Injeção de falhas de comunicação sobre implementações do protocolo EtherCAT

Luiz Gustavo A. Gomes[1], Taisy S. Weber[1], Sérgio L. Cechin[1], Rodrigo Dobler[1], João de Moraes[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

{lgagomes, taisy, cechin, rjdobler, joao.moraes}@inf.ufrgs.br

Abstract. *Ethernet-based protocols are used in different industrial areas. Among these protocols, EtherCAT is a popularly used one. For each new equipment, such protocol must be specially implemented and submitted to a validation process. This work proposes to develop a test suite using an existent fault injector, to provide support in this task. Thereby, such implementations will be tested against fault occurrences, helping in the process of validating its dependability.*

Resumo. *Protocolos baseados em Ethernet são usados em diferentes áreas industriais. Dentre estes protocolos, o EtherCAT é um popularmente usado. Para este protocolo, a cada novo dispositivo desenvolvido, é necessário que os desenvolvedores façam suas próprias implementações e as submetam para o processo de validação conforme a finalidade. Neste contexto, o trabalho propõe desenvolver um ambiente de testes utilizando um injetor de falhas existente para modificá-lo, de modo a servir de apoio neste processo. Com isto, tais implementações serão testadas frente à ocorrência de falhas, auxiliando no processo da validação de sua dependabilidade.*

1. Introdução

Com sistemas de controle digital se comportando cada vez mais como uma grande rede interconectada e o custo reduzido de equipamentos com interface Ethernet, surgiu um interesse econômico em introduzir sistemas de comunicação baseadas em Ethernet no domínio industrial [Neumann 2007]. Portanto, soluções em comunicação baseadas em Ethernet foram desenvolvidas especialmente para atender requisitos industriais como tempo e sincronismo.

Uma destas soluções é o protocolo EtherCAT [EtherCAT Technology Group 2016], planejado para dar suporte ao processamento de informações, monitoramento e controle de sistemas de controle para diversos setores industriais de diferentes domínios, de maneira simples.

Como o padrão EtherCAT não define uma implementação padrão pré-certificada [Zhou e Hu 2011] para qualquer dispositivo em geral, o protocolo deve ser implementado pelos desenvolvedores e sua validação feita pelo grupo órgão certificador responsável (EtherCAT Technology Group). Infelizmente não estão disponíveis ferramentas que auxiliem os desenvolvedores a testar seus códigos de forma automatizada frente à ocorrência de falhas previstas na especificação. Por conta disto, cada nova implementação deve ser enviada e validada pelo órgão certificador. Com o auxílio de uma ferramenta de injeção de falhas, o processo de validação se torna mais

rápido, visto que a implementação já terá sido testada na ocorrência de falhas antes de ser submetida ao órgão certificador.

Atualmente existem ferramentas que realizam monitoramento de dispositivos que já executam o protocolo EtherCAT, porém com nenhuma delas é possível determinar o comportamento destes dispositivos sob falhas de comunicação.

O objetivo principal deste trabalho é avaliar a possibilidade de usar o injetor de falhas FITT [Dobler 2016] para validar implementações de EtherCAT. O injetor FITT foi desenvolvido para validar PROFIsafe [PROFIsafe 2016], um protocolo de comunicação segura que executa sobre ProfiBUS ou ProfiNET e cuja especificação é completamente diferente de EtherCAT, exceto por alguns tipos de falhas que tanto PROFIsafe como EtherCAT devem tolerar. Assim será criado um ambiente experimental onde equipamentos já validados e que executam EtherCAT serão utilizados como mestre e escravo e nesses equipamentos serão injetadas falhas usando FITT. Este cenário servirá para que a portabilidade do injetor de falhas FITT seja avaliada, ou seja, quais funcionalidades presentes podem ser utilizadas e quais precisam ser modificadas a fim de que falhas possam ser injetadas no protocolo EtherCAT.

O segundo objetivo é aproveitar os resultados do experimento de portabilidade para criar um novo injetor específico para EtherCAT que será utilizado posteriormente como parte de uma arquitetura de validação de uma implementação do protocolo EtherCAT sendo desenvolvido pelo grupo de pesquisa. Tal implementação está sendo desenvolvida em um *Field Programmable Gate Array* (FPGA), e futuramente originará um Circuito Integrado (ASIC).

