

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

MARLON SAVERI SILVA

**Instrumentação via autocolimador eletrônico de mecanismos movidos a motores de
passo em linhas de luz síncrotron.**

São Carlos

2013

MARLON SAVERI SILVA

Instrumentação via autocolimador eletrônico de mecanismos movidos a motores de passo em
linhas de luz síncrotron.

Trabalho de Conclusão de Curso apresentado à
Escola de Engenharia de São Carlos da
Universidade de São Paulo para obtenção do
título de Engenheiro Mecânico.

Orientador:
Prof. Dr. Daniel Varela Magalhães

São Carlos

2013

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

S586i Saveri Silva, Marlon
Instrumentação via autocolimador eletrônico de
mecanismos movidos a motores de passo em linhas de luz
síncrotron / Marlon Saveri Silva; orientador Daniel
Varela Magalhães. São Carlos, 2013.

Monografia (Graduação em Engenharia Mecânica) --
Escola de Engenharia de São Carlos da Universidade de
São Paulo, 2013.

1. LabVIEW. 2. autocolimador. 3. motor de passo. 4.
metrologia. 5. instrumentação. 6. síncrotron. I.
Título.

FOLHA DE AVALIAÇÃO

Candidato: MARLON SAVERI SILVA


Título: Instrumentação via autocolimador eletrônico de mecanismos movidos a motores de passo em linhas de luz síncrotron.

Trabalho de Conclusão de Curso apresentado à
Escola de Engenharia de São Carlos da
Universidade de São Paulo
Curso de Engenharia Mecânica

BANCA EXAMINADORA

Prof. Dr. José Guilherme Sabe


Nota atribuída: (10,0) Dez



(assinatura)

Prof. Dr. Gherhardt Ribatski

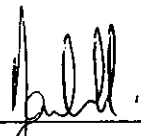
Nota atribuída: (10,0) Dez



(assinatura)

Prof. Dr. Jonas de Carvalho (Presidente)

Nota atribuída: (10,0) Dez.



(assinatura)

Média: (10,0) Dez.

Resultado: APROVADO!

Data: 22/Novembro/2013

Este trabalho tem condições de ser hospedado no Portal Digital da Biblioteca da
EESC:

SIM

NÃO,

Visto do orientador



Em memória de minha avó, Esaltina Lima da Silva,
Dona Fiúca,

AGRADECIMENTOS

A meus pais Marcos Tadeu da Silva e Silvane Aparecida Saveri Silva e a minha irmã Marília Saveri Silva.

A Adriano Trevelin Franoso, Alex Roger Almeida Colmanetti, Denys Eduardo Teixeira Marques, Douglas Martins Rocha, Lucas Silva Yoshida e Maurício Cardoso de Souza (Zhaogruppo).

Ao Centro Nacional de Pesquisa em Energia e Materiais (CNPEM) pela oportunidade em realizar o presente trabalho e tantos outros no objetivo de desenvolver a ciência brasileira. Em especial, ao apoio dos colaboradores Guilherme Lopes da Silva, coautor da primeira versão do aplicativo e parceiro nos testes dos mecanismos, Raul de Oliveira Freitas, Renan Ramalho Geraldes, Pedro Paulo de Souza Freitas, principal projetista dos mecanismos M1 a M5, Joacir Edmir dos Santos, Andre Luis Mesa e Elisa Silva Ferreira.

Aos membros das instituições que guiaram minha história acadêmica até este momento, Escola da Mônica II, Centro Educacional SESI 345, Colégio Anglo Ribeirão e Universidade de São Paulo.

“Respeito, disciplina e humildade”.

João Luiz Martins

RESUMO

SAVERI SILVA, M. **Instrumentação via autocolimador eletrônico de mecanismos movidos a motores de passo em linhas de luz síncrotron**. 68 f. Trabalho de Conclusão de Curso – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2013.

A construção de novas linhas de luz no Laboratório Nacional de Luz Síncrotron demanda a automatização de tarefas relacionadas à caracterização de equipamentos de elevada precisão projetados para trabalho em vácuo. Foi desenvolvido um aplicativo que usufrui de diversos recursos do LabVIEW para relacionar comandos enviados aos atuadores (motores de passo) às condições de *pitch* e *yaw* dos mecanismos, mensuradas através de um autocolimador. É apresentado um estudo de caso a fim de demonstrar que o programa auxiliou na detecção de erros, reduziu o tempo despendido em testes e disponibilizou dados para o controle dos mecanismos em campo.

Palavras-Chave: LabVIEW, autocolimador, motor de passo, metrologia, instrumentação, síncrotron

ABSTRACT

The construction of new beamlines at the Brazilian Synchrotron Light Laboratory demands the automatization of tasks related to the characterization of high precision equipment designed to work in vacuum. It has been developed an application that uses several LabVIEW resources to relate commands sent to the actuators (stepper motors) to pitch and yaw mechanisms behaviors, which was measured by an autocollimator. This report also presents a case study to demonstrate that the program helped on detecting errors, reducing the time spent on testing and provided data for the control of the mechanisms during their final uses.

Keywords: LabVIEW, Autocollimator, stepper motor, metrology, instrumentation, synchrotron

LISTA DE FIGURAS

Figura 1 - Princípio de acionamento de motores de passo de relutância variável.....	20
Figura 2 - Princípio de acionamento de motores de passo de ímã permanente.....	21
Figura 3 - Princípio de acionamento de motores de passo híbrido.....	21
Figura 4 - Modos de operação de motores de passo: única excitação e excitação dual	23
Figura 5 - Modo de operação meio passo.....	23
Figura 6 - Powerpack pp-electronics acoplado a hardware Phytron ZMX+	24
Figura 7 - Corte de uma chave de fim de curso comum.....	25
Figura 8 - Quando um par de engrenagens inverte seu sentido de rotação, a nova engrenagem motora percorre um trecho antes que a motriz passe a se deslocar. Isto se deve à folga existente entre seus dentes.....	26
Figura 9 - Painel frontal e controles básicos no LabVIEW	27
Figura 10 - Diagrama de Blocos com estruturas e blocos básicos	28
Figura 11 - Conector Pane (esquerda) e Ícone de um Instrumento Virtual.....	28
Figura 12 - Bloco VISA de configuração de porta serial	29
Figura 13 - Principais blocos VISA.....	29
Figura 14 - Figura representativa do groma	30
Figura 15 - Figura representativa da dioptra	31
Figura 16 - Caracterização de uma translação a partir de pontos lidos com Nível Óptico	31
Figura 17 - Princípio de medição do autocolimador eletrônico ELCOMAT 3000	32
Figura 18 – Princípio do uso de interferometria para medidas de deslocamento linear.....	33
Figura 19 - Esquema de um Síncrotron.....	34
Figura 20 - Esquema Óptico dos Espelhos da IR e projeto CAD da primeira câmara.....	35
Figura 21 - Movimentos permitidos para M1 a M5	36
Figura 22 - Elementos para transmissão de movimento Pitch do M2.....	36
Figura 23 – Elementos mais externos em detalhe	37
Figura 24 - Juntas flexíveis em M2	38
Figura 25 - Trio de prismas plano, cônico e V	38
Figura 26 - M1 em detalhe, molas para manutenção de contato.	39
Figura 27 - <i>Bellows</i> em M5. Peças (como o Flange) foram ocultadas na imagem ampliada...	39
Figura 28 - Montagem do sistema de medição: autocolimador (a) com unidade de controle (b), computador com aplicativo LabVIEW (c), <i>powerpack</i> (d) e mecanismo estudado (e)	41

Figura 29 – Adaptação para uso de espelho de testes na vertical.....	42
Figura 30 - Tela principal do aplicativo	43
Figura 31 - Sequência de episódios para a função "reiniciar"	44
Figura 32 - Trecho do diagrama de blocos no evento “Reiniciar MA”. No detalhe, ciclo <i>while</i> que aguarda o término do movimento à medida que informa as posições do atuador	45
Figura 33 - Sequência de episódios para a função "incrementar"	45
Figura 34 - Trecho do diagrama de blocos no evento “Incrementar MA”. Nota-se, no canto direito, o <i>loop</i> de verificação de final de movimento em função da estabilização da posição do motor.....	46
Figura 35 - Extrapolação do <i>looping while</i> de encerramento para movimentos simultâneos ..	46
Figura 36 - Sequência de episódios para a função "executar varredura"	47
Figura 37 - Trecho do diagrama de blocos do evento "executar varredura" e painel frontal do <i>SubVI</i> de salvamento de arquivo	47
Figura 38 - Trecho do comando "executar varredura". Em destaque, <i>SubVIs</i> relacionados à captura de pontos pelo autocolimador	48
Figura 39 - Diagrama de Blocos do Programa reduzido a uma única tela	50
Figura 40 - Desmembramento de um ícone representativo de <i>SubVI</i>	51
Figura 41 - Painel Frontal da nova versão. Observa-se que o motor é selecionado a partir de uma <i>combobox</i> (no “ <i>Config Cluster</i> ”) não mais por abas repetitivas.	52
Figura 42 - Evento no <i>loop</i> produtor associado à seleção de motor na <i>combobox</i> . No detalhe, <i>SubVI</i> para chamada do cluster referente ao motor selecionado na variável global, mostrada à esquerda.	52
Figura 43 – Máquina de Estados referente à função de varredura. A, configurações do arquivo e do <i>slider</i> ; B, armazenamento de dados e atualização do <i>Slider</i> ; C, movimentação do motor; D, término.....	54
Figura 44 - Trecho inicial de arquivo de texto produzido para uma varredura em M3.....	54
Figura 45 - Estabilidade do autocolimador em sala controlada (esquerda) e sala comum (direita)	56
Figura 46 - Análise de estabilidade do autocolimador em sistema isento de tripés	57
Figura 47 - Resultado da varredura em M1 para atuação do motor <i>pitch</i> com comportamento linear (azul) e erro <i>yaw</i> senoidal decrescente (vermelho) ~30x menor	57
Figura 48 - Resultado da varredura em M1 para atuação do motor <i>roll</i> com comportamento linear (verde) e erro <i>yaw</i> senoidal decrescente (vermelho) ~20x menor.....	58

Figura 49 - Resultado da varredura em M1 para atuação do motor <i>yaw</i> com comportamento linear (vermelho) e erro <i>pitch</i> também linear (azul) ~30x menor	58
Figura 50 - Resultado da varredura em M2 para atuação do motor <i>pitch</i> com comportamento linear (azul) e erro <i>yaw</i> senoidal (vermelho) ~40x menor.....	58
Figura 51 - Resultado da varredura em M2 para atuação do motor <i>yaw</i> com comportamento linear (vermelho) e erro <i>pitch</i> também linear (azul) ~25x menor	59
Figura 52 - Resultado da varredura em M3 para atuação do motor <i>pitch</i> com comportamento linear (azul) e erro <i>yaw</i> também linear (vermelho) ~40x menor	59
Figura 53 - Resultado da varredura em M3 para atuação do motor <i>yaw</i> com comportamento linear (vermelho) e erro <i>pitch</i> parabólico (azul) ~360x menor	59
Figura 54 - Resultado da varredura em M4 para atuação do motor <i>pitch</i> com comportamento linear (azul) e erro <i>yaw</i> senoidal crescente (vermelho) ~40x menor.....	60
Figura 55 Resultado da varredura em M4 para atuação do motor do movimento que combina <i>yaw</i> e <i>roll</i> com comportamento linear (alaranjado) e erro <i>pitch</i> parabólico (azul) ~40x menor	60
Figura 56 - Resultado da varredura em M5 para atuação do motor <i>pitch</i> com comportamento linear (azul) e erro <i>yaw</i> senoidal crescente (vermelho) ~15x menor.....	60
Figura 57 - Resultado da varredura em M5 para atuação do motor do movimento que combina <i>yaw</i> e <i>roll</i> com comportamento linear (alaranjado) e erro <i>pitch</i> parabólico (azul) ~40x menor	61
Figura 58 - Leituras "v" durante movimentação do motor <i>pitch</i> em 40 diferentes posições do motor <i>yaw</i>	62
Figura 59 Mecanismos 1 a 3 instantes antes da instalação no anel	63
Figura 60 - Varredura para verificação de <i>Pitch</i> e <i>Yaw</i> em movimento horizontal (X) de bancada para movimentação de amostras.....	63
Figura 61 - <i>Pitch</i> e <i>Yaw</i> para movimento horizontal (X)	64

LISTA DE SIGLAS

CAD	<i>Computer Aided Design</i>
CCD	<i>Charge Couple Device</i>
CNPEM	Centro Nacional de Pesquisa em Energia e Materiais
CNC	Comando Numérico Computadorizado
IR	<i>Infrared Beamline</i>
LNLS	Laboratório Nacional de Luz Síncrotron
NI	<i>National Instruments</i>
SNOM	<i>Scanning Near Field Optical Microscopy</i>
USB	<i>Universal Serial Bus</i>
USP	Universidade de São Paulo
VISA	<i>Virtual Instrument Software Architecture</i>

SUMÁRIO

1. INTRODUÇÃO	16
1.1 Apresentação	16
1.2 Objetivos	17
1.2.1 Objetivo Geral	17
1.2.2 Objetivos Específicos	17
1.3 Justificativa	17
1.4 Estrutura do trabalho	18
2. CONCEITUAÇÃO TEÓRICA	20
2.1 Motores de Passo	20
2.1.1 Descrição	20
2.1.2 Controle	24
2.1.3 Chave de Fim de Curso	24
2.1.4 Histerese (<i>backlash</i>) e outros conceitos envolvidos	25
2.2 NI LabVIEW	26
2.3 Autocolimador e óptica em metrologia	30
2.4 Contexto em que o trabalho foi aplicado	34
2.4.1 Fonte de Luz Síncrotron	34
2.4.2 Linha de Infravermelho	35
2.4.3 Mecanismos dos espelhos	35
3. RESULTADOS	41
3.1 Montagem do sistema de medição	41
3.2 Primeira versão do programa	42
3.3 Breve explicação dos eventos	44
3.4 Melhorias	49
3.4.1 Filas e Loops Paralelos	49
3.4.2 A Importância das <i>SubVIs</i>	50
3.4.3 Vantagens na utilização de <i>clusters</i> e variáveis globais	51
3.4.4 Expansão para oito motores	53
3.4.5 Outras alterações	53
3.5 Limitações	55

4. ESTUDO DE CASOS	56
4.1 Varreduras nos mecanismos da linha de infravermelho.....	56
4.2 Exemplo de utilização em outra aplicação	63
5. CONSIDERAÇÕES FINAIS	65

1. INTRODUÇÃO

1.1 Apresentação

O presente trabalho de conclusão de curso em Engenharia Mecânica tematiza a automatização de ensaios de metrologia em mecanismos de linhas de luz, para a qual emprega a plataforma NI LabVIEW. Classifica-se como um projeto de intervenção que inclui um estudo de caso e refere-se a uma das atividades exercidas pelo autor durante seu período de estágio no Centro Nacional de Pesquisa em Energia e Materiais (CNPEM).

O CNPEM sedia o Laboratório Nacional de Luz Síncrotron (LNLS), o qual dispõe da primeira fonte de luz síncrotron da América Latina e cujas contribuições abrangem diversas áreas da ciência e possibilitam avanços acadêmicos e industriais.

Elétrons em trânsito por uma câmara de vácuo têm suas trajetórias desviadas para emitirem a radiação síncrotron. Denominam-se Linhas de Luz sistemas que filtram faixas específicas do espectro eletromagnético dessa radiação para que sejam aplicadas em experimentos sobre diversos materiais.

Em alguns desses sistemas, são utilizados motores de passo para ajustes finos no posicionamento de espelhos. Por conseguinte, a construção de Linhas de Luz exige instrumentos de metrologia de alta precisão. Entre esses equipamentos, os autocolimadores eletrônicos efetuam medições angulares sem contato e apresentam interface de programação de aplicativos.

É possível e conveniente a construção de um programa que efetue a interação entre os motores de passo e o autocolimador eletrônico em uma linguagem de programação apropriada. Origina-se, portanto, o seguinte problema de pesquisa: “Como relacionar, através do programa computacional LabVIEW, o controle de motores de passo às leituras de instrumentos metrológicos na caracterização de mecanismos utilizados em linhas de luz síncrotron?”.

1.2 Objetivos

1.2.1 Objetivo Geral

Elaborar um programa computacional na plataforma NI LabVIEW para automatizar análises de metrologia por autocolimador em mecanismos movimentados por motores de passo.

1.2.2 Objetivos Específicos

- Compreender o funcionamento de linhas de luz síncrotron.
- Familiarização com a plataforma de programação NI LabVIEW.
- Desenvolver um aplicativo que correlacione posições de motores de passo a leituras angulares fornecidas por um autocolimador durante movimentação de mecanismos.
- Através do aplicativo desenvolvido, realizar ensaios em mecanismos utilizados em linhas de luz síncrotron e avaliar resultados.
- Discutir elementos mecânicos de tais mecanismos e associá-los aos resultados das varreduras.
- Efetuar melhorias no aplicativo para facilitar a automatização de tarefas e a análise dos dados obtidos.

1.3 Justificativa

A necessidade de aperfeiçoamento na mensuração de ângulos em mecanismos utilizados em linhas de luz do acelerador de elétrons do LCLS, tendo em vista a alta precisão exigida pelo anel em construção (Projeto Sirius), motivou os cientistas à aquisição de um autocolimador eletrônico em substituição às técnicas até então aplicadas, limitadas a teodolitos menos precisos e privados de comunicação computadorizada com os motores que originam os movimentos nos mecanismos.

A automatização de tarefas proporcionada pelos aplicativos desenvolvidos em NI LabVIEW permite simultaneamente facilitar a obtenção de dados, eliminar possíveis

falhas originadas em atividades manuais, reduzir consideravelmente o tempo de trabalho, aprimorar testes de repetibilidade e aumentar as possibilidades de comandos sobre as variáveis envolvidas.

Por conseguinte, uma vez estabelecido o programa computacional, ele pode ser reutilizado em aplicações futuras que envolvam a comunicação entre motores de passo e autocolimador, sem se limitar a mecanismos de linhas de luz síncrotron. Além disso, o presente trabalho serve de modelo para o desenvolvimento de aplicativos de comunicação entre motores e outros instrumentos de metrologia.

1.4 Estrutura do trabalho

A primeira etapa do presente documento possui caráter teórico, ela aponta conceitos relacionados a ferramentas utilizadas e ao contexto em que o trabalho foi realizado por intermédio de pesquisas bibliográficas.

Pesquisas de finalidades aplicadas objetivam o emprego de conhecimentos já disponíveis na resolução de problemas. Este relato possui, portanto, caráter descritivo e utiliza as mais variadas fontes na busca de seu objetivo.

Nesse sentido, o capítulo 2 destina-se à apresentação de conhecimentos relacionados a motores de passo, suas classificações e seus modos de operação no intuito de se compreender seus princípios de funcionamento e, posteriormente, seu controle. O capítulo abrange também os elementos associados denominados chaves de fim de curso.

O capítulo 3 introduz a plataforma de programação LabVIEW com destaque ao padrão VISA de configuração e programação de instrumentos em diversas interfaces. No entanto, a compreensão dos pormenores do aplicativo desenvolvido exige que o leitor possua conhecimento básico da linguagem.

O capítulo seguinte analisa brevemente a evolução de instrumentos ópticos em aplicações metrológicas, enfoca os autocolimadores eletrônicos e sua conexão a computadores.

Segue-se uma explanação a respeito dos equipamentos que primeiro motivaram a implementação do programa.

A segunda etapa deste relatório expõe os resultados obtidos e refere-se ao programa desenvolvido em LabVIEW capaz de associar dados concernentes aos

movimentos de motores de passo com leituras angulares importadas do autocolimador eletrônico.

