

Desenvolvimento de Mesa de Coordenadas Didática a Partir de Resíduos Eletroeletrônicos

Carlos Alberto de Souza Filho*
Ademar Virgolino da Silva Netto*

* Departamento de Engenharia Elétrica, Universidade Federal da Paraíba, PB, (e-mail: calberto@cear.ufpb.br, ademar@cear.ufpb.br).

Abstract: Each year the amount of electronic waste generated increases and its users are obliged to dispose of them correctly, making the treatment problematic. Based on the initiative of the lectures of the controle and automation laboratory at de Federal Universit of Paraíba, several educational kits are being developed to be used in the disciplines of the area. The main focus of the lectures is to be able to develop experiments of situations experienced in the daily life of an engineer, awakening the critical and analytical thoughts of the students. Another objective is to have low cost, using components and waste from the laboratory itself or collected by extesion project TREE (Electronic Waste Treatment) at UFPB. In this article we will present the didactic kit of a coordinate table (x and y) and a set of experiments that have already been applied in the classroom with the kit, such as developing an interface for system integration, control and positioning, defining trajectory and use communication protocols. It has been verified in the classroom that solutions of this nature improve students motivation and performance, and it can help to reduce students dropout and retention in the engineering course.

Resumo: A cada ano o montante de resíduos eletroeletrônicos gerados aumenta e seus usuários têm a obrigação de fazer a destinação correta dos mesmos, sendo uma problemática o tratamento destes resíduos. A partir da iniciativa dos professores do laboratório de controle e automação da universidade federal da paraíba estão sendo desenvolvidos diversos kit didáticos para serem utilizados nas disciplinas da área. O foco principal dos professores é desenvolver experimentos de situações vividas no cotidiano de um engenheiro, despertando o olhar crítico e analítico dos alunos. Outro objetivo é que tenha baixo custo, utilizando componentes e resíduos do próprio laboratório ou arrecadados pelo projeto de extensão da UFPB - TREE (Tratamento de Resíduos Eletroeletrônicos). Neste artigo será apresentado o kit didático de uma mesa de coordenadas (x e y) e um conjunto de experimentos que já foram aplicados em sala de aula com o kit, como desenvolvimento de interface para integração, controle e posicionamento do sistema, definição de trajetória e uso de protocolos de comunicação. Tem-se verificado em sala de aula que soluções desta natureza melhoram a motivação e rendimento dos alunos, e serve de medida para reduzir a evasão e retenção dos alunos no curso de engenharia.

Keywords: Waste; Electronics; Automation; Reuse; Educational kit; Low Cost.

Palavras-chaves: Resíduos; Eletroeletrônicos; Automação; Reaproveitamento; Kit didático; Baixo Custo.

1. INTRODUÇÃO

O crescimento na produção de resíduos sólidos nos aglomerados urbanos instala um grave problema socioambiental, resultado dos padrões atuais de produção e consumo, provocando impactos ambientais e de saúde pública que precisam ser enfrentados. Ainda assim, reduzir os impactos gerados pela produção dos resíduos sólidos tornou-se meta planetária para que se possa atingir um “modelo de desenvolvimento socialmente benéfico, ambientalmente sustentável e economicamente sustentado” (Jacobi and Bese, 2011).

Entre os resíduos sólidos gerados um tipo específico merece um foco especial pela sua característica de periculosidade

ao meio ambiente, que são os Resíduos de Equipamentos Elétricos e Eletrônicos, denominados Resíduos Eletroeletrônicos (REEs), Resíduos Tecnológicos, ou simplesmente lixo eletrônico. Os REEs mais comuns são os aparelhos tecnológicos como computadores, *tablets*, impressoras, celulares, TVs, notebooks e diversos outros equipamentos (Natume and SantAnna, 2011), que sofrem com a obsolescência programada e perceptiva, fazendo que os usuários sintam a necessidade de sempre estar consumindo novos produtos aumentando a quantidade de resíduos produzidos.

A indústria eletrônica é uma das maiores e das que mais crescem no mundo. Segundo a ONU estima-se que é produzido cerca de 50 milhões de toneladas de REEs por ano

no mundo (ONUBrasil, 2019). De acordo com a política nacional dos resíduos sólidos (PNRS), a lei nº 12.305 de 02 de agosto de 2010 indica que a responsabilidade sobre estes resíduos é compartilhada, sendo obrigação do usuário fazer a destinação correta dos mesmos. O descarte inadequado de REEs é um problema de escala mundial.

O uso do REEs no LAC (Laboratório de Automação e Controle da UFPB) surgiu a partir de um projeto de extensão denominado tratamento de resíduos eletroeletrônicos - TREE (mais detalhes nas redes sociais, instagram e facebook: @treeufpb) que coletaram equipamentos juntos aos alunos do curso visando seu reaproveitamento nos laboratórios, com reposição de componentes e fornecimento de matéria-prima para o desenvolvimento de projetos nas disciplinas do curso. Então os professores associados ao laboratório tiveram a iniciativa de refinar o uso dos equipamentos arrecadados visando a sustentabilidade e soluções verdes.

A cada dia a aquisição de novos equipamentos pelos laboratórios da universidade é dificultada seja pela burocracia para a sua aquisição, pela falta de recursos, ou pelo longo tempo para sua aquisição. Em decorrência de toda esta problemática os pesquisadores do LAC da UFPB, vem buscando soluções complementares para manter a excelência das suas atividades e atender as demandas do seu alunato. Recentemente foram desenvolvidos pela equipe do LAC, kits de baixo custo para o ensino de controle (Cunha et al., 2018) e de instrumentação industrial (Cardoso and Souza Filho, 2019). Além disto, o laboratório busca desenvolver medidas de sustentabilidade buscando contribuir com a redução da produção de REE. A partir disto, surgiu a proposta apresentada neste trabalho, que tem por objetivo o desenvolvimento de um kit didático (mesa de coordenadas) de baixo custo para as disciplinas da área de automação, como sistemas de aquisição de dados, sistemas de automação industrial e instrumentação industrial, o kit já vem sendo utilizado há quatro semestres e a cada semestre vai sendo aperfeiçoado para atender às demandas das disciplinas.

