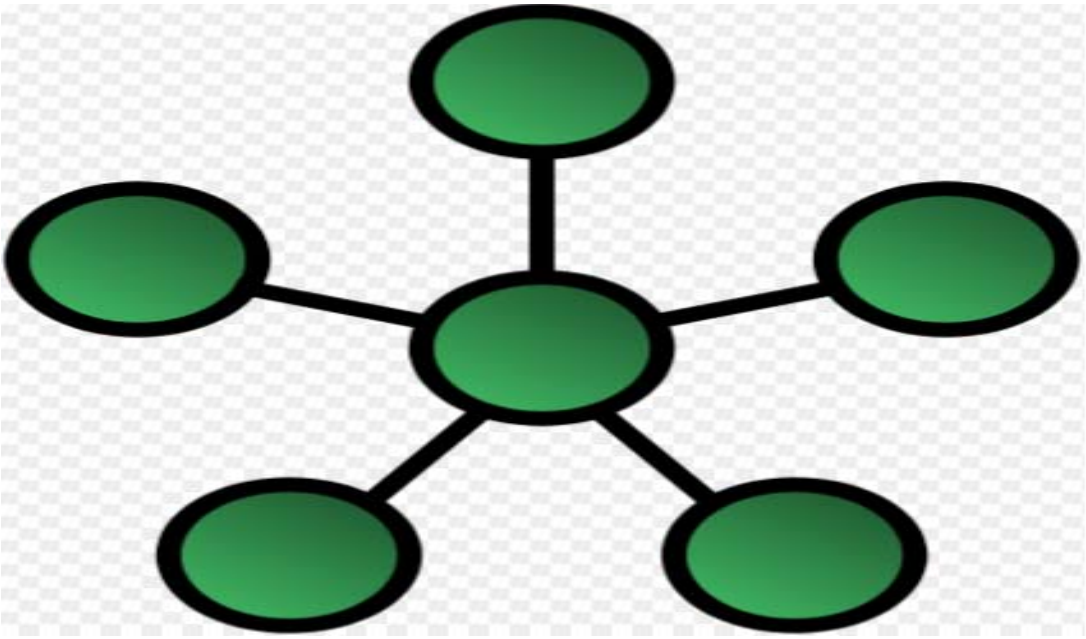

Figura 16 - Topologia de Rede Estrela

O Maxwell desempenha o papel de controlar o acesso e agendamento dos clientes que desejam usar o laboratório. O usuário utiliza a aplicação por intermedio do repositório, em nenhum momento se comunica diretamente com o servidor.

A comunicação da placa de controle com o servidor é feita por meio da entrada USB.

11. Web Server– Acesso Remoto

O LabVIEW nos dá várias opções de compartilhamento do controle de um painel frontal.

Porém, independente do meio que faremos essa comunicação, temos que configurar o servidor que vai hospedar a aplicação.

Três áreas devem ser configuradas: diretórios de arquivos e opções de rede, clientes permitidos e Vis visíveis.

Para o funcionamento do webserver do labview as configurações foram feitas segundo as figuras 17, 18 e 19.