Os resultados são apresentados em duas etapas. A primeira exhibe uma versão inicial do programa, a qual foi mantida neste texto para facilitar a compreensão do caminho traçado até a construção da versão final (segunda etapa), de maior complexidade.

O penúltimo capítulo é um estudo de caso que explora as informações colhidas pelo programa computacional produzido na análise de mecanismos de uma linha de radiação infravermelha de uma fonte de luz síncrotron.

As últimas páginas são dedicadas a uma discussão sobre o que foi realizado.

2. CONCEITUAÇÃO TEÓRICA

2.1 Motores de Passo

2.1.1 Descrição

Motores de passo são motores elétricos de corrente contínua cujo rotor gira em frações discretas de volta – passos – o que implica em alta precisão, porquanto são utilizados para controle de posicionamento.

Brites e Santos (2008, p. 3) assim os apresentam:

[...] dispositivos eletromecânicos que convertem pulsos elétricos em movimentos mecânicos que geram variações angulares discretas. O rotor ou eixo de um motor de passo é rotacionado em pequenos incrementos angulares, denominados “passos”, quando pulsos elétricos são aplicados em uma determinada sequência nos terminais deste. A rotação de tais motores é diretamente relacionada aos impulsos elétricos que são recebidos, bem como a sequência a qual tais pulsos são aplicados reflete diretamente na direção a qual o motor gira. A velocidade que o rotor gira é dada pela frequência de pulsos recebidos e o tamanho do ângulo rotacionado é diretamente relacionado com o número de pulsos aplicados.

Quanto à estrutura, podem ser divididos em três tipos. Os de relutância variável possuem rotor dentado e estator com enrolamentos que, quando energizados, atraem os dentes do rotor de modo que a relutância do circuito seja mínima – Figura 1. Sua resolução é aumentada com o acréscimo de dentes devido à diminuição do ângulo de passo.

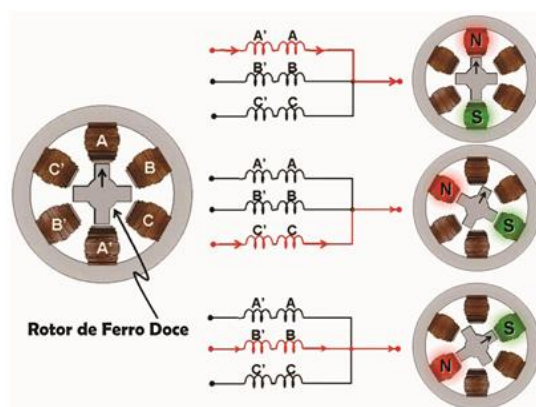


Figura 1 - Princípio de acionamento de motores de passo de relutância variável
Fonte: adaptado de MCCOMB (2007, p. 19-23)

Rotores de motores de passo de imã permanente – Figura 2 – não possuem dentes e são constituídos de ímãs permanentes. Como resultado, mantêm suas posições quando não energizados. O aumento de polos leva a um aumento de resolução.

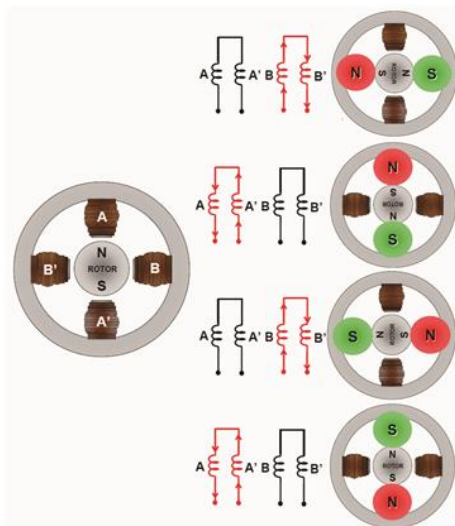


Figura 2 - Princípio de acionamento de motores de passo de ímã permanente
Fonte: adaptado de MCCOMB (2007, p. 12-18)

Motores híbridos reúnem qualidades dos anteriores, pois o rotor é multidentado e com ímãs permanentes – Figura 3.

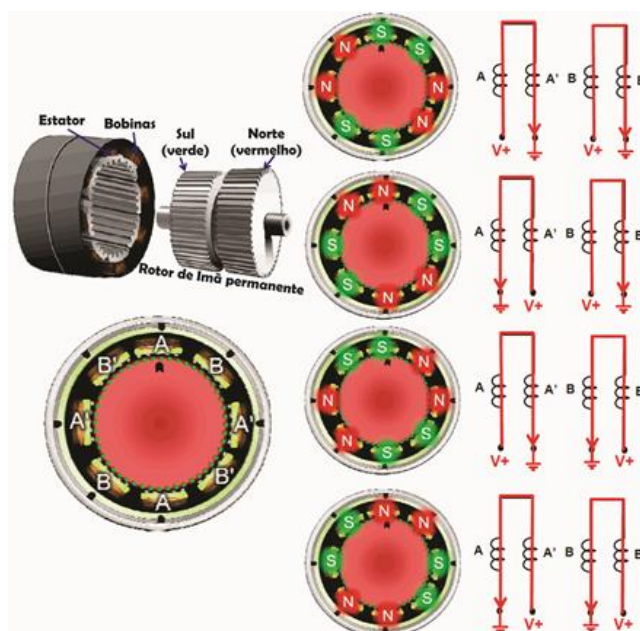


Figura 3 - Princípio de acionamento de motores de passo híbrido
Fonte: adaptado de MCCOMB (2007, p. 24-30)

Athani oferece uma análise histórica de motores de passo. Patentes de seus precursores datam de 1919 e 1920 por, respectivamente, Walker na Inglaterra e Chicken

e Thain nos Estados Unidos, aplicados em navios britânicos e estadunidenses. Motores de passo modernos híbridos de ímã permanente foram inventados por Feirtag e Donahoo da *General Electric* em 1952. Já os de relutância variável foram relatados somente em 1957 por Thomas e Fleischauer. No Japão, motores de passo foram utilizados pela Fanuc no início da década de 1970 em sistemas CNC (ATHANI, 2005, p. 1-3).

Visto que sua movimentação deve-se a uma sequência de pulsos elétricos, eles podem ser programados por equipamentos digitais e, portanto, definidos como atuadores que convertem pulsos digitais em movimentos angulares de modo diretamente proporcional.

Inicialmente acionados a partir de contatos mecânicos, acompanharam a evolução de dispositivos a estado sólido, como tiristores e transistores até que, na década de 1970, motores de passo passaram a ser utilizados em impressoras e outros periféricos de computadores (RICCI, 2005).

Todavia, são, na maioria das vezes, aplicados em malha aberta e, se por problemas mecânicos (como sobrecarga), não corresponderem exatamente às instruções digitais, ocorrerá a chamada “perda de passo”, que culmina em falha de posicionamento. Naturalmente, suas aplicações carecem de dimensionamentos com margem de segurança.

Não obstante, aplicações que não exijam velocidade e potência elevadas não justificam a opção por servomotores com *encoders* devido ao maior custo em comparação aos motores de passo, que, ademais, sofrem menor desgaste mecânico apresentam “torque elevado em baixas velocidades” (KORDIK, 2003, p. 26).

Há quatro modos de acionamento para motores de passo: completo de única excitação, completo com excitação dual, meio passo e micropasso. No primeiro caso, uma fase é energizada de cada vez, conforme ilustrado na Figura 4. Como poucas bobinas são acionadas concomitantemente, este modo é mais econômico e resulta em menor torque. É importante ressaltar que uma fase não coincide necessariamente com um enrolamento, uma vez que, na presença de várias bobinas, as tensões de alimentação se repetem de modo sequencial – A+ (enrolamento 1), B+ (enrolamento 2), A- (enrolamento 3), B- (enrolamento 4), A+ (enrolamento 5),... – e, neste caso, os enrolamentos de mesma fase são ligados de modo a receberem a mesma tensão (SOUZA, 2007, p. 13).

No modo de excitação dual, duas fases são ligadas ao mesmo tempo de forma que o motor preserve a quantidade de passos por volta do modo anterior, mas com torque e gasto de energia superiores.



Figura 4 - Modos de operação de motores de passo: única excitação e excitação dual
Fonte: Adaptado de MESSIAS (2006)

Uma combinação entre os modos anteriores possibilita o “meio passo”, Figura 5, com torque não uniforme. Consequentemente, a ressonância do motor é reduzida.

Caso as bobinas sejam energizadas com intensidades diferentes, o rotor tenderá para o polo mais energizado. Assim, é possível subdividir o movimento em “micropassos”. A quantidade de micropassos deve ser informada ao controlador.

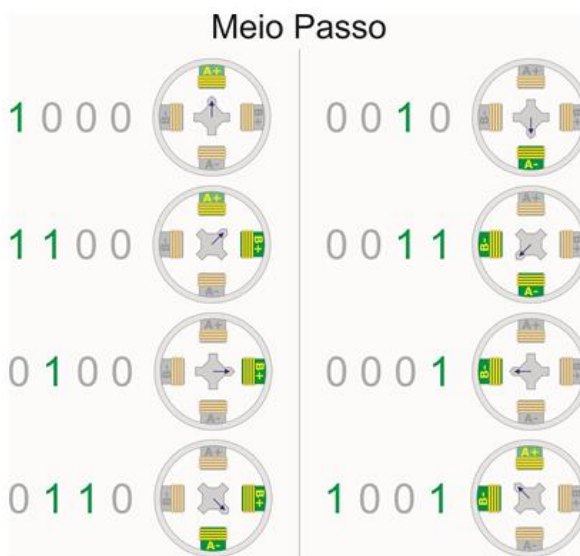


Figura 5 - Modo de operação meio passo
Fonte: Adaptado de MESSIAS (2006)

2.1.2 Controle

Controladores são dispositivos que recebem os sinais digitais (quantidade de passos e direção) originados, *e.g.*, por um computador, e gera a sequência de pulsos elétricos que acionam o motor por meio do *driver*. Portanto, o controlador define quais fases serão energizadas e o *driver* amplifica a corrente enviada ao motor.

A empresa alemã pp-electronic oferece uma solução chamada PowerPack para a movimentação de motores de passo. Trata-se de um conjunto de alimentação, Figura 6, que fornece energia a até oito motores e, acoplado a hardwares como DMC-41x3 ou ZMX+, respectivamente, das empresas Galil Motor Control e Phytron, permite o controle dos motores por meio de uma linguagem de programação específica.

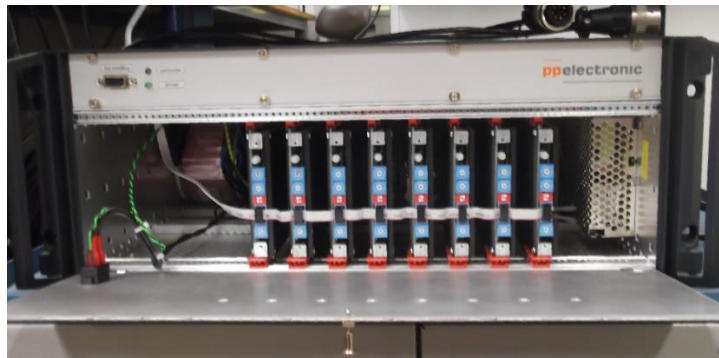


Figura 6 - Powerpack pp-electronics acoplado a hardware Phytron ZMX+
Fonte: proposta pelo autor

2.1.3 Chave de Fim de Curso

Motores de passo não apresentam um limite em quantidade de giros. Todavia, em muitas aplicações, eles são responsáveis por um movimento confinado, como em um carro que se desloca por uma guia de extensão limitada. Conseqüentemente, é necessário o emprego de sensores que detectem quando o curso alcançou as posições mínima e máxima convencionadas e que ordenem a interrupção do movimento.

Esta questão é resolvida com o uso de “chaves de fim de curso”, interruptores de ação rápida de baixo custo e alta durabilidade – Figura 7 – nos quais o toque em um botão modifica contatos elétricos no interior do dispositivo de modo a inverter o sinal enviado ao controlador, que então cessará o movimento.



Figura 7 - Corte de uma chave de fim de curso comum
Fonte: ESHAM (2007)

Chaves de fim de curso, em geral, disponibilizam dois modos de conexão. Para a chave ilustrada na Figura 7, o modo “Normalmente aberto” é obtido ao conectar o controlador por fios aos terminais da esquerda (comum) e central (positivo ou negativo). Nesta opção, nenhuma corrente passa pelo dispositivo até que o botão seja acionado e produza um curto entre os terminais.

Caso sejam utilizados os terminais das extremidades, a chave estará naturalmente em curto, que poderá ser interrompido pelo acionamento do botão – modo “normalmente fechado”.

Destas configurações, conclui-se que quando mais de um sensor for utilizado em ligações em paralelo, deve-se optar pelo modo “normalmente aberto”; em série, pelo modo “normalmente fechado” (MARTINS, 2008). Depreende-se também que este segundo modo é mais interessante caso se deseje uma interrupção do movimento na ocorrência de rompimento acidental de fios.

2.1.4 Histerese (*backlash*) e outros conceitos envolvidos

Quando se trabalha com sistemas mecânicos precisos, torna-se importante a compreensão de histerese (*backlash*). Este fenômeno ocorre quando folgas existentes entre peças acopladas provocam alterações nas medidas de posição de um mecanismo conforme se altera o sentido para que tal posição seja atingida. Exemplo de fácil compreensão é o de duas engrenagens que invertem seus sentidos de rotação – Figura 8.

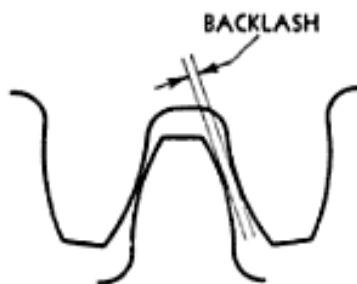


Figura 8 - Quando um par de engrenagens inverte seu sentido de rotação, a nova engrenagem motora percorre um trecho antes que a motriz passe a se deslocar. Isto se deve à folga existente entre seus dentes
 Fonte: JONES (1961, p. 20).

Além das folgas, deformações associadas ao atrito também podem originar histerese (ESTATCAMP, 2001).

Para um mecanismo em que um movimento é transmitido por uma série de diferentes peças, espera-se um erro por histerese superior aos provocados por cada par individualmente.

Porquanto, é necessário que leituras de posições de um mecanismo sejam consideradas apenas quando elas são atingidas via um único sentido de movimento. Conseqüentemente, os softwares que comandam os motores de passo devem ser programados com a chamada “correção de *backlash*”. Conhecido o grau de histerese e definido um sentido de rotação; o giro no sentido oposto deverá avançar à posição desejada (caso haja curso suficiente) na quantidade de passos suficiente para eliminar as folgas, retornar ao “sentido de medição” e enfim alcançar a posição pretendida.

Outros conceitos relacionados à caracterização de um mecanismo e importantes de serem lembrados são a repetibilidade (os resultados de sucessivas medições, realizadas nas mesmas condições, estão concordantes? O mecanismo sempre retorna à posição desejada?), a precisão (qual o grau de concordância entre os resultados? Com que incerteza atinge-se uma determinada posição?) e a reprodutibilidade (ao alterarmos as condições da medição, com que grau os valores concordam entre si?).

2.2 NI LabVIEW

O *Laboratory Virtual Instrument Engineering Workbench* (LabVIEW) é um ambiente de desenvolvimento de programas computacionais fornecido pela empresa estadunidense National Instruments (NI) desde 1986.

Destaca-se das demais linguagens de programação principalmente por sua capacidade de integração com diversos hardwares e de sua interface gráfica que facilita a compreensão dos códigos nele programados. Logo, é adequado para sistemas de medição e controle de diversos graus de complexidade.

Um programa em LabVIEW – Instrumento Virtual (VI) – é dividido em quatro partes principais: painel frontal, diagrama de blocos, *Conector* Pane e Ícone. O painel frontal exemplificado na Figura 9 é a interface do usuário e exibe variáveis de entrada e saída representadas, respectivamente, por janelas de controladores e indicadores.

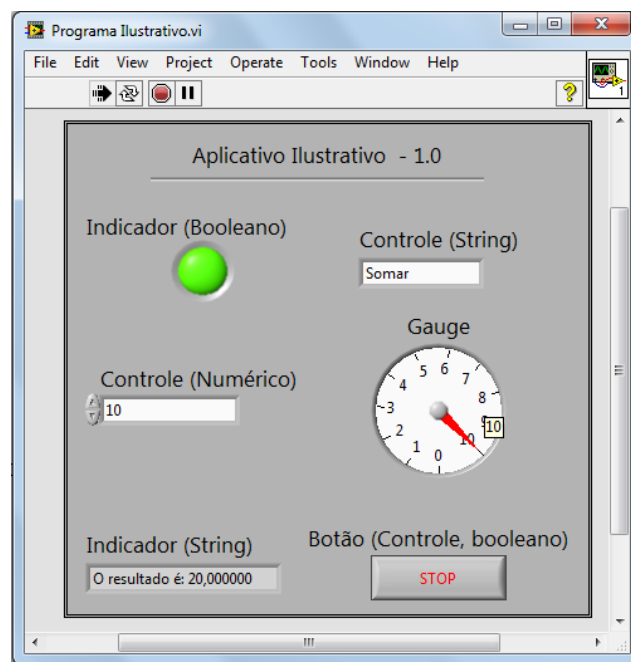


Figura 9 - Painel frontal e controles básicos no LabVIEW
Fonte: proposta pelo autor

O Diagrama de Blocos, Figura 10, corresponde ao código fonte do aplicativo, no qual ocorre a programação do fluxo de dados. Possui aparência didática, resumido a blocos representativos de variáveis (presentes ou ocultas no painel frontal) unidos por linhas e inseridos em estruturas, tais como controle de repetições e da sequência de acontecimentos. Uma estrutura em particular merece destaque: evento. Com uma linguagem orientada a eventos, o fluxo de dados do VI não é contínuo, porém dependente das interações do usuário com o painel frontal, *e.g.*, cliques com um mouse, ocorrência de erros ou sinais de um sensor de aquisição de dados.

No exemplo meramente ilustrativo das figuras 9 e 10, o VI verifica se o texto digitado na caixa de controle do tipo *string* é “somar”. Em caso positivo, os números informados pelos controles numéricos (digitados ou selecionados) são somados. O

resultado é então convertido em uma variável do tipo *string*, concatenado à frase “O resultado é: ” e exibido pelo indicador. Um clique sobre o botão de parada induz o término do *loop while* por enviar “verdadeiro” a seu condicional de término no canto inferior direito. Segue-se então a desativação do LED virtual e o VI é encerrado.

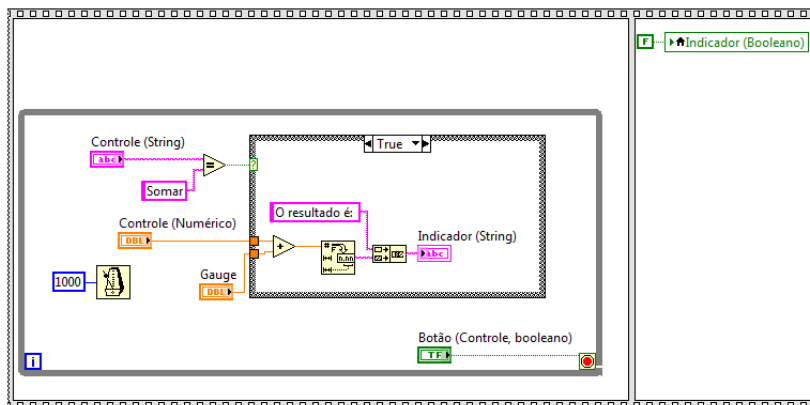


Figura 10 - Diagrama de Blocos com estruturas e blocos básicos
Fonte: proposta pelo autor

O *Conector Pane* localiza-se no canto superior direito do painel frontal quando o VI não se encontra em execução. Juntamente com o Ícone, Figura 11, permite que um VI possa ser convocado do interior de outro e, desse modo, funções sejam condensadas em um único bloco ou novas janelas sejam exibidas em um mesmo programa.



