

los em valores com unidades utilizáveis no sistema. Isto requer o ajuste da posição zero (*offset*) dos sensores e escala.

O ajuste do zero é simples e igual para o acelerômetro e giroscópio, necessitando que apenas seja verificado o valor de leitura dos sensores na posição em que o veículo deve-se manter em equilíbrio. Ao contrário do *offset*, a escala deve ser ajustada de acordo com as propriedades de cada sensor.

O sinal de saída do acelerômetro é uma tensão proporcional a aceleração no eixo a ser medido, é expressada em unidade de gravidades (g). A tensão pode então ser convertida em um sinal digital de 10 *bits* pelo conversor ADC do Arduino. Ajustando o fator de escala, este valor pode ser convertido para o valor de aceleração correspondente. o ângulo de inclinação θ_{accel} do veículo pode ser estimado utilizando trigonometria:

$$\theta_{accel} = \text{sen}^{-1} \left(\frac{V_{iacc} - V_{offset}}{\text{sensitividade}} \right) \quad (4)$$

Onde:

V_{iacc} → Tensão de leitura do acelerômetro;

V_{offset} → Tensão de *offset* do acelerômetro;

sensitividade → Sensitividade do acelerômetro conforme *datasheet* em *mV/g*.

Os ângulos mais importantes a serem medidos são os relacionados ao eixo vertical relativo a base do veículo. Pode-se assumir que se um ângulo acima de 30° for atingido, o controlador terá dificuldade para retornar a plataforma a posição vertical. Baseado nesta hipótese, há a possibilidade de se aplicar a aproximação para ângulos pequenos, que tem um erro de 1% a cada 14° e pode ser representado pela equação abaixo:

$$\text{sen}\theta_{accel} \simeq \theta_{accel} \quad (5)$$

O sinal de saída do giroscópio é uma tensão proporcional ao valor da velocidade angular em $^{\circ}/s$. Esta tensão é então convertida para um valor de 10 bits pelo ADC. A conversão do dado obtido da leitura para o valor correspondente a velocidade angular é similar ao caso do acelerômetro, porém o valor de *offset* é a tensão de saída do giroscópio no estado estacionário. A expressão para cálculo da velocidade angular pode ser descrita como:

$$\dot{\theta} = \frac{V_{giro} - V_{offset}}{sensitividade} \quad (6)$$

Onde:

$\dot{\theta}$ → Velocidade angular;

V_{giro} → Tensão de leitura do giroscópio;

V_{offset} → Tensão de *offset* do giroscópio;

sensitividade → Sensitividade do giroscópio conforme *datasheet* em $mV/^\circ/s$.

Para se obter o ângulo de inclinação do veículo através da velocidade angular calculada em (10) é necessário que esta seja integrada em relação ao tempo de medida.

3.3.2. FILTRO DE KALMAN

O problema conforme explicado anteriormente é que o acelerômetro gera muito ruído na sua leitura e o giroscópio tende a perder precisão ao longo do tempo. O Filtro de Kalman é um algoritmo que utiliza uma série de medições ruidosas dos sensores ao longo do tempo e a partir delas tenta estimar o estado do sistema baseando-se nos estados atual e anterior.

O estado do sistema no tempo k é dado como (FERDINANDO ET AL., 2012):

$$x_k = Ax_{k-1} + Bu_k + w_k \quad (7)$$

Os sinais de saída do filtro serão o ângulo θ e também o *bias* $\dot{\theta}_b$ baseados nas medições do acelerômetro e do giroscópio. Eles são definidos na matriz de estados x_k :

$$x_k = \begin{bmatrix} \theta_{accel} \\ \dot{\theta}_b \end{bmatrix}_k \quad (8)$$

Os sinais de saída do filtro serão o ângulo θ e também o *bias* $\dot{\theta}_b$ baseados nas medições do acelerômetro e do giroscópio.