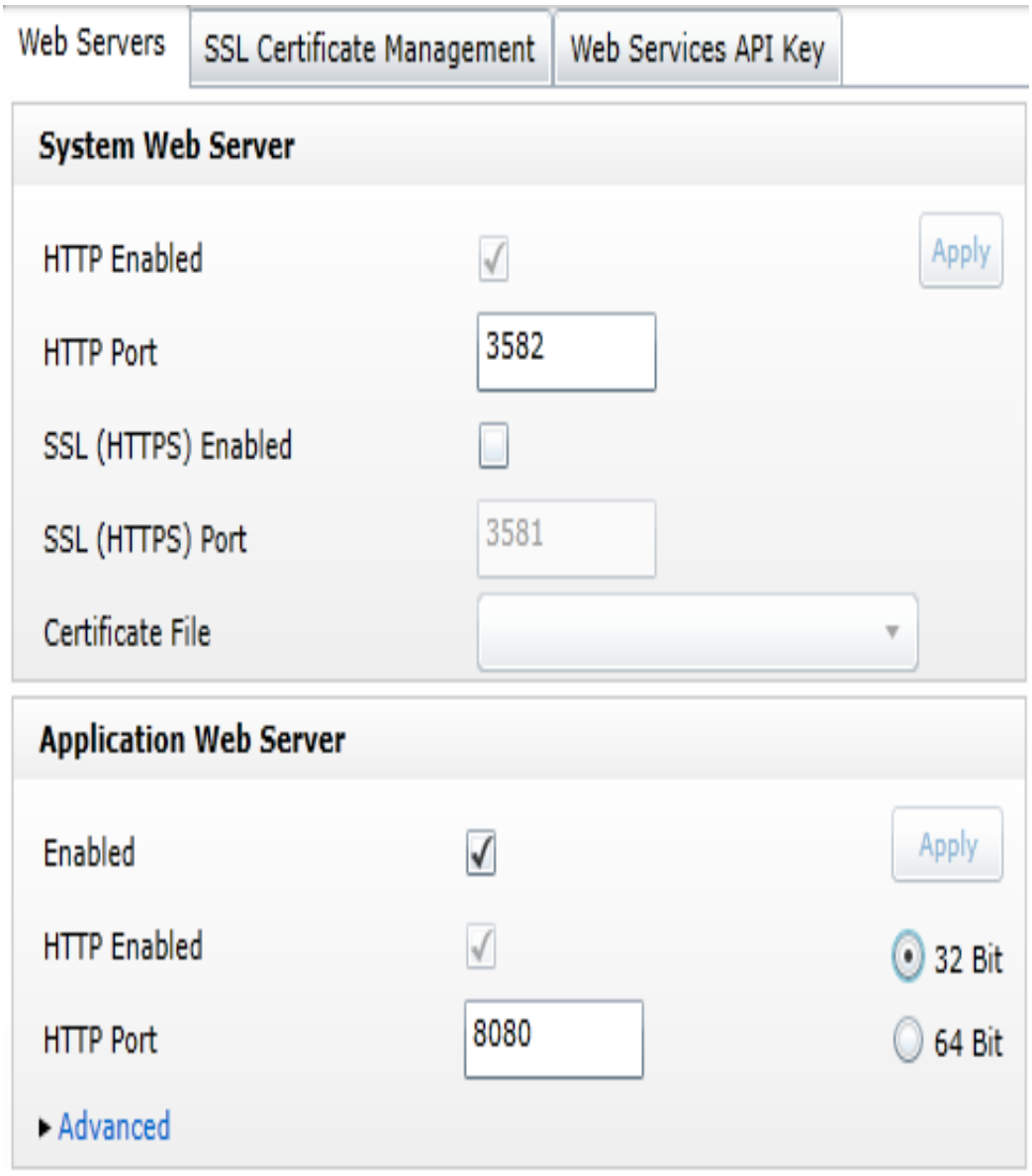

Figura 17 - Habilitando servidor

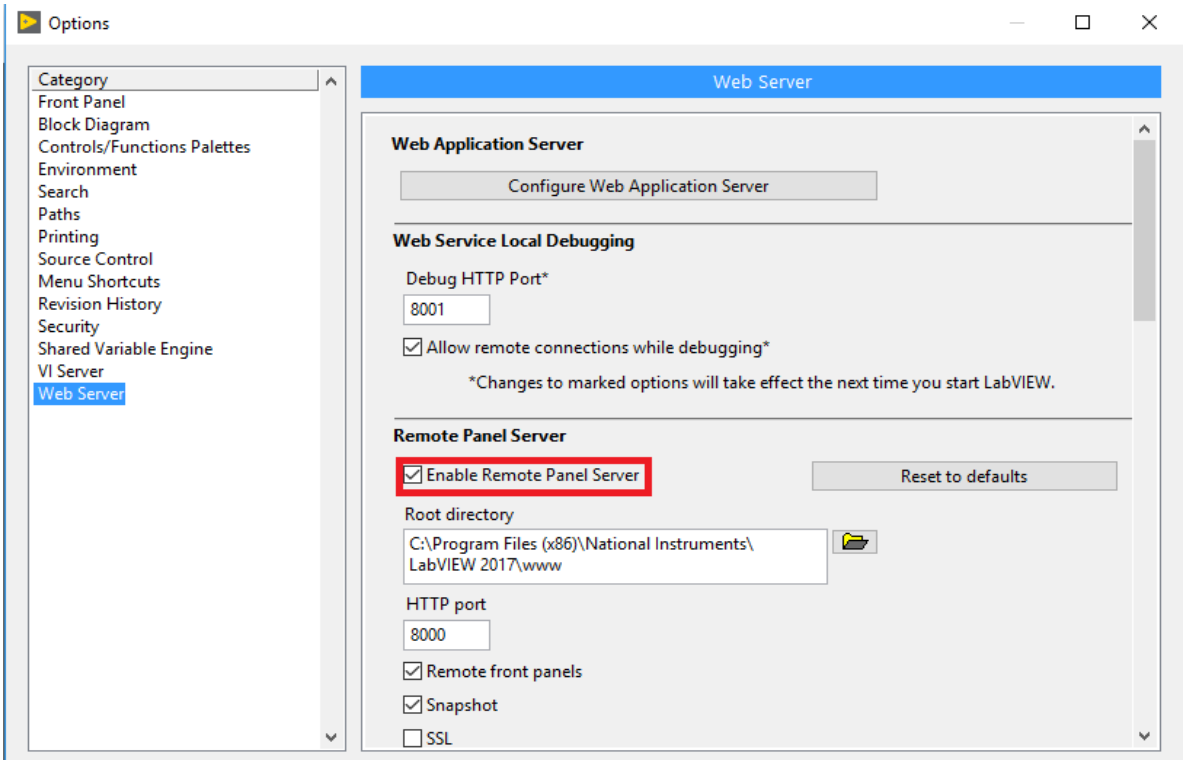

Figura 18 - Habilitando