Arquitetura de hardware de uma estação remota de entrada analógica em conformidade com a IEC 61508

Taisy S. Weber, Sérgio Cechin, João de Moraes, João C. Netto

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

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

Abstract. *The paper introduces the initial step of designing an analog input module for a remote input and output station used in safety related systems. The module was developed according to the IEC 61508 standard aiming at a high level of safety integrity (SIL 3). To achieve certification of compliance with the standard, the initial step is the development of the module architecture and its safety specification. Only after the approval of this step by the certification agency, the implementation of hardware and software can be initiated. The article presents the fault-tolerant hardware architecture of the input module.*

Resumo. *O artigo apresenta o resultado da etapa inicial do projeto de um módulo de entrada analógica para uma estação remota de entrada e saída usada em sistemas de segurança. O projeto foi desenvolvido de acordo com a norma IEC 61508 visando alto nível de integridade (SIL 3). Para almejar a certificação de conformidade com a norma, a etapa inicial consiste no desenvolvimento da arquitetura do módulo e a sua especificação de segurança. Apenas após a aprovação dessa etapa pela agência certificadora, a prototipação do hardware e software do módulo pode ser iniciada. O foco do artigo é a arquitetura de hardware tolerante a falhas do módulo de entrada.*

1 Introdução

Sistemas de segurança podem ser usados para interromper o fluxo de combustíveis e isolar equipamentos elétricos (através de atuadores) após detectar (através de sensores) alta pressão, alta temperatura, fogo ou vazamento de gás. Um exemplo são os sistemas de proteção contra alta pressão usados para prevenir acidentes em dutos de óleo ou gás (Lundteigen, Rausand, and Utne 2009).

O projeto de sistemas de segurança deve obedecer à rígida regulamentação. Em vários setores industriais, como extração e refino de petróleo, produção e distribuição de energia, e controle de máquinas e processos, regulamentações são baseadas na IEC 61508 (Gall and Wen 2010), que a partir do ano 2000, passou a permitir o uso de componentes programáveis no projeto de tais sistemas (Bell 2011).

Componentes programáveis estão sujeitos a falhas de hardware, interferências externas e erros de programação. Para garantir a integridade da função de segurança, o projeto do sistema deve ser conduzido obedecendo a rigorosos requisitos técnicos que impõem a escolha de mecanismos e estratégias adequados para manter a taxa de defeitos dentro dos limites adequados. A rigidez das normas de segurança limita as

opções de projeto, mas permite alcançar a integridade de segurança necessária, além de possibilitar certificação de integridade de segurança.

Um dos equipamentos de um sistema de segurança distribuído é a estação remota, formada por módulos de entrada e saída. Um módulo de entrada é responsável pela aquisição das informações dos sensores e o envio destas para um controlador programável (CP). Um módulo de saída recebe informações do CP e faz com que sejam aplicadas aos atuadores. A comunicação das estações remotas com o CP em um sistema de segurança é realizada através de um protocolo de comunicação seguro (Neumann 2007), sobre um *black channel*. A pilha do protocolo de comunicação deve ser implementada tanto na estação remota quanto no controlador programável, devendo também ser certificada quanto à integridade de segurança por órgão certificador competente.

Os módulos de entrada nas estações remotas podem ser digitais ou analógicos dependendo do tipo de sinal fornecido pelos sensores. Neste artigo, introduzimos o projeto do módulo de entrada SAIM (*Safety Analog Input Module*), que recebe oito sinais analógicos e os processa digitalmente antes de enviar ao controlador programável. SAIM está sendo projetado para ser usado com CPs seguros da Serie Nexto Safety, da empresa Altus SA, parceira no projeto. Depois de certificado, o produto resultante será aplicado para executar funções de segurança na indústria de máquinas e processos.

O projeto de um módulo de entrada analógica parece não representar grandes desafios. Mas o módulo não se restringe apenas a converter sinal analógico para digital. Ele deve executar os protocolos necessários para comunicação remota com o CP, detectar e corrigir erros, reportar os erros detectados e manter as máquinas ou processos sendo controlados em um estado seguro, caso os erros não possam ser corrigidos. Para atingir o nível de integridade de segurança almejado, o módulo deve ser projetado aplicando técnicas de tolerância a falhas e obedecendo todos os requisitos de segurança estabelecidos pela IEC 61508.