A matriz de transição de estados A , que é aplicada ao estado anterior x_{k-1} é definida como:

$$A = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \quad (9)$$

A matriz B é chamada de modelo das entradas de controle e é nela que aplica o vetor de controle u_k . Neste caso u_k é a medição do giroscópio $\dot{\theta}_{giro}$ no tempo k em graus por segundo ($^{\circ}/s$) e a matriz de controle:

$$B = \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \quad (10)$$

O estado do sistema, substituindo-se as matrizes se define então como:

$$\begin{bmatrix} \theta_{acel} \\ \dot{\theta}_b \end{bmatrix}_k = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_{acel} \\ \dot{\theta}_b \end{bmatrix}_{k-1} + \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \dot{\theta}_{giro} + \omega_k \quad (11)$$

O observador z_k do estado x_k é descrito por:

$$z_k = Hx_k + v_k \quad (12)$$

Como pode-se observar a leitura z_k é dada pelo estado atual x_k multiplicada pela matriz H e somada ao ruído v_k do observador. A matriz H é representada como:

$$H = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (13)$$

O ruído v_k deve ser gaussiano, com média zero e covariância R . Para o veículo auto-equilibrante assumiu-se que ele é o mesmo e independente do tempo k .

Com o modelo do sistema dinâmico definido partiu-se então para a aplicação das equações do Filtro de Kalman para estimar o estado do sistema no tempo x_k .

As duas equações seguintes fazem parte da fase de predição do Filtro de Kalman. A primeira tenta estimar o estado atual baseando-se nos estados anteriores e a leitura do giroscópio. A segunda tenta estimar a matriz de covariância do erro à priori $P_{k|k-1}$ baseando-se na matriz de covariância do erro anterior $P_{k-1|k-1}$.

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + B\dot{\theta}_k \quad (14)$$

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q_k \quad (15)$$

As próximas equações são pertencentes a fase de atualização do Filtro de Kalman. Primeiro é calculada a diferença entre a medição z_k e o estado à priori $x_{k|k-1}$:

$$\tilde{y}_k = z_k - H\hat{x}_{k|k-1} \quad (16)$$

Em seguida é calculado o resíduo da covariância (Equação 17) e a partir dela o ganho de Kalman (Equação 18):

$$S_k = HP_{k|k-1}H^T + R \quad (17)$$

$$K_k = P_{k|k-1}H^T S_k^{-1} \quad (18)$$

Por último, atualiza-se a estimativa a posteriori do estado atual através da equação 19 e o erro a posteriori da matriz de covariância através da equação 20:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\tilde{y}_k \quad (19)$$

$$P_{k|k} = (I - K_kH)P_{k|k-1} \quad (20)$$

3.3.3. CONTROLE DOS MOTORES DC

Dentre as opções de controle disponíveis no módulo *Sabertooth*, foi optado pelo modo de operação serial simplificado. Neste modo o processador envia os comandos para o controlador dos motores por comunicação serial. Ele necessita somente de um comando com um *byte* para ajustar a velocidade e direção de ambos os motores. Isto torna fácil a interface do controlador com o processador, pois não há a necessidade de se implementar um protocolo de comunicação baseado em pacotes.

Como explicado anteriormente, a escolha de modo de funcionamento é efetuada alterando a posição do conjunto de seis botões do *DIP switch* disponível na placa. Seguindo a informação disponível no *datasheet* do componente, o modo de funcionamento pretendido é estabelecido pela combinação de posições das chaves conforme a figura 12:

Devido ao *Sabertooth* controlar dois motores com um caractere de 8 *bits* quando operando no modo Serial simplificado, cada motor tem 7 bits de resolução. O envio de um

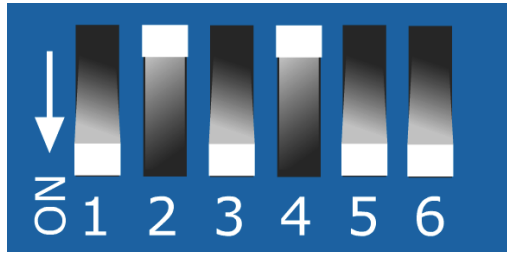


Figura 12: Posição das chaves do *DIP switch* para o modo de operação Serial Simplificado

caractere entre os valores de 1 a 127 controla o motor 1. Onde 1 ajusta a rotação completamente para trás, na posição neutra 64 o motor se mantém parado e, 127 ele se move completamente para a frente. O mesmo se aplica para o motor 2, porém este opera com caracteres entre 128 e 255. O caractere 0 tem uma função de botão de emergência, o seu envio implica no desligamento de ambos os motores.

