

INTERFACE HOMEM-MÁQUINA COM USO DE COMANDOS DE VOZ EMBARCADA EM PLATAFORMA ANDROID PARA CONTROLE DE DIRIGIBILIDADE DE UMA CADEIRA DE RODAS MOTORIZADA

ABÍLIO AZEVEDO, M. C., WALCIMAR DENADAI, Z., LUIS LIMA, E. M.

IFES – INSTITUTO FEDERAL DO ESPÍRITO SANTO
Av. Vitória, 1729 - Jucutuquara, Vitória - ES, 29046-020

E-mails: abilio10@gmail.com, walcimar@outlook.com, luisedu@ifes.edu.br

Abstract — Wheelchairs are the main mobility tools for people without ability to walk, whether by accident, illness or disability. With the advent of technology, motorized wheelchairs become more accessible and functional. This work aimed to describe the directional control of a motorized wheelchair through an application installed on a mobile device Android, whose control is effected by voice or touch of the chair user. The system developed consists of the interaction of a cell phone Android with a ATmega 2560 microcontroller through Bluetooth communication. The microcontroller processes the commands by PID, by scanning the gyroscope and encoders, and controls the drivers, which operates the wheelchair engines.

Keywords — mobile robotics, speech processing, wheelchair, automatic control, embedded system.

Resumo — Cadeiras de rodas são as principais ferramentas de locomoção para pessoas cuja capacidade de andar esteja comprometida, seja por acidente, doença ou deficiência. Com o advento da tecnologia, cadeiras de rodas motorizadas tornam-se mais acessíveis e funcionais. Este trabalho tem intuito de descrever o controle direcional de uma cadeira de rodas motorizada por meio de um aplicativo, instalado em um aparelho celular Android, cujo controle é efetuado por voz ou toque do usuário da cadeira. O sistema desenvolvido é composto pela interação de um celular Android com um microcontrolador ATmega 2560, por meio de comunicação Bluetooth. O microcontrolador processa os comandos por PID, fazendo a leitura do giroscópio e dos encoders, e controla a placa de acionamento, a qual opera os motores da cadeira.

Palavras-chave — robótica móvel, processamento de voz, cadeira de rodas, controle automático, sistema embarcado.

1 Introdução

A atual disponibilidade de recursos em software e hardware tem viabilizado soluções tecnológicas automáticas e programáveis com crescente abrangência no mercado. Esta conjuntura tem motivado relevantes esforços no sentido de proporcionar a melhoria da qualidade de vida do ser humano, sendo estes esforços ressaltados quando se trata da qualidade de vida de pessoas com deficiências físicas.

Visando a ampliar a acessibilidade e a inclusão social de pessoas com deficiências físicas (Bracciali, 2016), por uso de cadeira de rodas motorizada, com controle de baixo custo, neste trabalho é apresentado um sistema capaz de controlar automaticamente a mobilidade de uma cadeira de rodas a partir de comandos de voz.

Este sistema de controle aplicado a uma cadeira de rodas motorizada possibilita ao usuário a definição de comandos de voz para os movimentos desejados, por uso de uma interface homem-máquina (IHM) embarcada em plataforma ANDROID (Sistema Operacional Móvel). Assim sendo, é alcançada uma grande abrangência de usuários face à atual diversidade e disponibilidade de dispositivos (telefones celulares, *tablets* e outros) compatíveis com este sistema operacional. Além desta característica, soma-se o fato de a arquitetura de controle ser composta de hardware de controle e sensoramento de baixo custo computacional e aquisitivo.

Visando contribuir para o universo de soluções tecnológicas aplicadas à mobilidade, sendo atualmente disponíveis diversos dispositivos que proporcionam auxílio locomotor a pessoas deficientes, buscou-se desenvolver um método de interação homem-máquina funcional e flexível.