O artigo apresenta um breve resumo do protocolo EtherCAT, experimentos realizados, decisões de projeto tomadas e resultados alcançados.

2. Protocolo EtherCAT

EtherCAT [EtherCAT Technology Group 2016] é um protocolo industrial de tempo real baseado em Ethernet. Este protocolo não possui restrições quanto à topologia, podendo adotar topologias em linha, árvore, estrela ou uma combinação destas. Possui um mestre EtherCAT (ECM) e até 65535 escravos EtherCAT (ESCs), onde apenas o mestre pode criar e enviar *frames* e os escravos apenas podem realizar leituras e escritas nestes *frames* [EtherCAT Technology Group 2015].

Para comunicar o mestre e os escravos, EtherCAT encapsula seus dados (datagramas EtherCAT) dentro de um *frame* Ethernet padrão (Figura 1), sendo diferenciado de um frame Ethernet convencional através de um identificador (0x88A4) no campo *EtherType* [EtherCAT Technology Group 2015]. Cada datagrama EtherCAT é subdividido em um cabeçalho EtherCAT e um ou mais datagramas EtherCAT [EtherCAT Technology Group 2014], que indicam qual tipo de acesso o mestre deseja executar (leitura e / ou escrita) e quais escravos devem realizar tal tarefa [EtherCAT Technology Group 2015].

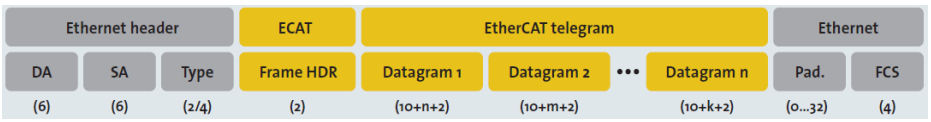

Figura 1. Dados EtherCAT encapsulados em um *frame* Ethernet padrão.

O mestre EtherCAT envia um *frame* que passa por cada nó pertencente à rede, onde cada dispositivo escravo lê o dado que lhe é endereçado e insere seus dados enquanto o *frame* está em trânsito [EtherCAT Technology Group 2015]. Um *frame* EtherCAT não para enquanto passa por um escravo, ocorrendo apenas um pequeno atraso enquanto a leitura ou escrita são realizadas [Orfanus et al. 2013]. Após o *frame* trafegar por toda a topologia, o último escravo EtherCAT a receber o *frame* é responsável por enviá-lo no sentido inverso, que será interpretado pelo mestre como um *frame* de resposta [EtherCAT Technology Group 2013a], com isto, vários nodos podem ser endereçados individualmente através de um único *frame* Ethernet, carregando diversos *Protocol Data Units* (PDUs) [EtherCAT Technology Group 2013a] – estrutura utilizada para configurar os escravos e transmitir dados [Orfanus et al. 2013].

2.1. Modelo de falhas

Erros de comunicação podem ser causados por diversos motivos, desde falhas de *hardware* até interferências do ambiente externo. Embora o protocolo trate de diversos tipos de falhas [EtherCAT Technology Group 2013a, 2013b, 2013c, 2013d, 2014], o escopo deste trabalho ficará reduzido a falhas de perda e atraso de pacotes, pois são falhas que afetam tanto EtherCAT quando PROFIsafe e estão implementadas no FITT.

Falhas de perda de pacotes ocorrem quando um ou mais pacotes não conseguem alcançar seu destino, podendo ser causadas por congestionamento do enlace, fazendo com que o protocolo descarte pacotes. Entre outras causas, pode-se citar mal funcionamento de equipamentos físicos, rompimento do meio físico de transmissão, etc.

Atraso de mensagens é observado quando restrições temporais não são respeitadas, isto é, o tempo esperado para um pacote ir de sua origem até seu destino é maior do que o esperado.