3.3.4. CONTROLE DE INCLINAÇÃO

Com uma das partes mais críticas concluídas, que é a posição angular do veículo no espaço pode-se partir para o estudo do sistema de controle de inclinação e locomoção do veículo. Conforme mostrado anteriormente, se o veículo está inclinado, as rodas devem locomover-se para direção da inclinação e quanto maior o ângulo de inclinação, maior deve ser a resposta das rodas.

Através da análise do sistema chega-se a seguinte conclusão quanto ao seu funcionamento:

- A variável de entrada é o ângulo da plataforma
- A variável de saída é a velocidade dos motores
- Tenta-se manter a variável de entrada igual ao *setpoint* (ângulo que o veículo se mantém estável) através do controle da variável de saída

Devido ao comportamento do sistema, um simples impulso aplicado na malha aberta implicaria no aumento do ângulo de inclinação e posição do veículo e o resultado seria a queda do veículo.

Como a planta a ser controlada é uma relação entre a entrada e a saída do sistema, foi decidido que para validar o projeto seria utilizado um controlador PID visto que para a sua

aplicação não há a necessidade do conhecimento do modelo matemático do sistema a ser controlado. Após implementada a lei de controle, exemplificada na equação abaixo, basta ajustar as variáveis relacionadas aos ganhos proporcional, integral e derivativo diretamente na planta e observar a sua resposta.

$$Saida = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (21)$$

Onde $e(t)$ é o erro entre o *setpoint* e o ângulo atual do veículo e K_p , K_i e K_d os ganhos a serem ajustados.

3.3.5. DESENVOLVIMENTO DA ROTINA DE CONTROLE

Após o estudo de todos os componentes necessários para o funcionamento correto do veículo, pode-se construir o diagrama de blocos completo da sua malha de controle:

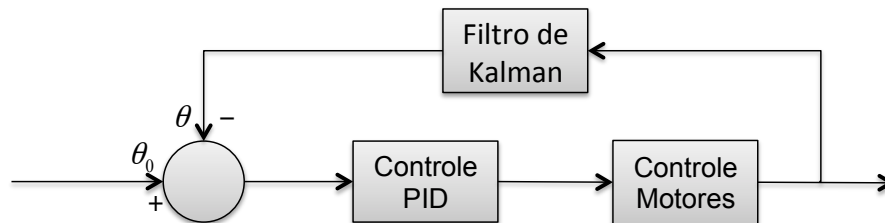


Figura 13: Diagrama de blocos do sistema de controle do veículo auto-equilibrante

O *setpoint* do sistema θ_0 é definido pela posição que a base se mantém na horizontal.

Definida a estratégia de controle a ser aplicada no veículo auto-equilibrante, iniciou-se o processo de transcrição do sistema projetado para o algoritmo suportado pelo Arduino, que é baseado na linguagem de programação C/C++.

3.4. MONTAGEM FINAL

3.4.1. Integração do sistema

Com os componentes necessários comprados e o projeto mecânico concluído, iniciou-se a construção física do veículo. O primeiro passo foi construir o projeto mecânico, trabalhando a matéria prima e posicionar os componentes especificados conforme indicado nos desenhos técnicos.