A temática já vem sendo utilizada em outros trabalhos como o uso dos resíduos para produção de jogos e materiais didáticos, voltados para o ensino da matemática e os conceitos básicos de informática (Paiva et al., 2016), para o ensino de química (Friedrich, 2014) e (Abreu, 2014), para a construção de pequenos protótipos para o ensino de formação básica a partir da robótica (Lima et al., 2017), além do seu uso em outras áreas como design (Damaceno, 2014). Para a automação e controle o grupo já vem desenvolvendo trabalhos desta natureza como desenvolvimento de kit didático sustentável para ensino de automação industrial (Vaz et al., 2018), porém, este atende a um conjunto maior de disciplinas e experimentos.

2. DESCRIÇÃO DA PLATAFORMA

Dispositivos que utilizam controle numérico computadorizado (CNC) são utilizados em diversas aplicações industriais. Pode-se encontrar dentre outras aplicações que utilizam CNC: tornos, fresadoras, Eletroerosão a Fio e por Penetração (EDM), retificadora, cortadoras a *laser*, fresadoras de placas de circuito. Recentemente, tem se tornado popular o uso de impressoras 3D para confecção de peças

por profissionais ou hobbistas. Alguns destes dispositivos CNC utilizam motores de passo que movimentam sistemas de coordenadas cartesianas. A plataforma proposta foi desenvolvida baseada neste princípio, sendo construída a partir de um *scanner* de mesa e uma impressora multifuncional, que estavam inutilizados e que seriam descartados como lixo. A plataforma desenvolvida está ilustrada na Figura 1.

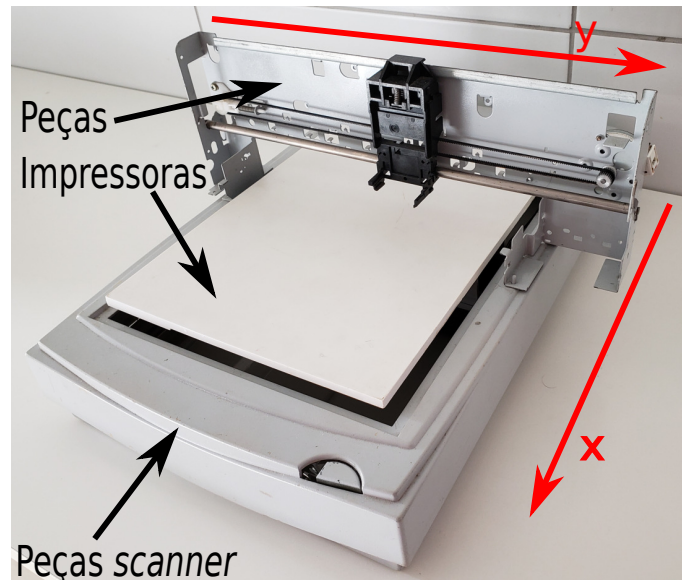


Figura 1. Estrutura Mecânica da Mesa de Coordenadas

O *scanner* de mesa TCE MK300s foi desmontado retirando-se o vidro da parte superior e placas eletrônicas existentes. Foi mantido apenas o motor de passo, unidade de leitura e estrutura mecânica para deslocamento da unidade de leitura. Na unidade de leitura foi mantida apenas a estrutura de plástico existente, fixando-se nessa estrutura duas cantoneiras de alumínio em L para sustentação do tampo da mesa. Para identificar a posição inicial e evitar que a mesa seja deslocada além do permitido, foram adicionados sensores de proximidade ópticos. Na posição de maior e menor deslocamento da unidade de leitura foram colocadas chaves ópticas retiradas de impressoras. Na unidade de leitura foi acoplada uma peça de plástico, recortada da estrutura da impressora, em formato adequado para passar através da chave óptica. O objetivo desta peça, em conjunto com a chave óptica, é possibilitar detectar quando a mesa atinge as posições de máximo e mínimo deslocamento. O motor de passo existente no *scanner* (M_1) é do tipo unipolar. Por meio do acionamento deste motor, pode-se deslocar a mesa de coordenadas na direção indicada pelo eixo x da Figura 1. Na Figura 2 são ilustradas as partes que compõem a plataforma isoladamente para melhor visualização.

Para realizar o movimento na direção y da mesa de coordenadas, utilizou-se a estrutura de uma impressora que realiza o deslocamento da unidade de impressão. Essa estrutura foi fixada na parte superior do *scanner* na posição central referente ao eixo x. A estrutura é composta por uma armação metálica onde são fixadas um eixo cilíndrico, correia dentada, unidade de impressão (UI) e um motor de passos bipolar (M_2). No eixo do motor de passo existe uma polia dentada que movimenta a correia. Como a unidade de impressão está acoplada à correia, ao

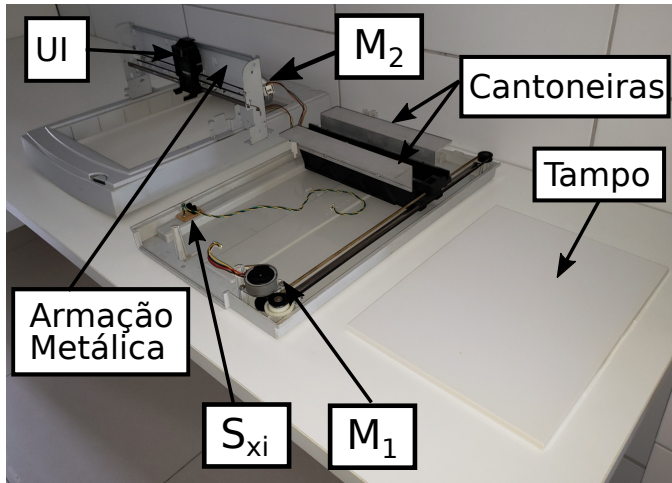


Figura 2. Estrutura Mecânica da Mesa de Coordenadas

acionar o motor de passos M_2 movimenta-se a unidade de impressão na direção do eixo y .