Figura 11 - Conector Pane (esquerda) e Ícone de um Instrumento Virtual
Fonte: proposta pelo autor

O LabVIEW possui várias funções empregadas, muitas vezes, em automação. Detalhamentos de tais funções fogem do escopo deste trabalho. Tutoriais, no entanto, podem ser encontrados nas apostilas dos diversos cursos oferecidos pela National Instruments.

É fundamental, todavia, apresentar noções sobre como ocorre a comunicação entre o programa e os outros dispositivos utilizados neste trabalho (motores de passos e autocolimador). Para tanto, utiliza-se o padrão NI-VISA.

VISA é a sigla para *Virtual Instrument Software Architecture*, uma interface de programação de aplicativos de entrada e saída. VISA pode ser entendido como um padrão para configuração e programação de instrumentos (*hardwares*) a partir de ambientes de desenvolvimento como LabVIEW e LabWindows/CVI por intermédio de

interfaces tais como GPIB, VXI, PXI, Serial, *Universal Serial Bus* (USB) e/ou Ethernet (NATIONAL INSTRUMENTS, 2012).

O envio de dados de um bit por vez é conhecido em ciência da computação como “Comunicação Serial”. Para seu correto funcionamento, é necessária a correspondência de parâmetros entre as duas portas comunicantes, dentre os quais, devem ser informados obrigatoriamente ao bloco de configuração serial VISA, Figura 12: velocidade de sinalização em bits por segundo (*Baud Rate*), quantos bits formam um caractere (*Data bits*) e o número de bits que indicam o final da transmissão (*Stop Bits*).

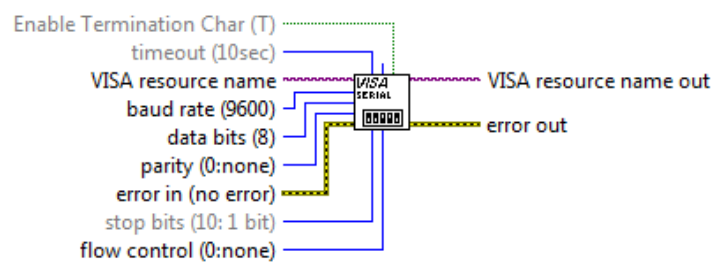


Figura 12 - Bloco VISA de configuração de porta serial
Fonte: software NI LabVIEW

Uma vez configurada a porta serial, utilizam-se blocos VISA para o envio (*Visa Write*) e recebimento (*Visa Read*) de informações a partir da lista de códigos que o *hardware* compreende. Outro bloco destina-se ao término da conexão (*Visa Close*). A Figura 13 exibe um diagrama de blocos que contém os principais ícones VISA; a *string* de entrada de *Visa Write* armazena o comando a ser enviado, que pode ter sido digitado a partir de um controle no painel frontal. Já a *string* de saída de *Visa Read* pode conter uma resposta emitida pelo *hardware* e ser armazenada em um indicador no painel frontal.

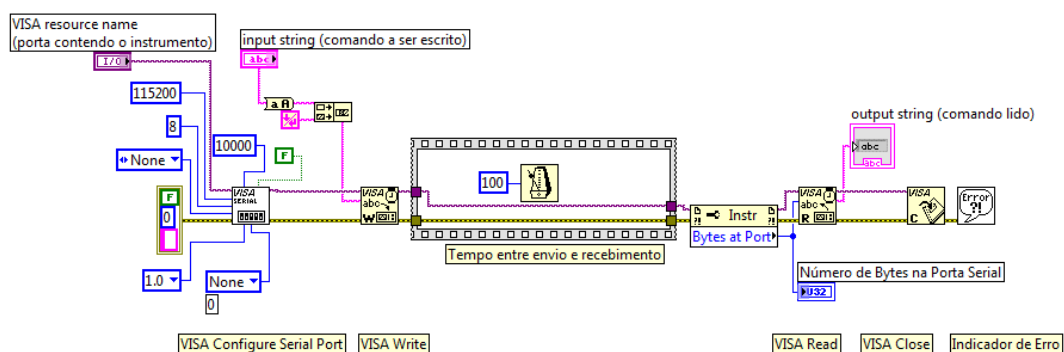


Figura 13 - Principais blocos VISA
Fonte: proposta pelo autor

2.3 Autocolimador e óptica em metrologia

A conjectura de que a história da metrologia tenha se iniciado assim que o homem se reuniu em sociedade e elaborou regras de convivência – como preceitos para trocas – é defendida por Irineu da Silva em sua obra *História dos Pesos e Medidas*. O autor sustenta a hipótese de que, no entanto, as unidades de medida podem ter surgido “a partir do momento em que foi preciso cultivar a terra ou transferir os animais para pastagens mais férteis” (DA SILVA, 2010, p. 38). Logo, é plausível que os primeiros instrumentos de medida tenham sido simples utensílios agrários, como réguas, fios de prumo e níveis topográficos primitivos.

Evidentemente, seus empregos eram manuais e os valores obtidos estavam sujeitos aos erros do operador (como a paralaxe), além daqueles provocados pela baixa precisão desses instrumentos apenas mecânicos.

Da Silva ressalta duas ferramentas de que alguns povos antigos dispunham. O groma romano, Figura 14, era composto por duas barras horizontais fixas e perpendiculares ligadas a um pedestal por meio de um braço e por fios que deveriam estar alinhados às estações (estacas verticais fixas à distância) para a demarcação de ângulos retos.

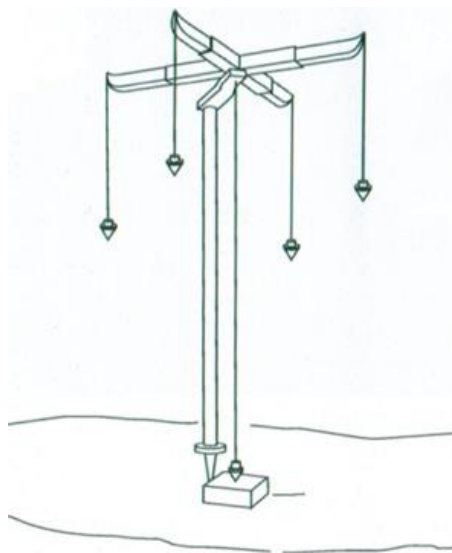


Figura 14 - Figura representativa do groma
Fonte: DA SILVA (2010, p. 189)

A dioptra de Heron de Alexandria, Figura 15, informava também distâncias horizontais, verticais e angulares através de uma mira graduada. Contudo, algo que permitiu um avanço ainda mais significativo foi a introdução da bússola.

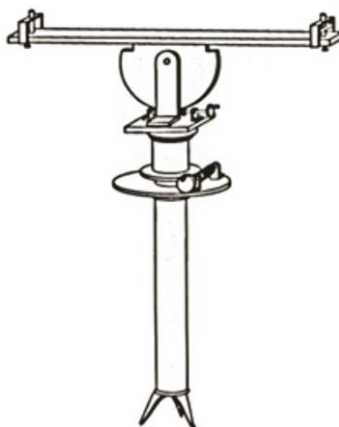


Figura 15 - Figura representativa da dioptra
Fonte: DA SILVA (2010, p. 190)

Os próximos marcos na história de instrumentos para alinhamentos foram as invenções do Teodolitus na Inglaterra do final do século XVI, do telescópio de Hans Lippershey em 1608 e do nível de bolha no século XIX.

O teodolito é um instrumento óptico para a medida de ângulos e, a partir deles, áreas e distâncias. Uma evolução do teodolito é a chamada “estação total”. Ambos são instrumentos topográficos; no entanto, quando se trata da medição automática com alta precisão de ângulos em mecanismos girantes, ainda não são as soluções ideais.

Com um Nível Óptico é possível caracterizar mecanismos em translação. Por exemplo, na Figura 16, a mira é posicionada em duas das extremidades da peça em cada uma de oito posições durante seu deslocamento vertical. As variações nas posições da mira são lidas no instrumento e anotadas em planilha para que se trace a curva do deslocamento real da peça. Por fim, as variações são comparadas à tolerância desejada para se concluir se o movimento é satisfatório.

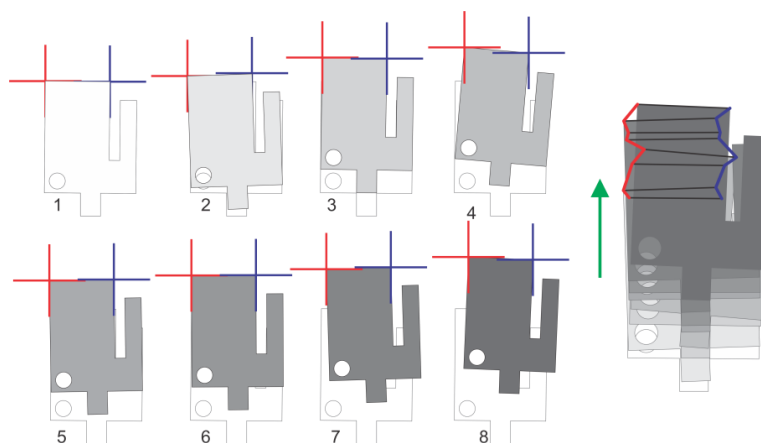


Figura 16 - Caracterização de uma translação a partir de pontos lidos com Nível Óptico
Fonte: proposta pelo autor

Consequentemente, poder-se-iam estimar valores angulares ao se considerarem as coordenadas das extremidades de um elemento em rotação. Todavia, é certo que para a necessidade apresentada no item 2.4, o autocolimador é o instrumento adequado.

Segundo Da Silva (2010, p. 127), ondas eletromagnéticas foram introduzidas na medição topográfica apenas em 1945.

O conceito Luz Colimada (do latim *collimare*, derivado de *collineare*, colinear) refere-se à luz que apresenta raios aproximadamente paralelos, que pouco se dispersam durante sua propagação.

Autocolimação é a composição óptica em que um espelho plano reflete uma luz colimada de volta ao sistema óptico de origem.

Autocolimadores são instrumentos que utilizam de tal composição no propósito de medir pequenas inclinações da superfície reflexiva. A fabricante japonesa Nikon (2013) em seu Centro de Informação na internet descreve algumas das aplicações do instrumento:

[...] em ambientes industriais e fabris para alinhamento com precisão de componentes mecânicos, detecção de movimento angular, monitoramento angular ao longo do tempo, para garantir que não haja algum erro angular em um sistema e para que haja cumprimento com especificações e padrões angulares.

A fabricante TriAngle (2012) exemplifica empregos na medição de pequenos deslocamentos angulares, tais como verificações de paralelismo e linearidade, nivelamento de superfícies e análise de vibração. A Figura 17 ilustra o princípio de medição do autocolimador eletrônico ELCOMAT 3000 produzido pela empresa alemã Möller-Wedel e utilizado neste trabalho.

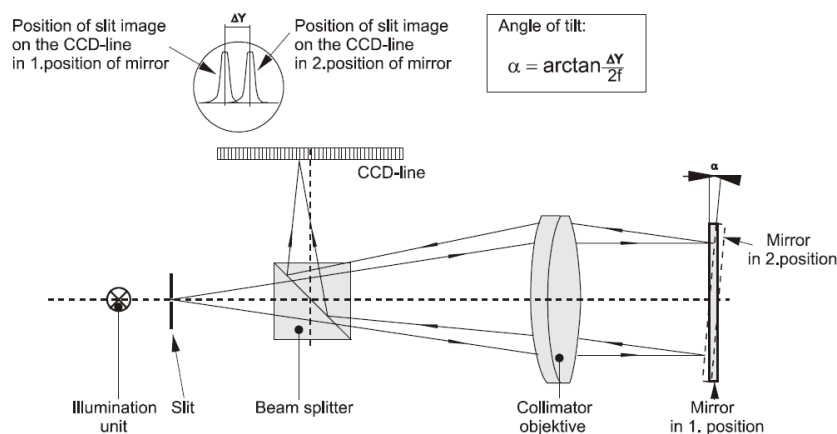


Figura 17 - Princípio de medição do autocolimador eletrônico ELCOMAT 3000
 Fonte: MÖLLER-WEDEL (2008, p. 7)

O ELCOMAT 3000 fundamenta-se em um divisor de feixe que direciona a luz refletida para um fotodetector, no qual são comparadas em arco-segundos as posições dos feixes projetado e refletido. Esses valores (horizontal, X; vertical, Y) podem ser enviados a um computador através dos padrões *Recommended Standard 232* (RS-232) ou USB.

Tais conexões permitem também o envio de comandos do computador para o instrumento – e.g., mensagens em ASCII – a fim de se capturarem suas leituras. Esses comandos são descritos no manual do fabricante (MÖLLER-WEDEL, 2008).

Outro instrumento de medição baseado em ondas eletromagnéticas e cujas leituras podem ser atualmente exportadas a computadores via comunicação serial é o interferômetro.

Seu princípio é, em parte, semelhante ao do autocolimador. Um feixe laser é dividido em um espelho semirrefletor e cada divisão é refletida por espelhos de reflexão total. Então o instrumento se vale da interferência entre os feixes finais para obter medidas. Por exemplo, a montagem da Figura 18 executa leitura de deslocamento linear.

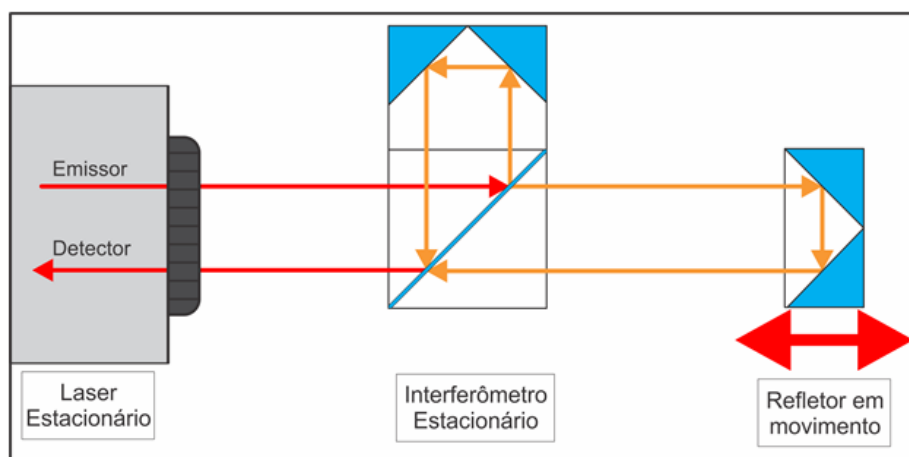


Figura 18 – Princípio do uso de interferometria para medidas de deslocamento linear
Fonte: adaptado de RENISHAW (2013)

Caso se importem os valores obtidos para o LabVIEW, pode-se criar um aplicativo que mapeie a posição do refletor em função de outros parâmetros, como a posição do motor que o desloca ou mesmo condições de temperatura e humidade, as quais podem ser facilmente exportadas ao LabVIEW com a utilização de sensores e *hardwares* de aquisição.

2.4 Contexto em que o trabalho foi aplicado

O aplicativo, embora possa ser utilizado para análise de repetibilidade e precisão de múltiplos produtos, foi desenvolvido, a princípio, para caracterizar os movimentos dos mecanismos construídos para a nova linha de infravermelho do LNLS.

2.4.1 Fonte de Luz Síncrotron

Segundo o portal lightsource.org, que reúne informações sobre fontes de luz ao redor do mundo, as fontes de luz são aceleradores que produzem feixes intensos de Raio-X a Infravermelho em vácuo a fim de promoverem pesquisas básicas e aplicadas em campos desde física a biologia e tecnologia, as quais não são possíveis por intermédio de equipamentos convencionais (LIGHTSOURCE, 2013).

Uma fonte de luz síncrotron é um tipo de acelerador cíclico no qual são utilizados eletromagnetos para alterar o curso dos elétrons que, desse modo, emitem uma luz brilhante denominada “luz síncrotron”, a qual é direcionada a caminhos conhecidos como “linhas de luz”, nas quais os pesquisadores selecionam o comprimento de onda desejado para a análise de suas amostras (CANADIAN, 2013).

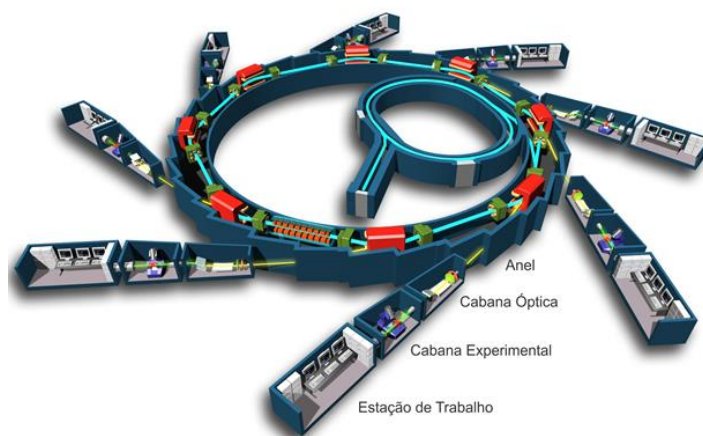


Figura 19 - Esquema de um Síncrotron
Fonte: adaptado de EPSIM (2005)

As linhas de luz, em geral, consistem em uma cabana óptica para focalização e seleção do comprimento de onda; uma cabana experimental, onde são realizados os experimentos (como difração de Raios-X, cristalografia de macromoléculas e espectroscopias de Raios-X e ultravioleta) e uma estação de trabalho, de onde os cientistas operam a linha e analisam os resultados sobre as amostras.

2.4.2 Linha de Infravermelho

A linha de infravermelho (IR) do LNLS objetiva a utilização de um microscópio de varredura de campo próximo (SNOM) para espectroscopia na faixa de radiação infravermelha.

Ela consiste em duas câmaras de espelhos que isolam a faixa desejada e controlam propriedades do feixe, Figura 20, a cabana óptica climatizada contendo o microscópio e a estação de trabalho. O aplicativo aqui desenvolvido destina-se justamente à caracterização dos mecanismos que movimentam esses espelhos M1 a M5 para garantir que estejam nas posições corretas.

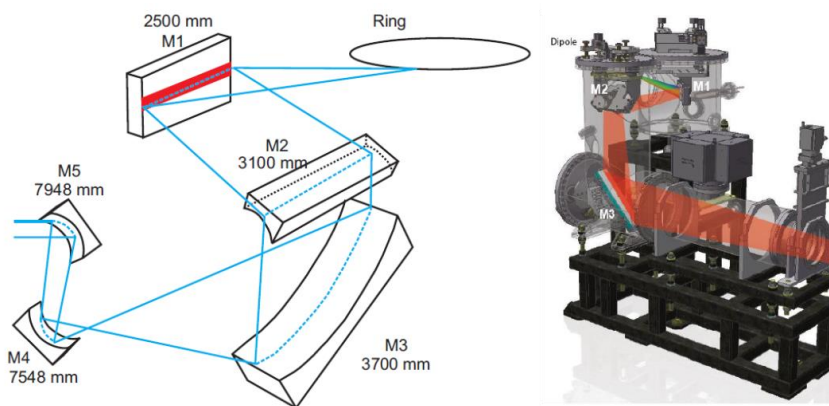


Figura 20 - Esquema Óptico dos Espelhos da IR e projeto CAD da primeira câmara
Fonte: Freitas (2012).