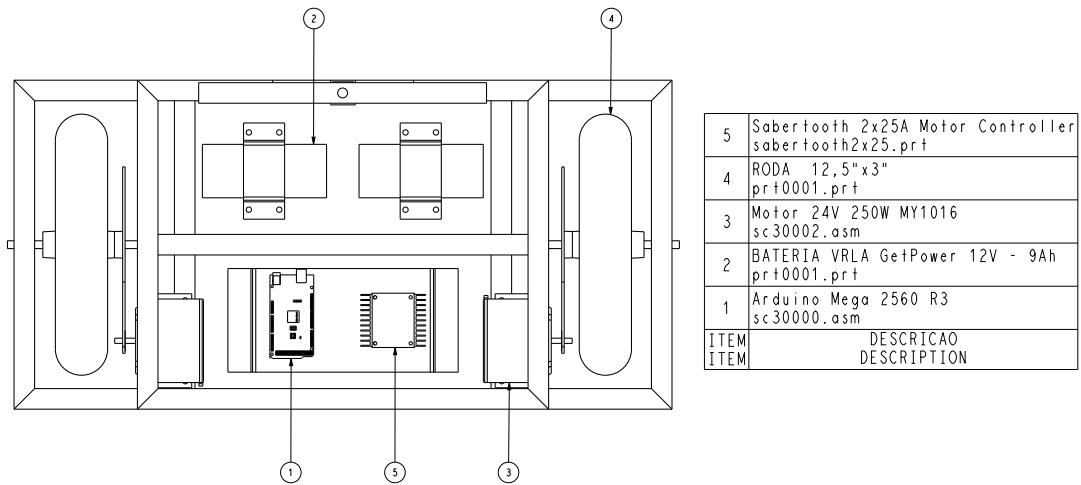


Figura 14: Distribuição dos principais componentes na estrutura do veículo

O segundo passo para integrar o sistema foi a montagem do projeto elétrico para o funcionamento do veículo. O diagrama elétrico conforme figura 15 demonstra as conexões dos principais componentes:

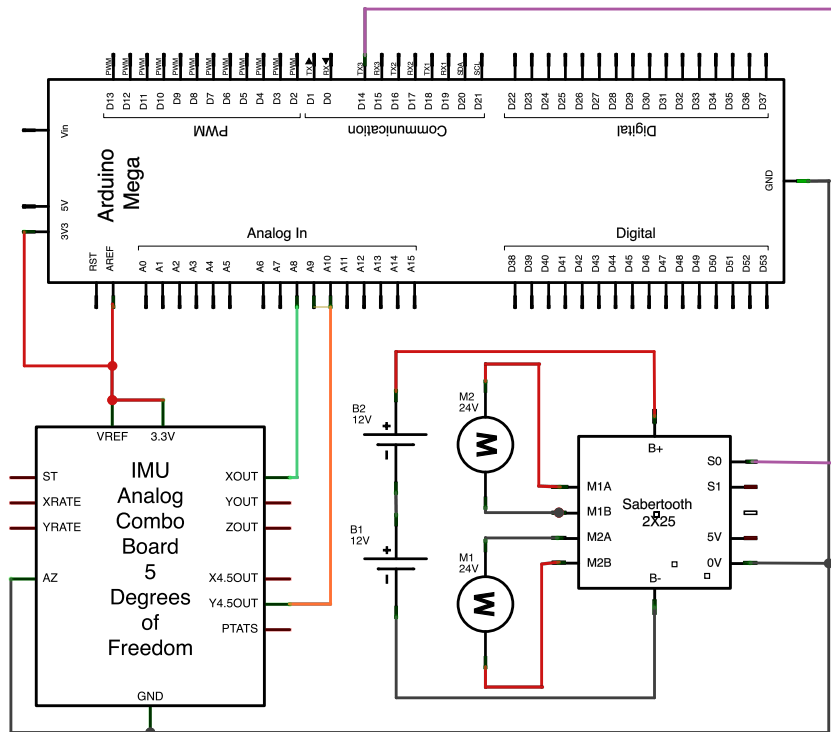


Figura 15: Diagrama elétrico do veículo

3.4.2. Validação

O primeiro passo para validação do veículo é a realização dos testes dos sinais obtidos do acelerômetro e giroscópio, para isto foi desenvolvida uma rotina no processador para captura das medições dos sensores e através de comunicação serial com o computador, verificar suas leituras. A implementação do Filtro de Kalman e do sistema de controle também foram testadas por este mesmo método.