A mesa de coordenada possui várias aplicações, o que muda de uma solução para outra é apenas a ferramenta que é acoplada. Por exemplo, ao acoplar um laser de corte é possível utilizar como uma *router* CNC fazendo cortes em acrílicos, MDF e até papel. Outro exemplo de aplicação é prender uma caneta que pode ser utilizada para escrever, fazer imagens vetorizadas entre outros. Outra aplicação é com o uso de uma broca ou fresa servindo como fresadora ou *router* CNC para fazer, por exemplo, a confecção de placas de circuitos impressos (PCI) ou esculturas em madeira.

Ao se projetar a plataforma didática optou-se por duas formas de utilização. Na primeira opção (**modo I**) a plataforma é operada remotamente, enviando-se comandos por um dispositivo computacional. Logo, deve existir um sistema que seja capaz de realizar a comunicação com o dispositivo computacional e controlar os motores existentes. Na segunda opção (**modo E**) a plataforma é totalmente controlada por uma unidade externa. Nesta opção o aluno(a) tem que desenvolver os mecanismos necessários para controlar a posição da mesa de coordenadas. Através da interpretação dos sinais dos sensores e acionamento dos motores existentes, o aluno pode usar um microcontrolador, FPGA, plataformas *single board*, placas de aquisição ou outro dispositivo similar para controlar a mesa de coordenadas.

Para permitir os dois modos de operação, foi desenvolvido o *hardware* e *firmware* necessários ao controle da plataforma proposta.

2.1 Hardware

O desenvolvimento do *hardware* teve como objetivo um custo reduzido utilizando componentes presentes no laboratório. O *hardware* pode ser representado pelo diagrama de blocos ilustrado na Figura 3.

Para identificar as posições iniciais e finais do deslocamento de cada eixo foram utilizadas chaves ópticas retirados de impressoras (S_{ni} e S_{nf} onde $n = x$ e y).

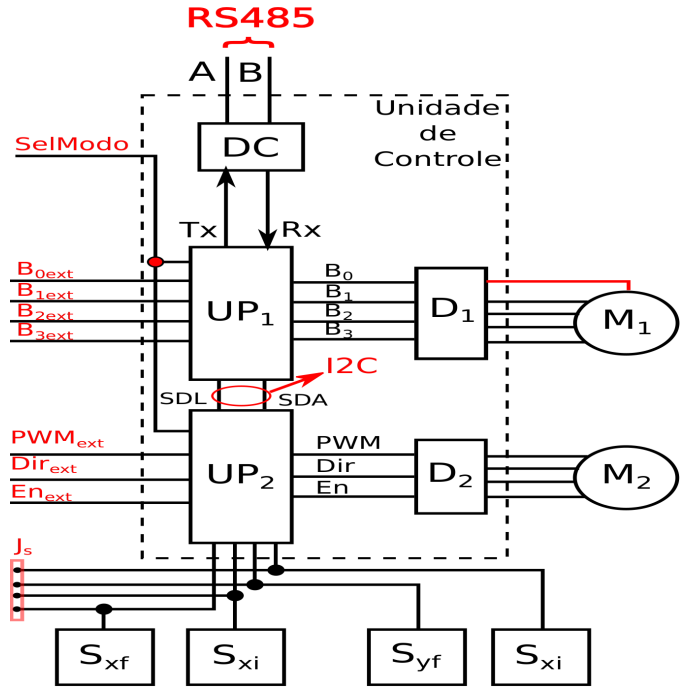


Figura 3. Diagrama de Blocos do Hardware desenvolvido

O deslocamento dos eixos x e y é realizado por motores de passo de diferentes princípios. No eixo x existe um motor unipolar e no eixo y um bipolar. Apesar de existir a possibilidade de transformar um motor unipolar em bipolar, o que poderia aumentar a precisão no deslocamento, o acionamento dos motores foi mantido na forma original. Isto permite ao aluno aprender diferentes princípios de acionamento e utilizar recursos distintos do *hardware* de controle. Para acionar o motor (M_1) foi utilizado como *driver* (D_1) o circuito integrado ULN2300. O motor M_2 foi acionado por meio do *driver* A4988 (D_2).

Os sinais de controle necessários para acionar os motores são gerados por duas unidades de processamento (UP_1 e UP_2) diferentes. A escolha por duas unidades distintas foi realizada em função do custo reduzido do microcontrolador escolhido, o MSP430G2553, que já estava disponível no laboratório. Apesar da quantidade de pinos disponíveis ser insuficiente para a necessidade, o custo de aquisição de duas unidades era inferior a outro microcontrolador com capacidade suficiente. A unidade UP_1 é responsável por realizar a comunicação com dispositivos externos através de uma porta serial, interpretando os comandos enviados e gerando os sinais necessários para o acionamento do motor M_1 (B_0 , B_1 , B_2 e B_3). Para converter os sinais da porta serial de UP_1 para o barramento RS485, foi utilizado o transceptor MAX485. A comunicação por RS485 foi escolhida por ser um padrão que permite conexão de dispositivos em rede, fazendo parte da camada física de protocolos industriais já bastante consolidados como MODBUS e PROFIBUS. Isto permitirá a conexão da mesa desenvolvida com os CLPs existentes no LAC. O microcontrolador UP_1 se comunica com UP_2 por meio da interface I2c.

A unidade UP_2 gera os sinais necessários para controlar o motor M_2 . O deslocamento de M_2 é controlado por um sinal PWM onde, para cada pulso, é realizado um desloca-

mento de um passo do motor. O sentido de giro do motor é controlado através de um pino de saída de UP_2 (Dir). O valor do pino de saída (En) determina a habilitação de D_2 . Em cada microcontrolador foi utilizado um pino de entrada para definir o modo de operação ($SelModo$). Quando $SelModo$ é ativado a plataforma opera através de sinais enviados de um dispositivo externo (B_{0ext} - B_{3ext} , PWM_{ext} , Dir_{ext} e En_{ext}). Os sinais dos sensores de proximidade estão disponíveis ao dispositivo externo por meio do conector J_s .

2.2 Firmware

Para controlar o *hardware* da mesa de coordenadas foi implementado o *firmware* representado pelo diagrama de estados ilustrado na Figura 4. Após energizado o sistema entra no modo de operação (m) Interno (I) ($m = i$).