2.2. Mecanismos de detecção / correção de erros do protocolo

Por ser um protocolo focado em velocidade de comunicação, EtherCAT não possui muitos mecanismos de detecção e correção de erros. Porém são capazes de lidar com alguns dos tipos mais comuns de falhas existentes: corrupção, atraso e perda de pacotes.

O primeiro mecanismo é *o Frame Check Sequence* (FCS) presente no *frame* Ethernet, sendo utilizado para detectar falhas de corrupção de pacotes [EtherCAT Technology Group 2013c]. Este é checado tanto pelo mestre quanto pelos escravos para determinar se um *frame* foi recebido corretamente. Como vários escravos podem alterar o *frame* durante seu percurso pela topologia, o FCS é recalculado por cada nó escravo na recepção e na retransmissão. Se um erro de *checksum* é encontrado, o escravo não repara o FCS, mas sinaliza o mestre incrementando um contador de erros interno para que o ponto de falha seja precisamente localizado na topologia.

O *Working Counter,* último campo do datagrama EtherCAT, possui um valor esperado calculado pelo mestre que é incrementado pelos escravos a cada leitura ou escrita bem-sucedida realizada no datagrama, sendo utilizado para detectar falhas de perda de pacotes [EtherCAT Technology Group 2014]. Escravos que apenas estão encaminhando o datagrama, não alteram o *working counter*. Comparando o *working counter* com o número esperado de escravos que deveriam acessar dados, o mestre pode saber quantos escravos processaram seus dados correspondentes.

Para lidar com falhas de atraso de pacotes, os dispositivos escravos possuem *watchdogs* internos, que podem ser monitorados por uma aplicação. Tais *watchdogs* são reiniciados a cada comunicação bem-sucedida (escrita ou leitura) entre o escravo e o mestre [EtherCAT Technology Group 2014]. Caso nenhuma comunicação ocorra dentro do tempo pré-estabelecido, o *watchdog* é disparado, seu contador é incrementado e seus sinais de saída não são entregues.

3. Trabalhos relacionados

Durante as pesquisas realizadas, foram encontradas ferramentas que fazem apenas monitoramento de dispositivos EtherCAT, como *TwinCAT* [TwinCAT 2016] e *EtherCAT Device Monitor* [EtherCAT Technology Group 2016], porém nenhuma que possa verificar novas implementações do protocolo EtherCAT em desenvolvimento.

TwinCAT é uma ferramenta utilizada para programar e monitorar Controladores Lógicos Programáveis (CLPs) que implementem o protocolo EtherCAT. Por contar apenas com monitoramento e programação, não é possível utilizá-lo para injetar falhas e verificar o funcionamento de um novo dispositivo em condições adversas.

EtherCAT Device Monitor é uma ferramenta com funcionalidades semelhantes à *TwinCAT*, porém desenvolvida por outra empresa e já descontinuada.

Ambas as ferramentas oferecem apenas monitoramento, não são gratuitas e não nos permitem avaliar o comportamento de uma dada implementação perante a ocorrência de falhas, por isto a necessidade da solução proposta neste trabalho.

4. Realização dos experimentos

A empresa parceira no projeto pretende desenvolver uma nova linha de produtos, que conta com um ASIC que implementa o protocolo EtherCAT. Para tal, é necessário que o ASIC passe por uma bateria de testes de campo em condições normais de operação e em condições de falhas antes de ser enviado para o processo de validação e certificação do protocolo pelo órgão responsável.

O uso de um injetor de falhas que ponha o produto à prova irá auxiliar a equipe a definir se a implementação está correta de acordo com a especificação. Com isso as chances de que outras implementações do protocolo EtherCAT sejam certificadas aumentará. Este cenário também irá avaliar a portabilidade da ferramenta FITT [Dobler 2016], definindo quais funcionalidades presentes podem ser utilizadas para que falhas possam ser injetadas no protocolo EtherCAT de modo geral.

Nas próximas subseções serão discutidos detalhes da arquitetura utilizada nos experimentos e decisões pertinentes de projeto.