Para executar suas várias funções, SAIM emprega microcontroladores (MCU). Várias arquiteturas tolerantes a falhas podem ser implementadas com esses componentes, com diferentes níveis de redundância. Para SAIM foi escolhida uma arquitetura de 2 canais com diagnóstico cruzado de erros. Cada canal tem sua MCU. Uma terceira MCU executa funções não relacionadas à segurança, ou seja, funções que não precisam ser certificadas.

Esse artigo apresenta a arquitetura de hardware do módulo remoto de entrada analógica, SAIM, construído em conformidade a norma IEC 61508 como parte do projeto de transferência tecnológica SDCD financiado pelo BNDES, cuja empresa parceira, Altus SA, é fabricante de Controladores Programáveis. O objetivo do artigo é apresentar a experiência do grupo de pesquisadores do laboratório LAIS, da UFRGS, no desenvolvimento de sistemas de segurança visando certificação.

Existem vários fabricantes internacionais com equipamentos semelhantes certificados. Não é do nosso conhecimento a existência de produtos nacionais neste nicho de mercado. A empresa Altus é pioneira nacional no projeto de sistemas de segurança e, através de cooperação com universidades, desenvolve tecnologia para atuar como fornecedora de equipamentos certificados, não apenas localmente, mas como competidora internacional.

SAIM teve seu projeto de arquitetura concluído e se encontra na fase de aprovação da especificação dos requisitos de segurança por uma agência certificadora. O resultado da primeira fase de projeto é o diagrama de blocos, com a descrição de todos os componentes, e o documento que contém a especificação dos requisitos de segurança.

O artigo está organizado como segue: inicialmente são apresentados os fundamentos da norma IEC 61508. Em seguida são apresentados os conceitos relacionados a uma estação remota de E/S e os componentes básicos de SAIM. Depois é mostrado como são tratados os sinais analógicos de entrada e finalmente o artigo é concluído.

2 Norma para segurança funcional IEC 61508

A IEC 61508 (Bell 2006), foi a primeira norma a permitir o emprego de componentes programáveis para a construção de sistemas de segurança. Ela estabelece técnicas e métodos para evitar e controlar falhas aleatórias de hardware e erros de projeto e de software.

Um sistema de segurança desenvolvido criteriosamente segundo a norma pode almejar certificação para um determinado nível de integridade de segurança, SIL. Os níveis variam de 1 a 4 (Joannou and Wassyng 2014). Sistemas com menor probabilidade de apresentar defeitos são classificados em SIL 4. SAIM almeja alcançar certificação para SIL 3, nível de integridade de segurança usual na área de controle de máquinas e processos.

A norma visa aumentar a garantia de que a função de segurança esteja livre de falhas que possam conduzir a defeitos perigosos (Jin, Lundteigen, and Rausand 2011). Dessa forma, é natural que obrigue o emprego tolerância a falhas (Lundteigen and Rausand 2009). Adicionalmente, caso o sistema não possa ser mantido operacional diante da ocorrência de falhas, suas saídas devem ser comutadas para um estado seguro, geralmente parando a máquina ou processo que está sendo controlado.

A IEC 61508 foi publicada originalmente em meados do ano 2000. Em 2010 a norma foi revisada consolidando 10 anos de experiência na sua aplicação (Bell 2011). No meio industrial brasileiro, é crescente o interesse pelo desenvolvimento de produtos em conformidade com essa norma devido à regulamentação de setores como energia, petróleo e transporte público.

2.1 Certificação de segurança

O objetivo de uma certificação de segurança é prover a garantia, reconhecida pelo órgão regulador, que o sistema foi considerado seguro pelo órgão certificador (Panesar-Walawege et al. 2010).

A certificação de conformidade com a IEC 61508 é dada a um produto específico, e envolve análise do processo de desenvolvimento e produção para aquele produto e também da competência da equipe de desenvolvimento.

O conjunto de técnicas e medidas adotadas em todas as fases do ciclo de vida deve convencer o órgão certificador que o projeto apresenta o nível de integridade de segurança almejado.

O produto é certificado, principalmente, através da documentação gerada, mas a norma não estabelece o formato dos documentos. Todas as fases do ciclo de vida do produto devem ser auditadas. Os certificadores devem ter acesso a toda documentação e pessoas envolvidas no desenvolvimento. As ferramentas também devem ser avaliadas, assim como o todo o material usado durante o desenvolvimento está sujeito à avaliação.

A IEC 61508 é uma norma extensa, com centenas de páginas e centenas de medidas prescritivas baseadas em uma vasta experiência prática, pesquisas e discussões. Como consequência, desenvolver um projeto em conformidade com a norma e obter certificação é uma tarefa árdua (Bilich and Hu 2009).