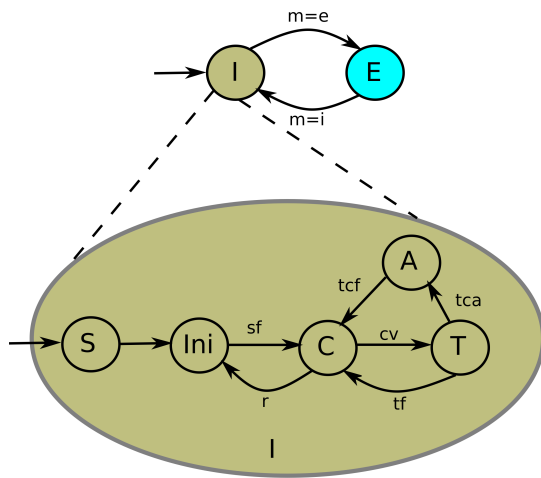


Figura 4. Diagrama de Estados do *firmware* desenvolvido

Neste modo o controle da mesa de coordenadas é realizado pelo *hardware* da própria plataforma. Inicialmente é realizada a configuração de todos os periféricos do microcontrolador utilizados no estado (S). Em seguida o sistema entra no estado de Inicialização (Ini), onde é executada a rotina para deslocamento dos motores até encontrar a posição de início indicadas pelos sensores (S_{xi} e S_{yi}). Quando a inicialização é finalizada (sf) o sistema entra no estado de controle. Nesse estado as UP geram os sinais de controle dos motores de forma a posicionar os eixos até atingir a posição desejada ($PosDes$).

Se for recebido um comando válido enviado pelo barramento RS485 o sistema entra no estado de execução de tarefas (T). O *firmware* foi idealizado para aceitar comandos enviados no protocolo MODBUS. Para facilitar a execução dos experimentos iniciais, também são aceitos comandos simples formados com caracteres ASCII.

A mudança da posição desejada pode ser realizada por meio dos comandos $Xvalor$ (ex: $X100$) ou $Yvalor$, em que a variável 'valor' representa o valor da posição em número de passos. Após a tarefa ser concluída o sistema retorna ao estado de controle. Foi criada uma tarefa de ajuste e calibração da mesa (A), onde os eixos são deslocados desde a posição inicial até a posição final retornado em seguida para posição inicial. Esta tarefa foi estabelecida para identificar o número de passos necessários ao deslocamento

total da mesa, para calibrar a quantidade necessária de passos e realizar o ajuste em relação ao valor da posição em (mm). Quando a tarefa de controle e ajuste é finalizada (tcf) o sistema retorna ao estado de controle. Se for recebido um comando de *reset* (r) o sistema retorna ao estado de inicialização, para realizar novamente o deslocamento da mesa à posição inicial.

No estado de operação E ($m = e$) o *firmware* faz apenas a leitura dos valores enviados pelo microcontrolador externo, reproduzindo os sinais nos pinos correspondentes para acionar os motores M_1 e M_2 . Um dos pinos da UP_1 , conectado a uma chave ($SelModo$), determina a mudança do modo de operação através de uma rotina de interrupção. Quando $SelModo = 1$ o modo de operação E ($m = e$) é selecionado. Quando $SelModo = 0$ o modo de operação I ($m = i$) é selecionado.

2.3 Análise de Custos

Foi realizada uma análise do custo total dos materiais necessários para confecção da plataforma. Na análise realizada, foram utilizados valores médios encontrados em lojas brasileiras. O MSP430G2553 foi cotado diretamente na loja *online* da *Texas Instruments*. Estes valores são referentes a cotação realizada no dia 19/06/2020, considerado-se o preço de aquisição de apenas uma unidade. Os circuitos integrados utilizados podem ter um custo consideravelmente reduzido quando comprado em quantidades mais elevadas. Isto reduziria ainda mais o custo total da plataforma. Como pode-se observar na Tabela 1, o custo total dos componentes foi de R\$51,00, o que é bem inferior ao valor de uma placa de aquisição de dados ou de um controlador lógico programável.

Tabela 1. Relação e custo dos materiais utilizados

Componente	Qtd	Valor Unitário	Valor Total
MSP430G2553	2	R\$10,00	R\$20,00
A4988	1	R\$15,00	R\$15,00
ULN2003	1	R\$2,00	R\$2,00
MAX485	1	R\$4,00	R\$4,00
Placa de circuito	1	R\$10,00	R\$10,00
Total			R\$51,00

3. EXPERIMENTOS REALIZADOS

A estrutura desenvolvida é bastante versátil, podendo ser utilizada para ajudar a assimilação de conteúdos de diversas disciplinas. É possível realizar experimentos básicos destinados a iniciantes e até experimentos avançados (alto grau de dificuldade). Desde o início do desenvolvimento da plataforma foram realizados experimentos em diversas aplicações, das quais foram elencados neste artigo alguns exemplos.

3.1 Desenvolvimento de Interface com Usuário

Com a plataforma configurada no modo de operação com microcontrolador interno (modo I), pode-se utilizá-la para auxiliar o aluno no desenvolvimento de interfaces gráficas ou sistemas supervisórios. Desta forma, o aluno pode por em prática conhecimentos adquiridos em disciplinas de

programação, de *softwares* supervisórios em disciplinas de sistemas de automação ou interface homem máquina, de protocolos ou interfaces de comunicação na disciplina de instrumentação industrial.

Neste experimento é solicitado ao aluno que desenvolva uma interface capaz de enviar o valor da posição desejada nos dois eixos, além de ilustrar a evolução temporal do valor da posição dos eixos através de gráficos. Na Figura 5 é ilustrado uma interface desenvolvida no *software* Labview da National Instruments.