O segundo passo é a implementação do programa final de controle na montagem completa do veículo. A rotina conta com a aplicação do Filtro de Kalman, ajuste do controle PID e implementação das funções de movimento dos motores.

4. RESULTADOS E DISCUSSÕES

A figura 16 mostra o protótipo finalizado:

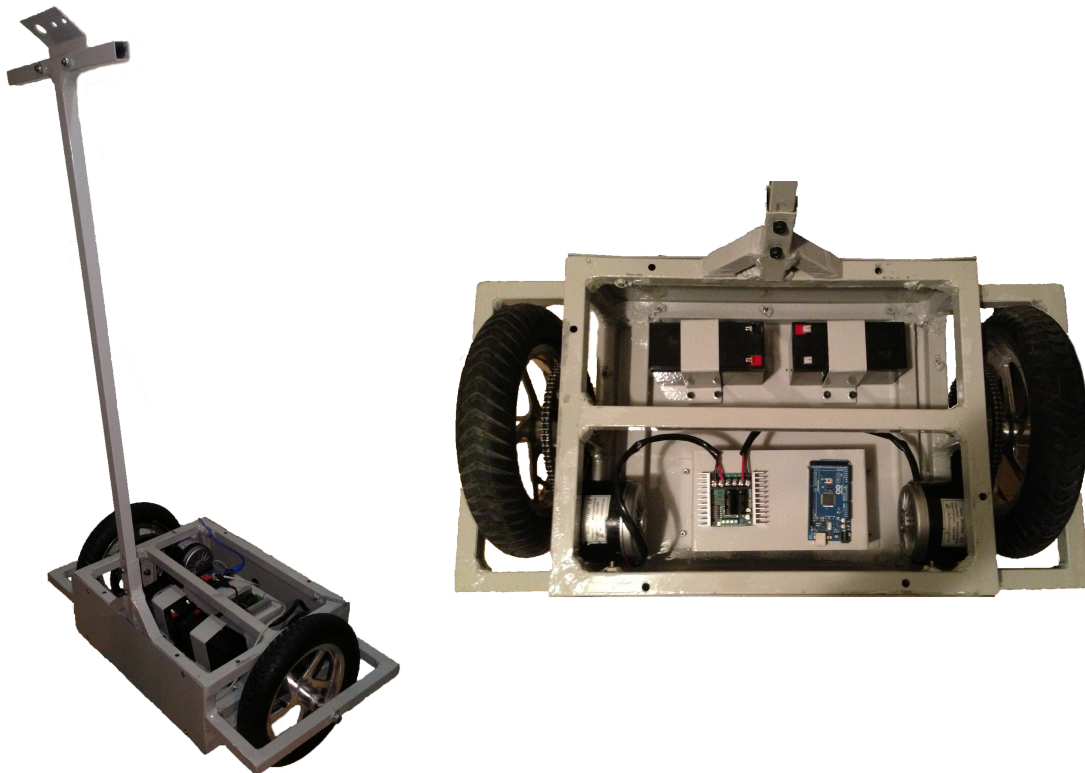


Figura 16: Protótipo finalizado

A tabela 1 descreve os principais custos para montagem do protótipo. Os valores em Dólar dos produtos importados foram convertidos para Real na taxa de câmbio conforme cotação na data de compra.

Tabela 1: Custos para construção do veículo

Descrição	Preço Unitário	QTD	Custo total (R\$)
Arduino Mega 2560 R3	\$ 58,95	1	R\$ 121,44
IMU Combo Board - 5 Deg IDG500/ADXL335	\$ 35,95	1	R\$ 74,06
Motor 24V 250W MY1016	\$ 26,80	2	R\$ 110,42
Sabertooth 2x25A Motor Controller	\$ 124,99	1	R\$ 257,48
Bateria VRLA GetPower 12V - 9Ah	\$ 69,90	2	R\$ 139,80
Roda 12,5"	\$ 47,99	2	R\$ 197,72
Estrutura metálica	R\$ 700,00	1	R\$ 700,00
Demais componentes elétricos	R\$ 150,00	1	R\$ 150,00
			R\$ 1.750,91

