

Universidade Federal de Minas Gerais

Departamento de Ciência da Computação

TP1 de Introdução a Robótica – Programação utilizando a HandyBoard

Alunos:

Daniel Balbino de Mesquita - 2008041306

Pedro Henrique Silva Faria - 2005017021

Tiago de Lima Pereira - 2005017200

Professor: Mario Fernando Montenegro Campos

1 Introdução

O objetivo desse trabalho prático é familiarizar os alunos com o desenvolvimento de software para a HandyBoard utilizando o Interactive C. Além disso, os alunos devem desenvolver um controle simples para o robô e fazer uma avaliação experimental dos erros de atuação.

2 Objetivos

O robô deve ser capaz de:

1. Efetuar translação linear para cada um dos 3 valores de distância: 10cm, 20cm e 30cm. O valor de distância desejado deve ser informado por meio do potenciômetro e o valor correspondente mostrado no display.
2. Efetuar rotação para cada um dos 3 valores de ângulo: 30 graus, 60 graus e 90 graus, o valor de ângulo desejado deve ser informado por meio do potenciômetro e o valor correspondente mostrado no display.
3. Fazer um grupo de 4 leds funcionar em seqüência. Utilizando os botões start e stop para inicializar e parar a seqüência dos leds, o robô deve executar uma seqüência diferente a cada inicialização.
4. Utilizar a característica multitarefa do IC para ligar os leds enquanto o robô se move (trajetória livre) . O robô e os leds devem ficar em execução durante exatos 30 segundos, interrompendo ambas tarefas após esse tempo.
5. Realizar autonomamente uma trajetória quadrada de 30cm de lado. Logo após, uma trajetória circular de 25 cm de raio, ambos por 3 vezes consecutivas(sem utilizar sensores). Além disso, o robô deve desenhar sua trajetória numa cartolina sobre o chão.

3 Desenvolvimento do Robô

Estrutura e Controle

O robô foi projetado para ser robusto e efetuar com precisão as tarefas designadas. Para isso, ele ficou com uma aparência que remete a um cubo totalmente intertravado e seguro. Tivemos alguns problemas no que se refere à fixação dos motores ao LEGO, o que nos levou a problemas de calibração e a várias tentativas de travamento, uma vez que isso se tornou um quesito fundamental para a correta realização das tarefas exigidas.

Foram utilizados dois motores de 9v, cada um conectado a uma roda. Para realizar a redução usamos uma pequena caixa de engrenagens. Tivemos uma enorme dificuldade em fazer com que ele permanecesse em linha reta, já que as caixas de engrenagens de cada lado não tinham exatamente a mesma resistência; o motores estavam em alturas sutilmente diferentes, mas que alteravam o torque em algumas situações. Observe as figuras a seguir:



Figura 1 : Vista frontal do robô

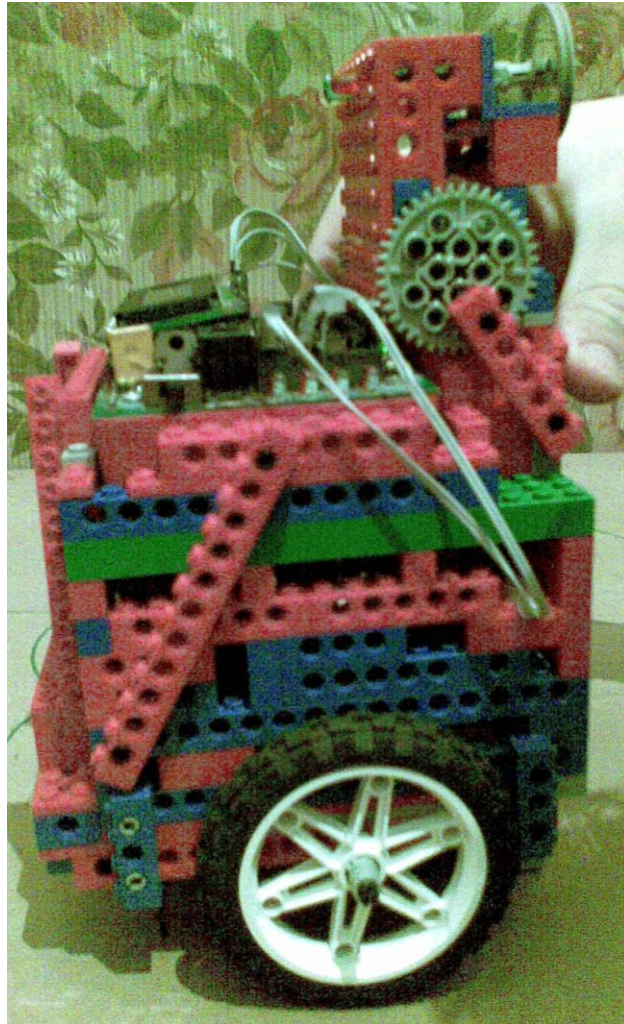


Figura 2: Vista lateral do robô

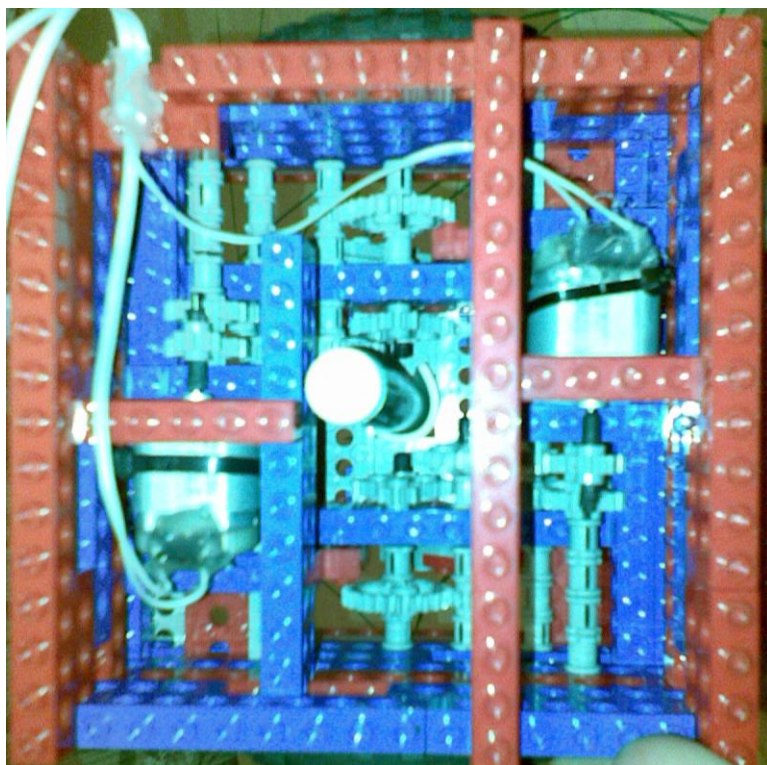


Figura 3: Vista do travamento dos motores



Figura 4: Vista do suporte onde estão os LEDs(vermelho, amarelo, verde, vermelho)