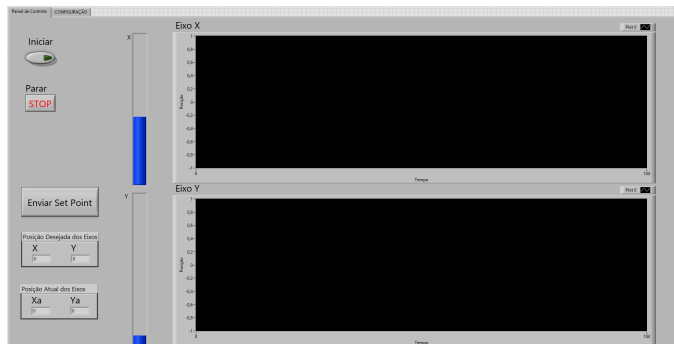


Figura 5. Interface Gráfica para controle e monitoramento da plataforma

Na interface desenvolvida existem controles para escolher a posição desejada nos dois eixos. O valor da posição desejada é transmitido através da interface serial para a plataforma assim que é pressionado o botão enviar.

Além de habilidades necessárias para desenvolvimento da parte gráfica, o aluno precisa ter conhecimentos sobre o protocolo de comunicação existente para enviar o comando de maneira correta até a plataforma. No exemplo ilustrado, o experimento foi direcionado para alunos de instrumentação industrial, em que o foco era a assimilação de interfaces de comunicação seriais (RS485) e de sistemas supervisórios.

Entretanto, este experimento pode ser utilizado para validar conteúdos de programação básicos. Um aluno do segundo período que está cursando uma disciplina de programação, pode ficar responsável em implementar apenas o *software* da interface em qualquer que seja a linguagem.

É possível também utilizar um microcontrolador para enviar os comandos para plataforma. Neste cenário, o aluno pode desenvolver um servidor *Web* no próprio microcontrolador para enviar e receber os valores da plataforma acessando esta página remotamente em um navegador *Web*. Outra possibilidade é utilizar o conceito de Internet das coisas (IOT), onde o aluno desenvolve no microcontrolador um *firmware* que realize a comunicação com um servidor na nuvem. Através deste servidor seria possível monitorar e controlar o valor da posição dos eixos.

3.2 Padrões, Interfaces e Rede de Comunicação

Na disciplina de sistemas de aquisição de dados e instrumentação industrial são abordados conteúdos de interfaces de comunicações seriais RS232 e RS485. O aluno precisa compreender os padrões existentes, identificando entre outros conceitos: as vantagens e desvantagens da

forma de codificação de *bit* (nível de tensão absoluto ou diferencial), interferência do ruído, formato na transmissão do sinal (*bit* de início, dados, paridade e parada). Além disso, o padrão RS485 permite a ligação de dispositivos em rede, fazendo parte da interface dos protocolos industriais bastante utilizados como *MODBUS* e *Profibus*.

Esses conteúdos são de difícil assimilação se não forem acompanhados de experimentos práticos. Para facilitar a aprendizagem destes conteúdos pode-se utilizar a plataforma desenvolvida, com o experimento representado pelo diagrama ilustrado na Figura 6.

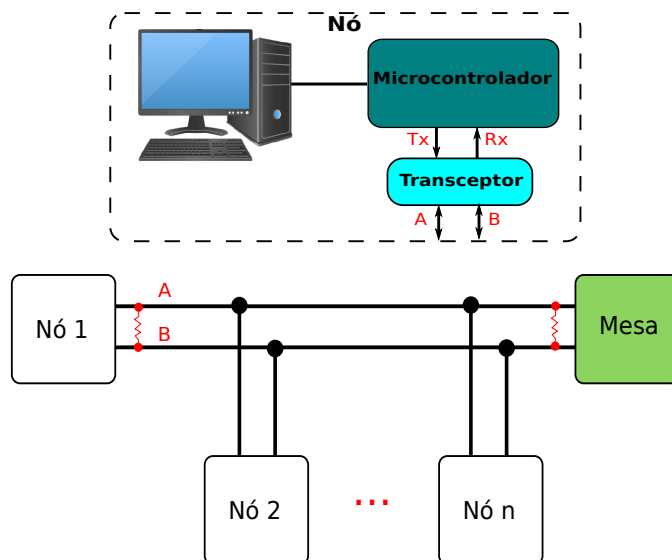


Figura 6. Estrutura da Rede de Comunicação RS485

Durante este experimento os alunos montaram a estrutura de dispositivos conectados em uma rede RS485, com objetivo de fixar conteúdos de interfaces, padrões e redes de comunicação. Cada nó da rede é composto por um computador pessoal, um microcontrolador e um transceptor para RS485. O microcontrolador utilizado é o TM4C1294NCPDT contido na placa de desenvolvimento TIVA 1294 da Texas Instruments. O transceptor utilizado é o MAX485 que é conectado a uma UART (*Universal Asynchronous Receiver/Transmitter*) da Tiva, com objetivo de converter os sinais da UART para o padrão RS485.

Os alunos são divididos em dupla e inicialmente realizam a conexão dos elementos que constituem um nó em *protoboard*. Em seguida, realizam o teste do nó programando o microcontrolador para enviar um caractere pela UART e verificando os sinais elétricos na saída da UART e do transceptor. O microcontrolador se comunica com o PC através da UART0, sendo possível enviar e receber dados através de qualquer terminal serial no PC. O programa desenvolvido deve então realizar a ponte entre a UART0 que está conectada ao PC e a UARTn escolhida para se comunicar com o microcontrolador. Os dados recebidos pela UART0 são enviados para UARTn e os recebidos pela UARTn enviados para UART0. Desta forma, ao digitar um caractere pelo terminal do PC ele é enviado pelo barramento RS485. Os dados recebidos transmitidos pelo barramento RS485 são então monitorados pelo terminal do PC. Nesta etapa, o aluno já utiliza os conhecimentos

adquiridos na programação da interface serial do microcontrolador, assim como dos padrões de comunicação.

Quando todos os nós estão funcionando, eles são conectados ao barramento composto por um par trançado instalado nas bancadas do laboratório. A plataforma desenvolvida é conectada em uma das extremidades do barramento devendo estar configurada no modo I. Neste momento, apenas um dos nós é configurado para transmissão por meio da habilitação do pino correspondente no MAX485 (pino DE). Em seguida, o nó transmissor deve enviar o comando para deslocar a posição da mesa em um dos eixos (ex: para posição 100 do eixo X enviar X100). Os demais nós devem ser capazes de receber este comando e monitorar no terminal do PC. Se tudo estiver correto o aluno verifica a mudança do eixo da mesa para a posição desejada. Neste momento é possível demonstrar, com um auxílio do osciloscópio, a importância da transmissão de sinais complementares no padrão RS485. O aluno pode visualizar que o sinal individual (A ou B) pode ser bastante ruidoso, enquanto que a diferença entre eles é capaz de minimizar consideravelmente este ruído.