o servidor do painel remoto

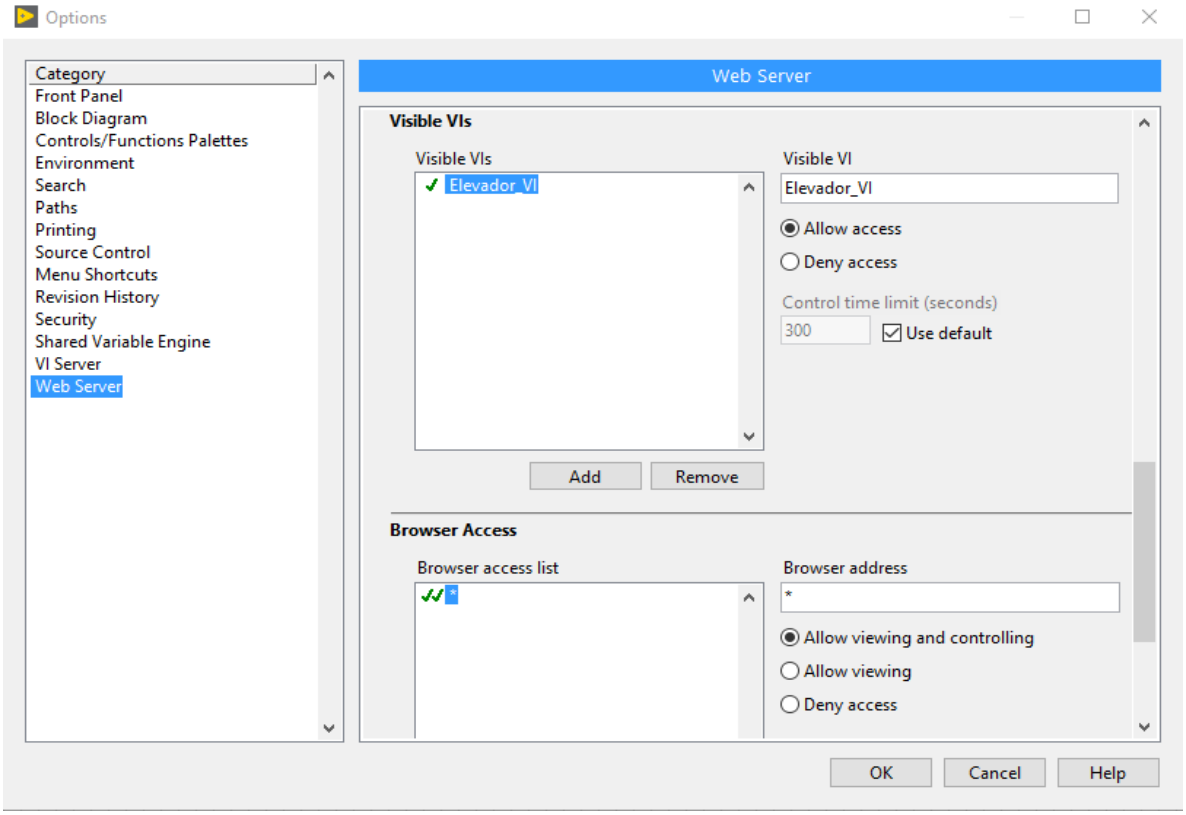

Figura 19 - VIs acessíveis e lista de clientes permitidos.

Para prover acesso pelo web browser é preciso mais um passo na configuração de uma ferramenta, Web Publishing Tool, que cria um documento HTML.