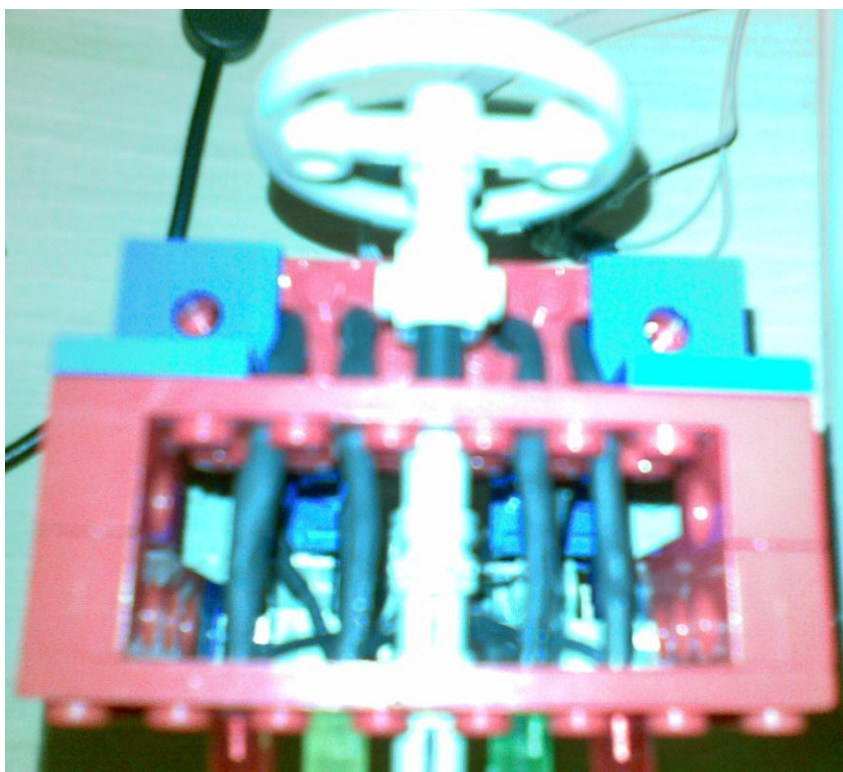


Figura 5: Detalhe acabamento LEDs

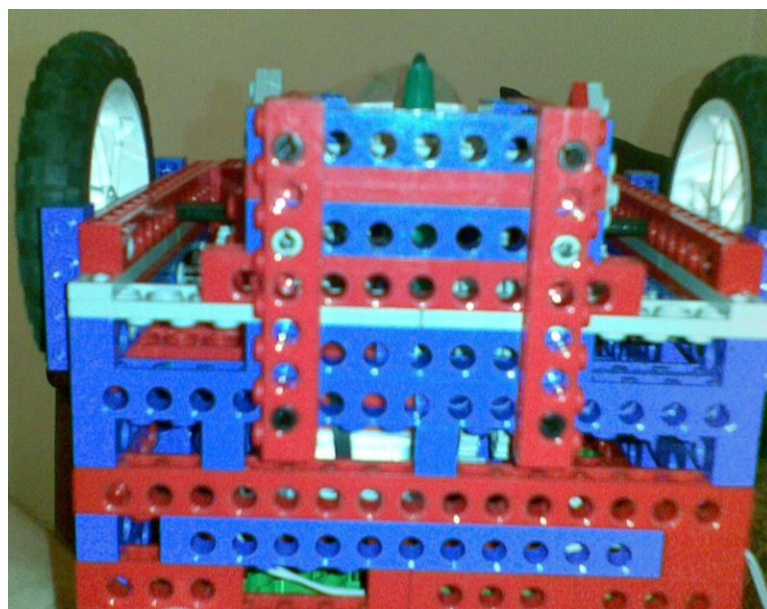


Figura 6: Detalhe do pincel para deixar rastro no papel

Já a parte de controle ficou concentrada na programação da HandyBoard, na qual a dificuldade maior não foi a programação em si mas sim a relação desta com o comportamento esperado e nem sempre obtido dos atuadores, muitas vezes prejudicados por contatos mal encaixados e engrenagens desalinhadas.

Menu

Criamos um menu de utilização intuitiva onde o potenciômetro serve para navegar entre as funções do menu e para determinar valores numéricos. Já os botões de 'start' e 'stop' servem, respectivamente, para selecionar/ativar e voltar/parar as funções. Veja as fotos a seguir:

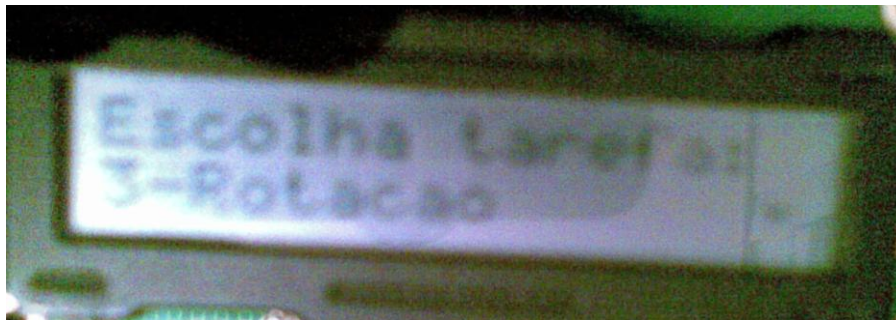


Figura 7: Menu de tarefas

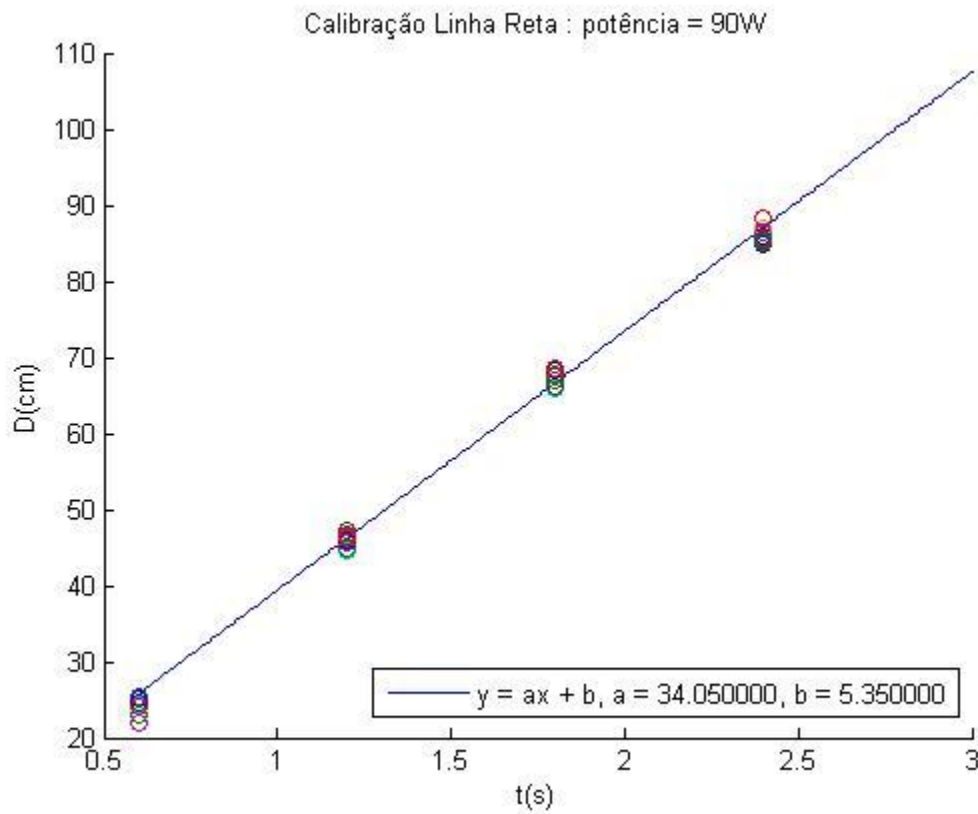


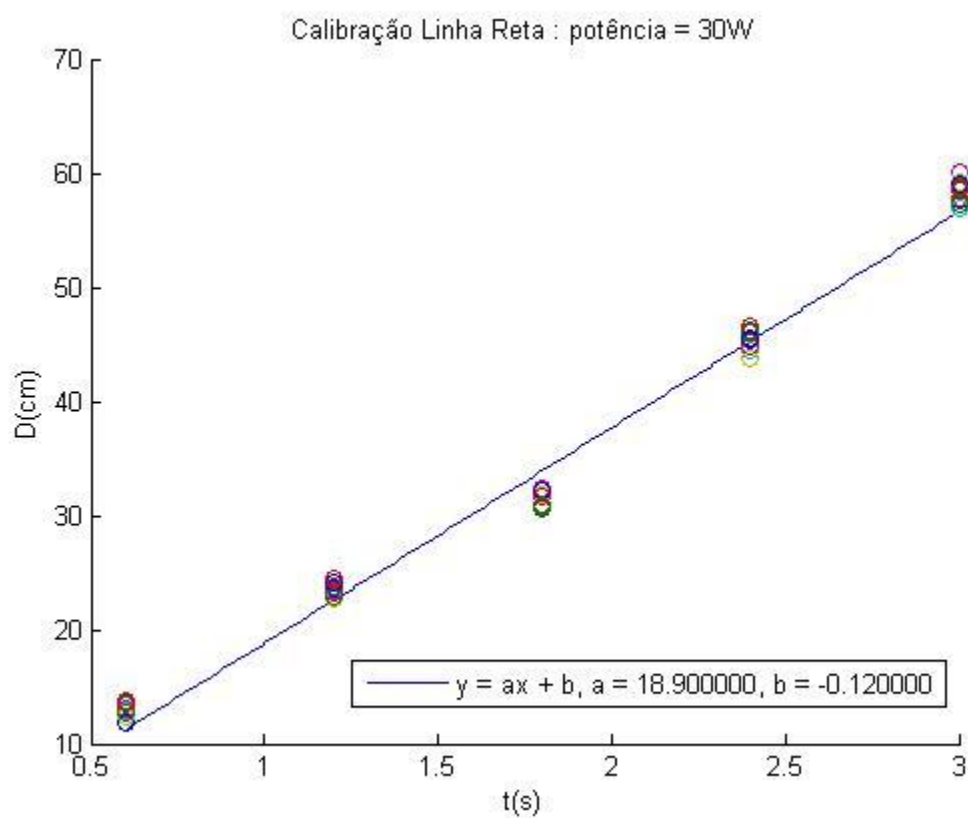
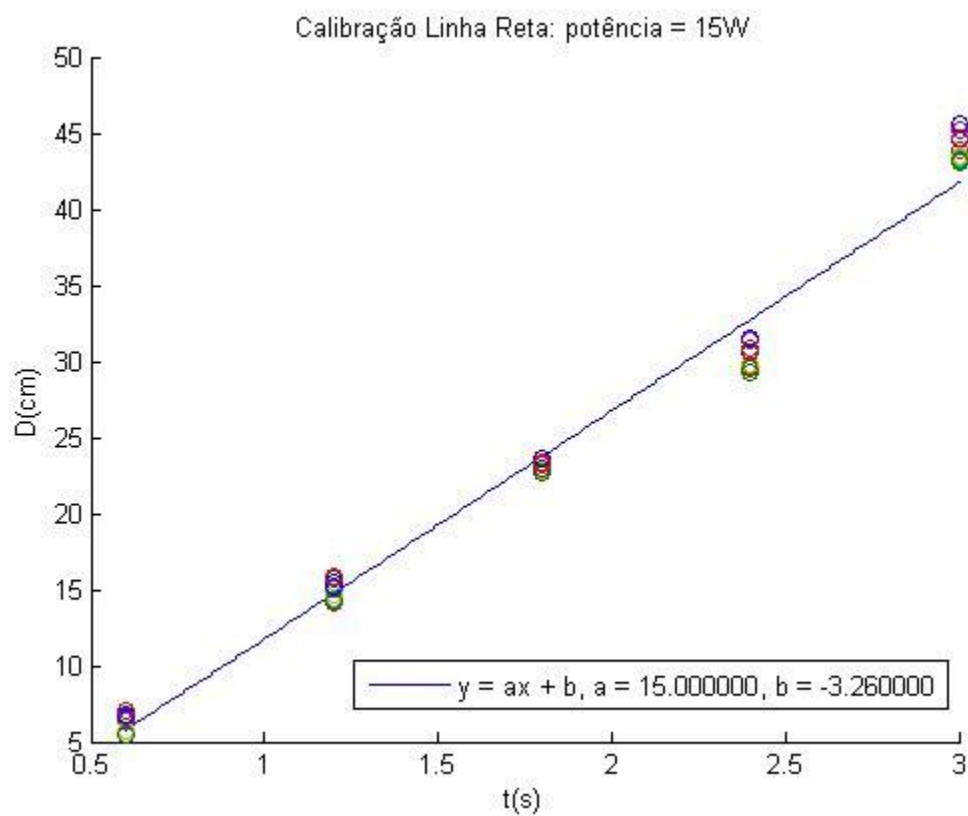
Figura 8: Menu translação

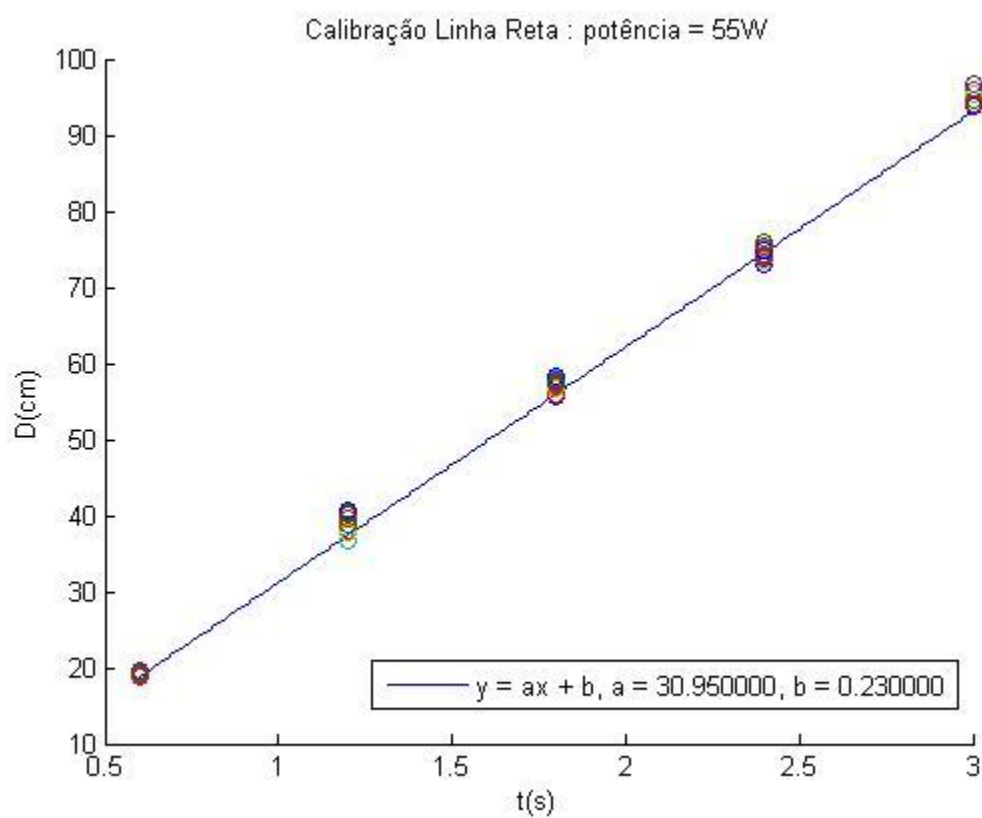
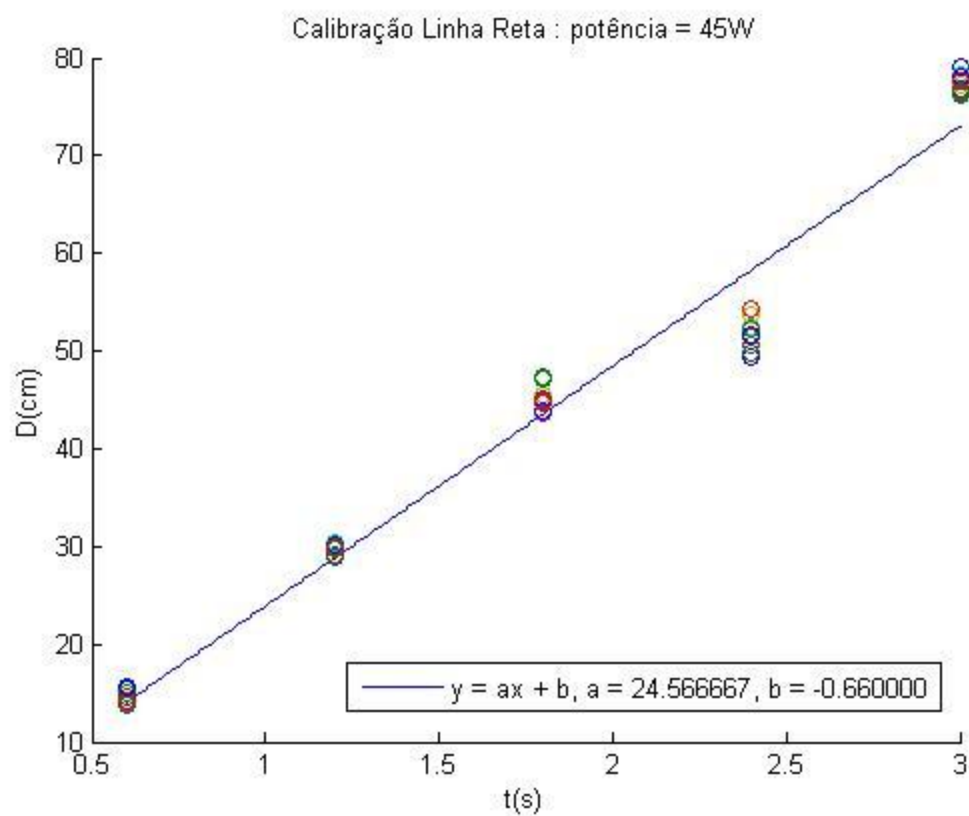
Erro de Translação