Considerando-se que os dispositivos celulares móveis ou simplesmente celulares são na atualidade de grande abrangência de uso, e ainda que integram diversos recursos e sensores para captação de comandos externos (voz, toque, movimento, posição, e outros), esta plataforma apresenta-se como uma ferramenta robusta para suportar a interface-homem máquina no sistema proposto.

Neste trabalho, a solução definida por um processamento de voz totalmente embarcado (local), estabelece uma contribuição em relação aos trabalhos de (Tadvil et al, 2016) que limita-se ao processamento de comandos de toque-tela para definição de movimentos da cadeira, bem como, é obtido melhor desempenho em tempo de resposta relativo ao trabalho de (Purushothaman et al., 2015) que faz uso da “nuvem” (servidores *on-line* via internet) para o processamento dos comandos de voz.

Para a obtenção da solução desejada será apresentada a arquitetura de software e hardware utilizada, assim como, uma análise de desempenho a partir dos resultados obtidos com base em tempo de resposta e precisão de execução de movimentos de rotação e translação.

2 Arquitetura do Sistema

A partir da interface homem-máquina processada no celular, é estabelecida a comunicação dos comandos de locomoção via um link Bluetooth para a unidade de processamento (microcontrolador ATmega 2560), na qual são definidos os comandos a serem enviados via serial para *driver* de acionamento dos motores. Sensores de aceleração, orientação e *encoders* incrementais realimentam a unidade processadora com as informações referentes aos movimentos realizados, sendo toda esta arquitetura ilustrada na Figura 1.

2.1 Hardware Utilizado

- Placa de Controle com Microcontrolador (ATmega 2560 R3);
- Placa de Acionamento (RoboClaw 2x60A);
- Telefone Android (Smartphone);
- Modulo Bluetooth (RS232 HC-05);
- Sensor Giroscópio e Acelerômetro (MPU6050);
- 2 x Encoders (HTR-W2-1024-3-L-5V);
- 2 x Motores DC;
- Cadeira de rodas adaptada.

2.2 Software Utilizado

- Plataforma online MIT App Inventor;
- Integrated Development Environment Arduino;
- Software supervisorio da placa Roboclaw.

3 Implementação

3.1 Placa de Controle Arduino Mega

Trata-se de uma placa de desenvolvimento microprocessada, baseada no microcontrolador ATmega2560, que utiliza frequência de *clock* de 16MHz, e disponibiliza vias de entrada e saída digitais e analógicas, fontes de interrupção, *timers* e outros recursos conforme detalha (Datasheet Atmel, 2014).

3.2 Placa de Acionamento RoboClaw 2x60A

Esta é uma placa que contém os *drivers* de potência para acionamento dos motores da cadeira, sendo disponíveis dois canais com capacidade para suprir 60A cada um. Esta placa também possui uma unidade processadora com interface USB, que viabiliza o uso de software supervisorio para monitoramento e modificação de parâmetros, como ganhos do controlador PID e leitura de pulsos dos *encoders*. Há ainda disponível uma interface serial pela qual podem ser recebidos comandos pré-definidos para acionamento dos motores e consequente movimentação da cadeira. Esta placa é alimentada em 24V (alimentação que supre os motores) e disponibiliza alimentação em nível de 5V

que é utilizada para alimentar a placa de controle Arduino Mega.

3.3 Sistema Operacional Android

A IHM desenvolvida é um aplicativo processado sobre o sistema operacional Android. Esta plataforma viabiliza um processamento embarcado, de baixo custo computacional para implementação e com grande abrangência de usuários. Vale ainda ressaltar que com esta solução torna-se acessível os diversos recursos disponíveis na plataforma móvel (Smartphone).