4.1. Arquitetura do sistema

FITT [Dobler 2016] é um injetor de falhas para protocolos seguros, em especial o PROFIsafe [PROFIsafe 2016], onde falhas previstas na norma IEC 61508 são injetadas. Este injetor age interceptando pacotes trocados entre dois dispositivos ligados em série, tornando-se assim um componente independente do sistema.

Figura 2. Arquitetura proposta

O ambiente experimental utilizado é semelhante ao descrito por [Dobler 2016] e ilustrado pela Figura 2, mas contando com dispositivos que executam o protocolo EtherCAT cedidos pela empresa parceira: um mestre EtherCAT que é monitorado em tempo real por um computador separado, um escravo EtherCAT que recebe comandos vindos do mestre e um segundo computador que executa o injetor, sendo colocado entre o mestre e o escravo, interceptando os pacotes trocados.

O computador onde o injetor foi executado conta com processador *Intel Core* i7-3770 3.4 GHz com 16 GB de memória RAM, com sistema operacional *Ubuntu* 12.04 LTS 64 *bits* e duas placas de rede *Intel Gigabit* modelo e1000e 82574L que são responsáveis por interligar os dispositivos mestre e escravo ao injetor de falhas. Já o computador que monitorava o mestre EtherCAT conta com um processador *Intel Core 2 Duo* E4000 2.4 GHz, 4 GB de memória RAM e sistema operacional *Windows*® 7 32 *bits*.

FITT utiliza uma biblioteca chamada PF_RING [PF_RING 2013] para captura e envio de pacotes em alta velocidade, a qual torna possível a análise e manipulação de pacotes e tráfego ativo de rede. Essa biblioteca possui um módulo que deve ser carregado no *Kernel* do Linux, permitindo que os pacotes sejam trocados diretamente entre a interface de rede e aplicações do usuário sem utilizar mecanismos do *Kernel* do Linux. Com isto os ciclos de processamento da CPU são usados unicamente para consumir pacotes e não para movê-los para dentro ou para fora do adaptador de rede, não inserindo atrasos adicionais no sistema.

Ainda sobre o injetor FITT, vale destacar que é possível escolher um sentido para as falhas serem injetadas, seja na direção mestre para escravo ou escravo para mestre. Isto oferece uma facilidade, pois sabendo o sentido de injeção de falhas, facilitando a procura por resultados anômalos.

O ambiente de testes final é mostrado na Figura 3 (com exceção dos computadores) e está estruturado da seguinte forma: dispositivos “A” e “D” são fontes de alimentação 30 W e 24 Vdc. Dispositivo “B” é um módulo a relé servindo como escravo EtherCAT. Dispositivos “C” e “G” são expansores de barramento que convertem sinais LVDS (*Low-voltage differential signaling*) em pacotes Ethernet 100BASE-TX. O dispositivo “C” está conectado a uma das placas de rede do computador executando o injetor de falhas e o dispositivo “G” está conectado à outra placa de rede do mesmo computador. O dispositivo “E” é um controlador programável que age como mestre. E o dispositivo “F” é um controlador programável adaptado apenas para monitorar o fluxo de pacotes que transita pelo mestre.

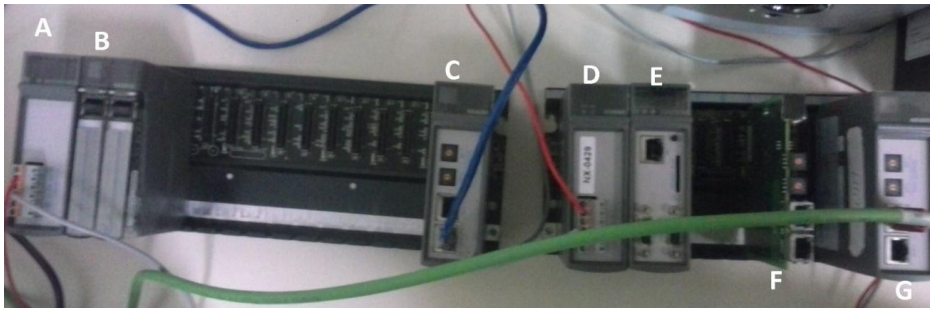

Figura 3. Dispositivos utilizados nos testes

4.2. Metodologia de testes

A partir do modelo de falhas levantado e dos mecanismos de detecção de falhas definidos pela especificação do protocolo, ficou decidido que os testes de falhas de perda, atraso de pacotes e funcionamento normal seriam utilizados.