2.4.3 Mecanismos dos espelhos

A Figura 21 exibe imagens CAD dos mecanismos M1 a M5, nas quais foram introduzidos vetores para compreensão dos movimentos realizados.

O espelho do mecanismo M1 translada quando seus três motores verticais são acionados. Dois deles, separadamente, são responsáveis pelos movimentos *roll* e *pitch*, enquanto seu motor horizontal provoca o deslocamento *yaw*.

O espelho de M2 é inclinado 45° em relação ao plano horizontal, no qual está inserido o feixe incidente. Seus motores são responsáveis por *pitch* e *yaw*. M3 é semelhante: dois motores, inclinação de 45° , *pitch* e *yaw* e presença de *weaklinks*. No entanto, a definição de *yaw* em M2 considera o giro do plano horizontal que contém o

feixe incidente, já para M3, o motor *yaw* atua na condição de promover um giro no plano do próprio espelho, conforme ilustrado na figura.

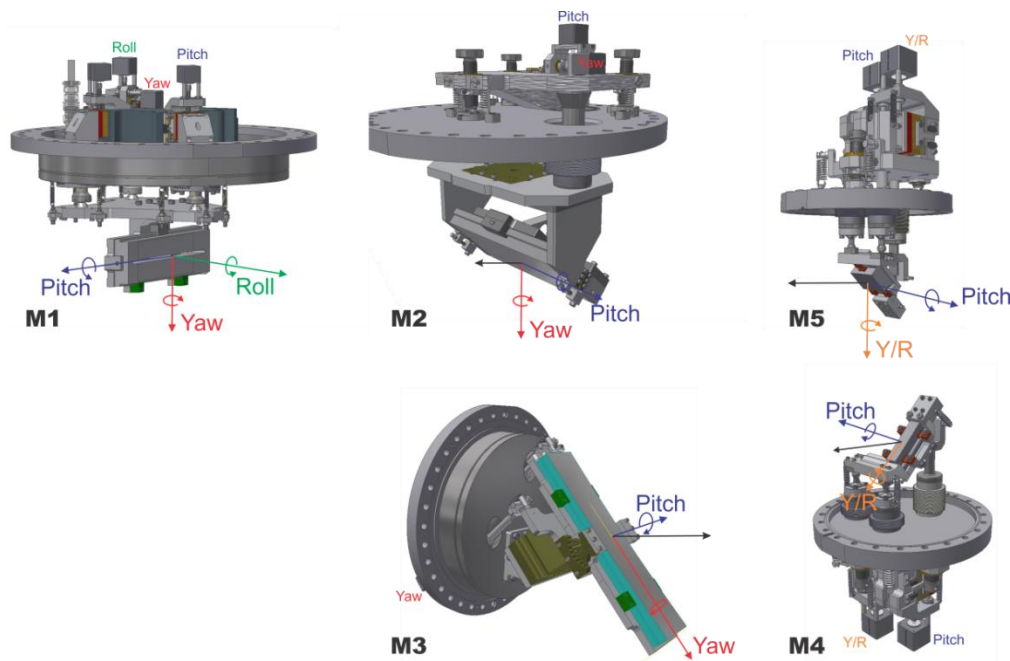


Figura 21 - Movimentos permitidos para M1 a M5
Fonte: adaptada sobre projeto de Pedro Paulo de Souza Freitas

M4 e M5 são semelhantes, promovem *Pitch* e um segundo movimento que combina *roll* e *yaw*.

Cada movimento tem origem no giro de um motor de passo. Como engrenagens exigem elementos lubrificados, elas não são utilizadas em um ambiente de vácuo. Assim, as rotações são proporcionadas por braços de giro, Figura 22, sobre os quais atuam eixos em translação.

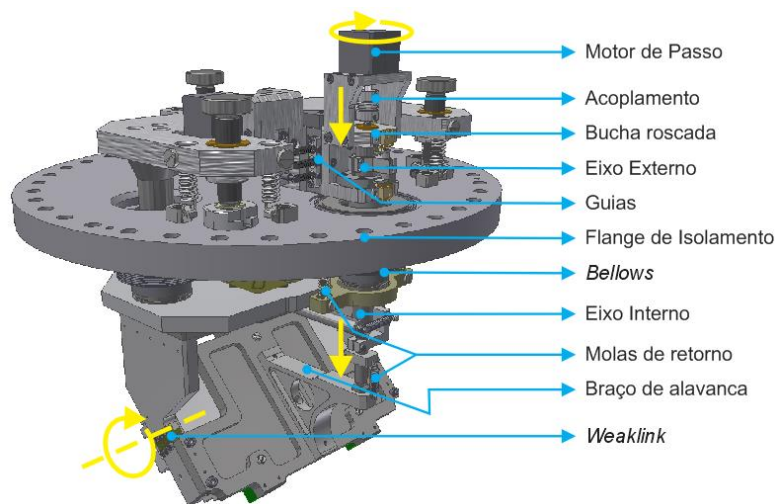


Figura 22 - Elementos para transmissão de movimento Pitch do M2
Fonte: adaptada sobre projeto de Pedro Paulo de Souza Freitas

Da figura, destacam-se os elementos mecânicos apresentados a seguir, os quais justificam os desvios detectados nos movimentos.

2.4.3.1 Acoplamentos, Porcas e Guias Lineares

A princípio, os cinco mecanismos contariam com uniões simples, rígidas, entre eixo do motor e fuso. Entretanto, tal união não garante alinhamento entre esses eixos, o que resulta na precessão do segundo conforme rotação do primeiro.

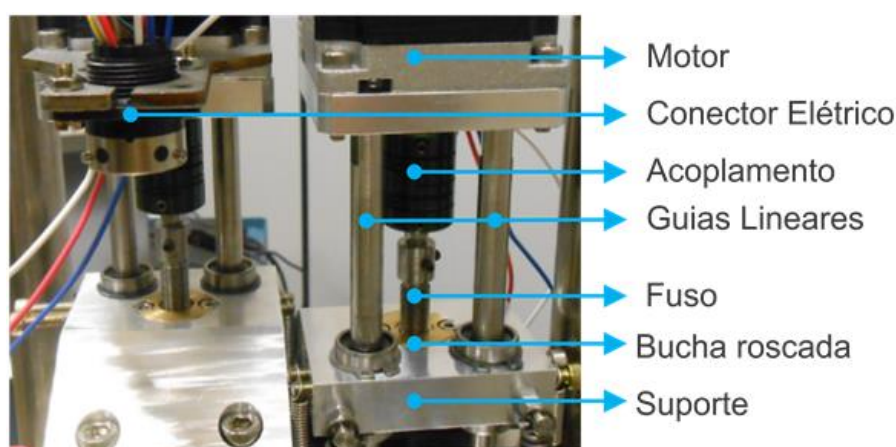


Figura 23 – Elementos mais externos em detalhe

Conseqüentemente, os resultados das varreduras através do aplicativo em LabVIEW apontaram desvios grosseiros. Também foram notados ruído e alta temperatura nos motores. Isso levou à utilização de acoplamentos flexíveis, Figura 23, para compensação de desalinhamento.

Esta solução evita, portanto, que o fuso force a rosca e que as guias lineares forcem suas buchas, o que reduz o ruído e as folgas entre esses elementos.

2.4.3.2 *Weaklinks*, Prismas e Molas

Folgas, baixa precisão em movimento e desgastes são os problemas mais apontados contra o emprego de juntas rígidas. Além disso, elas geralmente exigem lubrificação, o que deve ser evitado em ambientes de alto vácuo.

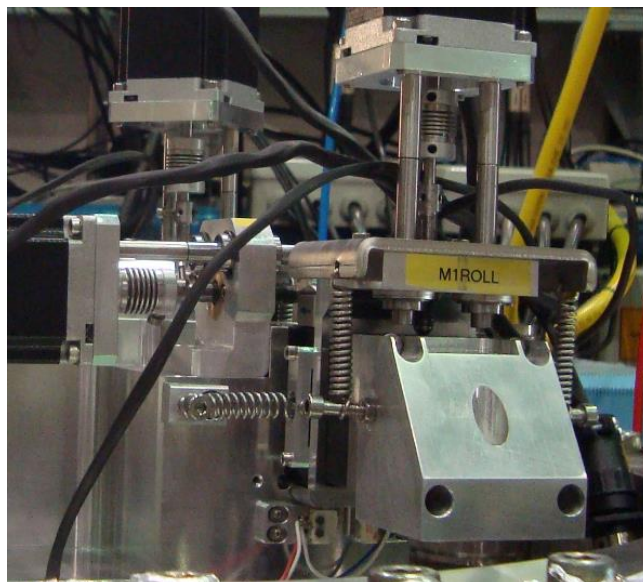


Figura 26 - M1 em detalhe, molas para manutenção de contato.
Fonte: proposta pelo autor

2.4.3.3 *Bellows*

Os *Bellows* são excelentes soluções na transmissão de movimento entre um ambiente aberto e o interior de uma cabana com vácuo. Possuem o formato de sanfona soldada entre dois flanges e admitem compressão entre eles.

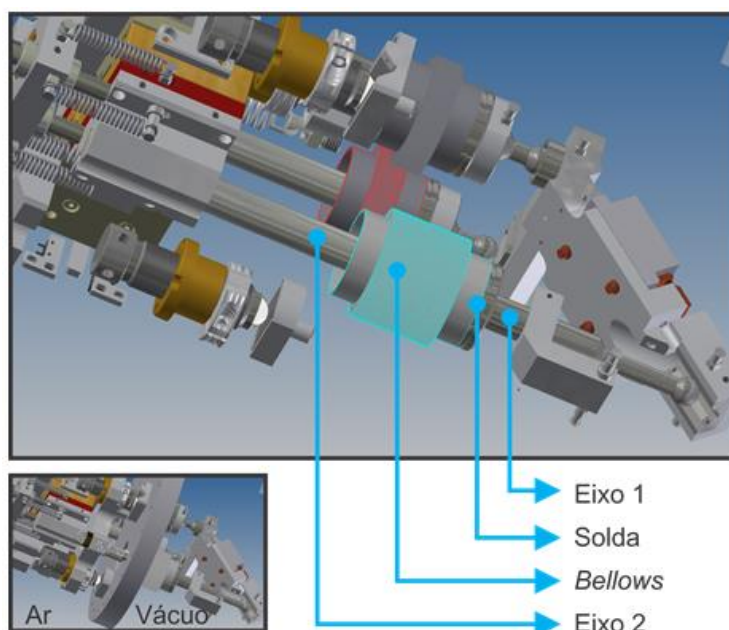


Figura 27 - *Bellows* em M5. Peças (como o Flange) foram ocultadas na imagem ampliada
Fonte: adaptada sobre projeto de Pedro Paulo de Souza Freitas

Naturalmente, o vácuo reduz a pressão em um dos lados do *bellows*, o que altera sua extensão. Consequentemente, os esforços sobre as molas e motores diferem entre as condições de testes (com ambas as extremidades à pressão ambiente) e de aplicação em vácuo, o que pode afetar o curso total, mas não as relações entre graus e passos para esses mecanismos.

3. RESULTADOS

3.1 Montagem do sistema de medição

A Figura 28 esclarece as relações entre os componentes da instalação. Inicialmente, o mecanismo é disposto em sua posição de projeto, a qual é alcançada com o auxílio de um braço de medição tridimensional.

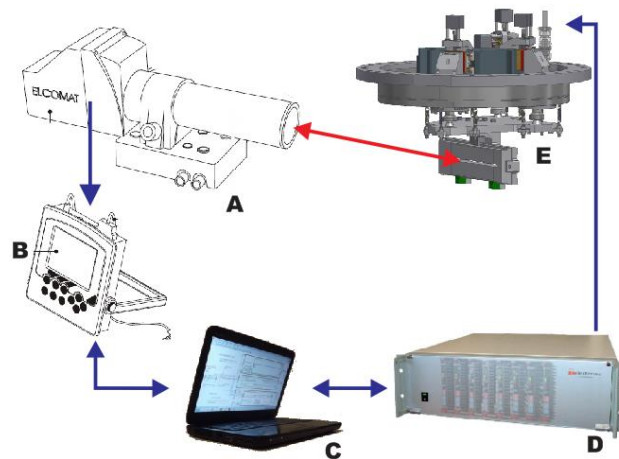


Figura 28 - Montagem do sistema de medição: autocolimador (a) com unidade de controle (b), computador com aplicativo LabVIEW (c), *powerpack* (d) e mecanismo estudado (e)
Fonte: proposta pelo autor

O *powerpack* é responsável por alimentar e controlar os motores do mecanismo segundo comandos do computador. Ordenada a rotação de um motor durante um intervalo centrado na posição de perpendicularidade entre feixe e espelho (os demais atuadores permanecem na posição zero), o autocolimador passa a detectar variações angulares e enviá-las a seu sistema de controle, o qual atende ao pedido do computador pelos dados.

Os espelhos utilizados nos ensaios devem ser posicionados de tal modo que um dos deslocamentos do mecanismo coincida com uma direção de leitura do instrumento. Na maioria dos casos, foram posicionados perpendicularmente ao solo, Figura 29, de modo que a leitura Y – doravante denominada “v” – do autocolimador correspondeu ao *Pitch* e, a medição horizontal (indicação X, a ser referenciada por “h”) ao *yaw*.

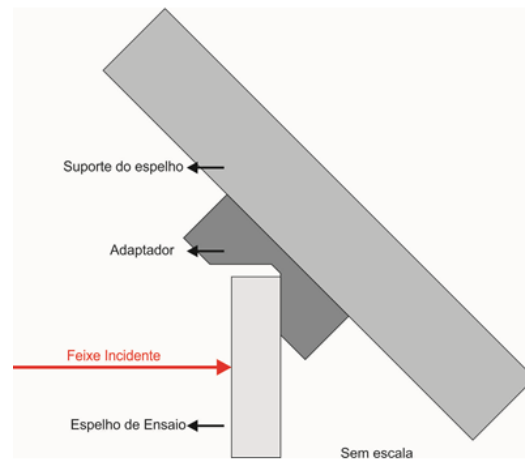


Figura 29 – Adaptação para uso de espelho de testes na vertical
Fonte: proposta pelo autor

3.2 Primeira versão do programa

A Figura 30 mostra a tela principal da primeira versão satisfatória do aplicativo. Essa tela é dividida em duas partes. A metade esquerda contém controles nos quais devem ser selecionadas as portas de comunicação USB em que os instrumentos (controlador e autocolimador) estão conectados. Logo abaixo, há um botão com a função de habilitar ou desabilitar as chaves de final de curso, indicadores das posições de até quatro motores, indicadores das leituras X e Y do autocolimador e variáveis booleanas no formato LED que indicam (I) se o programa está em funcionamento e (II) os momentos em que dados são importados do autocolimador. Contém ainda um controle do tipo *string* no qual o usuário digita o diretório em que os arquivos serão salvos e um indicador que assinala qual o último arquivo salvo. Tal método será substituído (vide item 3.4.5) por uma caixa padrão para salvamento de arquivo.

Esses arquivos apresentam extensão *.txt e guardam os valores dos desvios angulares captados pelo autocolimador em função da posição dos motores durante a movimentação de um deles.

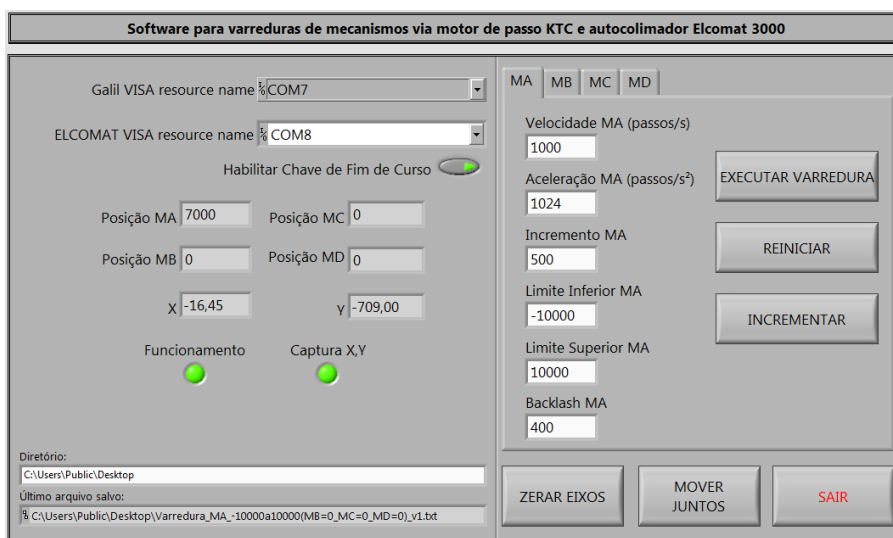


Figura 30 - Tela principal do aplicativo
Fonte: proposta pelo autor

A metade direita do aplicativo apresenta um *Tab Control* com quatro abas, indicativas dos quatro motores com que o programa trabalha, pois fora projetado em função do número máximo de atuadores nos mecanismos da IR. Em cada uma das abas encontram-se controles numéricos para configuração dos movimentos. A saber: velocidade, aceleração, quantidade de passos por incremento, limites superior e inferior do curso e número de passos a ser usado para correção de folga no sistema (*backlash*). No lado direito encontram-se também botões de ação aos quais estão associados eventos conforme ilustrado na Tabela 1.

Tabela 1 - Eventos disponíveis no VI

Botão	Descrição do evento
Executar Varredura	Movimenta o atuador selecionado em incrementos entre os quais importa dados do autocolimador e salva em arquivo de texto
Reiniciar	Situa o atuador selecionado em sua posição inicial
Incrementar	O atuador gira segundo quantidade de passos informada
Zerar Eixos	Todos os atuadores retornam à posição zero
Mover Juntos	Todos os atuadores se movimentam segundo os incrementos informados
Sair	Encerra o aplicativo

3.3 Breve explicação dos eventos

O evento referente à modificação do estado do botão “reiniciar” segue o algoritmo representado na Figura 31.

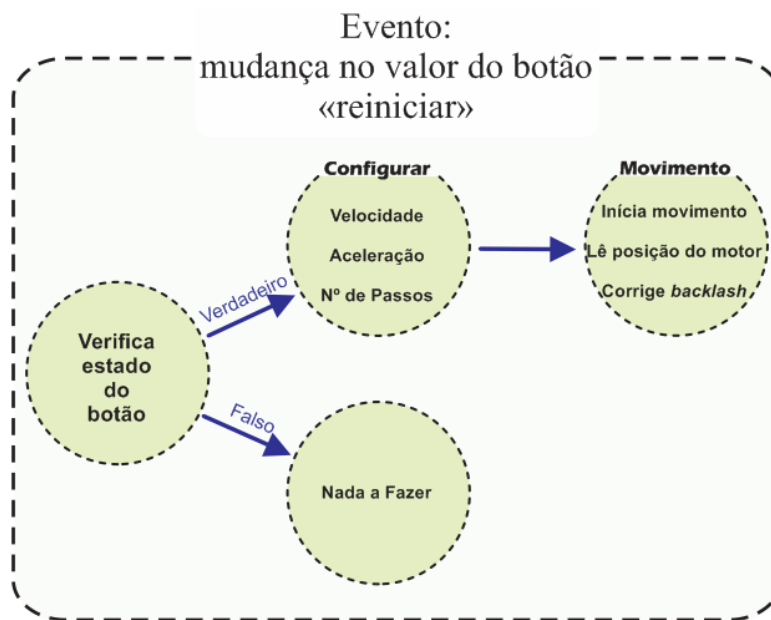


Figura 31 - Sequência de episódios para a função "reiniciar"
Fonte: proposta pelo autor

O clique sobre o botão altera seu valor de falso para verdadeiro. Quando falso, nada deve ocorrer; quando verdadeiro, terá início o movimento do motor selecionado até uma posição estipulada pelo controle “Limite Inferior MX” (em que X é utilizado aqui para generalizar referências aos motores A, B, C ou D).