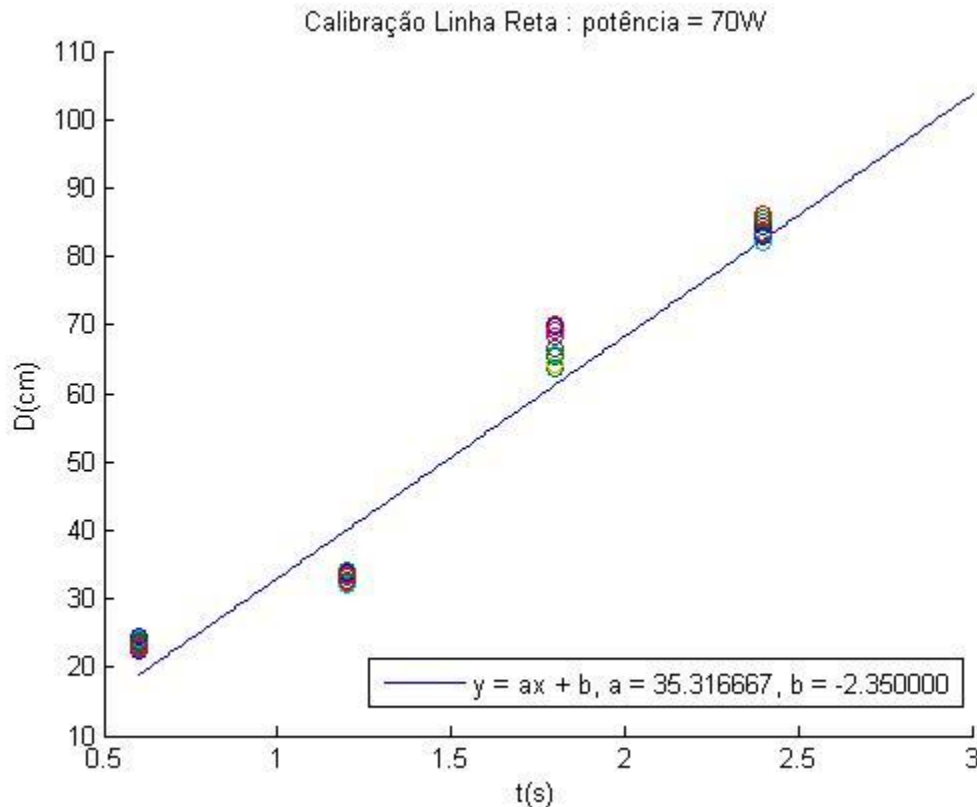
Para cada potência P deveriam ser feitos 10 testes onde seriam medidos o tempo t e a distância percorrida d com o intuito de traçarmos um gráfico $t \times d$ (t -tempo; d -distância percorrida pelo robô). Depois de uma cuidadosa observação, vimos que a handyboard registra apenas 8 potências, onde uma delas é nula. Montamos assim uma tabela para cada uma das 7 potências, variando o tempo em cada uma delas. Observamos também que em maiores potências a velocidade desenvolvida pelo robô era muito alta, o que após alguns testes acabou por prejudicar o desempenho da bateria e, por consequência, afetou nossa calibração. Para realizar a calibração, colocamos um piloto na parte traseira do robô para facilitar a coleta de dados.

Realizamos os testes variando o tempo de 0 a 3 segundos, com intervalos de 0.6 segundos, para cada uma das 7 potências. Porém, para efeito de realização das tarefas, foi priorizado o primeiro nível de potência, que apresenta a melhor velocidade para apresentação do trabalho. Assim os gráficos obtidos foram:







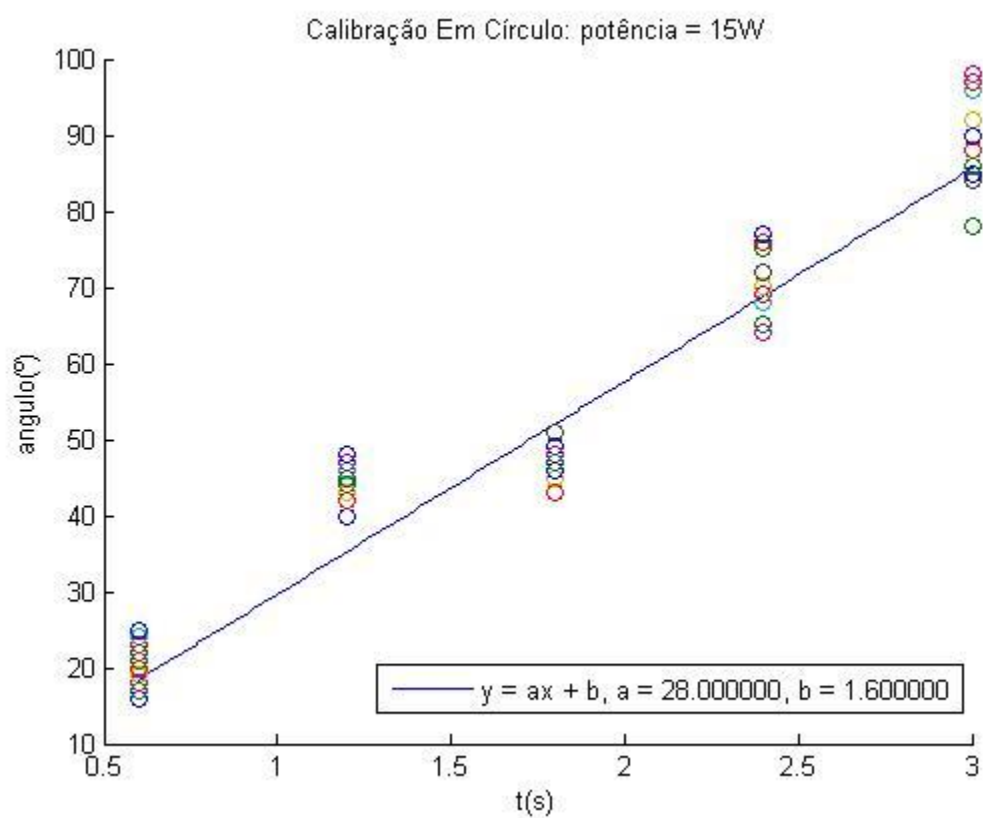
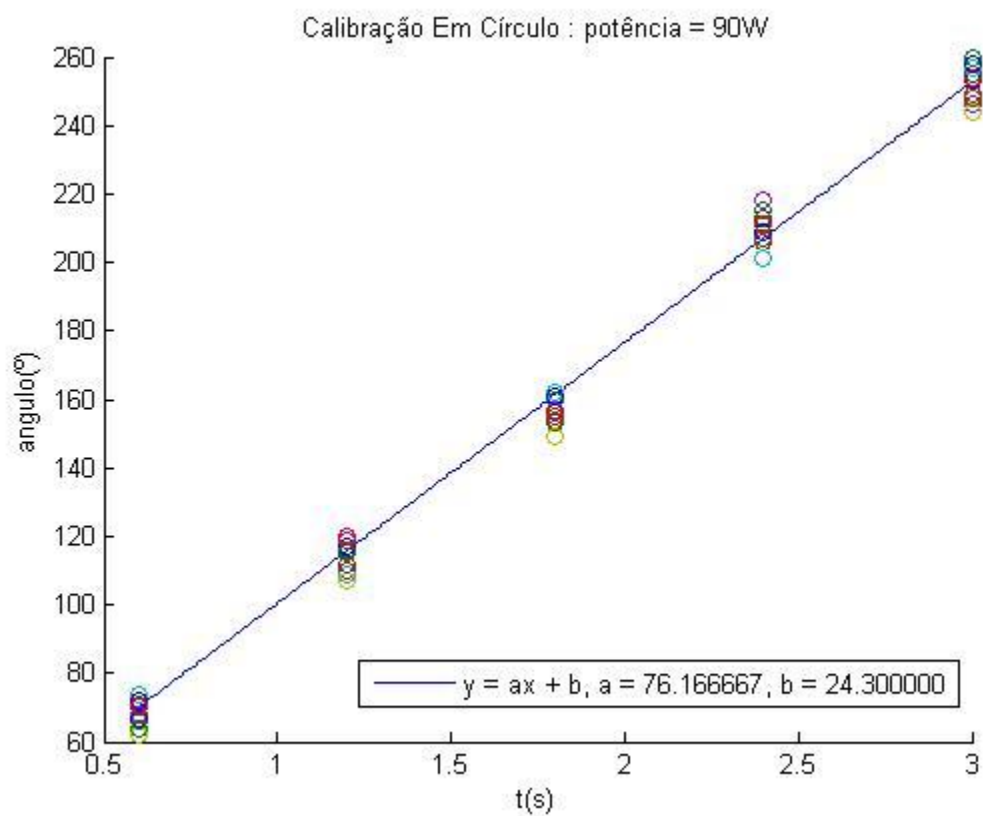