Os testes preliminares para avaliar a possibilidade de se utilizar o injetor para o protocolo EtherCAT foram realizados conectando o dispositivo mestre a uma das placas de rede do computador executando o injetor por um cabo Ethernet comum e, com outro cabo, o dispositivo escravo foi conectado ao mesmo computador, porém na segunda placa de rede.

Como a ferramenta FITT [Dobler 2016] foi desenvolvida exclusivamente para protocolos seguros e EtherCAT não é um protocolo seguro, é esperado que nem todas as funções de injeção de falhas possam ser reaproveitadas.

O primeiro teste foi apenas para testar a comunicação entre os dispositivos. Aproveitando uma das funções já existentes no injetor, de execução normal sem injeção de falhas, foram enviados comandos do mestre para o escravo e foi observado no computador que monitorava o mestre, o estado das variáveis de leitura e escrita no escravo.

Para que os testes com injeção de falhas pudessem ser realizados, foi necessária uma mudança no injetor. Originalmente se filtrava qualquer pacote que não seguisse o modelo imposto pelo protocolo PROFIsafe. Com isso qualquer pacote que fosse diferente deste modelo não era enviado para o injetor, sendo encaminhado automaticamente. Para contornar isto foi feita uma alteração neste filtro, fazendo com que apenas pacotes pertinentes ao mestre e escravo, ou seja, pacotes com o *EtherType* igual a 0x88a4 fossem enviados para o injetor.

O próximo teste foi o de perda de pacotes. O objetivo deste teste é avaliar a robustez dos dispositivos mestre e escravo em situações que apresentam perda de dados e verificar se seu comportamento é condizente com o previsto pela especificação. Neste cenário, o injetor é configurado de forma a não enviar alguns pacotes que estão transitando em um dos sentidos da comunicação entre o mestre e o escravo. O usuário entra com dois parâmetros de configuração: um sentido para injetar as falhas e um tempo em segundos. Este tempo é o tempo de escolha do pacote que será descartado, ou seja, um contador de tempo será incrementado constantemente e quando for igual ao valor informado pelo usuário, o próximo pacote a chegar pela direção selecionada, será

descartado. Quando os próximos pacotes chegarem, eles serão enviados normalmente ao seu destino, até que o contador atinja novamente o tempo especificado e o ciclo de injeção de falhas se repita.

Em seguida foi realizado o teste de atraso de pacotes. Este teste tem por finalidade verificar o funcionamento dos *watchdogs* internos dos escravos. Neste teste o usuário seleciona uma direção para injetar as falhas e dois valores de tempo: o tempo de escolha do pacote e o tempo para atrasar este pacote. O funcionamento é similar ao teste de perda de pacotes, porém ao invés do pacote selecionado ser descartado, aqui ele é posto em um *buffer,* aguardando até que passe o tempo de atraso definido. Depois de decorrido este tempo, o pacote é enviado ao seu destino. Vale notar que enquanto um pacote está no *buffer*, os demais trafegam normalmente.

4.3. Resultados

O processo de adaptação do injetor teve que considerar aspectos relativos à diferença entre os protocolos PROFIsafe e EtherCAT.

Para o teste de execução normal, sem injeção de falhas, após alguns minutos de troca de pacotes, verificou-se que o funcionamento ocorreu dentro do esperado, com limites temporais previstos na especificação do protocolo sendo respeitados. Logo o injetor não interferiu na comunicação entre mestre e escravo.

No teste de perda de pacotes, foi observado um comportamento esperado para situações com pacotes não chegando ao seu destino, onde algumas respostas do escravo não chegavam até o mestre e comandos do mestre não chegavam até o escravo.