3.3 Acionamento de Motores de Passo

Em disciplinas de sistemas de aquisição de dados ou microcontroladores, o aluno precisa entender e saber programar os periféricos do microcontrolador utilizado. A plataforma desenvolvida pode auxiliar bastante no entendimento destes periféricos. Pode-se configurar a plataforma no modo E e utilizar qualquer microcontrolador externo para controlá-la. O acionamento dos motores de passo que movimentam cada eixo é um exemplo de aplicação para utilização de microcontroladores.

Pode-se utilizar conceitos básicos, como o uso de pinos de entrada e saída GPIO (*General Purpose Input/Output*), para controlar a posição do eixo x acionando o motor de passo M_1 . Como o motor M_1 é um motor de passo unipolar, pode-se utilizar a sequência de passos completos descrita na Tabela 2.

Tabela 2. Sequência de Passo completo.

Passo	B_0	B_1	B_2	B_3
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

Este tipo de acionamento consiste em se energizar de forma sequencial as bobinas do motor de passo. Dependendo do sentido dos passos realizados (de 1 até 4 ou de 4 até 1) pode-se controlar o sentido de giro. A velocidade do motor é determinada pelo intervalo de tempo entre a energização das bobinas. Desta forma, pode-se facilmente implementar a sequência descrita na Tabela 2 apenas realizando-se um deslocamento de bits em uma variável selecionada.

No experimento de acionamento do motor M_1 é solicitado ao aluno que crie uma função de nome *girarMotorX(int sentido)* que recebe o sentido de giro. Esta função deve realizar o deslocamento de um passo de acordo com o sentido recebido. A mudança entre os passos pode ser realizada através de um deslocamento de bit em uma variável escolhida. Cada bit da variável representa uma

bobina do motor de passo. De acordo com o sentido estabelecido o deslocamento é realizado para direita ou para esquerda, respeitando-se os valores limites que devem ser atribuídos a variável estabelecidos na Tabela 2. Para acionar o motor é necessário configurar 4 pinos GPIO como saída. Ao final da função *girarMotorX* deve-se atribuir o valor da variável escolhida ao registrador correspondente à porta dos pinos GPIO. Para girar o motor continuamente em um determinado sentido é preciso chamar esta função dentro de um laço de repetição.

Outro conceito importante que tem sido abordado com o uso da plataforma proposta é o uso de interrupções. Utilizando-se os sinais provenientes dos sensores de início e fim do eixo x (S_{xi} e S_{xf}) conectados a pinos de GPIO, habilita-se as interrupções correspondentes para identificar quando o eixo x atinge a posição inicial ou final. Este experimento é uma continuação do acionamento do motor M_1 , onde aqui é implementada uma função de inicialização. Nesta função, o motor deve girar até atingir a posição inicial que é identificada através de uma interrupção.

O conceito de gerador de PWM, existente na maioria dos microcontroladores, é utilizado na plataforma para o acionamento do eixo y. O motor M_2 que movimenta o eixo y é um motor de passos bipolar acionado por intermédio do drive D2. O controle deste drive é realizado através de um sinal PWM que determina o deslocamento do motor a cada pulso da onda quadrada. A direção do movimento pode ser controlada por um pino de GPIO, onde o nível lógico selecionado determina o sentido de giro. A habilitação do drive é realizada pela entrada *Enable* (*En*) ativado em nível baixo, que pode ser controlado através de um pino de GPIO do microcontrolador. Neste experimento o aluno deve configurar um pino de PWM e dois pinos de GPIO para controlar o motor de passos. Assim como no acionamento do motor M_1 , solicita-se que seja implementada uma função para girar o motor de acordo com o sentido estabelecido. A diferença está na forma com que a função é implementada. Para o motor M_2 é necessário apenas mudar o valor de *Dir* para modificar o sentido de giro. Quando deseja-se parar o motor deve-se desativar D2 colocando-se *En* em nível alto. A função de inicialização do eixo y é similar a do eixo x.

3.4 Controle de Posição

Após ter conhecimento de como acionar os motores de passo, o aluno pode desenvolver o *firmware* para controlar a posição desejada da mesa de coordenadas. Para desenvolver esta atividade o(a) estudante necessita de conhecimentos sobre GPIO, PWM e interrupções. O fluxograma ilustrado na Figura 7 representa o *firmware* desenvolvido para o controle de posição.

Quando deseja-se controlar a posição de um eixo através de motor de passo em malha aberta, a posição atual é determinada em função do número de passos realizados desde uma posição de referência. Para esta finalidade, utiliza-se sensores de proximidade para determinar a posição inicial do deslocamento. Logo, no início do funcionamento do sistema, deve-se executar uma rotina de inicialização para determinar onde está localizada a posição inicial.