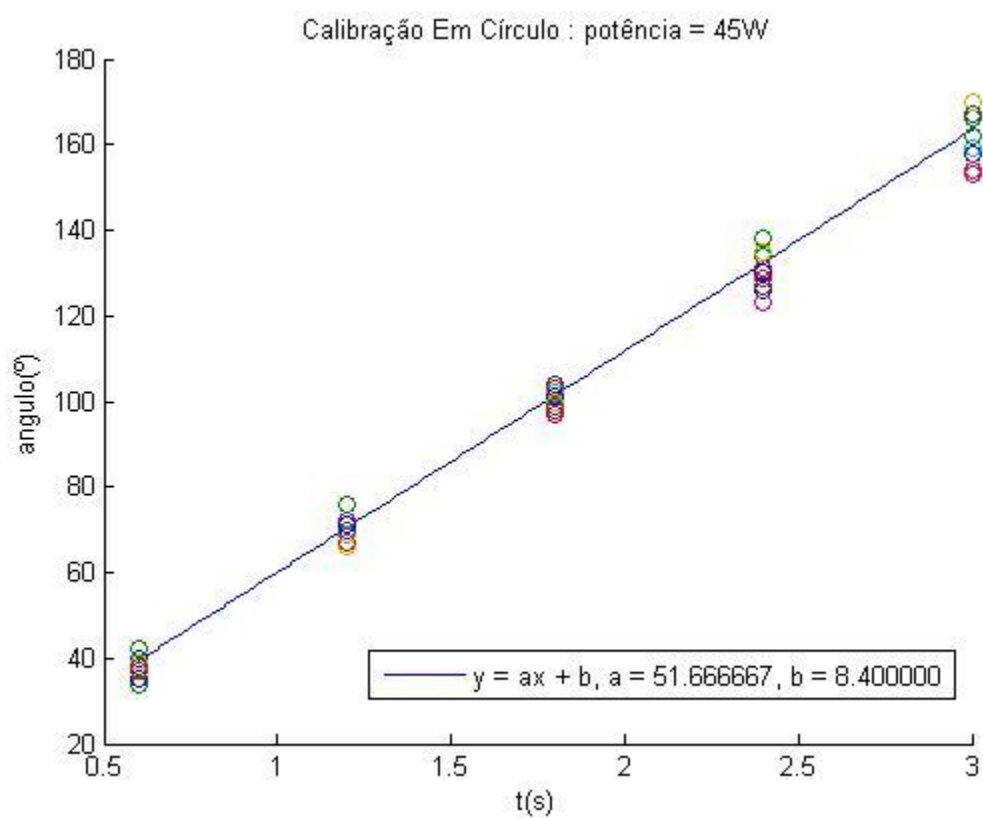
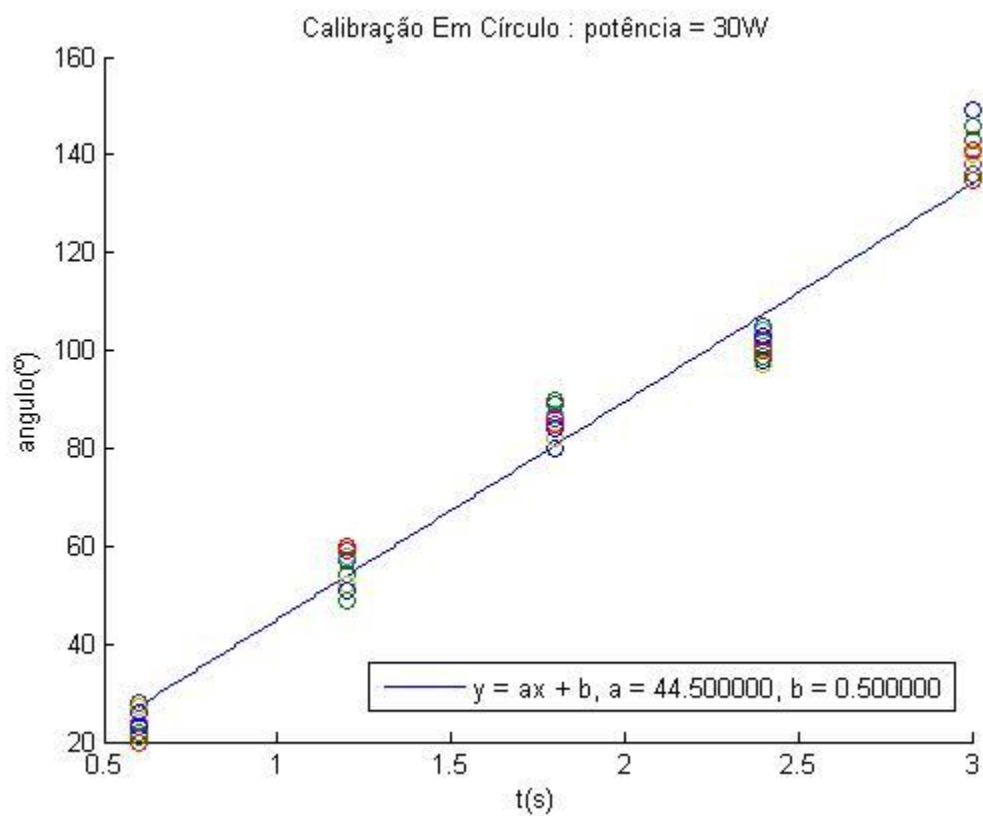
No caso em que a potência era alta, não realizamos os testes para tempos acima de 2 segundos, já que a distância percorrida (1 m) era maior que a cartolina utilizada.