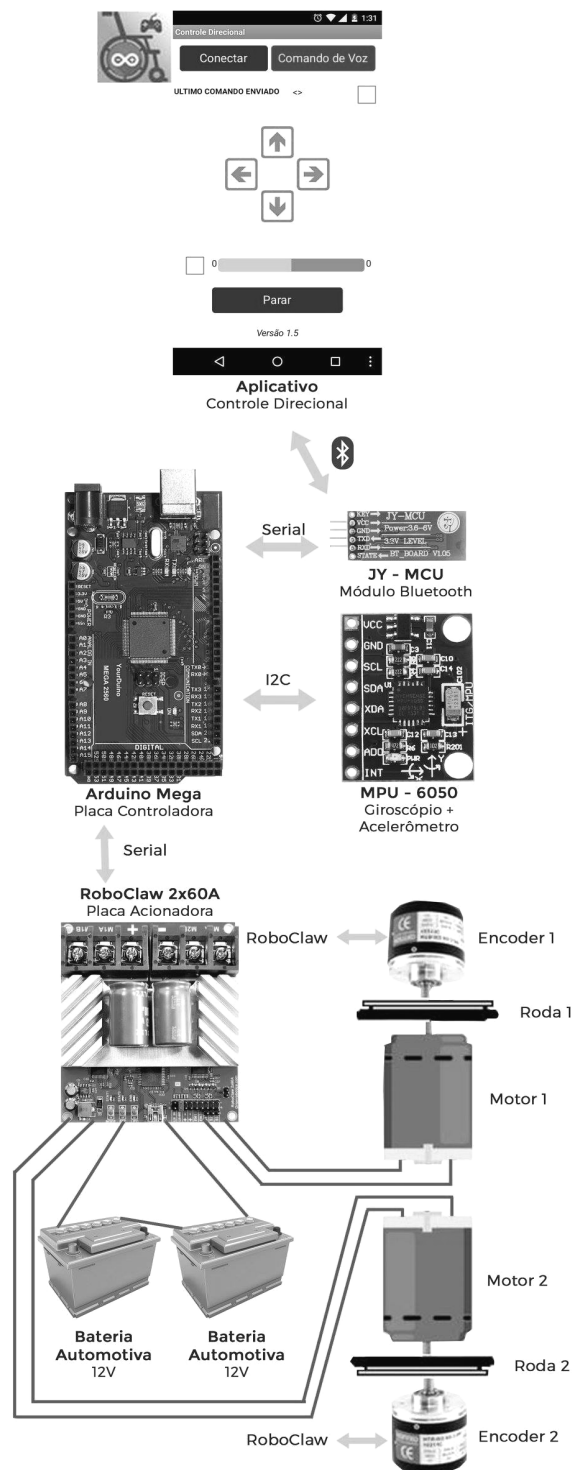


Figura 1. Diagrama de blocos da cadeira.

3.4 Módulo Bluetooth HC-05

O módulo HC-05 viabiliza comunicação Bluetooth SPP (Serial Port Protocol), que caracterizada por ser *wireless* flexibiliza o uso ainda que à distância, limitada pelo alcance de 10m.

3.5 Módulo MPU-6050

Este módulo possui dois sensores integrados: giroscópio e acelerômetro, e internamente um recurso chamado DMP (Digital Motion Processor), com o qual obtém-se a orientação da cadeira, sem necessariamente ocupar tempo de processamento do microcontrolador. O resultado é lido diretamente ou colocado em um buffer do tipo FIFO (First In, First Out), sendo acessado via comunicação no padrão I2C.

3.6 Plataforma MIT App Inventor

A plataforma online MIT (*Massachusetts Institute of Technology*) App Inventor é uma ferramenta de código aberto que permite o desenvolvimento de programas aplicativos para o sistema operacional Android. A possibilidade de programação por uso de diagramas de blocos funcionais facilitou o uso de ferramentas necessárias para implementação da IHM desejada, bem como, reduziu o tempo de desenvolvimento do aplicativo, o que viabiliza a redução no custo de correções para aperfeiçoamento da solução.

4 Metodologia

O desenvolvimento deste trabalho divide-se em quatro etapas.

