CRIAÇÃO E IMPLEMENTAÇÃO DO FIRMWARE DA PLACA PRINCIPAL PARA FUTEBOL DE ROBOS

Danilo Pilotto[1], Prof. Dr. Flávio Tonidandel[3]
[1,2] Departamentos de Ciência da Computação, Centro Universitário FEI
[1]dpilotto@uol.com.br, [2]flaviot@fei.edu.br

Resumo: Este projeto consiste no desenvolvimento de um novo firmware relacionado à placa principal utilizada no robô F-180 da equipe RoboFEI. Com o novo firmware, pretende-se reduzir o número elevado de falhas encontradas no atual. O novo código será modelado com base no conceito das Redes de Petri. Com isso, espera-se facilitar a manutenção e melhorar o desempenho do código. Dentre as melhorias já implementadas no projeto vale ressaltar o Debug, que permite analisar o código enquanto está rodando, e o controle relacionado à movimentação do robô.

1. Introdução

Iniciando seus trabalhos em 2003, a equipe de Futebol de Robôs é o principal grupo de pesquisas e projetos robóticos do Centro Universitário FEI Ao longo de seus 15 anos de existência, muitas experiências e conhecimentos tecnológicos foram adquiridos, desenvolvendo robôs capazes de participar de diversas competições pelo Brasil e pelo mundo [1].

O Futebol de robôs é uma modalidade que requer um amplo desenvolvimento das áreas de Inteligência Artificial e Robótica. As partidas são dinâmicas, imprevisíveis e autônomas, resultando em complexos desafios que podem ser desenvolvidos através de múltiplas soluções [2].

Para tal feito os robôs devem ter um sistema para situá-los e controlá-los em campo. Para fazer essa situação, existem câmeras sobre o campo que fazem a captura de tudo o que acontece ali. Com essa informação, o software da estratégia, após tê-la desenvolvido, envia os dados gerados ao robô através de um sinal de radio. Com esses dados o firmware da placa eletrônica irá gerar todos os valores necessários (como velocidade dos motores ou potência de chute) para que a ação desenvolvida na estratégia seja executada [3].

2. Referencial Teórico

2.1 Firmware

Também conhecidos pela nomenclatura "software embarcado", os Firmware são um conjunto de instruções operacionais que são programadas diretamente no hardware de equipamentos eletrônicos. Estes códigos são fundamentais para iniciar e executar os hardwares e os seus recursos, fornecendo informações idênticas sempre que o dispositivo for ligado.

Desta forma, este sistema responsável por operar aparelhos eletrônicos estará gravado diretamente no chip de memória de seus hardwares, mais especificamente nas memórias PROM e EPROM (memória somente leitura programável e memória somente leitura apagável, respectivamente) [4].

2.2 Firmware Atual

O algoritmo atual utilizado pela equipe é dividido em 10 partes. No geral, essas partes deveriam fazer com que o firmware operasse da melhor maneira esperada.

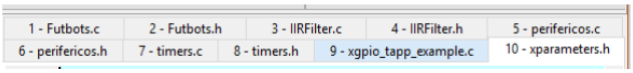

Figura 1 - Componentes do Firmware da equipe.

Contudo, algumas dessas partes não estão sendo mais utilizadas. Isso se deve ao fato de não haver nenhuma documentação sobre esse código e também por sua difícil compreensão. Ainda, o código presenta maiores problemas como, por exemplo, o fato de não haver controle algum sobre os motores.

2.3 Redes de Petri

Uma rede de Petri ou rede de transição é uma das várias representações matemáticas para sistemas distribuídos discretos. Como uma linguagem de modelagem, ela define graficamente a estrutura de um sistema distribuído como um grafo direcionado com comentários.

Possui nós de posição, nós de transição, e arcos direcionados conectando posições com transições. Uma transição possui lugares de entrada e saída (1 ou vários).

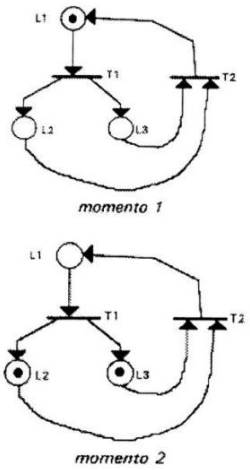

Figura 2 - uso dos tokens em uma rede de Petri.

Para representar o dinamismo dos fenômenos, é necessário que se coloquem marcas (fichas ou, do inglês, tokens) em determinados lugares da rede. A circulação destas marcas através da rede é que nos vai dar a noção deste dinamismo. Uma transição é dita como habilitada quando os seus lugares de entrada estão marcados. Ocorrendo isto, a transição pode ser realizada, como pode ser visto na figura 2 [5].

3. Metodologia

Em primeiro momento, será estudado o funcionamento do firmware utilizado atualmente nas placas. Posteriormente, será desenvolvido o código que ocupará o lugar do antigo e analisada a melhora obtida. Um ponto positivo é que o método de desenvolvimento utilizado nas Redes de Petri sobre lugares, transições e fichas faz com que o código tenha uma fácil compreensão, o que ajudará em reparos futuros.