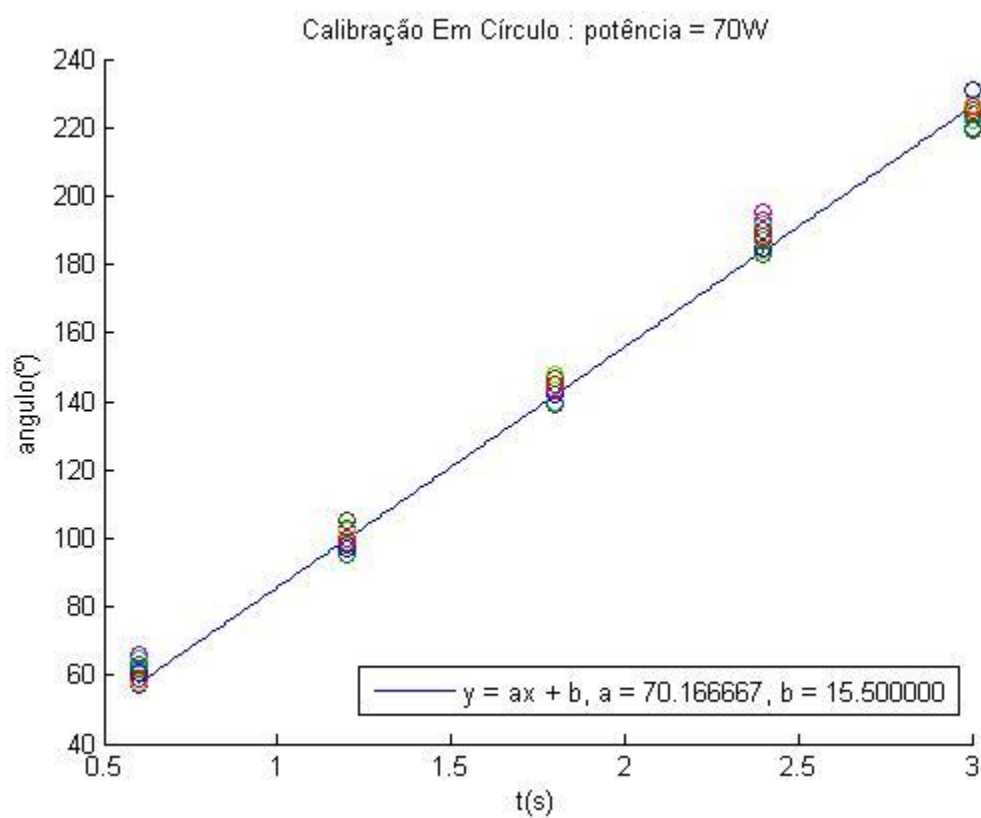
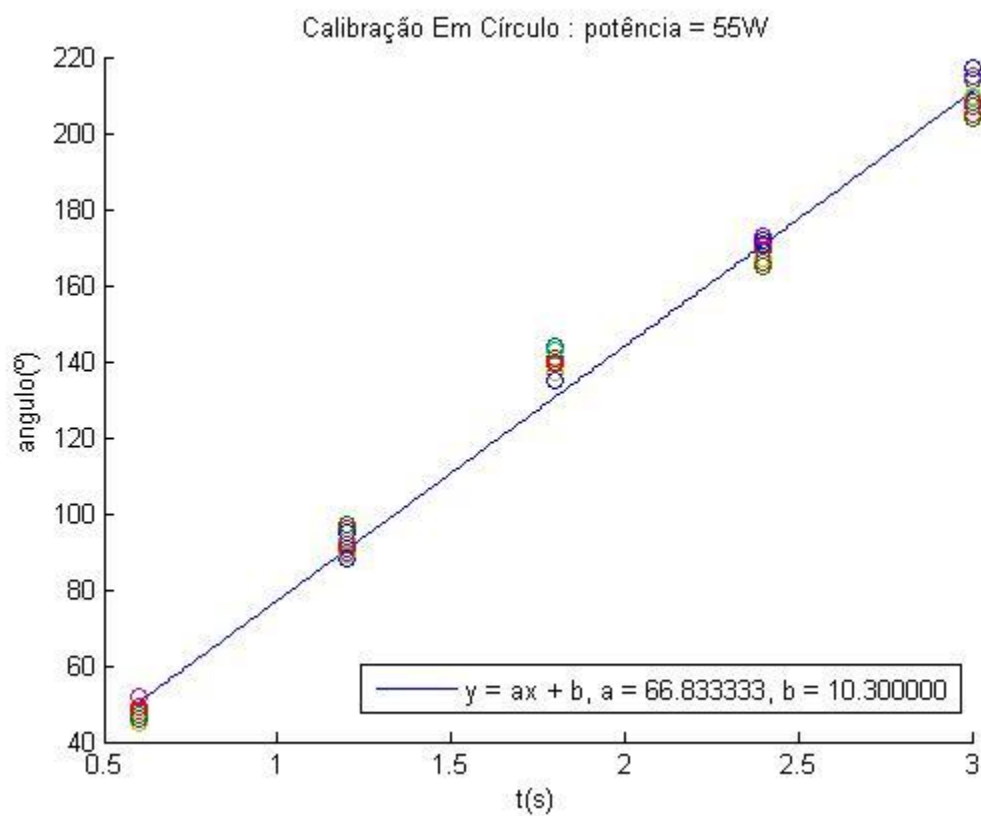
Erro de Rotação

De modo análogo ao erro de translação, para cada potência P deveriam ser feitos 10 testes onde seriam medidos o tempo t e o ângulo de rotação do robô d com o intuito de traçarmos um gráfico $t \times d$ (t -tempo; d - theta - ângulo de rotação do robô). Depois de uma cuidadosa observação, vimos que a handyboard registra apenas 8 potências, onde uma delas é nula. Montamos assim uma tabela para cada uma das 7 potências, variando o tempo em cada uma delas. Observamos também que em maiores potências a velocidade desenvolvida pelo robô era muito alta, o que após alguns testes acabou por prejudicar o desempenho da bateria e, por consequência, afetou nossa calibração. Para realizar a calibração, colocamos um piloto na parte traseira do robô para facilitar a coleta de dados.

Realizamos os testes variando o tempo de 0 a 1 segundo, com intervalos de 0.2 segundos, para cada uma das 7 potências. Porém, para efeito de realização das tarefas, foi priorizado o primeiro nível de potência, que apresenta a melhor velocidade para apresentação do trabalho. Assim os gráficos obtidos foram:







4 Utilizando a calibração

Depois de uma extensa análise dos resultados, estimamos que a relação entre o tempo e a distância é aproximadamente linear, onde o maior desvio está na derrapagem da largada. Então, partindo de uma regressão linear encontramos os coeficientes (a,b) da relação:

$D = at + b$, onde D é a distância percorrida e t é o tempo do motor.