A primeira etapa correspondeu a criação da IHM, sendo o *layout* definido para suportar aos comandos de toque-tela e voz.

Na segunda etapa foi definido o *link* de comunicação Bluetooth entre o Smartphone e a unidade de processamento (ATmega 2560). Para uso deste *link* foi utilizado RS232 HC-05.

A terceira etapa corresponde ao sensoramento angular para definição da orientação da cadeira, na qual foi utilizado um módulo integrado com giroscópio e acelerômetro (MPU-6050). O processamento para obtenção da orientação é realizado com uso da integração da velocidade angular, obtendo-se assim o ângulo relativo da cadeira.

A quarta etapa corresponde à programação do microcontrolador em conjunto com a placa de acionamento. Nesta etapa foi estabelecido um controle em malha fechada. A partir dos comandos de movimentos desejados, recebidos por Bluetooth, define-se direção e distância a ser percorrida, ou deslocamento angular desejado. Com as informações lidas dos sensores referentes a posição angular e distância, é gerado o acionamento controlado dos motores.

5 Desenvolvimento

5.1 Aplicativo Android

A IHM desenvolvida com a ferramenta MIT APP Inventor é ilustrada na Figura 2 e foi desenvolvida para viabilizar o uso de dois tipos de comandos, o toque-tela e o comando de voz.

Abaixo segue uma descrição sucinta dos elementos presentes no display e suas funções:

- **Conectar:** permite a conexão e desconexão de forma manual entre o aparelho celular e o módulo bluetooth RS232 HC-05, desde que ambos, celular e módulo bluetooth, estejam previamente pareados;
- **Comando de Voz:** possibilita o recebimento de comandos de voz. Os comandos aceitos são: conectar, desconectar, frente, traz, direita, esquerda e parar. Os comandos frente e traz podem ser acompanhados do parâmetro distância a ser percorrida, já os comandos direita e esquerda podem ser acompanhados pelo parâmetro deslocamento angular;
- **Último Comando Enviado:** apresenta as informações do comando anterior enviado à cadeira pelo usuário. Acompanhado com uma caixa de seleção, esta opção pode ser ativada ou desativada;
- **Setas Direcionais:** enviam à unidade processadora do controle dos motores a informação da direção que se deseja que a cadeira siga (frente, trás, direita e esquerda);
- **Barra de Rolagem:** possibilita um controle angular da cadeira (direita e esquerda) em modo contínuo, e pode atuar em conjunto com as setas direcionais frente e trás. Acompanhado com uma caixa de seleção, esta opção pode ser ativada ou desativada;
- **Parar:** envia o comando para que a cadeira pare seu trajeto independente do comando anterior. Tem prioridade sobre os demais comandos.

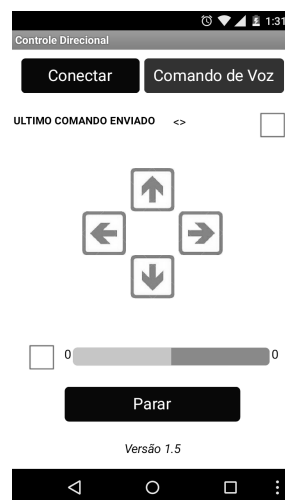


Figura 2. Interface Homem-Máquina

Em ambientes com acesso à internet ao tocar o ícone “Comando de Voz” o aplicativo receberá o chamado através do microfone do aparelho celular, enviará a gravação para que o processamento em nuvem nos servidores da Google (empresa desenvolvedora do sistema Android), seja feito e o áudio seja convertido em texto. O texto é enviado ao aplicativo.

O texto recebido é processado, sendo extraídas as informações pertinentes como: direção, distância ou ângulo. Comandos são convertidos em *caracteres* únicos. Distâncias e ângulos são enviados como *inteiros*.