A construção do veículo foi realizada conforme o projeto e testes foram realizados para verificação da eficiência nos movimentos mecânicos. O veículo demonstrou bom deslocamento em movimentos para frente e para trás, entretanto durante os experimentos do controle de balanço, foi percebida certa dificuldade para o mesmo se manter no centro de gravidade. Isto foi facilmente corrigido realocando as baterias em uma posição mais próxima do eixo de rotação das rodas.

Os eixos para monitoramento do ângulo de inclinação e taxa de rotação dependem de como a *breakout board* dos sensores foi posicionada no veículo. Então do modo que o veículo foi projetado foram utilizados o eixo X do acelerômetro e o eixo Y do giroscópio para esta aplicação.

Para que o Arduino capture as medições dos sensores é necessário somente que as portas dos eixos a serem utilizados sejam conectados nas entradas analógicas do microprocessador. A comunicação serial entre o *Sabertooth* e o controlador também é muito simples, bastando conectar a porta TX do microprocessador na porta S1 do *driver* dos motores. A implementação das funções para curva do veículo foram feitas através do incremento do ganho aplicado para cada motor de acordo com a direção a ser aplicada.

4.1. VERIFICAÇÃO DOS SINAIS DOS SENSORES E FILTRO DE KALMAN

Para verificação das equações para conversão dos sinais analógicos dos sensores e filtro de Kalman, o veículo em operação foi conectado ao computador através da porta serial do Arduino e foi criada uma função para o envio das medições capturadas. A figura 17 representa graficamente os dados obtidos:

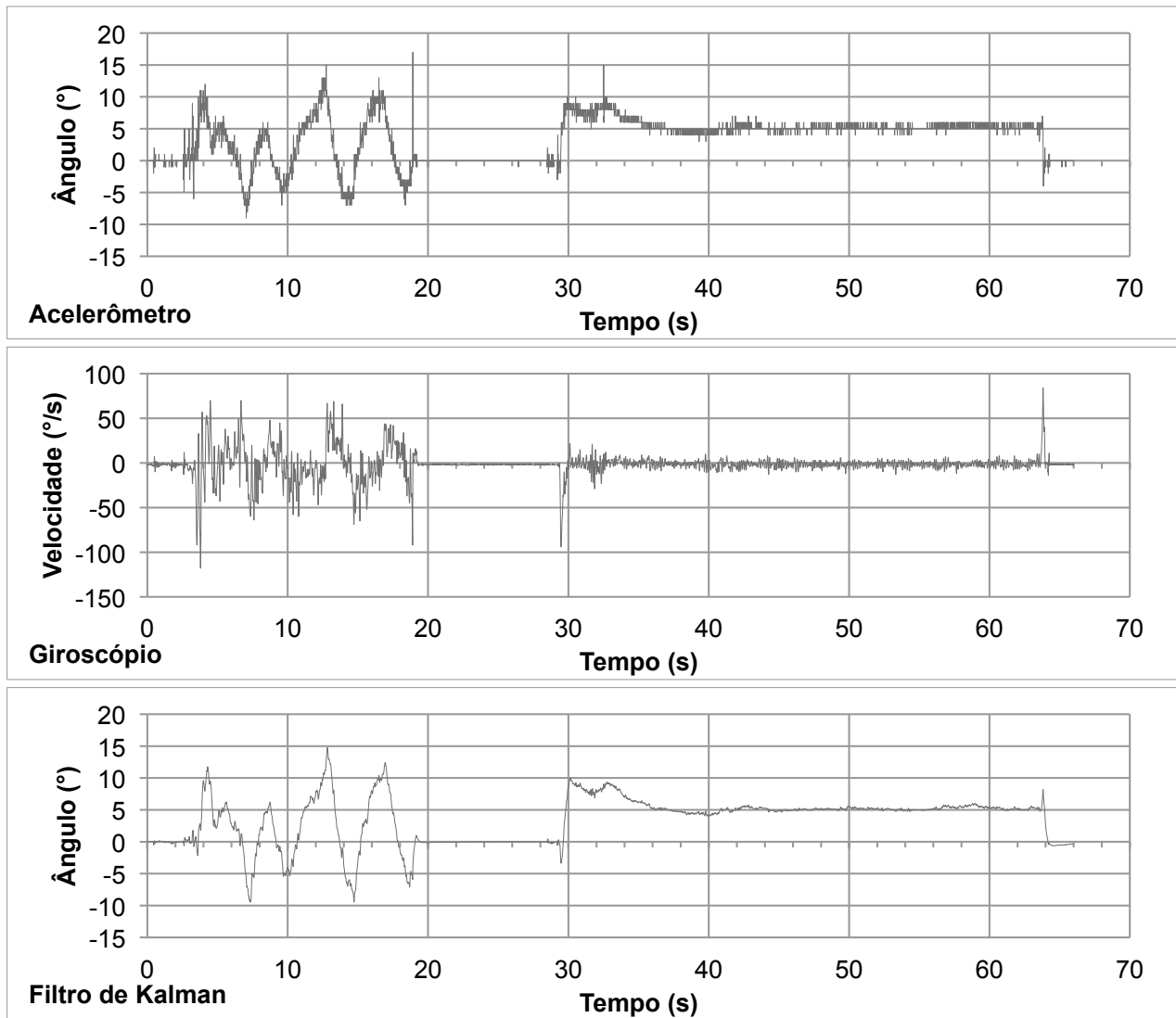


Figura 17: Medição dos sensores e do Filtro de Kalman

Analisando os dois primeiros gráficos, pode-se observar o comportamento ruído do acelerômetro devido a vibração do veículo e movimento, e a tendência ao *drift* do giroscópio quando em repouso. O filtro de Kalman, apresentado no terceiro gráfico, demonstra a suavização do ruído do acelerômetro e praticamente eliminação o erro do giroscópio.

4.2. SINTONIZAÇÃO DO CONTROLE DE EQUILÍBRIO DO VEÍCULO

Após os testes dos sinais de sensores, calibração das suas medições e validação do funcionamento do Filtro de Kalman foram iniciados os testes para o objetivo final do trabalho que é manter o veículo em equilíbrio sem interferência externa. O método mais

simples que poderia ter sido empregado seria tentar rotacionar as rodas na direção de queda do veículo até que o ângulo de inclinação atinja o valor que o veículo se equilibre. Este é efetivamente o método mais simples de implementação de um controle PID com os termos derivativo e integral zerados. Na prática isto não se mostrou muito eficiente e foi necessária a implementação e sintonização do controle PID.

Foi verificado que não houve necessidade da aplicação do termo integral do controlador, somente aplicação dos ganhos proporcional e derivativo. Os ganhos do controlador PD foram sintonizados manualmente pelo método de Ziegler–Nichols descrito abaixo:

- Primeiro todos os ganhos foram sintonizados em zero;
- O segundo passo foi incrementar o ganho proporcional até que os motores girassem as rodas na direção que o veículo estava caindo e começar a oscilar;
- O terceiro passo foi reduzir o ganho proporcional para que o veículo ficasse um pouco abaixo do ponto de oscilação;
- O último passo foi incrementar o ganho derivativo para amortecer a oscilação até que o veículo se auto-equilibrasse.

Após a sintonização pode-se observar a influência de cada ganho para que o veículo se equilibre:

- Proporcional - Ação corretiva proporcional ao ângulo do veículo. Funciona como a constante da mola de um sistema, onde é aplicada uma força na medida que o veículo se move na vertical.
- Derivativo - Ação corretiva com escala proporcional a taxa de variação do ângulo, ou seja, a sua velocidade angular. Esta ação é similar ao coeficiente de amortecimento de um sistema aplicando uma força que resiste à rotação em ambas as direções.