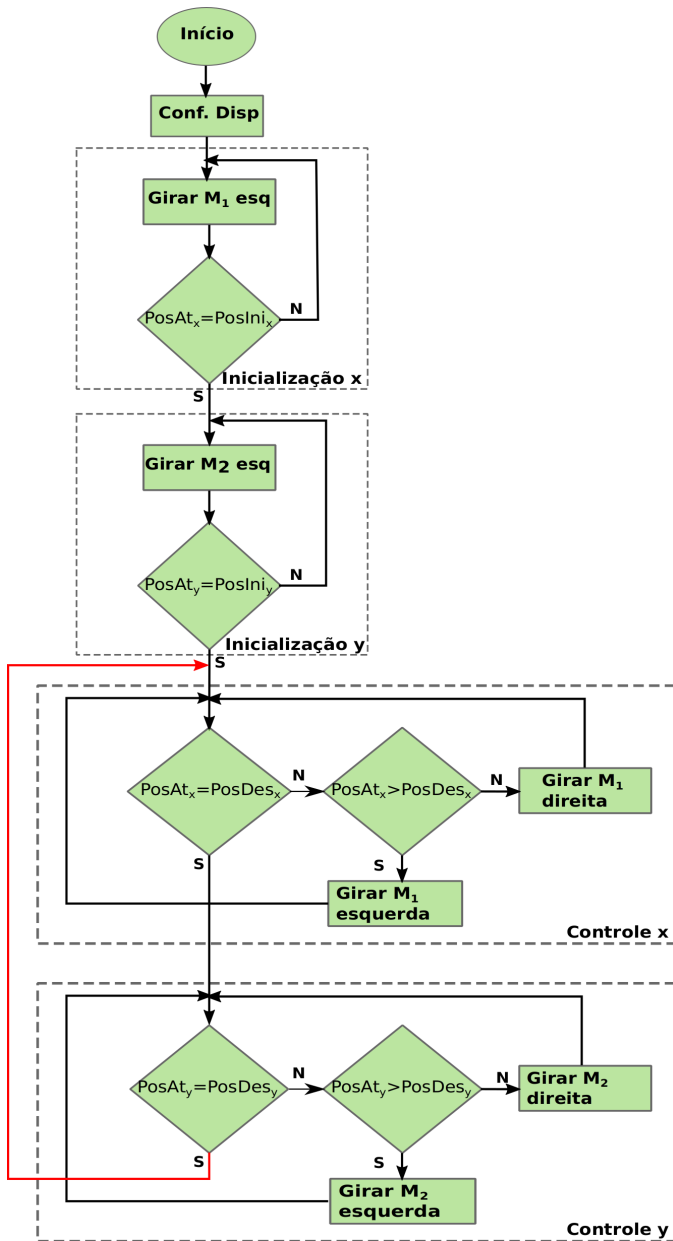


Figura 7. Fluxograma para controle de posição

Para facilitar a implementação do controle de posição, optou-se por realizar o ajuste de posição dos eixos de forma sequencial. Inicialmente, controla-se a posição do eixo x e em seguida a do eixo y. Inicialmente, o *firmware* desenvolvido deve configurar os periféricos utilizados (GPIO, PWM, interrupções). Em seguida realiza-se na sequência a rotina de inicialização para o eixo x e eixo y. Na rotina de inicialização o microcontrolador deve girar o motor correspondente até encontrar a posição inicial, determinada através das interrupções geradas pelos pinos de GPIO conectados aos sensores S_{xi} ou S_{yi} . O aluno já havia implementado esta rotina no experimento relacionado a interrupção descrito na seção 3.3. Quando o eixo atinge a posição inicial, a posição atual de cada eixo ($PosAtual_n$ com $n = x$ ou y) deve ser igualada a zero. A partir deste instante o *firmware* passa a executar a rotina de controle do eixo x.

No controle do eixo x quando a posição atual ($PosAt_x$) é maior do que a posição desejada ($PosDes_x$), aciona-se o motor M_1 fazendo-o girar para esquerda. Quando a $PosAt_x < PosDes_x$ aciona-se o motor para direita. Utiliza-se a função *giraMotor* desenvolvida no experimento de acionamento com GPIO, com a modificação de atualizar o valor da posição atual de acordo com o sentido de giro. Ao atingir a posição desejada ($PosAt_x = PosDes_x$) executa-se a rotina de controle do eixo y. Para o eixo y a lógica de programação é a mesma, mudando-se apenas a forma com que o motor é acionado e como a posição atual é atualizada. Como cada passo de M_2 é realizado através de um pulso do PWM gerado, a atualização da posição atual do eixo y ($PosAt_y$) pode ser realizada na rotina de interrupção do PWM. Ao atingir a posição desejada do eixo y o *firmware* retorna ao início do controle do eixo x.

O *software* utilizado para programar a TIVA1294 é o *Code Composer Studio* (CCS). Através do depurador existente no CCS é possível modificar em tempo de execução o valor da posição desejada para os dois eixos. Além disto, também pode-se monitorar a evolução da posição atual ou qualquer outra variável desejada. A mudança da posição desejada também pode ser feita através de comandos enviados pela porta serial do microcontrolador utilizado.

3.5 Seguimento de trajetória

O *firmware* implementado para controle de posição pode ser expandido para realizar um controle de trajetória da mesa de coordenadas. Uma primeira opção é realizar o controle de trajetória em outro dispositivo. Pode-se desenvolver um *software* em um computador pessoal, que envie comandos para mudança de posição desejada através da comunicação serial.

Outra opção é realizar o controle no próprio microcontrolador externo. A trajetória desejada pode ser previamente armazenada em um vetor contido no próprio *firmware*. Em intervalos de tempos fixos incrementa-se a posição do vetor que será indexada. Para esta finalidade utiliza-se um temporizador (“*Timer*”) do microcontrolador, que é configurado para estourar no intervalo de tempo desejado. Ao atingir o limite, a interrupção do temporizador é ativada e desloca o *firmware* para rotina de interrupção atribuída. Nesta rotina muda-se o valor de $PosDes$ para o próximo valor armazenado no vetor. Ao atingir a última posição do vetor retorna-se o valor da variável indexadora para a primeira posição. Desta forma, a trajetória é realizada de maneira cíclica.

Motivado pelo cenário atual de pandemia causado pela COVID-19 em que se encontra o mundo, a plataforma desenvolvida foi utilizada para reproduzir o perfil de respiração de um paciente. No mundo todo têm surgido diversos projetos para desenvolver respiradores de baixo custo, com o objetivo de suprir a carência existente nos hospitais destes equipamentos. Este é um problema mundial e em alguns países morreram diversos pacientes por falta de leitos e equipamentos. Projetos promissores, que já foram testados em pacientes, utilizam respiradores manuais pressionados por algum dispositivo automático. Alguns destes projetos fazem uso de motor de passo para acionar o dispositivo que pressiona o respirador manual, como por

exemplo o projeto INSPIRE desenvolvido pela USP (Poli-USP, 2020).