2.2 Esquema básico de um sistema de segurança

Em sistemas de segurança construídos seguindo a IEC 61508, o hardware e o software, assim como a aplicação que implementa a função de segurança, devem ser construídos e validados usando estratégias e técnicas da área de tolerância a falhas e engenharia de software embarcado.

A Figura 1 mostra um sistema formado por estações remotas de entrada e saída, um componente eletrônico programável central (no caso um CP), e a comunicação entre eles. As estações remotas podem conter vários módulos de E/S. Cada módulo tem sua lógica de leitura e escrita de dados, comunicação e detecção de erros implementada por seus próprios microcontroladores (MCU). Um sistema de segurança pode apresentar alguns módulos E/S seguros (em amarelo na Figura 1) e outros módulos de E/S convencionais, mas o CP deve ser seguro.

Na área de segurança, é necessária uma alta probabilidade de uma resposta segura do sistema a defeitos tanto do processo ou equipamento sendo controlado como do próprio sistema de segurança. Entretanto, estes sistemas podem introduzir novos defeitos. Se a taxa de falsos alarmes ou ativações indevidas for muito alta (Lundteigen, Rausand, and Utne 2009), os operadores podem perder a confiança no sistema.

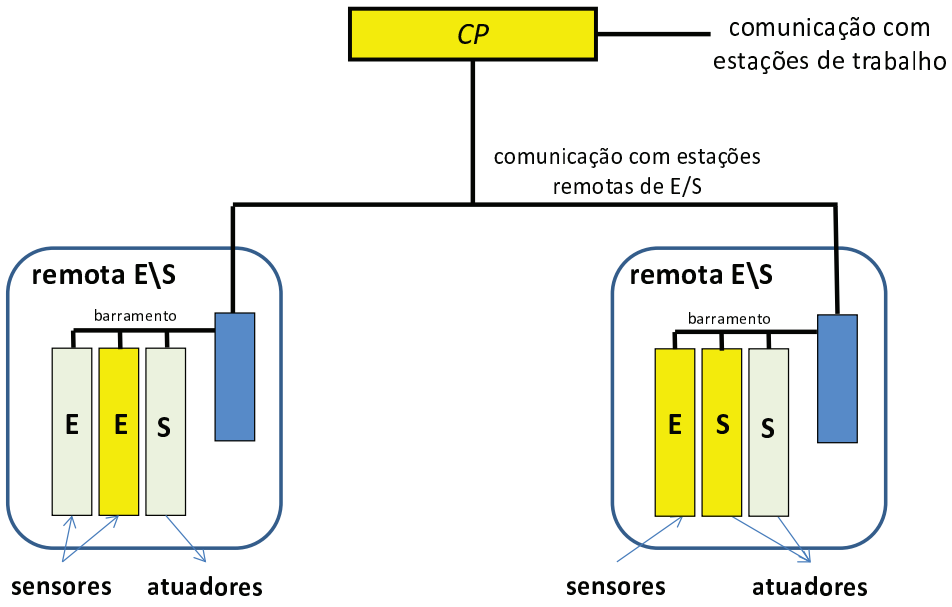

Figura 1 – Sistema distribuído com módulos de segurança

2.3 Comunicação segura e *black channel*

A IEC 61508 define que os sistemas de comunicação não podem ser responsáveis por mais do que 1% dos defeitos provocados por falhas não cobertas pelos mecanismos de detecção e diagnóstico.

A primeira versão da IEC 61508 estabelecia que todos os subsistemas de comunicação deveriam ser certificados. Essa exigência impunha sérias limitações ao desenvolvimento de estações remotas, pois sistemas de segurança distribuídos dependem de comunicação. Era impossível usar protocolos já consolidados na área de controle e automação, pois tais protocolos não são certificados pela IEC 61508. Uma possibilidade seria desenvolver e certificar tanto a pilha dos protocolos já em uso como os dispositivos de rede. Mas essa possibilidade tem um custo proibitivo e poucos fornecedores interessados.

A melhor solução veio com o conceito de *black channel*: toda a pilha de protocolos e o meio físico que transporta o sinal elétrico, magnético ou ótico são considerados um canal opaco. Sobre essa camada de comunicação insegura, cujos detalhes de implementação não precisam ser conhecidos, é implementado um protocolo seguro. O protocolo seguro é uma camada extra que reduz a probabilidade de erro na comunicação através de diversos mecanismos para detecção e correção de erros. Basta que a implementação dessa nova camada seja certificada como segura e toda a comunicação será considerada segura.