Os ganhos obtidos foram sintonizados para o veículo sem carga alguma na base, ou seja, a potência utilizada dos motores para que o veículo se equilibre e o torque aplicado foram bem baixos. Para o transporte de pessoas ou objetos pesados há a necessidade de se aumentar os ganhos proporcional e derivativo de acordo com a carga a ser aplicada na base do veículo.

5. CONCLUSÃO

O objetivo deste trabalho foi a construção de um protótipo funcional de um veículo auto-equilibrante. Pode-se verificar através da aplicação prática que com os materiais utilizados, o veículo conseguiu se manter em equilíbrio sem interferência externa.

O sucesso desse tipo de veículo está relacionado diretamente com o controle que se tem sobre a inclinação da plataforma. Devido a estimação adequada do ângulo através do Filtro de Kalman, a sintonia do controle proporcional derivativo aplicado foi simples de ser efetivada. Em funcionamento a sua aplicação se comportou como se adicionássemos uma mola e um amortecedor virtuais e ajustáveis no veículo.

Uma vez que o conceito do protótipo é funcional, pode-se partir para o estudo de outras técnicas para apuração do ângulo da plataforma, também a possibilidade de se desenvolver diferentes leis de controle e fazer um estudo comparativo abordando vantagens e desvantagens de cada caso.

Agradecimentos

Agradeço aos professores e colegas por todo o suporte e motivação dados no desenvolvimento do trabalho, aos meus amigos e principalmente à minha família pelo apoio incondicional.

6. REFERÊNCIAS BIBLIOGRÁFICAS

SEGWAY BRASIL. Disponível em: <<http://www.segwaybrasil.com.br/historia.html>>. Acesso em: 25 de março de 2013.

YEONGYUN, Kim, Se-Han. Lee, D.H. Kim. **Dynamic equations of a wheeled inverted pendulum with changing its center of gravity**. In Control, Automation and Systems (IC-CAS), 2011 11th International Conference on, pages 853–854. IEEE, 2011.

OUYANG, L., Du, B., Peng, A., Ou, Y.. **Design of a Wheeled Inverted Pendulum as a platform for learning based control** (pp. 418–421). Presented at the Information and Automation (ICIA), 2012.

CARDOZO, Gary; VERA, Luis. Prototype for a Self-Balanced Personal Transporter. **2012 Workshop on Engineering Applications**, May 2012.

BARSHAN, B. B., Durrant-Whyte, H. F.. **Inertial navigation systems for mobile robots. Robotics and Automation**, IEEE Transactions on, 11(3), 328–342, 1995.

SABATELLI, S.; Sechi, F.; Fanucci, L.; Rocchi, A. **A sensor fusion algorithm for an integrated angular position estimation with inertial measurement units** . Presented at the Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011. doi:10.1109/DATE.2011.5763273

FERDINANDO, H.; Khoswanto, H.; Purwanto, D.. **Embedded Kalman Filter for Inertial Measurement Unit (IMU) on the ATmega8535**. Innovations in Intelligent Systems and Applications (INISTA), 2012 International Symposium on, 1–5. doi:10.1109/INISTA.2012.6246978

WALCHKO, K. J.; Mason, P.. **Inertial navigation**. Presented at the Proceedings of Florida Conference. Florida, 2012

DIMENSIONENGINEERING, **A beginner's guide to accelerometers**. Disponível em: <<http://www.dimensionengineering.com/info/accelerometers>>. Acesso em: 02 de abril de 2013.

BISHOP, G., & Welch, G. (2001). **An introduction to the kalman filter**. Proc of SIGGRAPH, Course, 8, 27599–23175.

LEE, H.-J. J.;Jung, S. (2009). **Gyro sensor drift compensation by Kalman filter to control a mobile inverted pendulum robot system (pp. 1–6)**. Presented at the Industrial Technology, 2009. ICIT 2009. IEEE International Conference on. doi:10.1109/ICIT.2009.

ARDUINO. **Getting Started with Arduino**. Disponível em: <<http://www.arduino.cc/>>. Acesso em: 25 de março de 2013.