O passo seguinte é a configuração de velocidade, aceleração e número de passos a serem utilizados no deslocamento entre as posições atual e zero subtraído da quantidade de passos necessários para a eliminação da folga (*backlash*).

Para cumprir esse objetivo, comandos são enviados a blocos *Visa Write* em *SubVIs*. A descrição de cada comando, como já antecipado, torna a redação cansativa e muito específica, uma vez que eles variam conforme o controlador utilizado e podem ser encontrados em seus manuais.

Após as configurações, é enviado pelo *Visa Write* o comando que inicia o movimento (para o controlador utilizado, é a *string* BG seguida pela letra do motor, sem espaços).

Durante o deslocamento, a posição (em passos) é lida e exibida no indicador correspondente. Todavia, mais que proporcionar essa informação ao usuário, o ciclo

While, em detalhe na Figura 32, é utilizado para que o programa aguarde o término da rotação antes de prosseguir com novos comandos; o que se faz necessário, uma vez que o controlador enviaria um sinal de erro caso uma nova ordem de movimentação fosse empreendida. Porquanto o ciclo apenas se encerra quando as posições – lidas constantemente – começarem a se repetir, o que significa que o motor parou de se movimentar.

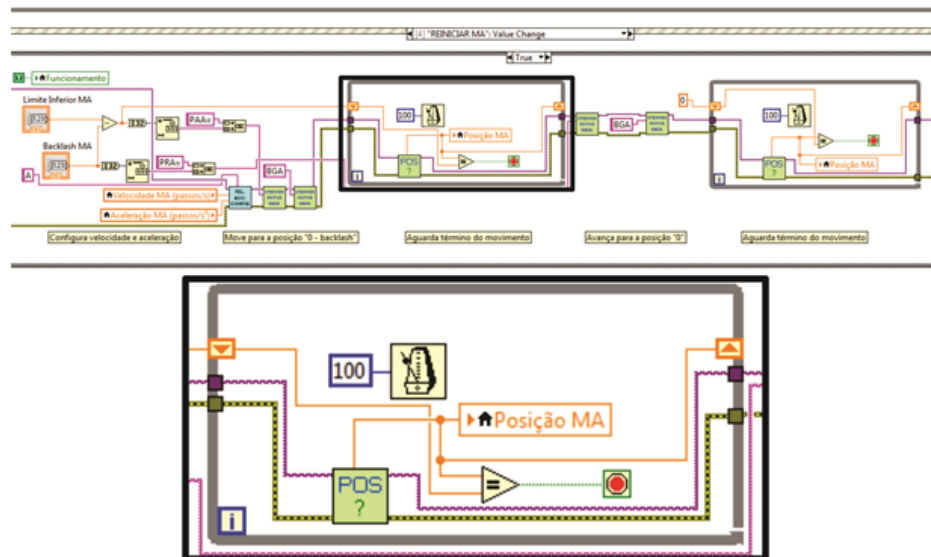


Figura 32 - Trecho do diagrama de blocos no evento “Reiniciar MA”. No detalhe, ciclo *while* que aguarda o término do movimento à medida que informa as posições do atuador
Fonte: proposta pelo autor

Finalmente, é ordenado que o motor desloque-se com o valor de *backlash*, para que retorne à posição zero.

A função “incrementar” respeita lógica semelhante e pode ser explicada pelo roteiro da Figura 33.

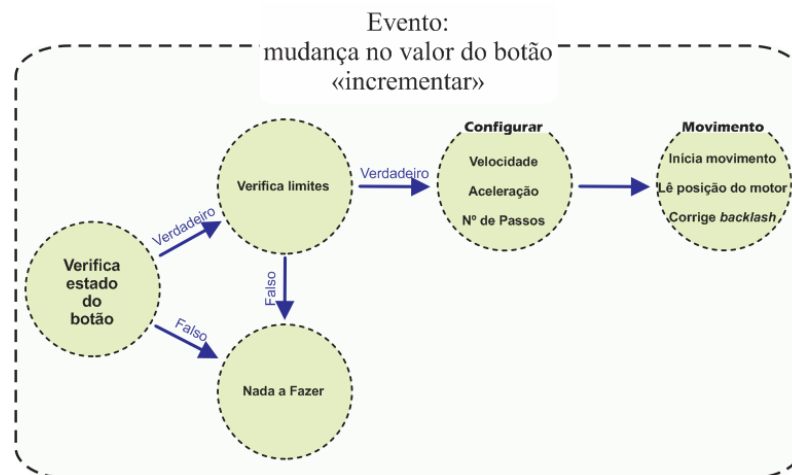


Figura 33 - Sequência de episódios para a função "incrementar"
Fonte: proposta pelo autor

O programa verifica se a posição pretendida é inferior ao limite, configura velocidade, aceleração e quantidade de passos e executa o movimento com correção de folga. A Figura 34 apresenta os blocos principais deste evento.

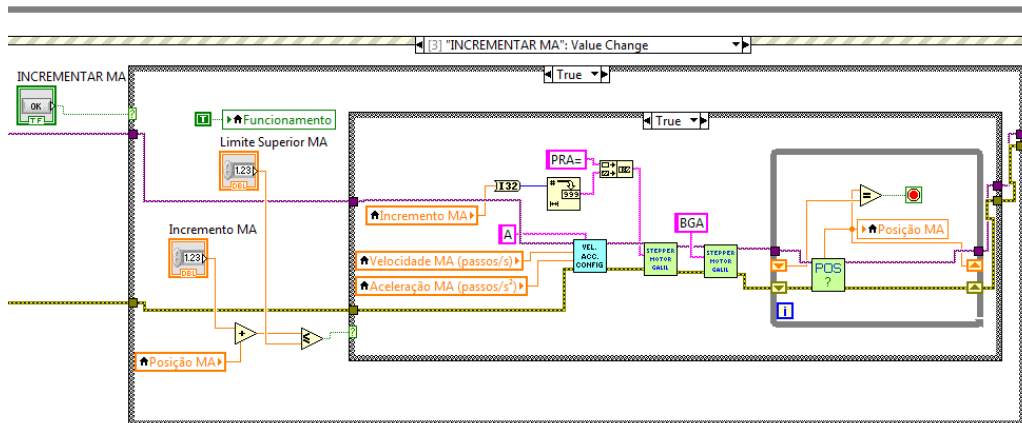


Figura 34 - Trecho do diagrama de blocos no evento “Incrementar MA”. Nota-se, no canto direito, o *loop* de verificação de final de movimento em função da estabilização da posição do motor.

Fonte: proposta pelo autor

O evento associado à alteração da condição do botão “Mover Juntos” é semelhante ao anterior. Nele, ocorrem os ajustes de velocidade, aceleração e giros para ambos os quatro atuadores. Por conseguinte, é necessário verificar os pontos de término todos os eixos conjuntamente, o que é solucionado pelo ciclo *while* da Figura 35. Nele, a condição de parada é que os quatro eixos estejam estacionários.

Esse mesmo ciclo é utilizado duas vezes (antes e após correção de *backlash*) no evento associado ao botão “zerar eixos”, que difere do anterior pelo fato de objetivar, não incrementos, mas a posição zero, associada, na maioria das vezes, à condição central dos mecanismos.

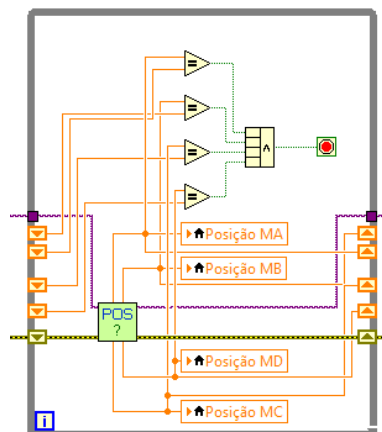


Figura 35 - Extrapolação do *looping while* de encerramento para movimentos simultâneos

Fonte: proposta pelo autor

As serventias das ações supracitadas restringem-se basicamente a posicionamentos e ajustes no intuito de preparar o sistema para o evento denominado “Executar Varredura”, que relaciona o controle de motores de passo às leituras praticadas pelo instrumento de medição e é esclarecido pelo esquema da Figura 36.

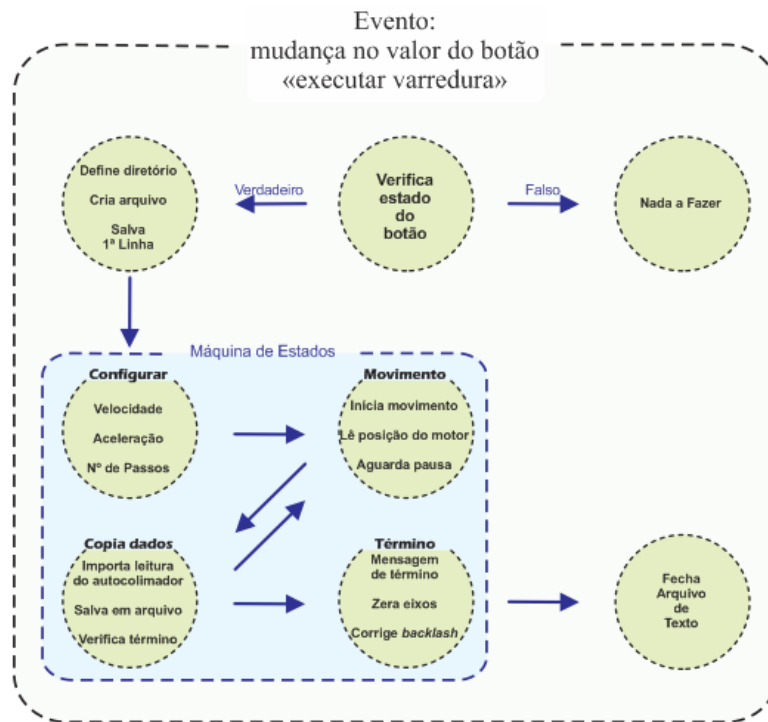


Figura 36 - Sequência de episódios para a função "executar varredura"
Fonte: proposta pelo autor

A primeira ação do evento é a criação do arquivo de texto em que as informações serão armazenadas – Figura 37 – cujo título é determinado primeiramente pelas localizações dos eixos, porém pode ser modificado em uma caixa de confirmação.

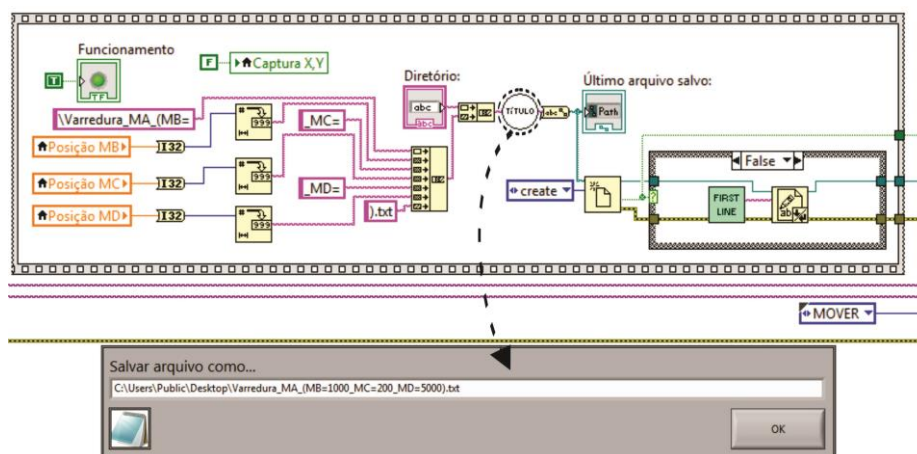


Figura 37 - Trecho do diagrama de blocos do evento "executar varredura" e painel frontal do SubVI de salvamento de arquivo
Fonte: proposta pelo autor

Posteriormente é inicializada uma máquina de estados que altera entre as ocasiões de rotação do motor e de leitura e salvamento dos dados. Os valores são lidos e salvos no estado “exporta”, conforme elucidado pela Figura 38. Estado esse indicado pelo acionamento da variável booleana do tipo LED “Captura X, Y” no painel frontal.

Os *SubVIs* que executam a leitura – detalhados na figura – foram adaptados do driver para LabVIEW disponibilizado pela fornecedora Vermont Photonics.

O *SubVI* “Criar Linhas Arquivo.vi” foi desenvolvido para reunir as variáveis referentes às posições dos atuadores e as leituras X e Y do autocolimador em uma única *string* que será armazenada no arquivo de texto.

Ao término da varredura, uma mensagem de término surge na tela e os eixos retornam para a posição zero.

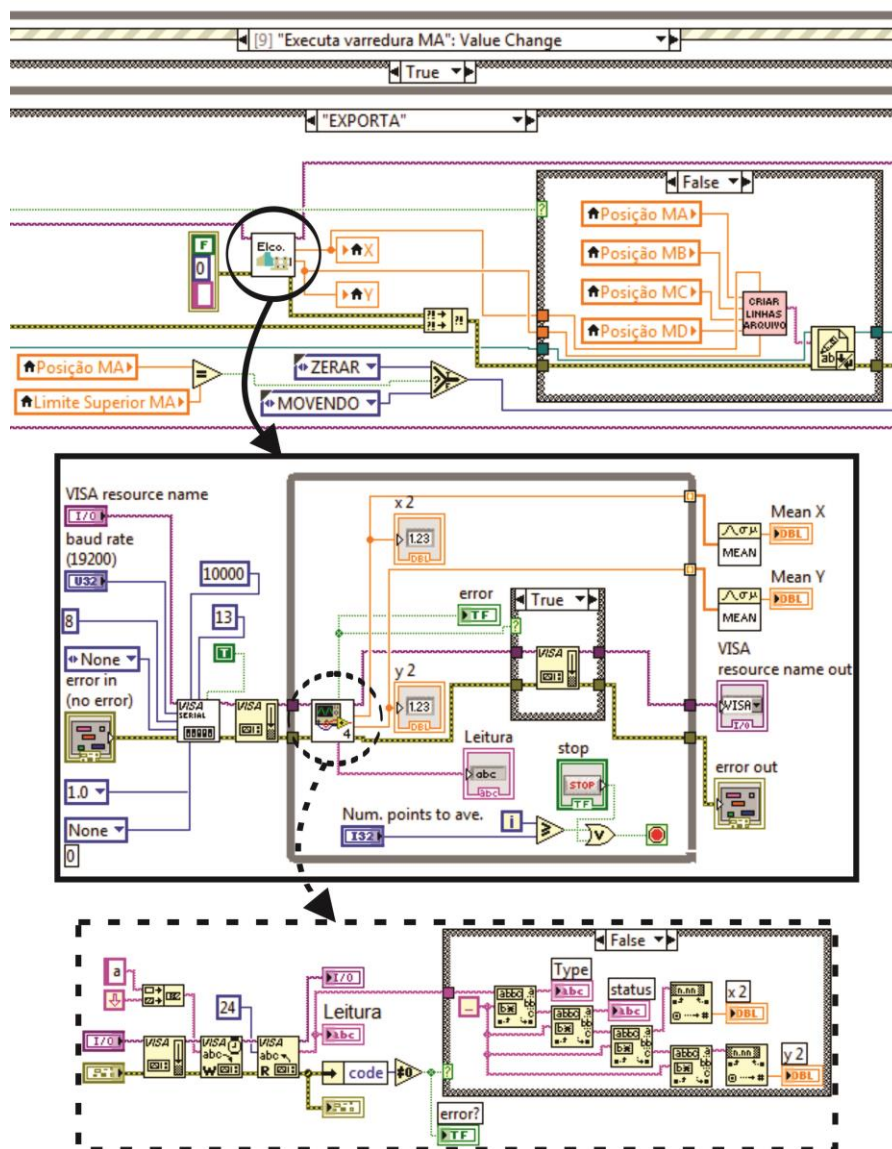


Figura 38 - Trecho do comando "executar varredura". Em destaque, *SubVIs* relacionados à captura de pontos pelo autocolimador
Fonte: proposta pelo autor

3.4 Melhorias

Embora o programa tenha atendido às exigências dos ensaios nos mecanismos. É desejável que ele seja o mais abrangente possível, visando outras aplicações. O código mostrou-se muito extenso, o que leva a uma interpretação demorada por outros programadores, além disso, o acionamento de uma função travava o painel frontal e impedia que outras pudessem ser “enfileiradas”. Deste modo, foram listadas adaptações que poderiam ser efetuadas:

- Melhora na visualização do código no Diagrama de Blocos
- Aperfeiçoamento na tela de salvamento de arquivo
- Introdução de barra de status no acompanhamento das varreduras.
- Possibilitar o enfileiramento de comandos
- Expansão do programa para oito motores
- Inserção das VIs em um único arquivo de projeto
- Controle, pelo usuário, da habilitação da correção de *backlash* e da escolha sobre o sentido em que isso é feito.
- Inserir dados de *setup* no arquivo de texto

3.4.1 Filas e Loops Paralelos

Para que a tela do painel frontal não “congele” durante a execução de um aplicativo, deve-se tornar seu manuseio independente das funções em andamento. Isso é possível com a separação das atividades em ciclos paralelos, que possibilitam que suas execuções sejam concomitantes quando necessário.

Para o programa desenvolvido, Figura 39, os eventos associados a alterações no painel frontal foram alojados em um *loop while* produtor. Dentro desse ciclo, os eventos enviam comandos ao *loop while* consumidor através do sistema de *queues* (filas) do LabVIEW, o que “descongela” o painel frontal e propicia o enfileiramento de comandos.

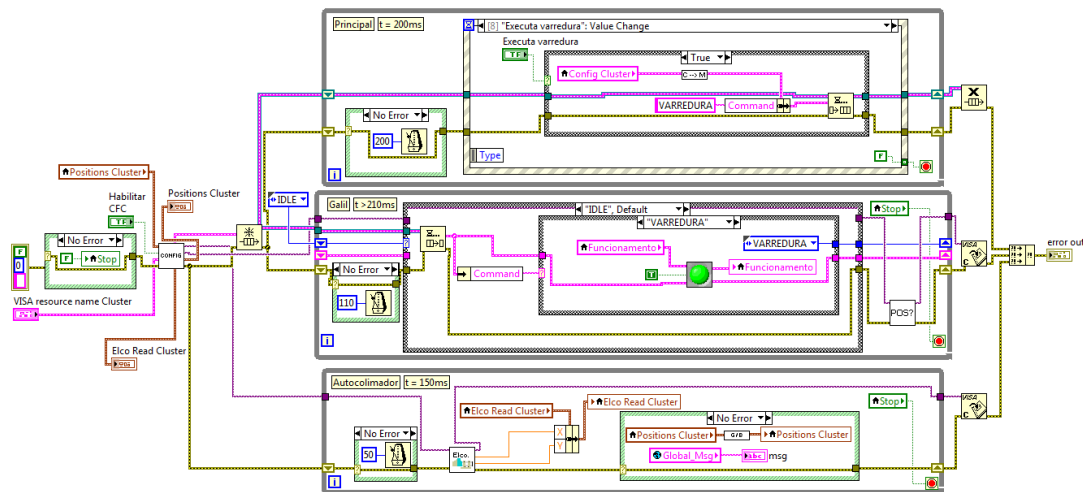


Figura 39 - Diagrama de Blocos do Programa reduzido a uma única tela
Fonte: proposta pelo autor

Por exemplo, suponha que o mecanismo I utilize os motores A, B e C e que se deseje planejar uma série de atividades ao longo do dia sem a presença constante do usuário. O aplicativo, ao enfileirar em sua memória uma sequência de eventos (originários nos cliques sobre os botões), oferece a oportunidade de “agendar”, e.g., uma varredura para o motor A seguida de outras para os motores B e C.