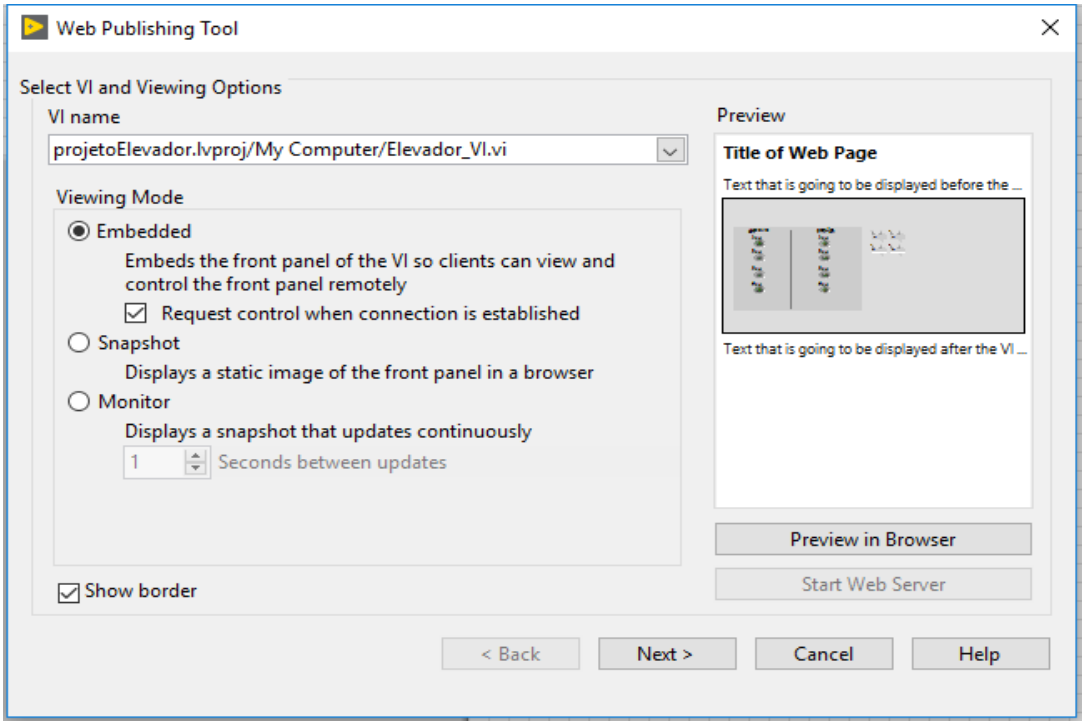

Figura 20 - Web Publishing Tool

Terminada a configuração do servidor web e do acesso pelo browser, a máquina do cliente web também deve ser configurada corretamente.

Caso o cliente tenha o LabVIEW instalado em seu computador basta abrir a janela para operar conexões remotas, figura 21. O painel frontal abre na máquina que fez requisição da mesma forma que se o VI estivesse instalado.

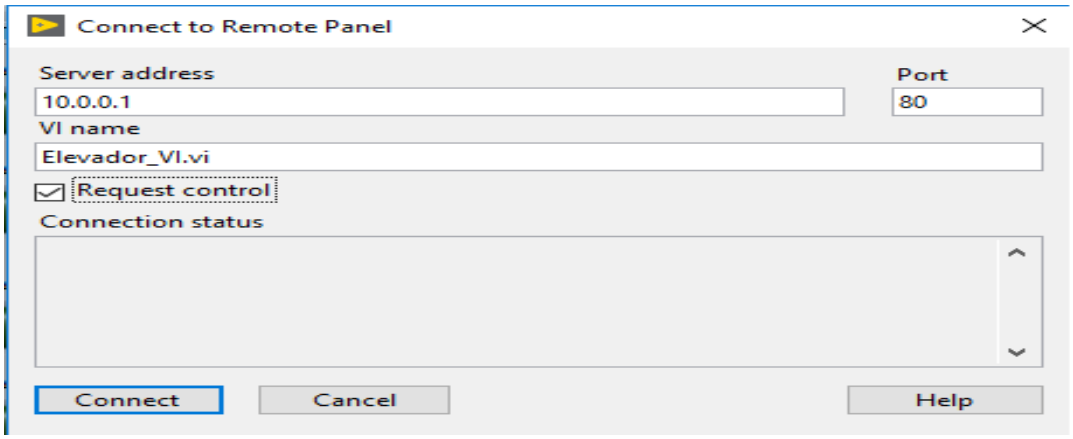

Figura 21 - Operar painel remoto

Para conexão através do web browser, o cliente precisa ter instalado o LabVIEW run-time engine, pacote que permite a execução de aplicativos, e plugins instalados no browser. O pacote run-time precisa ser da mesma versão que o pacote do web server.

O endereço para realizar a conexão é composto pelo IP do servidor, nome da rede e nome do arquivo HTML que contém o VI que queremos controlar.

As últimas versões do LabVIEW não tem suporte para browsers mais avançados, alguns itens de plugins recentes travam o processo. Somente os browser Internet Explores 11 e Safari conseguem abrir painéis frontais remotos.