A curva de respiração do paciente foi inicialmente reproduzida no *software Scilab*, para obter o conjunto de pontos de referência. Utilizou-se então a opção de controle implementada no próprio microcontrolador descrita anteriormente. Os valores de referência foram armazenados no vetor dentro do *firmware*. Os resultados obtidos para o deslocamento realizado estão ilustrados na Figura 8.

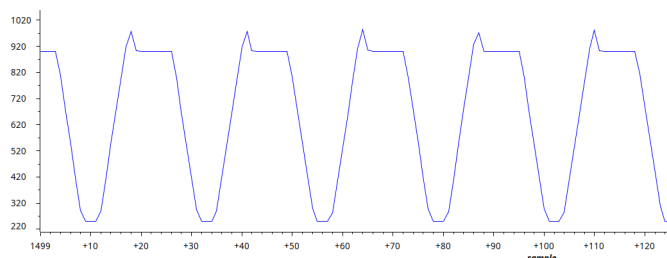


Figura 8. Posição do eixo x para reproduzir a curva de pressão necessária de um respirador mecânico

É possível observar que o eixo controlado seguiu a trajetória que reproduz o perfil de respiração do paciente. Este perfil só seria convertido na pressão de ar desejada se o respirador automático tivesse uma relação linear com o deslocamento do eixo. Para um controle eficaz, seria preciso realimentar a malha de controle com valor da pressão medida. Entretanto, este experimento serve de fase inicial de testes para desenvolvimento do respirador, que podem ser validados na plataforma desenvolvida.

4. CONCLUSÃO

Neste artigo foi apresentado uma mesa de coordenadas desenvolvida com o reuso dos materiais visando diminuir o descarte e aumentar o tempo de vida útil de peças. Esta solução tende a dirimir a falta de prática nas disciplinas de engenharia, devido ao orçamento limitado da instituição na compra de equipamentos novos para o laboratório. Outra vantagem do uso da solução é ter aplicações de problemas reais fazendo com que o aluno tenha uma maior desenvolvimento de senso crítico, análise de problemas, depuração de erros e proposição de soluções. Além disto, é observado que problemas desta natureza aumentam a motivação dos alunos ao cursar as disciplinas e consequentemente podem reduzir a evasão e retenção, que é uma barreira recorrente dos cursos de engenharia.

Espera-se que o compartilhamento desta experiência e da solução de forma detalhada sirva de estímulo para que outros consigam reproduzir ou desenvolver kits semelhantes sempre com segurança e sustentabilidade. Ademais os alunos ao verem as soluções de reuso, se sentem estimulados a desenvolverem projetos para solucionar pequenos problema ou de reutilizarem os componentes para desenvolvimento de projetos das disciplinas de graduação. A plataforma se mostra útil para o estudo do sistema de coordenadas, tanto para análise de posição (leitura de sensores), quanto controle, e insere o aprendizado de conceitos práticos que não são fáceis de se observar e aprender apenas com estudos teóricos, como latência, calibração e uso de interrupções. Este estudo hoje é aplicado em

impressoras 3D, CNC, além das comunicações que são utilizadas em ambientes industriais.

Pretende-se dar continuidade ao projeto com a colocação de ferramentas acopladas na plataforma para realização de diferentes experimentos, como cortes e marcações. Além disto, pretende-se acionar a plataforma por meio de controladores lógicos programáveis via rede MODBUS.

AGRADECIMENTOS

Gostaríamos de agradecer ao Técnico Pedro Marcelo do LAC - UFPB por ser sempre muito solícito e auxiliar no desenvolvimento das soluções propostas pelo grupo de trabalho e o TREE/PROBEX por fornecer REEs arrecadados em suas atividades.

REFERÊNCIAS

- Abreu, D.C. (2014). *Resíduo eletrônico e o ensino de química: uma proposta de educação para a cidadania*. Master's thesis, Universidade Federal de Brasília - UNB, DF.
- Cardoso, J.T. and Souza Filho, C. (2019). Development of a didactic ultrasonic level sensor through a modular approach. In *Congresso Brasileiro de Educação em Engenharia*.
- Cunha, M.F., da Silva, T.C.B., and Souza Filho, C. (2018). Plataforma didática térmica multidisciplinar para controle, automação e redes de sensores. In *Congresso Brasileiro de Automática*.
- Damaceno, A.S. (2014). *Reutilização de lixo eletrônico no design de novos produtos*. Master's thesis, Universidade Tecnológica federal do Paraná, Curitiba - PR.
- Friedrich, L.S. (2014). *O lixo eletrônico como a possibilidade para o ensino de química na formação de professores*. Master's thesis, Universidade Federal de Santa Maria, RS.
- Jacobi, P.R. and Bese, G.R. (2011). Gestão de resíduos em são paulo: desafios da sustentabilidade. *Estudos Avançados*, 25, 135 – 158.
- Lima, S.F., Albuquerque, M.F.A., Oliveira, L.S., de Lima, E.P., and Silva, P.H.G. (2017). Robô eco-sustentável para aplicação em robótica educativa utilizando lixo tecnológico. *Caderno de Graduação de Ciências Exatas e Tecnológicas - UNIT, AL*.
- Natume, R.Y. and Sant'Anna, F.S.P. (2011). Resíduos eletroeletrônicos: Um desafio para o desenvolvimento sustentável e a nova lei da política nacional de resíduos sólidos. *3º International Workshop Advances in Cleaner Production*.
- ONUBrasil (2019). Oit: somente 20% do lixo eletrônico é reciclado formalmente. <https://news.un.org/pt/story/2019/04/1668641>.
- Paiva, D., Gonzaga, G., Oliveira, F., Jasbick, D., and Miranda, J. (2016). Utilização de lixo eletrônico para a produção de jogos e materiais didático-pedagógicos. *Revista Educação Ambiental em Ação*, 58.
- Poli-USP (2020). Inspire ventilador pulmonar aberto de baixo custo. <https://www.poli.usp.br/inspire>.
- Vaz, I.S., Flores, T.K.S., de F. Medeiros, D., and Netto, A.V.S. (2018). Desenvolvimento de kit didático sustentável para ensino de automação industrial. In *Congresso Brasileiro de Automática*.