Um terceiro *loop* paralelo e independente dos demais é utilizado para leitura constante das posições indicadas pelo autocolimador e armazenamento nos indicadores correspondentes, os quais são chamados nos outros *loops* quando necessário.

Em resumo, um primeiro ciclo administra os eventos no painel frontal; um segundo, as funções que incluem comandos VISA ao controlador dos atuadores e; um terceiro, abrange as configurações VISA relacionadas ao instrumento de medida.

3.4.2 A Importância das *SubVIs*

Quando trechos do código são repetidamente utilizados em locais diferentes, é preferível que eles sejam condensados em sub-rotinas; no LabVIEW, *SubVIs*. Isso facilita a visualização do diagrama de blocos e permite que a alteração de trechos idênticos seja efetuada uma única vez. Por exemplo, caso sejam acoplados mais motores, não é necessário que todo o código seja adaptado para a presença desse novo atuador.

Outra vantagem importante na utilização de *SubVIs* é que elas facilitam o descobrimento de erros, pois, na maioria das vezes, podem ser testadas independentemente do VI principal.

Uma amostra da simplificação proporcionada por *SubVIs* é fornecida pela Figura 40. A configuração de velocidade e aceleração é exigida em numerosos momentos. Portanto, ela pode ser facilitada pela utilização de um único ícone cujas variáveis de entrada – instituídas no *Conector Pane* – sejam a velocidade e a aceleração estipuladas, qual motor é referenciado e qual a porta em que o controlador está conectado.

Analogamente, o envio de informações ao controlador pode ser sintetizado em um terceiro *SubVI*, o qual recebe as informações a serem escritas no *VISA Write* e devolve aquelas fornecidas pelo *VISA Read*.

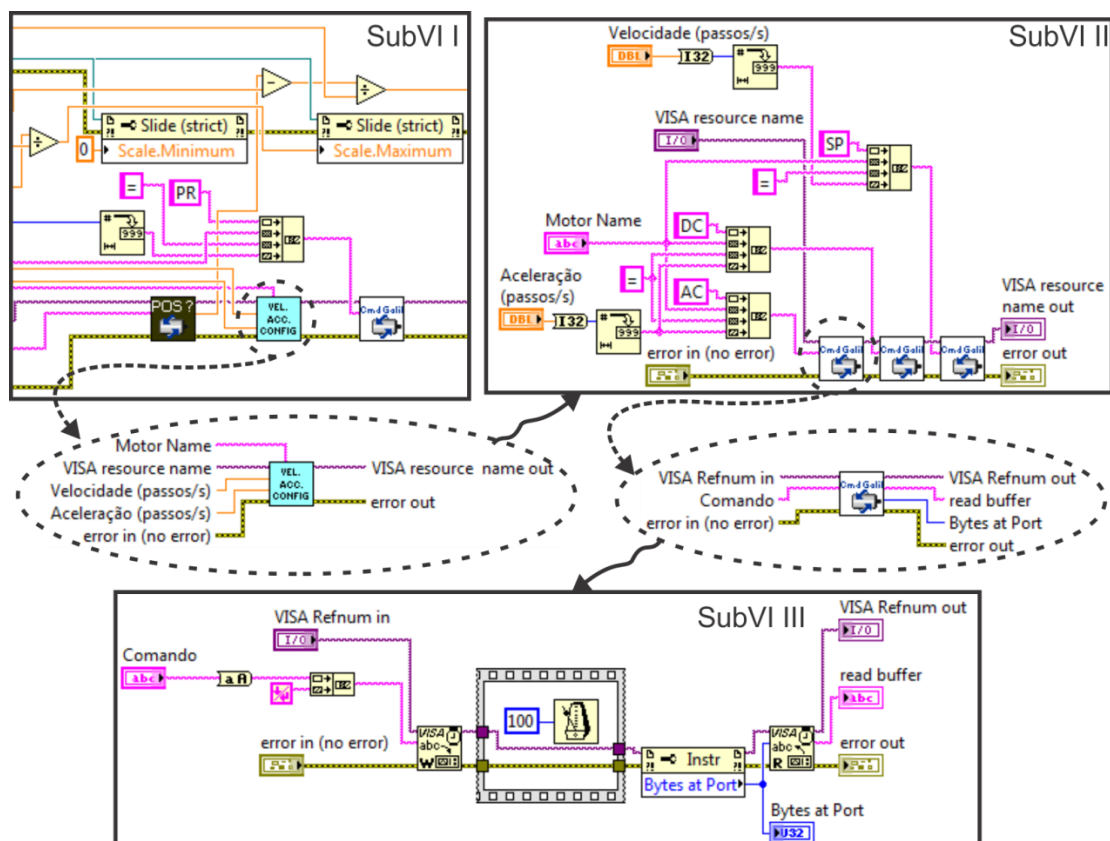


Figura 40 - Desmembramento de um ícone representativo de *SubVI*.

Fonte: proposta pelo autor

3.4.3 Vantagens na utilização de *clusters* e variáveis globais

Na nova versão do programa, Figura 41, controles e indicadores foram distribuídos entre *clusters* segundo suas funções a fim de facilitarem a programação e a visualização tanto do painel frontal quanto do diagrama de blocos.

Os controles das propriedades dos motores (como velocidade e aceleração permaneceram os mesmos e têm seus valores alterados conforme o usuário seleciona o

motor em uma caixa de combinação localizada no “*Config Cluster*” ou reescreva os valores nas caixas de texto no cluster denominado “*Motor Properties*”.

No momento em que um motor é selecionado na *combobox*, os dados referentes a ele são exibidos no “*Motor Properties*”. Para que isso seja possível, as propriedades de cada atuador são armazenadas em *clusters* localizados em variáveis globais, Figura 42.

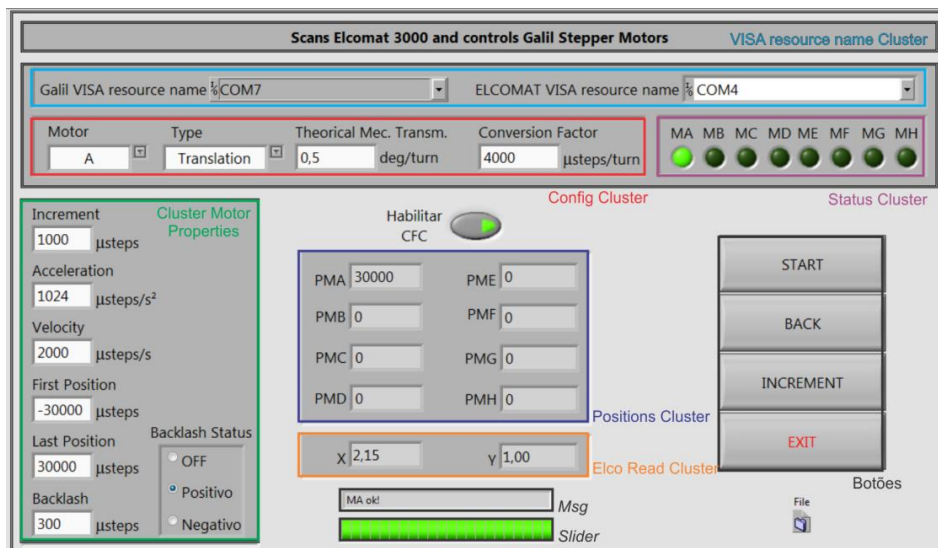


Figura 41 - Painel Frontal da nova versão. Observa-se que o motor é selecionado a partir de uma *combobox* (no “*Config Cluster*”) não mais por abas repetitivas.
Fonte: proposta pelo autor

Tais variáveis guardam esses e outros dados que são chamados em diversos momentos do programa. A solução de um cluster por motor dentro da variável global é análoga à criação de uma matriz com as características deles.

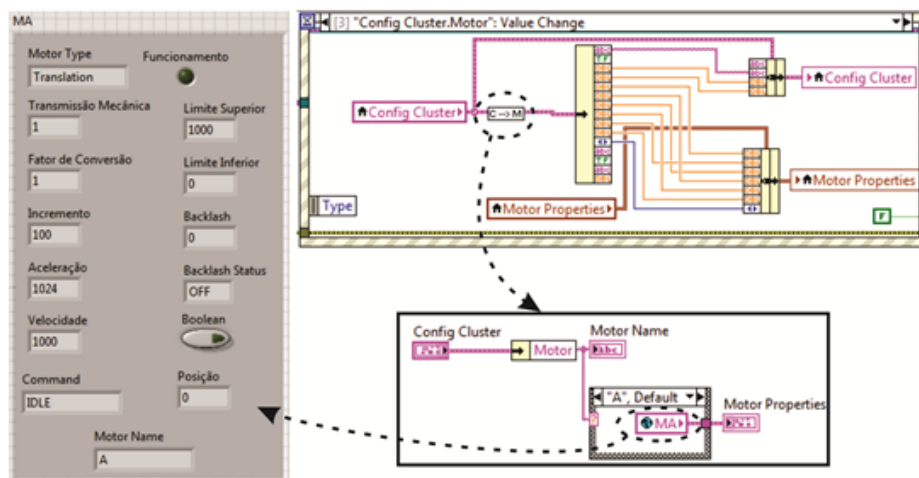


Figura 42 - Evento no *loop* produtor associado à seleção de motor na *combobox*. No detalhe, *SubVI* para chamada do cluster referente ao motor selecionado na variável global, mostrada à esquerda.
Fonte: proposta pelo autor

Uma vez que as configurações dos atuadores ficam armazenadas em VIs independentes, elas podem ser convocadas dentro de quaisquer funções em quaisquer *SubVIs* ou *loops* distintos.

3.4.4 Expansão para oito motores

Da maneira como o programa se apresentava anteriormente (com divisão por abas), a edição dos códigos era custosa, pois os comandos deveriam ser alterados quatro vezes (uma para cada motor).

Portanto, era conveniente que o motor a ser controlado fosse mais um parâmetro de entrada nas funções, independentemente da aba selecionada e de modo que, caso seja necessário editar o programa ou inserir mais motores, o programador tenha trabalho mínimo.

3.4.5 Outras alterações

A nova metodologia adotada para o armazenamento de dados em arquivo *.txt elimina a necessidade do desenvolvimento de tela específica para salvamento, uma vez que aproveita da tela padrão do Windows devido à utilização da caixa “*File Dialog*”, disponível no menu de funções de arquivo avançadas do LabVIEW.

Foi introduzida também uma barra de progresso (vide *slider* na Figura 41) a fim de que o usuário possa acompanhar os tempos percorrido e restante em sua análise.

A Figura 43 mostra os blocos da “máquina de estados” referentes à função “Executar Varredura” (apresentada na Figura 36), em que *SubVIs* relacionadas a arquivo (VARREDURA/*Create Title*, VARREDURA/*Create File*, EXPORTA/*Save to File*) e ao *slider* (VARREDURA/*Start Sweeping*, EXPORTA/*Refresh Slider*) foram utilizadas. Não convém ao presente texto exibir pormenores dos códigos em cada *SubVI*.

Para facilitar a administração dos diversos arquivos, aproveitou-se da opção do LabVIEW de criação de um arquivo de projeto, no qual os demais são inseridos.

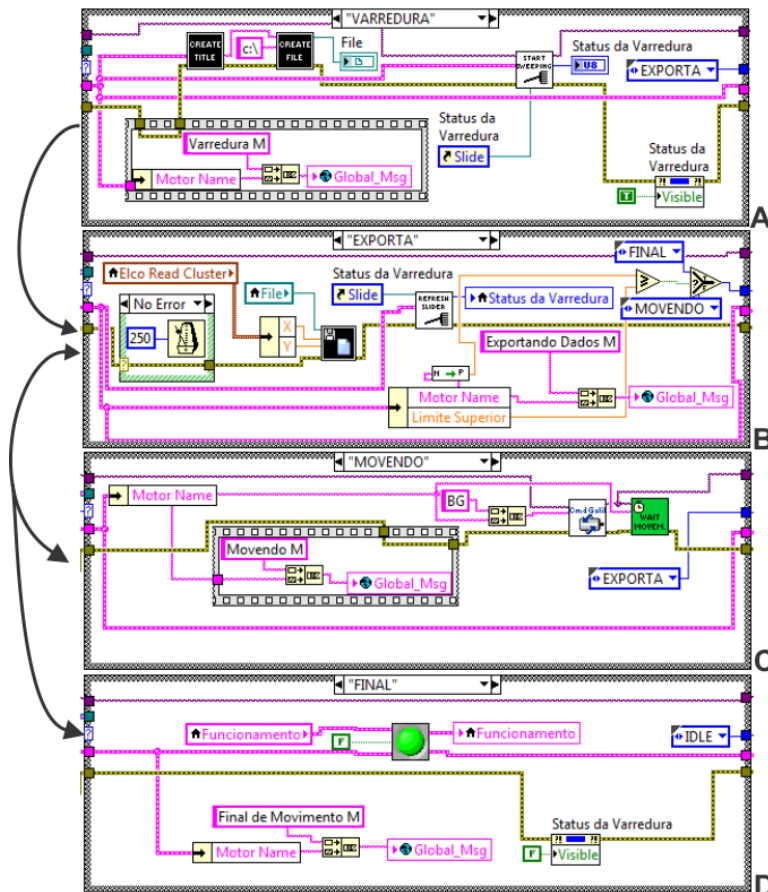


Figura 43 – Máquina de Estados referente à função de varredura. A, configurações do arquivo e do *slider*; B, armazenamento de dados e atualização do *Slider*; C, movimentação do motor; D, término
 Fonte: proposta pelo autor

Em uma etapa do trabalho, foram desenvolvidas funções para análise automática dos dados (criação de gráficos, máximos e mínimos, cálculos das relações passo do motor/desvios no movimento). Contudo, dada a facilidade em se adotar uma solução (de fácil compreensão e melhor aparência) através de macros em Microsoft Excel, tais atividades foram desconsideradas.

Desse modo, o arquivo de texto é gerado conforme a Figura 44.

```

Motor Type: Translation
Theoretical Mechanical Transm. (unit/turn): 0,500000
Conversion Factor (usteps/turn): 4000,000000
Increment (usteps): 500,000000
Acc (ustep/s^2): 1024,000000
Velocity (usteps/s) 2000,000000
Superior Limit (usteps) 28500,000000
Inferior Limit (usteps) -28500,000000
Start Position (usteps) -28500,000000
Backlash (usteps) 200,000000

Start: 16/10/2013 21:56

PMA PMB PMC PMD PME PMF PMG PMH X Y
-28500 0 0 0 0 0 0 0 0,006111 0,256389
-28000 0 0 0 0 0 0 0 0,005505 0,241919
-27500 0 0 0 0 0 0 0 0,005051 0,226667
-27000 0 0 0 0 0 0 0 0,004722 0,211389
-26500 0 0 0 0 0 0 0 0,004444 0,196288
-26000 0 0 0 0 0 0 0 0,004167 0,181111
-25500 0 0 0 0 0 0 0 0,003889 0,166035
-25000 0 0 0 0 0 0 0 0,003434 0,151086
  
```

Figura 44 - Trecho inicial de arquivo de texto produzido para uma varredura em M3
 Fonte: proposta pelo autor

3.5 Limitações

Embora a versão atualizada permita “enfileirar eventos”, ela não admite que funções sejam concomitantes, por exemplo, duas varreduras em motores de mecanismos diferentes conectados ao mesmo *powerpack* ou incrementos em distintos motores com correção de *backlash*, porque o *loop* que interage com o *powerpack* – no qual existe a função que aguarda o término de movimento – deve ser finalizado antes de receber novos elementos da fila.

Foi desenvolvida uma versão que solucionava essa necessidade para os casos em que a correção de *backlash* não fosse necessária. Nessa circunstância, bastava enfileirar as ordens a serem enviadas ao controlador. Todavia, ou o programa ficaria isento de *feedback* e sujeito a falhas quando um motor ainda em movimento recebesse nova ordem ou estaria restrito a uma única função (incrementar ou varrer) caso a verificação de término de deslocamento fosse efetuada no caso IDLE de uma máquina de estados no *loop* produtor (logo, o *loop* consumidor seria exclusivo para envio e recebimento de comandos VISA).

Uma sugestão para a solução dessa questão é a adoção de outra lógica de programação que incluía, por exemplo, a implementação de rotinas através da própria linguagem de programação do controlador.

4. ESTUDO DE CASOS

4.1 Varreduras nos mecanismos da linha de infravermelho

Antes de se obterem conclusões a respeito da precisão de equipamentos, é necessário conhecer a margem de erro dos estudos. Com esse objetivo, foi desenvolvido um aplicativo em LabVIEW a fim de examinar a estabilidade do autocolimador quando executa a leitura de dados sobre um espelho fixo e perpendicular ao feixe projetado.

A Figura 45 demonstra que o instrumento aponta para menores variações quando em ambiente de temperatura controlada e solo isolado de vibrações quando comparado a uma sala comum.

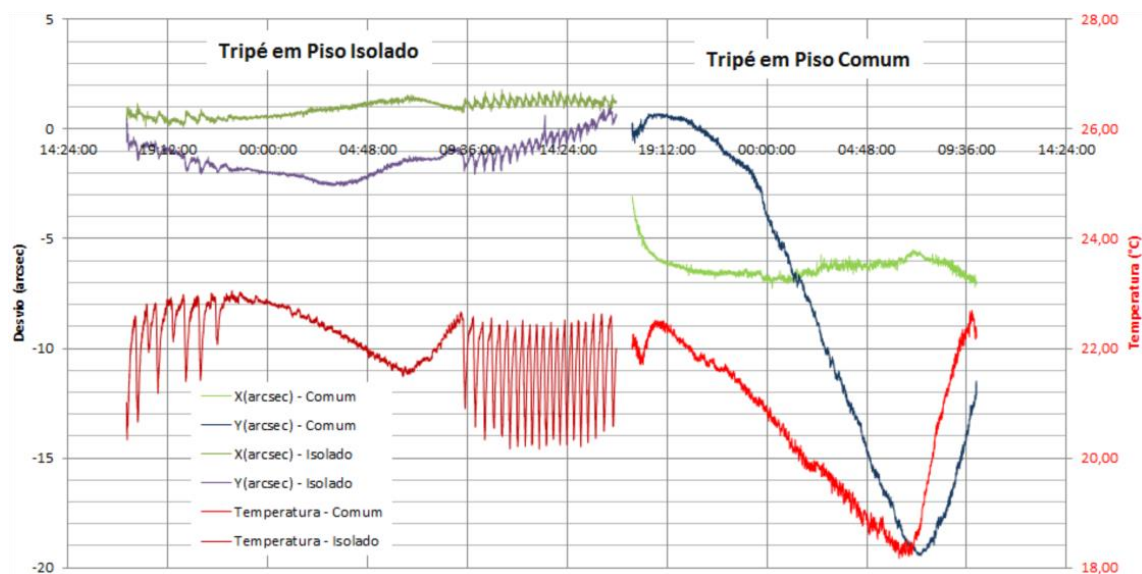


Figura 45 - Estabilidade do autocolimador em sala controlada (esquerda) e sala comum (direita)
Fonte: proposta pelo autor

O experimento foi repetido durante um final de semana e, nesse caso, o Elcomat 3000 não foi fixado a um tripé, mas a um desempenho. Percebeu-se, Figura 46, que após um período de acomodação, o sistema permaneceu estável dentro de tolerâncias mais estreitas e menos dependente de oscilações de temperatura (responsável por dilatações nos tripés que sustentavam espelho e instrumento nos casos anteriores).