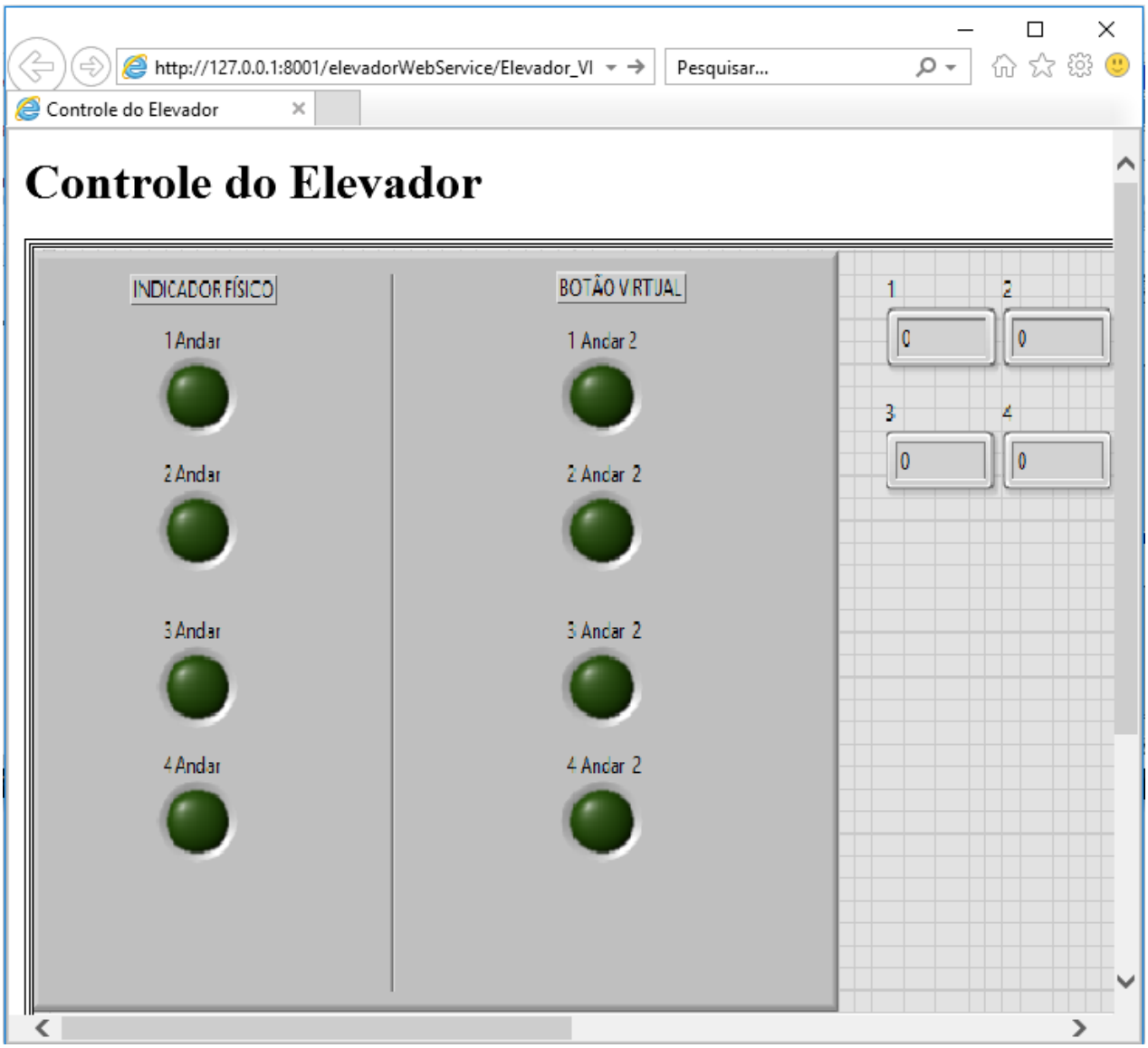

Figura 22 - Acesso do painel remoto via browser

12. Conclusão

Na concepção do projeto e em seu desenvolvimento, várias teorias e dúvidas não pensadas no início surgiram.

A solução encontrada foi considerada pensando principalmente na adaptação do que já existia, dos custos para essa adaptação e do tempo disponível para conclusão.

O projeto de laboratório remoto de elevador de carga engloba matérias dadas nos cursos de Engenharia Elétrica, Engenharia de Controle e Automação e Engenharia de Computação.

Tanto o desenvolvimento do projeto como a sua leitura e entendimento levam ao conhecimento de importantes teorias da tecnologia atual.

A realização desse projeto obteve resultados satisfatórios no controle virtual do elevador, vide controlevirtualElevador.mp4, e na comunicação entre o servidor e o cliente.