Como no nosso robô será importante ter o tempo de atuação no motor em função da distância, temos:

$$\mathbf{t = (D - b) / a}$$

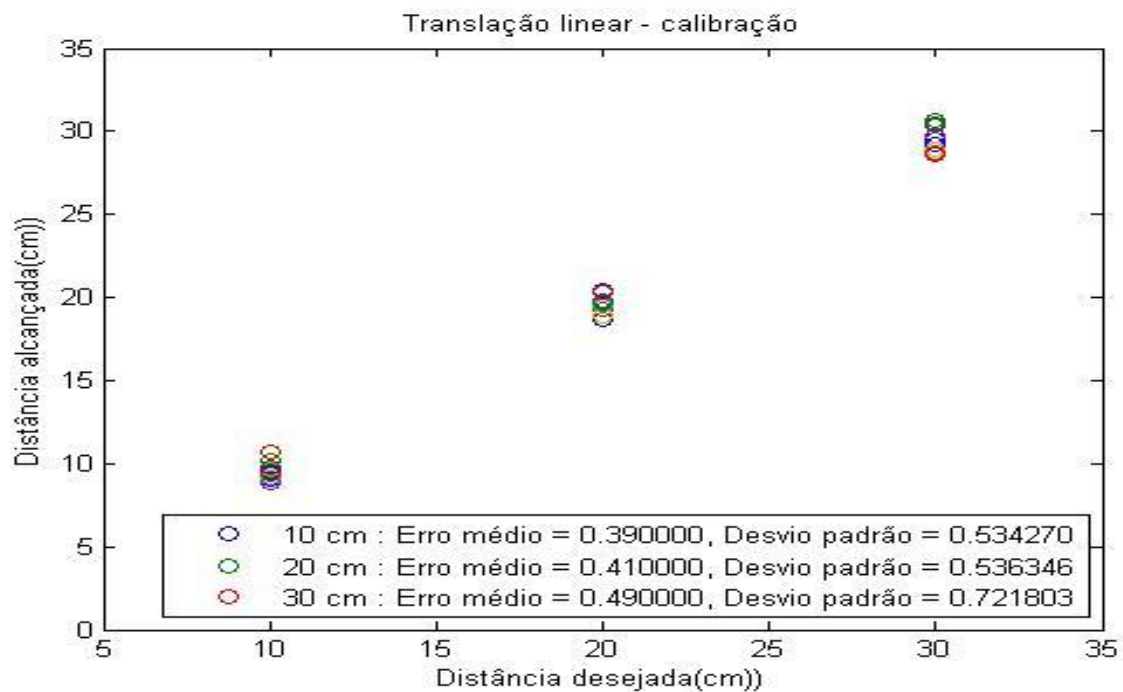
Agora que temos o tempo função da distância, basta apenas armazenar na memória do robô os coeficientes (a,b) encontrados para cada potência.

Obs: As mesmas relações foram efetuadas para os ângulos, bastando substituir D pelo ângulo.

5 Cumprindo as tarefas

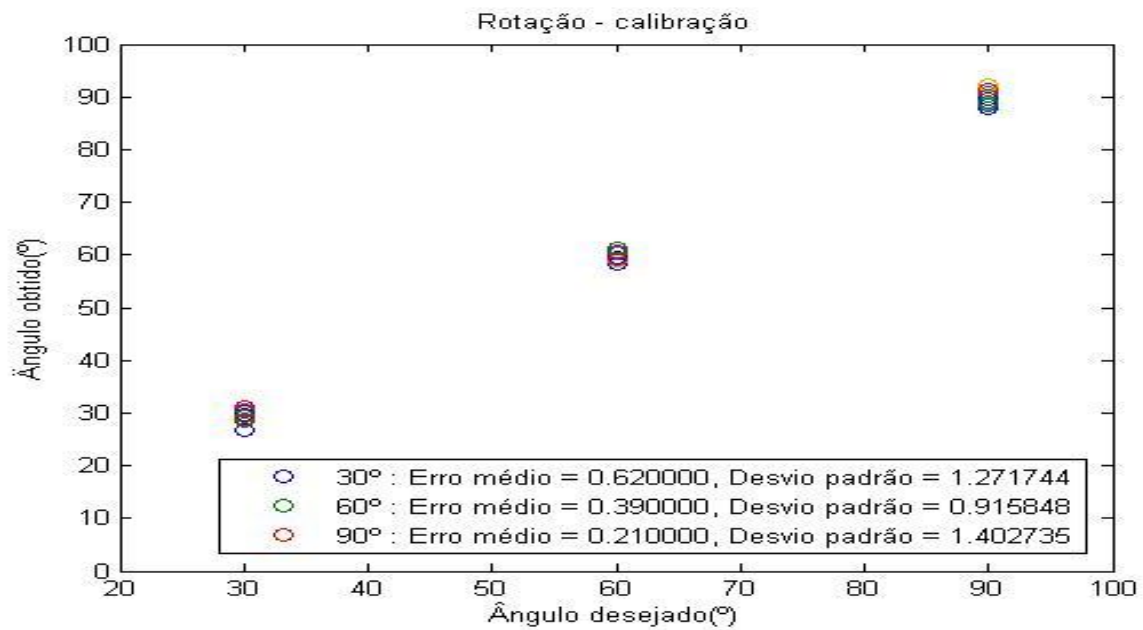
5.1- Translação Linear

Para realizar a tarefa utilizamos a tabela de calibração com o tempo em função da distância pedida para a potência mínima, em seguida anotamos os valores e montamos um gráfico com os resultados:



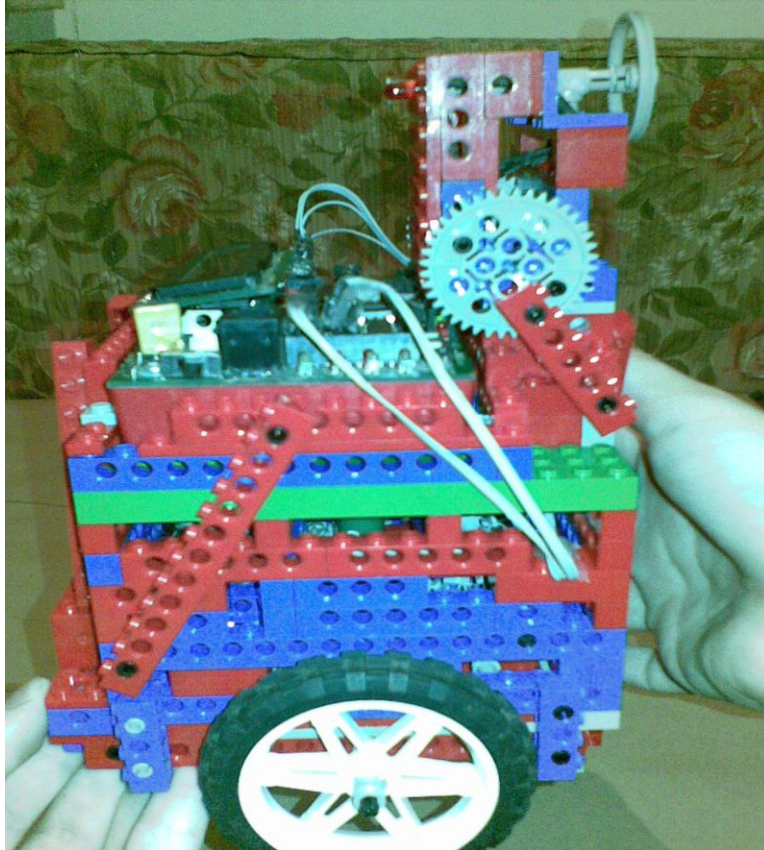
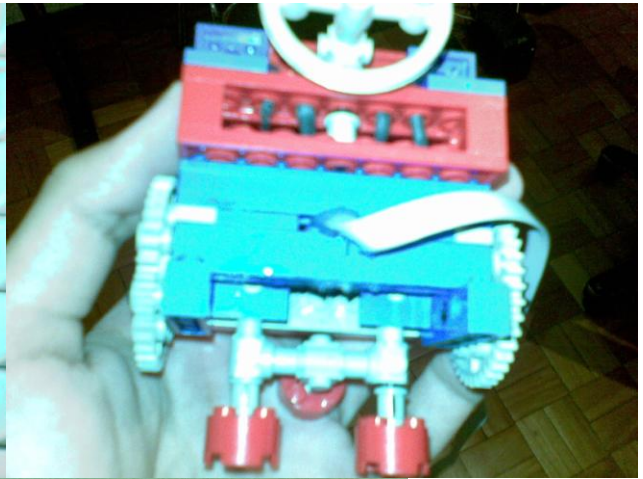
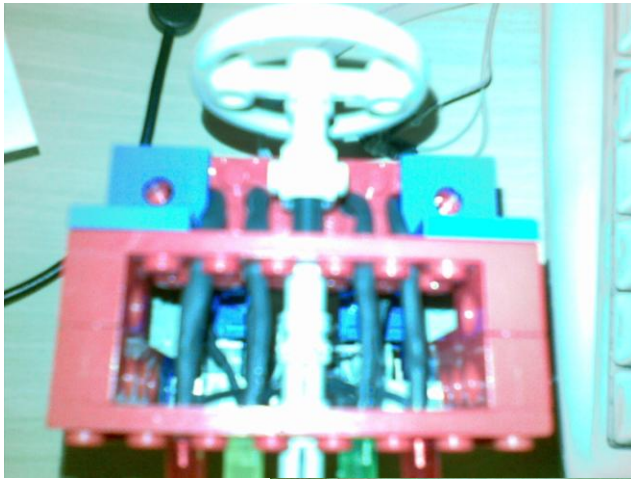
5.2- Rotação