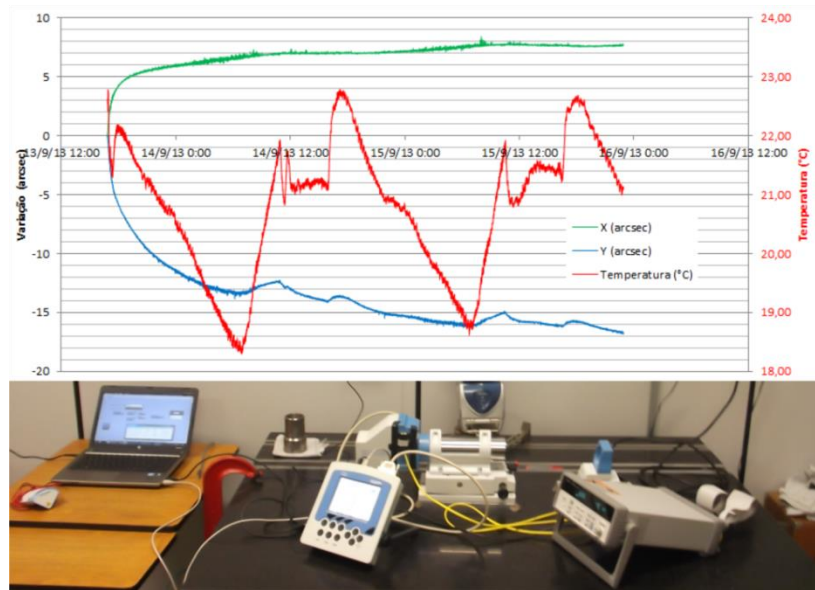


Figura 46 - Análise de estabilidade do autocolimador em sistema isento de tripés
Fonte: proposta pelo autor

Conhecidos os limites da instrumentação, será visto que as oscilações do autocolimador poderão ser desprezadas nas varreduras dos mecanismos M1 a M5, uma vez que os ensaios são de curta duração e os erros do instrumento são de ordem inferior aos dos sistemas.

Os gráficos presentes nas figuras 47 a 57 indicam as leituras efetuadas pelo autocolimador em seus eixos “v” e “h” em função da posição do motor de passo “p”, em μ passos, para os ensaios finais de cada mecanismo.

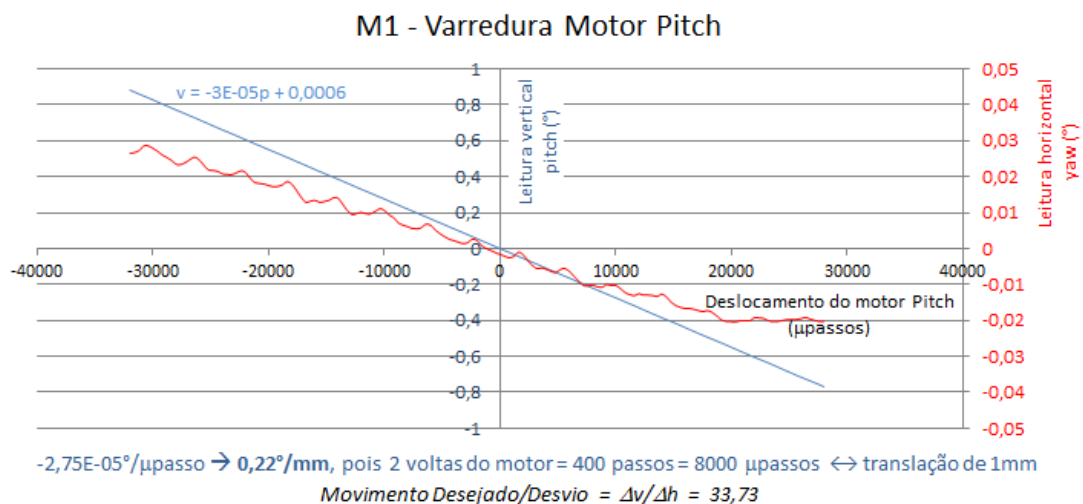


Figura 47 - Resultado da varredura em M1 para atuação do motor *pitch* com comportamento linear (azul) e erro *yaw* senoidal decrescente (vermelho) $\sim 30x$ menor

M1 - Varredura Motor Roll

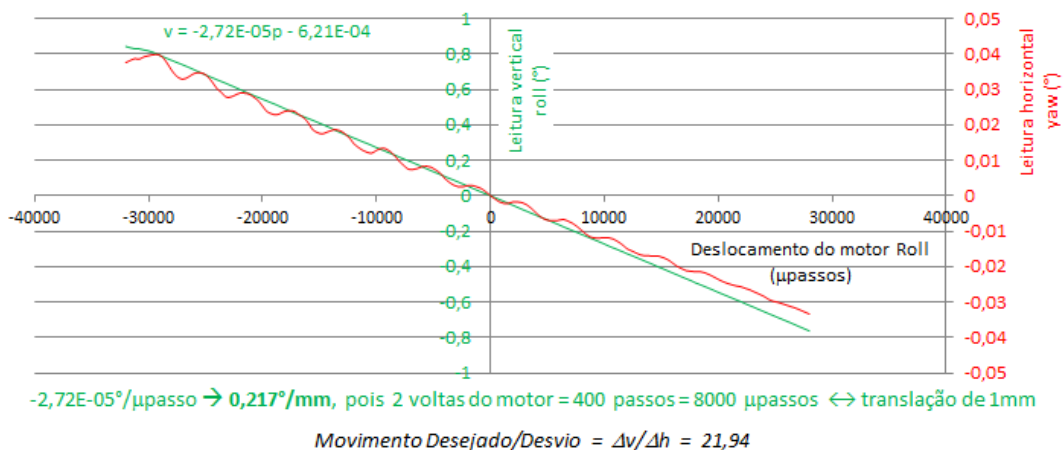


Figura 48 - Resultado da varredura em M1 para atuação do motor *roll* com comportamento linear (verde) e erro *yaw* senoidal decrescente (vermelho) ~20x menor.

M1 - Varredura Motor Yaw

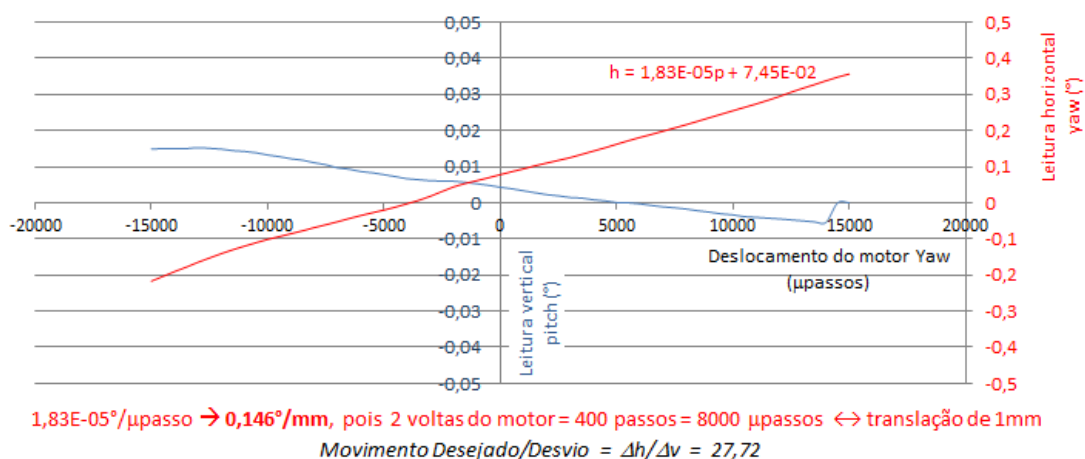


Figura 49 - Resultado da varredura em M1 para atuação do motor *yaw* com comportamento linear (vermelho) e erro *pitch* também linear (azul) ~30x menor

M2 - Varredura Motor Pitch

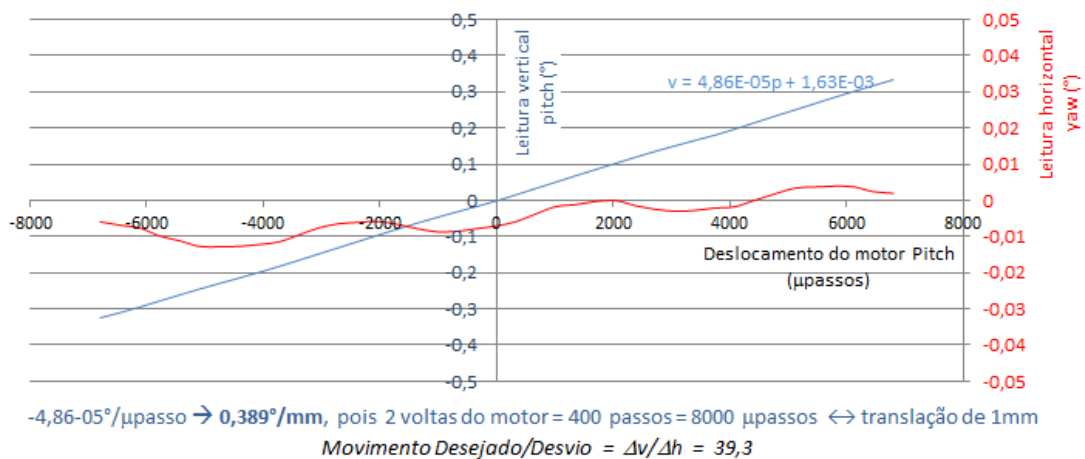


Figura 50 - Resultado da varredura em M2 para atuação do motor *pitch* com comportamento linear (azul) e erro *yaw* senoidal (vermelho) ~40x menor

M2 - Varredura Motor Yaw

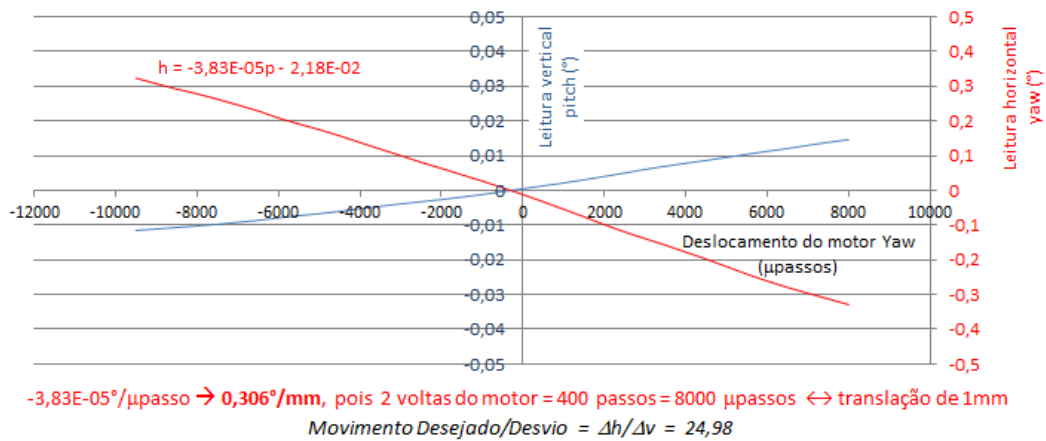


Figura 51 - Resultado da varredura em M2 para atuação do motor *yaw* com comportamento linear (vermelho) e erro *pitch* também linear (azul) ~25x menor

M3 - Varredura Motor Pitch

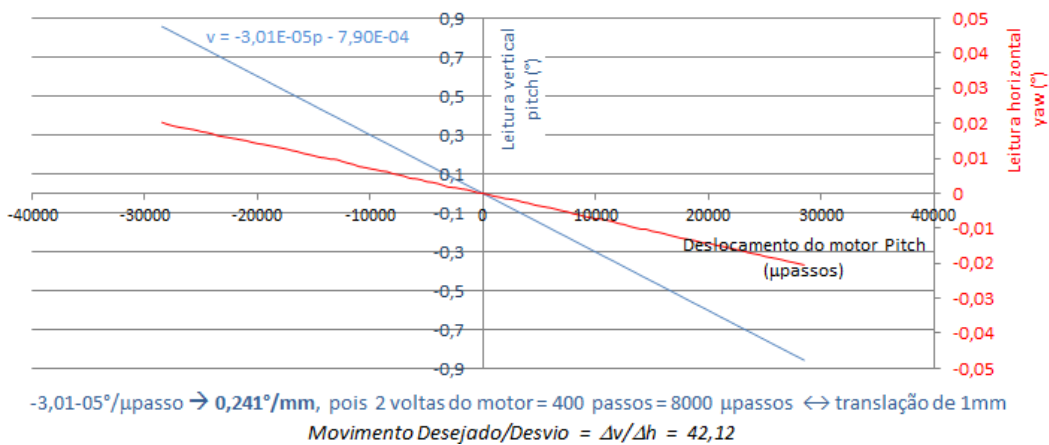


Figura 52 - Resultado da varredura em M3 para atuação do motor *pitch* com comportamento linear (azul) e erro *yaw* também linear (vermelho) ~40x menor

M3 - Varredura Motor Yaw

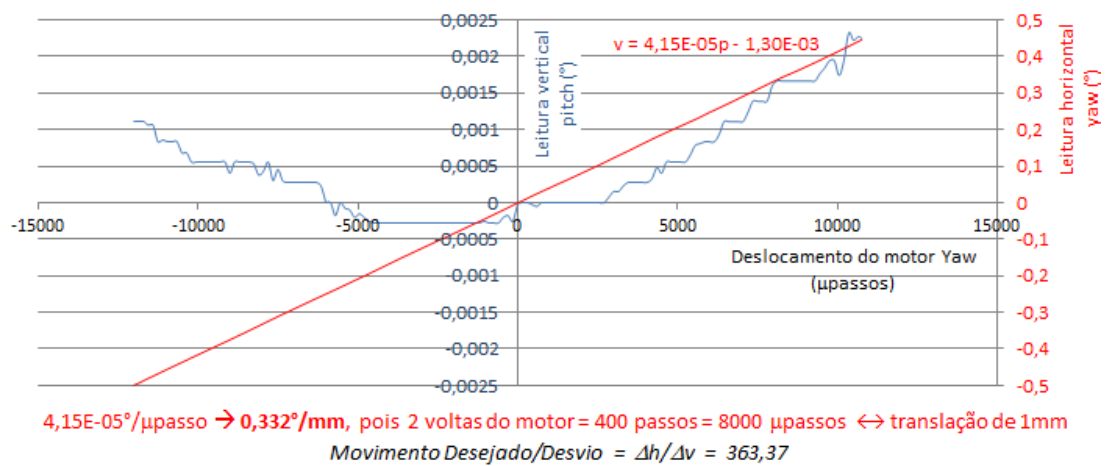


Figura 53 - Resultado da varredura em M3 para atuação do motor *yaw* com comportamento linear (vermelho) e erro *pitch* parabólico (azul) ~360x menor

M4 - Varredura Motor Pitch

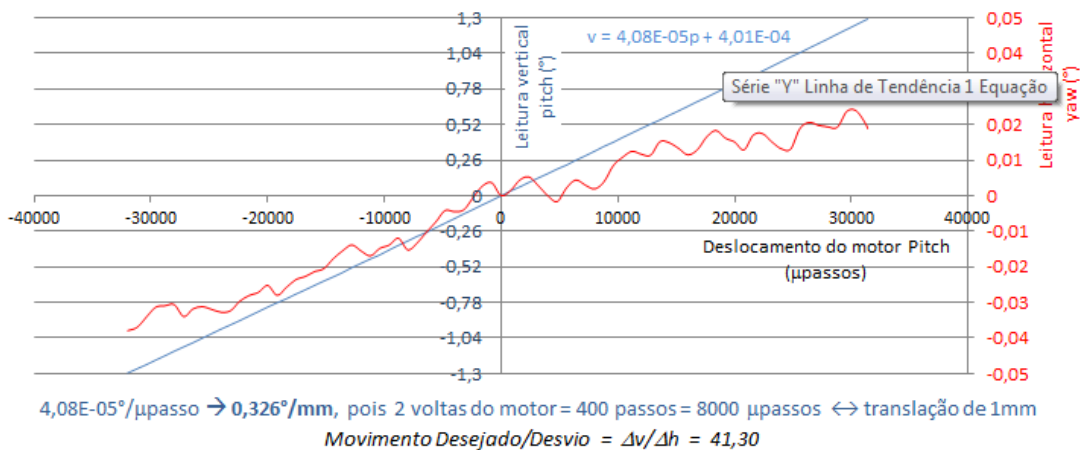


Figura 54 - Resultado da varredura em M4 para atuação do motor *pitch* com comportamento linear (azul) e erro *yaw* senoidal crescente (vermelho) ~40x menor

M4 - Varredura Motor Y/R

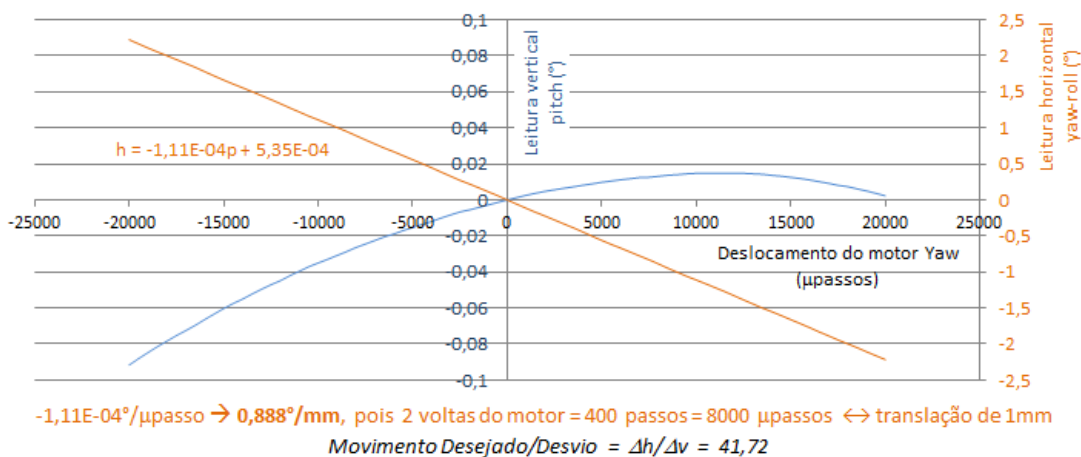


Figura 55 Resultado da varredura em M4 para atuação do motor do movimento que combina *yaw* e *roll* com comportamento linear (alaranjado) e erro *pitch* parabólico (azul) ~40x menor