Exemplo: Para a cadeira seguir em frente por 100 centímetros, basta que o usuário pronuncie **FRENTE 100**, que será processado e convertido como f 100 (char, int). O resultado é enviado ao módulo Bluetooth RS232 HC-05.

Com o intuito de possibilitar o uso de comandos de voz em qualquer ambiente ainda que na ausência de acesso à internet, o processamento de voz foi embarcado no dispositivo celular. Esta solução foi obtida por meio da compatibilização entre o aplicativo desenvolvido e a versão 4.4 do Android e superiores, que apresentam processamento de voz entre seus recursos nativos. Mediante ao download de uma biblioteca desenvolvida pela Google e sua correta instalação, o celular passa a processar comando de voz off-line.

5.2 Comunicação Bluetooth

O transceptor Bluetooth utilizado para estabelecer o link celular-microcontrolador, atua com frequência de 2,4GHz e largura de banda 2,4GHz a 2,8GHz, banda ISM (Industrial, Scientific and Medical). A modulação do sinal é GFSK (Gaussian Frequency Shift Keying), taxa de 8 bits de dado e 1 de parada, sem paridade. O alcance de comunicação é de 10 metros segundo o fabricante, todavia, distâncias maiores foram obtidas durante o manuseio prático. (Datasheet RS232 HC-05, 2013).

5.3 Programação Placas de Controle e Acionamento

Inicialmente a placa de acionamento RoboClaw foi configurada com os ganhos PID em testes de locomoção para controle de velocidade, através do software supervisor. Este software possui a função de “AutoTune”, a qual define os ganhos PID para o controle de velocidade dos motores (aceleração, manutenção e desaceleração), e assim podemos passar instruções de velocidade via comunicação serial para que a placa acione os motores da cadeira.

Em seguida, foi desenvolvido um programa para o microcontrolador ATmega 2560, o qual inclui a recepção dos comandos via Bluetooth, leitura da posição angular da cadeira, leitura dos encoders (com uso da placa RoboClaw) e controle PID da velocidade.

Assim, o microcontrolador recebe o comando via Bluetooth (referências), e com uma malha fechada de controle PID, processa esse comando e

controla a velocidade dos atuadores para realização do movimento desejado. Na Figura 3 é ilustrado o processamento lógico do programa executado pelo microcontrolador. (User Manual RoboClaw, 2015).

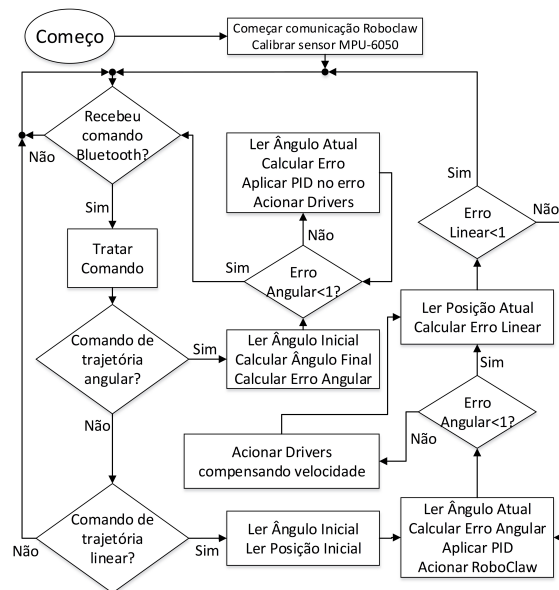


Figura 3. Fluxograma simplificado do programa que roda no microcontrolador Arduino.

5.4 Movimentos de Rotação

Para o controle dos movimentos e rotação foi utilizado o módulo MPU-6050, o qual fornece a orientação da cadeira a partir de uma orientação referência inicial. Assim pode-se fazer um controle em malha fechada, ilustrado na Figura 4, do movimento de rotação desejado. Por exemplo, pronunciado o comando de voz “**DIREITA 30**”, a cadeira gira 30 graus para direita (**sentido horário**).