Para o teste de atraso de pacotes, não foram detectadas alterações significativas no funcionamento normal do mestre e escravo. Após uma análise do tráfego do mestre pelo computador monitor utilizando o programa *Wireshark*, notou-se que seguidamente os pacotes atrasados não tinham tanta relevância para o funcionamento correto do escravo, como por exemplo, um *broadcast* para se detectar novos escravos conectados na rede. Devido a isso, uma alteração no injetor foi proposta, de atrasar todos os pacotes ao invés de apenas um. Com isto será possível estabelecer uma margem de operacionalidade de um dispositivo mestre da seguinte forma: um dispositivo escravo introduz um atraso conhecido (ainda que mínimo) na comunicação ao repassar pacotes. Injetar um atraso em todos os pacotes da rede assemelha-se a conectar mais e mais dispositivos escravos a um mestre. Assim pode se ter uma estimativa de quantos escravos um mestre pode ter antes que se tenha um atraso considerável na comunicação.

5. Conclusão

O injetor de falhas escolhido como base do trabalho provou-se capaz de operar sobre um tráfego de dados EtherCAT com algumas modificações, apesar de ter sido desenvolvido especificamente para PROFIsafe. O uso dos dispositivos fornecidos pela empresa parceira foi de grande auxílio, pois o fato de já serem certificados para o protocolo EtherCAT serviu como validação do experimento de portabilidade.

Com base nos resultados positivos alcançados, as próximas etapas do trabalho consistirão em realizar modificações no injetor para que este insira falhas de corrupção de pacotes, atraso de todo o fluxo de dados e realizar a avaliação de futuras implementações sendo desenvolvidas pelo grupo de pesquisa pertencente ao projeto.

Testes futuros irão incluir falhas de corrupção de dados, para avaliar o funcionamento dos dispositivos em ambientes que oferecem interferência ao meio de comunicação. Como os dispositivos que implementam o protocolo podem ser utilizados em ambientes do tipo "chão de fábrica", que tendem a apresentar interferências, este teste é considerado essencial.

A experiência adquirida no desenvolvimento do ambiente de testes bem como adaptações realizadas no injetor permitirá que implementações do protocolo EtherCAT sejam testadas de maneira automatizada perante falhas antes que sejam enviadas para o processo de certificação.

6. Bibliografia

Dobler, R. J. (2016). "FITT: Uma Ferramenta de Injeção de Falhas para Validar Protocolos de Comunicação Seguros", Brasil, Dissertação (Mestrado), Universidade Federal do Rio Grande do Sul.

EtherCAT Technology Group (2013a). "EtherCAT Specification – Part 4", EtherCAT Technology Group.

EtherCAT Technology Group (2013b). "EtherCAT Specification – Part 1", EtherCAT Technology Group.

EtherCAT Technology Group (2013c). "EtherCAT Specification – Part 3", EtherCAT Technology Group.

EtherCAT Technology Group (2013d). "EtherCAT Specification – Part 6", EtherCAT Technology Group.

EtherCAT Technology Group (2014). "EtherCAT Slave Controller – Hardware Data Sheet Section 1".

EtherCAT Technology Group (2015). "EtherCAT – The Ethernet Fieldbus", EtherCAT Technology Group. Disponível em: <https://www.ethercat.org/en/downloads.html>.

EtherCAT Technology Group | HOME (2016). Disponível em: <https://www.ethercat.org/default.htm>, Acesso em 30 Mai. 2016.

Neumann, P. (2007). "Communication in industrial automation—What is going on?", *Control Engineering Practice*, v. 15, n. 11, p. 1332–1347.

Orfanus, D. et al. (2013). "EtherCAT-based platform for distributed control in high-performance industrial applications", em *Emerging Technologies & Factory Automation*. IEEE.

PF_RING 2013. Disponível em:< http://www.ntop.org/products/ pf_ring/> Acessado em: Março 2016.

PROFIsafe (2016). Disponível em: <http://www.profibus.com/technology/profisafe>, acesso em 30 Mai. 2016.

TwinCAT (2016). Disponível em: <https://www.beckhoff.com/english.asp?twincat/default.htm>, acesso em 8 Ago. 2016,

Zhou, T. e Hu, J. (2011). "Design and realization of EtherCAT master", Em *Electronic and Mechanical Engineering and Information Technology*. IEEE.