M5 - Varredura Motor Pitch

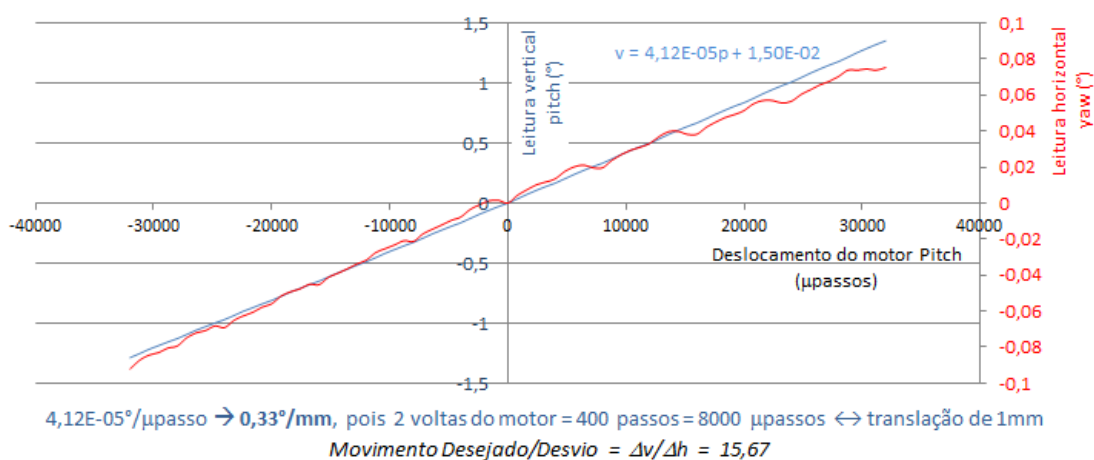


Figura 56 - Resultado da varredura em M5 para atuação do motor *pitch* com comportamento linear (azul) e erro *yaw* senoidal crescente (vermelho) ~15x menor

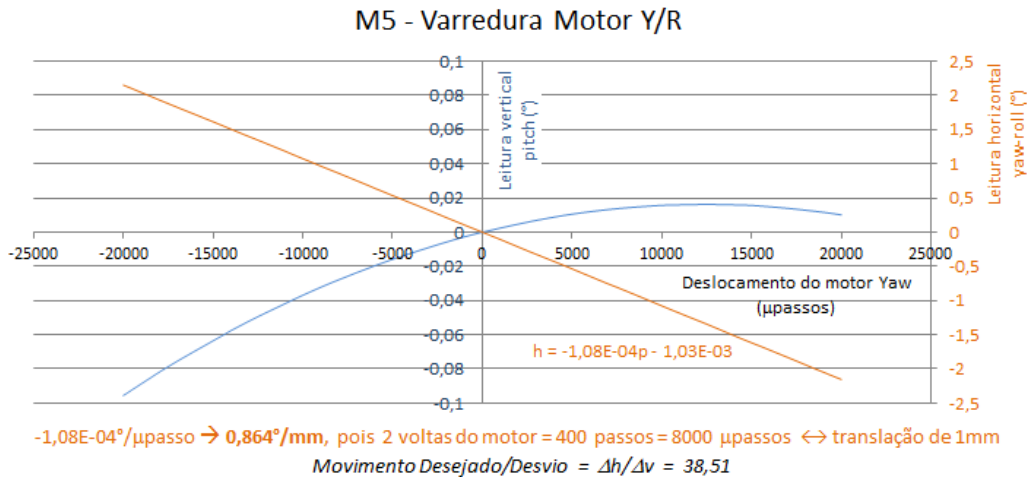


Figura 57 - Resultado da varredura em M5 para atuação do motor do movimento que combina *yaw* e *roll* com comportamento linear (alaranjado) e erro pitch parabólico (azul) ~40x menor

Em relação a essa sequência de gráficos, a primeira constatação é que os erros do instrumento são de ordem inferior aos desvios dos ensaios; logo, podem ser desprezados.

Nas equações presentes nas figuras, “p” indica a posição em micropassos do motor ativo e, “v” e “h”, as leituras efetuadas pelo autocolimador, respectivamente, vertical e horizontal. Os coeficientes angulares nessas relações indicam quantos micropassos são necessários para inclinar o espelho em um grau.

Como esperado, os movimentos desejados apresentaram comportamento linear, os movimentos parasitas foram entre 15 e 360 vezes inferiores e apresentaram, muitas vezes, comportamento senoidal de períodos próximos aos passos dos fusos. Em muitos, devido à baixa amplitude e ao amplo *range* de medição, tais ondulações apenas são visualizadas em gráficos ampliados.

Uma ligeira diferença de inclinação em uma das extremidades da reta da Figura 48 aponta que aquela varredura partiu de um ponto sem se ter efetuado correção de *backlash*.

Gráficos de varreduras intermediárias que indicaram falhas nos mecanismos não serão aqui exibidos. Na maioria delas, eram visualizadas interrupções bruscas seguidas de valores constantes, o que se justifica pela perda de contato entre fuso e eixos ou entre prismas e eixos. Nesses casos, as molas foram esticadas ou trocadas.

Ao se repetir os ensaios e sobrepor os gráficos resultantes, testemunhou-se que os equipamentos apresentavam excelente repetibilidade. A análise da precisão envolve o alcance de uma mesma posição a partir das chaves de final de curso, o que também se mostrou satisfatório.

Além disso, dada a facilidade em se executar esses testes com o aplicativo em LabVIEW, é possível realizar uma caracterização ainda mais completa, a partir de varreduras de um motor para diferentes posições dos demais atuadores.

Esse estudo foi realizado para M2 e a Figura 58 corresponde a um de seus resultados. Para atingi-lo, utilizaram-se sequencialmente as funções “executar varredura” e “reiniciar” no motor *pitch* e a função “incrementar” no motor *yaw*. O comportamento percebido é de manutenção no coeficiente angular das retas, ou seja, independentemente da posição do outro atuador, o movimento de um permanece com a mesma relação graus/passos.

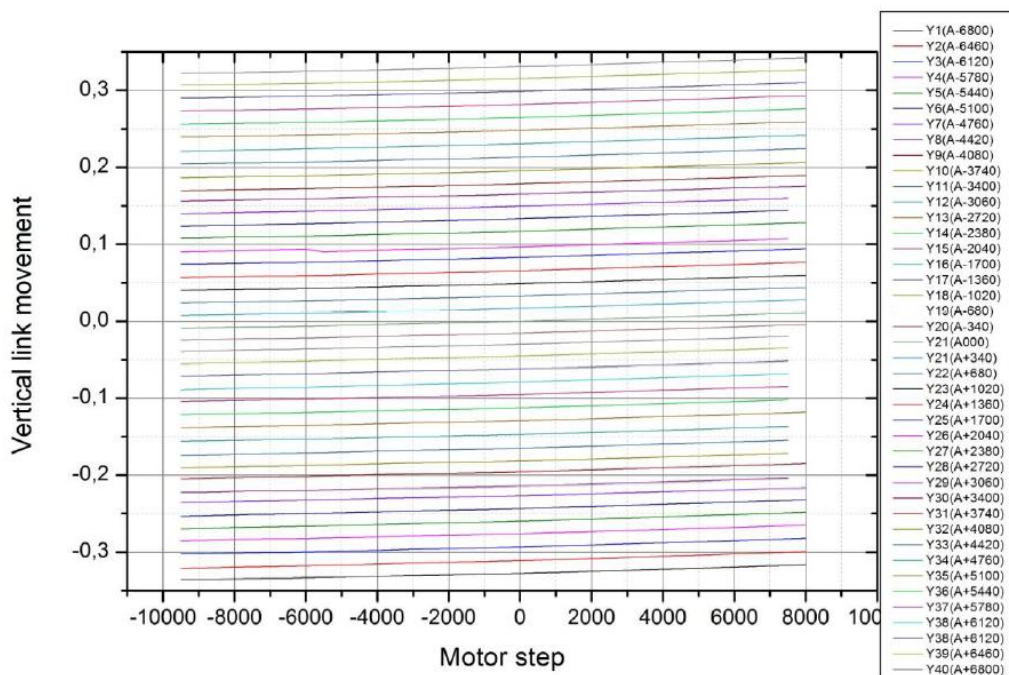


Figura 58 - Leituras "v" durante movimentação do motor *pitch* em 40 diferentes posições do motor *yaw*

Concluídos os ensaios, os mecanismos foram limpos para vácuo e montados com seus respectivos espelhos no interior da câmara óptica, Figura 59.

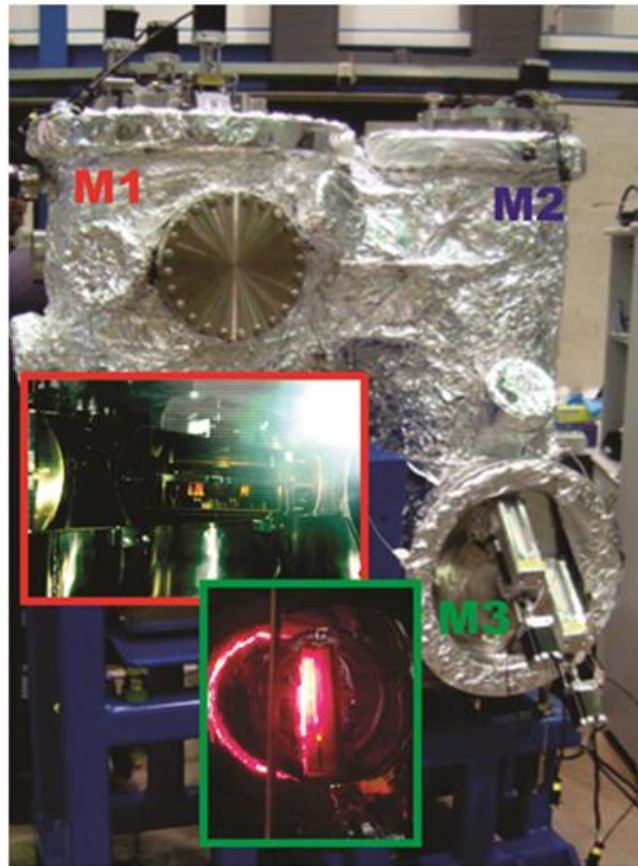


Figura 59 Mecanismos 1 a 3 instantes antes da instalação no anel
Fonte: fotografias de Raul de Oliveira Freitas

4.2 Exemplo de utilização em outra aplicação

A estrutura apresentada na Figura 60 também exemplifica a utilização do programa para caracterização de movimentos em uma estrutura projetada para movimentar corpos de prova em uma linha de luz.

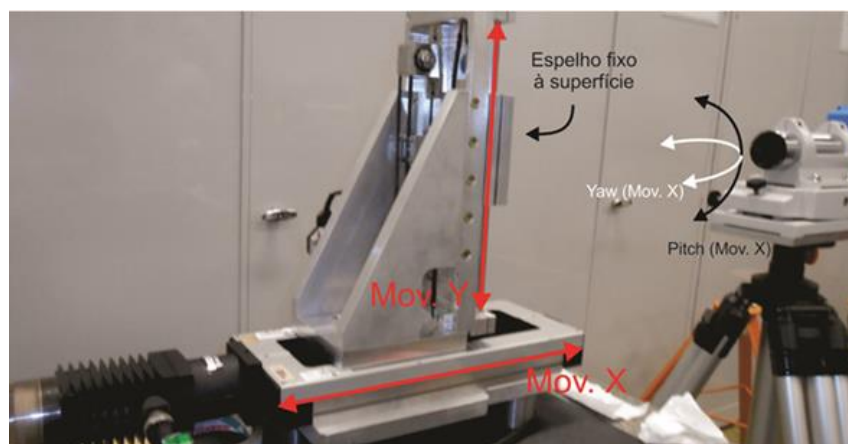


Figura 60 - Varredura para verificação de *Pitch* e *Yaw* em movimento horizontal (X) de bancada para movimentação de amostras
Fonte: proposta pelo autor

Vê-se que o programa não se limita a uma única aplicação. Nesse caso, ambas as leituras do autocolimador referem-se a desvios, pois o movimento desejado é linear, o qual pode ser medido, por exemplo, com um interferômetro.

Com o uso de paquímetro e da função “incrementar”, pôde-se avaliar os cursos X e Y e caracterizar as relações de transmissão e redução do mecanismo. Com a função “Varredura”, pôde-se avaliar *pitch* e *yaw* nos deslocamentos ao longo dos eixos. As varreduras foram reproduzidas algumas vezes e apresentaram boa repetibilidade. A Figura 61 resume os resultados de uma delas.

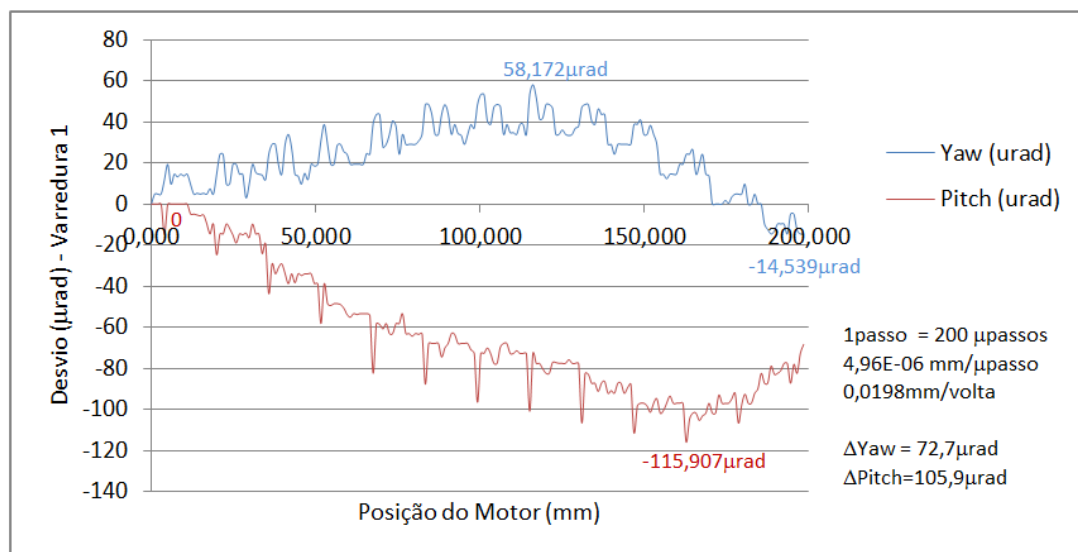


Figura 61 - *Pitch* e *Yaw* para movimento horizontal (X)

Fonte: proposta pelo autor

5. CONSIDERAÇÕES FINAIS

A construção de novas linhas de luz no Centro Nacional de Pesquisa em Energia e Materiais impulsiona a aquisição de instrumentos de metrologia de alta precisão e fomenta a automatização de tarefas relacionadas à caracterização de seus mecanismos.

Muitos desses engenhos, em particular, os suportes dos espelhos da linha de infravermelho, geram pequenos deslocamentos a partir da rotação de motores de passo.

Foi desenvolvido um aplicativo em LabVIEW para correlacionar as posições desses motores aos erros (*pitch* e *yaw*) e deslocamentos reais desses equipamentos, mensurados por um autocolimador.

Através de uma linguagem de alto nível, o trio *Powerpack-LabVIEW-VISA* dispensa o programador do tratamento das combinações binárias para o controle de motores de passo e oferece uma gama de funções que auxiliam na obtenção de dados sobre o sistema em estudo.

O aplicativo fornece rapidamente a seus usuários informações mais precisas que aquelas obtidas por métodos convencionais na tentativa de se caracterizar o comportamento real daquilo que foi projetado.

No estudo de caso apresentado, a partir de repetidos testes, algumas peças foram alteradas até se atingirem condições aceitáveis de desvios e repetibilidade de movimento.

REFERÊNCIAS¹

ATHANI, V. V. **Stepper motors: fundamentals applications and design**. 1. Ed. New Delhi: New Age International (P) Ltd. 2005.

BRITES, F. G. SANTOS, V. P. de A. **Motor de passo**. Formato PDF. Universidade Federal Fluminense Niterói, 2008.

CANADIAN Light Source. **Synchrotron science classroom resources**. Canadian Light Source Outreach 2012. Formato PDF. Disponível em: <<http://www.lightsource.ca/education/pdf/SSCR2013Jul23-Formatted-for-Web.pdf>>. Acesso em 29 Setembro 2013.

DA SILVA, Irineu. **História dos pesos e medidas**. 2. ed. São Carlos: EdUFSCar, 2010

EPSIM 3D, JF Santarelli. **Schéma de principe du synchrotron**. Formato PNG. Color. Disponível em: <http://commons.wikimedia.org/wiki/File:Sch%C3%A9ma_de_principe_du_synchrotron.jpg>. Acesso em: 29 Setembro 2013.

ESHAM, B. D. **The inside of a microswitch**. Fotografia. Color. Junho de 2007. Disponível em: < <http://commons.wikimedia.org/wiki/File:Microswitch.jpg>>. Acesso em 20 outubro 2013.

ESTATCAMP. **Parâmetros característicos de um sistema de medição**. Disponível em: <<http://www.portalação.com.br/1003-4-par%C3%A2metros-caracter%C3%ADsticos-de-um-sistema-de-medi%C3%A7%C3%A3o>>. Acesso em 17 Setembro 2013.

FREITAS, R. DENEKE, C. et al. **Integration of a comercial a-NSOM setup into an IR beamline**. Pôster. Laboratório Nacional de Nanotecnologia. Laboratório Nacional de Luz Síncrotron. CNPEM. Campinas, 2013.

JONES, F. D. **Gear design simplified**. 3rd ed, Industrial Press: New York, 1961.

KORDIK, Jeff. **Comparação de sistemas com motores de passo e servomotores**. Mecatrônica Atual. São Paulo, n. 10, p. 20-26. Junho de 2003.

¹ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023

LIGHTSOURCE. **What is a Light Source.** Disponível em: <<http://www.lightsources.org/what-light-source>>. Acesso em 29 Setembro 2013.

MARTINS, F. **Almanaque: eletrônica, informática e programação.** Normalmente aberto, normalmente fechado. Março, 2008. Disponível em: <<http://www.fcmartins.net/blog/?p=21>>. Acesso em 28 Maio 2013.

MESSIAS, A. R. **Controle de motor de passo através da porta paralela.** Rogercom. Disponível em: <<http://www.rogercom.com/pparalela/IntroMotorPasso.htm>>. Acesso em 1 outubro 2013.

MCCOMB, M. **Introduction to stepper motors.** Part 1: Types of Stepper Motors. 2007. 49 slides. Color. PDF. Microship Technology Incorporated. Disponível em: <http://www.microchip.com/stellent/groups/SiteComm_sg/documents/DeviceDoc/en543047.pdf> Acesso em 19 Maio 2013.

MÖLLER-WEDEL OPTICAL. **Instruction manual Elcomat 3000.** Electronic Autocollimator, 2-axis. Formato PDF. Rosengarten. Dezembro de 2008.

NATIONAL INSTRUMENTS. **LabVIEW™ core 1:** manual do curso. Software do curso versão 2009. Novembro 2009

NATIONAL INSTRUMENTS. **LabVIEW™ core 2:** manual do curso. Software do curso versão 2009. Novembro 2009

NATIONAL INSTRUMENTS. **What is VISA?** 2012. Disponível em: <<http://www.ni.com/visa/>>. Acesso em 31 Maio 2013.

NIKON.**Autocolimador.** Centro de Informações Nikon Instruments. 2013. Disponível em: <http://www.nikoninstruments.com/pr_BR/Centro-de-Info/mao/Autocolimador>. Acesso em: 20 outubro 2013.

RENISHAW. **A interferometria explicada.** Disponível em: <<http://www.renishaw.com.br/pt/a-interferometria-explicada--7854>>. Acesso em 12 Outubro 2013.

RICCI, M. C. **Motores elétricos e acionamento em mecânica fina.** Notas de Aula. Universidade São Judas Tadeu. São Paulo: 2005

SOUZA, M. A. A. **Implementação de sistema controlador de motor de passo em malha fechada utilizando tecnologia baseada em controlador digital de sinais.** Trabalho de Conclusão de Curso (TCC). Escola de Engenharia de São Carlos. Universidade de São Paulo. São Carlos: 2007.

TRIANGLE. **Precision electronic autocollimator.** 2012. Disponível em: <<http://www.trioptics.com/triangle/description.php>>. Acesso em 20 outubro 2013.