O controle proporcional integral derivativo - PID (Åström et al, 1995)(Ogata, 2003), foi fundamental para suavizar os movimentos e impedir deslizamentos por atrito das rodas.

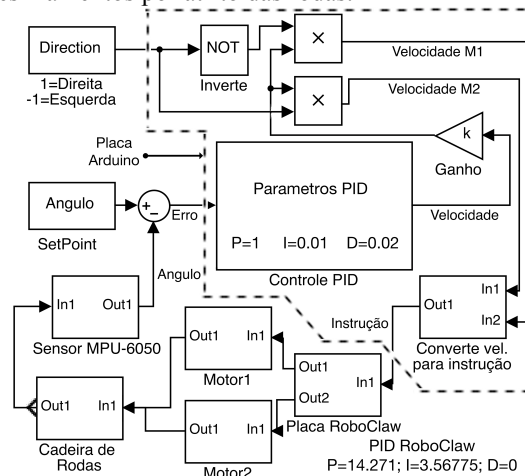


Figura 4. Malha de Controle para Posição Angular

5.5 Movimentos Retilíneos

Para a movimentação em linha reta usou-se o valor da posição angular inicial para manter a cadeira na mesma direção. A parada da cadeira é realizada a

partir da leitura dos *encoders*, sendo a contagem de pulsos lida pela placa RoboClaw e realimentada para o microcontrolador ATmega2560. A malha de controle é ilustrada na Figura 5.

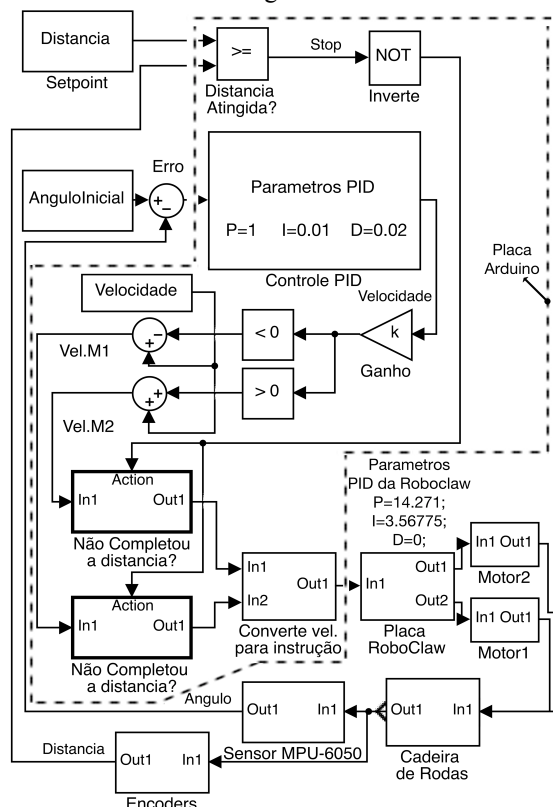


Figura 5. Malha de Controle para trajeto em linha reta.

6 Análise de Performance

6.1 Tempo de resposta do sistema

A análise do tempo de resposta do sistema foi realizada para três intervalos de tempo: o primeiro medido durante o processamento no aplicativo, entre o início do comando (voz) e o envio via Bluetooth; o segundo intervalo medido entre o instante de recebimento do comando pelo microcontrolador e o envio da instrução de velocidade para placa de acionamento; e o terceiro, sendo o tempo entre o envio da instrução do aplicativo e o recebimento de uma confirmação de sucesso, subtraindo o valor dos intervalos um e dois e dividindo por dois (referente à ida e volta do sinal), temos o tempo de transmissão.