Para conseguir o melhor resultado possível, será feito um estudo sobre o firmware utilizado por equipes de alto nível para ver qual a chance de programar algo parecido. E, ainda, será feito uma pesquisa sobre programações em placas parecidas com a utilizada, para, com isso, diminuir ao máximo o número de falhas. Em seguida serão realizados testes para comprovar essa melhora.

4. Resultados

4.1 Outras Equipes

Ao analisar as soluções propostas por equipes que utilizam da mesma linguagem de programação utilizada pelo RoboFEI (C/C++).

Uma das equipes analisadas foi a ITAndroids. Além de eles possuírem um doutorando no tema, eles passaram por um problema parecido. A solução apresentada por eles foi desenvolver uma ferramenta de controle, o que solucionaria o problema da movimentação do robô, e separar o código em blocos destinados a cada estrutura a ser programada na placa.

4.2 Proposta

Para atingir os objetivos e conseguir desenvolver o que foi proposto, o primeiro passo se deu com a longa pesquisa e aprendizado do software da Xilinx, o ISE Design Suite.

Em seguida, foi gerada a estrutura de hardware da placa eletrônica do projeto dentro deste software. Concluindo isso, já é possível gerar o Workspace (local) onde o código que constituirá o firmware será desenvolvido.

Com o Workspace gerado, basta adicionar os arquivos em linguagem C/C++ que irão compor o firmware e baixar para a placa. Caso o código não se comporte como esperado é possível debuggar o código e analisa-lo enquanto ele roda para corrigir esses eventuais erros.

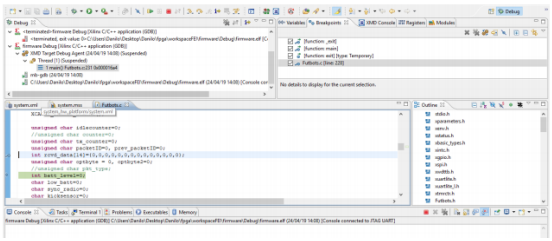

Figura 3 - Janela de Debugger gerada no SDK

Além disso, chegou-se à conclusão de que um código segmentado e organizado traria um melhor entendimento e uma maior facilidade na hora de encontrar erros e falhas. Por isso, o código foi reestruturado e criaram-se blocos para cada parte a ser controlada.

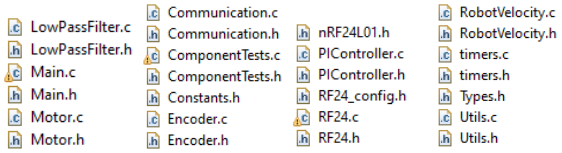

Figura 4 – Nova estrutura do firmware.

5. Conclusões

Um dos grandes problemas existentes no firmware atual é a falta de documentação. Graças a isso, não se tem acesso à boa parte dele, como a parte de debug e indexação de variáveis. Esta falta de informações faz com que boa parte do código atual não seja entendida pelos programadores, fazendo com que essas funções sejam excluídas.

Como o novo firmware será uma reformulação do antigo, além da equipe voltar a ter acesso a essas estruturas perdidas o que aumentando ainda mais o controle do robô. Por se tratar do mesmo código apenas com algumas reformulações, o entendimento do mesmo será bem mais direto, por parte dos programadores mais antigos, devido à familiaridade.

Além disso, outro benefício será na hora da manutenção. Devido à utilização das Redes de Petri, o código contará com diversas condições para executar determinadas funções. Ao avaliar essas condições será possível determinar (durante os testes) exatamente onde os erros ocorrem. O debug também traz vantagens nesta etapa, pois localiza os erros.

6. Referências

[1] ROBOFEI Centro Universitário da FEI. Página do time de futebol de robôs da FEI. Disponível em: <https://portal.fei.edu.br/Pagina/robo-fei>.

[2] RoboCup Federation. RoboCup, 2016. Disponivel em: <http://www.robocup.org>.

[3] RoboCup Federetion. Small Size League, 2019. Disponível em: <https://www.robocup.org/leagues/7>.

[4] OLIVEIRA, André Schneider de; ANDRADE, Fernando Souza de. Sistemas Embarcados: Hardware e Firmware na Prática. São Paulo: Érica Ltda., 2006.

[5] CARDOSO, Janette; VALETTE, Robert. Redes de Petri. Florianópolis, 1997.

Agradecimentos

Ao Centro Universitário FEI pela realização das medidas, empréstimo de equipamentos e financiamento do Projeto de Robótica RoboFEI

[1] Aluno de IC do Centro Universitário FEI (ou FAPESP, CNPq ou outra). Projeto com vigência de 11/18 a 10/19.

[2] Professor orientador do projeto do Centro Universitário FEI.