Outras possibilidades de melhoria no sistema podem vir no futuro. A instalação de uma câmera filmando o movimento do elevador de acordo com a manipulação do painel frontal remoto, a integração de todo o sistema numa única placa controladora e o acoplamento de um shield de conexão Wifi para substituição de um pc fazendo o papel de servidor web são algumas opções para um próximo melhoramento do sistema.

Porém, o maior ganho que o projeto pode proporcionar é motivar os alunos e fazer com que aprendam teorias das engenharias mais facilmente.

A multidisciplinaridade ressaltada no terceiro paragráfo dessa seção também é ponto alto do projeto.

13. Referências Bibliográficas

[1]http://www.desafiosdaeducacao.com.br/entenda-por-crescimento-dos-moocs-desperta-interesse-pela-criacao-de-laboratorios-remotos/

[2]http://www.oxbridgeessays.com/blog/the-pros-and-cons-of-e-learning-and-education-technology-717/

[3]http://www.maxwell.vrac.puc-rio.br/23180/23180.PDF

[4]http://www.ni.com/white-paper/7108/pt/

[5]ftp.ni.com/pub/branches/latam/brazil/academic_day/aquisicao_de_dados_e_instrumentacao_virtual_com_mydaq.pdf

[6]http://pt.wikipedia.org/wiki/Sistemas_de_Supervis%C3%A3o_e_Aquisi%C3%A7%C3%A3o_de_Dados

[7]http://www.ebah.com.br/content/ABAAAA1uMAK/sistema-scada#

[8]http://s3.amazonaws.com/academia.edu.documents/30915693/Advances_on_remote_labs.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1435454420&Signature=C3GW9kAd2vMfiNeidSFsYzpWJOg%3D&response-content-disposition=inline

[9]http://www.lila-project.org/resources/Documents/files/Grube_et_al_Metadata_EDUCON2011.pdf

[10] http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6970267

[11] http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5654493

[12] www.arduino.cc

[13] http://digital.ni.com/public.nsf/allkb/5874283E968ADB33862573210071D43A

[14] https://learn.sparkfun.com/tutorials/arduino-shields

[15] http://www.wikiwand.com/pt/Aceler%C3%B4metro

[16] CEAM_ModeloElevador.pdf – Prof.Mauro Esperanza Neto

[17] https://pt.wikipedia.org/wiki/Arduino

[18] https://pt.wikipedia.org/wiki/Topologia_de_rede#/media/File:StarNetwork.svg

[19] http://arduinomais.blogspot.com.br/2014/05/resistor-pull-up-e-pull-down.html

[20] https://pt.wikipedia.org/wiki/Servidor_web

[21] http://www.ni.com/tutorial/4791/en/

[22] http://zone.ni.com/reference/en-XX/help/371361P-01/lvconcepts/viewing_fp_remote/

[23] http://zone.ni.com/reference/en-XX/help/371361P-01/lvhowto/build_web_service/

[24] https://www.youtube.com/watch?v=RIrBCeJmwWA

[25]https://forums.ni.com/t5/LabVIEW/Accessing-vi-front-panel-using-labview-webserver/td-p/3192117

[26]http://zone.ni.com/reference/en-XX/help/371361N-01/lvhowto/creating_html_documents_an/

[27] http://zone.ni.com/reference/en-XX/help/371361N-01/lvhowto/root_directory/

[28] https://knowledge.ni.com/KnowledgeArticleDetails?id=kA00Z000000P7YuSAK

[29] http://zone.ni.com/reference/en-XX/help/371361L-01/lvhowto/allowingdenyingwebbrowser/

[30]http://home.hit.no/~hansha/documents/labview/documents/web_services/LabVIEW%20Web%20Services.pdf

[31] http://www.ni.com/product-documentation/53635/en/?OpenDocument

[32] https://www.labviewmakerhub.com/doku.php?id=libraries:linx:start

[33] http://vi-lib.com/vision/CapturingVideofrmUSBCamer.html

[34] http://www.ni.com/white-paper/7747/en/#toc13