Teste	Tempo no App (ms)	Tempo no Microcontrolador ATmega (ms)	Tempo de Transmissão Bluetooth (ms)	Tempo Total (ms)
1	740	500	270	1510
2	1080	490	210	1780
3	800	510	260	1570
4	800	500	180	1480
5	810	500	250	1560
6	830	490	230	1550
7	860	480	220	1560
8	840	510	240	1590
9	780	470	220	1470
10	810	500	230	1540
Media	835	495	231	1561

Tabela 1. Tempo de Resposta para comando "FRENTE 100".

6.2 Taxa de acerto no reconhecimento de voz

O sistema de reconhecimento de voz possibilita as opções online e off-line (com e sem acesso à nuvem). Testes foram realizados para obtenção da precisão no reconhecimento dos comandos de voz, tendo sido utilizadas três vozes diferentes. Sendo gerados comandos de forma aleatória, em um total de 50 comandos de voz por usuário, os resultados obtidos seguem nas tabelas abaixo.

Voz	Acertos	Erros	Percentual de Acerto
1	47	3	94%
2	46	4	92%
3	44	6	88%
Total	137	13	91,3%

Tabela 2. Resultados do reconhecimento de voz Offline.

Voz	Acertos	Erros	Percentual de Acerto
1	48	2	96%
2	47	3	94%
3	47	3	94%
Total	142	8	94,67%

Tabela 3. Resultados do reconhecimento de voz Online.

6.3 Erro de trajeto angular

Os erros nas rotações foram menores em giros menores, todavia quanto maior o deslocamento angular desejado, maior o erro. Porém, esses erros se mostraram bem pequenos, devido à precisão do sensor MPU-6050. Em uma rotação de 30 graus (ilustrado no Gráfico 1) a média de erro foi de 0,08%, e para rotação de 90 graus (ilustrada no Gráfico 2) obteve-se um erro de 0,241%.

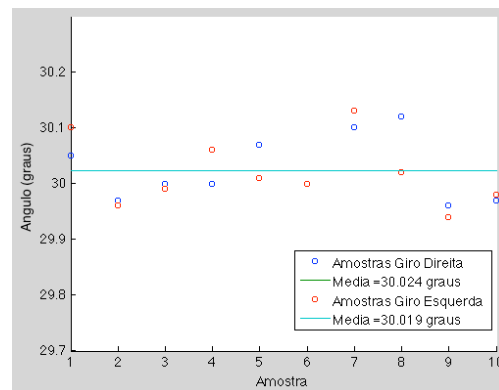


Gráfico 1. Erros angulares para giro em 30 graus

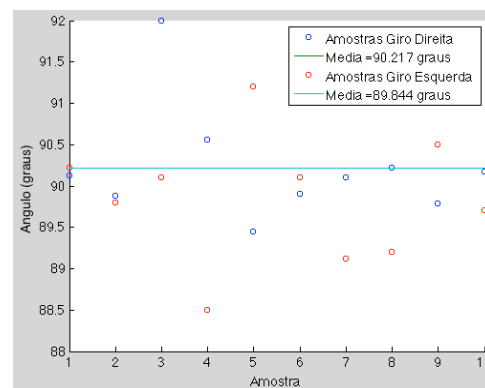


Gráfico 2. Erros angulares para giro em 90 graus

6.3 Erro de deslocamento retilíneo

Nos deslocamentos retilíneos verifica-se uma precisão satisfatória para distâncias curtas (50cm), porém para distâncias maiores (300cm) os erros não foram satisfatórios. Uma sintonia mais refinada dos ganhos PID proporcionará melhores resultados. Na Tabela 3 pode-se conferir os erros nos pulsos dos encoders, na distância em centímetros convertida dos encoders, na medição com uma trena e a variação angular medida com o módulo MPU-6050

Teste	Valor Final Enc.1 (pulsos)	Valor Final Enc.2 (pulsos)	Distância Percorrida - Encoder (cm)	Distância Percorrida - Trena (cm)	Variação Angular (graus)
1	38528	38126	50,47	51,5	1
2	38120	38212	50,26	51	1
3	38100	38063	50,15	50,5	2
4	38472	39027	51,03	52	1
5	39295	39269	51,73	53	2
6	39088	39617	51,82	53	2
7	38592	39125	51,17	52	1
8	39152	39104	51,42	52,5	2
9	37908	38333	50,20	50	1
10	38637	39314	51,32	51,5	2
Media	38589,2	38819	50,957	51,7	1,5
Erro Medio	618,7	848,5	0,957	1,7	1,5

Tabela 3. Erros para trajeto de 50 cm.