Para realizar a tarefa utilizamos a tabela de calibração com o tempo em função do ângulo pedido para a potência mínima, em seguida anotamos os valores e montamos um gráfico com os resultados:



5.3- Led

Seguimos as instruções dadas pelo manual da handyboard e utilizamos as portas SP1. Colocamos um led em cada porta com uma resistência, em seguida ligamos todos no terra. Para fixar os leds montamos uma estrutura fechada no robô, já que tínhamos receio de que algum fio se partisse.



Na parte de programação, criamos um gerador aleatório de sequências, então cada vez que o botão start fosse pressionado uma nova sequência era gerada.

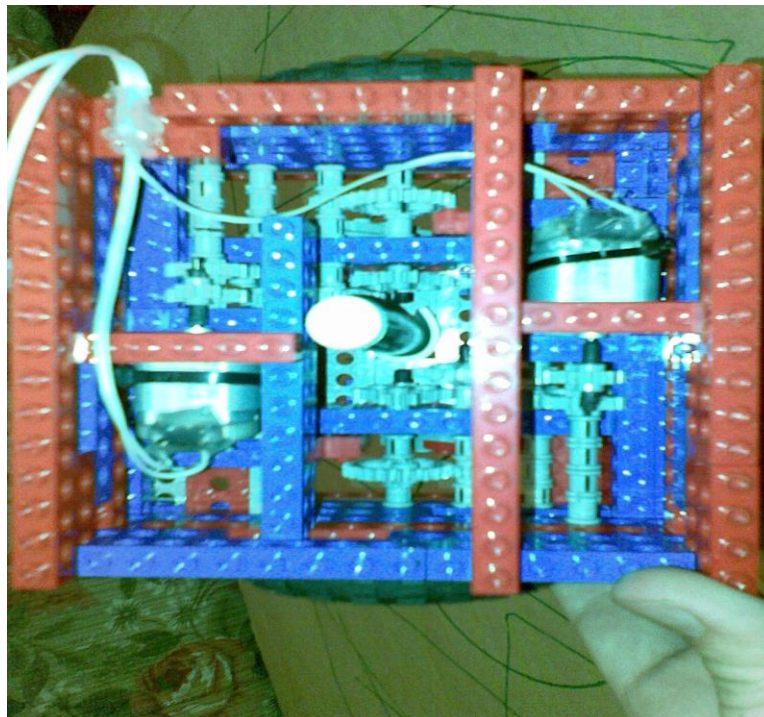
5.4- Multitarefa

Criamos duas funções simples, uma para mover os motores por tempo ilimitado e outra para fazer uma sequência com os leds e, em seguida, chamamos a função `start_process()` para cada uma destas funções. Para encerrar, utilizamos a função `sleep()` com 30 segundos seguida da função `kill_process()` para cada um dos processos iniciados. Feito isso, o programa encerra os motores e o led.

5.5- Quadrados e Circulos

Para montar os 3 quadrados bastava repetir 12 vezes o procedimento de ir reto 30 cm em seguida virar 90 graus. Porém vimos que os erros de rotação prejudicaram que o robô mantivesse a trajetória.

Para o círculo fizemos alguns cálculos simples que permitiram a escolha das diferentes potências para cada roda, a partir da foto abaixo podemos demonstrar que a caneta se encontra no centro do robô entre as duas rodas:



A partir do projeto do robô, queríamos que o seu centro percorresse por um círculo de raio 25 cm. Como a distância entre uma roda e outra é de 16,6 cm, então a roda interna deverá percorrer um círculo de raio 16,7 cm (25 - 8,3) e a roda externa de raio 33,3 cm (25 + 8,3). Sabemos que as duas rodas deverão percorrer suas respectivas circunferências no mesmo intervalo de tempo, onde essa distância percorrida por cada roda é descrita por:

$D = 2 \times \pi \times R$, onde D é a distância e R é o raio.

Como a relação é linear, a razão entre essas distâncias será igual à razão entre os raios. Então a razão entre as distâncias será:

$$\text{Dext}/\text{Dint} = \text{Rext}/\text{Rint} = 33,3 / 16,7 \approx 2 ,$$

Sabendo que o tempo do percurso de cada roda deverá ser igual, e que $D = V.T$, então temos:

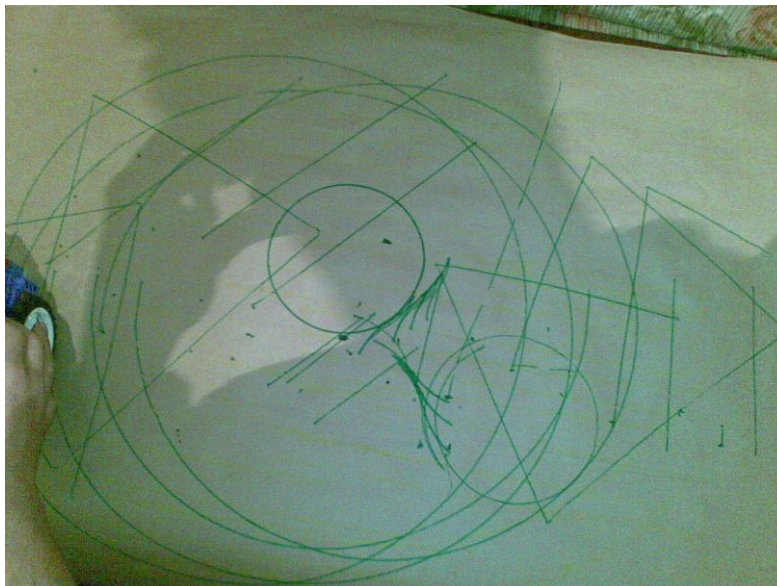
$$\text{Text} = \text{Tint} \Rightarrow \text{Dext}/\text{Vext} = \text{Dint}/\text{Vint} \Rightarrow \text{Dext} / \text{Dint} = \text{Vext}/\text{Vint} = 2 ,$$

Concluimos então que a velocidade de uma roda deverá ser igual ao dobro da outra. Analizando as regressões lineares podemos notar que o coeficiente (a) da equação $d=at+b$ seria o equivalente à velocidade; se observarmos a tabela de calibração feita inicialmente temos:

Potência 15 -- $a = 15.000000$

Potência 55 -- $a = 30.950000$

Como uma velocidade é aproximadamente o dobro da outra, que era o que estávamos buscando, temos então uma boa possibilidade para cumprir a tarefa. Na prática conseguimos um círculo com raio próximo de 25 cm, porém foi difícil se manter na mesma rota.



4 Conclusões

Os objetivos do trabalho foram concluídos apesar das dificuldades encontradas como, por exemplo, na construção do robô e adaptação dos motores na estrutura LEGO. Além disso, foi uma experiência válida para a constatação de que existe uma grande diferença de resposta entre os atuadores, neste caso os motores, que mesmo sendo aparentemente iguais exigiram tempo e esforço para que fossem satisfatoriamente calibrados.

5 Referências

- [1] Fred G. Martin. The art of lego design. 1995.
- [2] Fred G. Martin. The Handy Board Technical Reference. MIT Media Laboratory, 1998.