7 Desenvolvimento Futuro

Como primeira etapa de desenvolvimento deste projeto, foram testadas apenas trajetórias para curtas distâncias e rotações para deslocamentos fixos e pré-definidos. Os resultados obtidos podem ser melhorados a partir de uma sintonia mais precisa nos ganhos dos controladores, e algumas adequações de projeto mecânico, por exemplo, uso de rodas livre omnidirecionais, que minimizam as interferências na orientação da cadeira no início dos movimentos.

O uso de outras tecnologias de comunicação sem fio também podem ser testadas, assim como suas interferências: ZigBee (Stoll, G. R. 2008), Shields com comunicação SPI/UART direta e o módulo WiFi ESP8266 (Mehta, M 2015).

Visando a aplicação desta solução em ambiente escolar, além dos ajustes para melhoria de precisão na realização dos movimentos, uma das etapas futuras é a realização de trajetórias entre setores da instituição a partir do planejamento de trajetórias e ainda da detecção de obstáculos para realização de movimentos evitando colisões.

8 Conclusão

A execução deste trabalho até o presente momento permitiu a verificação da viabilidade de implementação de uma IHM sobre plataforma ANDROID, com baixo custo computacional e totalmente embarcada em dispositivo celular.

Conclui-se que já é possível a realização da locomoção em ambientes de dimensões residenciais de pequeno e médio porte com realização de trajetórias satisfatórias. Esta contribuição permite a ampliação na abrangência de usuários e aplicação desta solução com independência aos locais de uso vinculados à disponibilidade de internet.

Agradecimentos

Agradecimento ao IFES – Instituto Federal do Espírito Santo – pela infra-estrutura disponibilizada para realização deste trabalho e à todos que contribuíram financeiramente com o projeto.

Referências Bibliográficas

- Bracciali, L. M. P. (2016). "Tecnologia Assistiva e Produção do Conhecimento no Brasil".
- Tadvi1, E. S., Adarkar P., Yadav A., Siddik M. H. S. (2016), "Automated Wheelchair using Android Technology", Imperial Journal of Interdisciplinary Research (IJIR), Vol.2, No.4.
- Purushothaman V., Nagaraj D., Mahadir M., Selvakumar C., Sabarirajan M. (2015), "Automated Multi Specialized Wheelchair for Differently Abled Person Using Smartphone"- International Journal of Trend in Research and Development (IJTRD), Vol.3, No.3.
- ATmel Corporation, "Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V" data sheet, 2014.
- Ion Motion Control Company, "RoboClaw Series - Brushed DC Motor Controllers" User Manual, 2015.
- Guangzhou HC Information Technology Company, "Module Data Sheet" datasheet, 2011.
- Åström, K.J; Hägglund, T. (1995) "PID controllers: theory, design and tuning:". Instrument Society of America. 343p.
- Ogata, K. (2003) Engenharia de Controle Moderno – 4ª edição, 2003, Prentice-Hall.
- Cunha, A. R. (2013) "On the use of IEEE 802.15.4/ZigBee as federating communication protocols for Wireless Sensor Networks"
- Stoll, G. R. (2008) "O QUE É ESTE TAL DO ZigBee?" - UTC Journal 2008 Special Issue – Smart Utilities Networks
- Mehta, M (2015) "ESP 8266: A BREAKTHROUGH IN WIRELESS SENSOR NETWORKS AND INTERNET OF THINGS"