O conceito de *black channel* é extremamente útil, pois torna possível o uso de uma pilha convencional de protocolos de comunicação sobre a qual basta adicionar um protocolo seguro implementado em software. Esse conceito foi responsável pelo aparecimento de vários protocolos seguros (Neumann, 2007).

Perfis de protocolos seguros foram normatizados pela IEC 61784-3 (Bell, 2011). Uma dada implementação em software destes protocolos é passível de ser certificada até o nível SIL3. No projeto, foi escolhido o PROFIsafe, um dos protocolos padronizados pela IEC 61784-3, principalmente devido a sua popularidade, qualidade da documentação e a pré-aprovação para SIL 3. A pré-aprovação da especificação para SIL 3 não implica em garantias que a implementação em software para um dado hardware será certificada, mas facilita o processo de certificação do sistema de segurança distribuído que emprega o protocolo.

Um protocolo seguro forma uma camada acima de outro protocolo de mais baixo nível que reside no *black channel*. O PROFIsafe (Malik and Mühlfeld 2003), que é um protocolo seguro, opera sobre o protocolo PROFIBUS, um protocolo de comunicação popular no setor industrial, não certificado quanto a segurança.

3 O módulo de entrada analógica SAIM

SAIM é um módulo de entrada analógica que faz parte de uma estação remota. A função executada por SAIM é a de receber e executar os comandos enviados pelo CP, obter os dados dos sensores, processá-los e enviá-los para o CP. Para ser usado em sistemas de segurança, todo o projeto do módulo deve passar por um processo de certificação de conformidade com a IEC 61508. Esse processo inicia junto com a avaliação da especificação de segurança do módulo e sua arquitetura de hardware por uma agência certificadora.

3.1 Módulo de entrada SAIM em um sistema de segurança

A estação remota, mostrada na metade direita da Figura 2, é formada por uma cabeça PROFIBUS, uma interface para comunicação interna entre os módulos de E\S de uma mesma estação e um ou mais módulos de entrada e saída.

A comunicação segura da estação remota com o CP ocorre através de um canal seguro sobre PROFIBUS, que transporta os quadros PROFIsafe. O *black channel* engloba as cabeças PROFIBUS, que se encontram nos dois lados da comunicação, e a comunicação pelo barramento EtherCat entre os diversos módulos de E/S da estação remota.

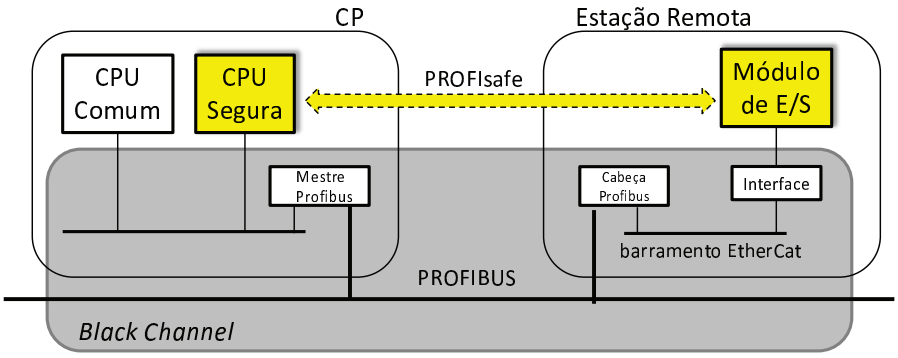

Figura 2 – *Black channel* entre estação remota e o controlador

EtherCat (Rostan, Stubbs, and Dzilno 2010) é um barramento muito usado em ambiente industrial, e é construído sobre Ethernet (Lin and Pearson 2013).

O CP, como pode ser observado na Figura 2, opera com duas unidades de processamento: um processador convencional, a CPU Comum, não relacionada à segurança; e uma CPU Segura, que processa o protocolo de comunicação segura e executa a função de segurança.

A estação remota deve contribuir com apenas uma baixa percentagem de erros residuais. No máximo 5% dos defeitos de todo o sistema de segurança podem ter sido gerados por erros na estação remota. Para isso, os módulos que executam o protocolo seguro devem ter incorporados mecanismos para detecção de erros e diagnósticos com alta cobertura de falhas. Assim, além de executar o protocolo de comunicação segura, o módulo deve conduzir todos os testes necessários para alcançar a cobertura de diagnóstico e de detecção de erros especificada para o nível de SIL desejado.

O hardware da estação remota é separado em dois grupos de circuitos: um dentro e outro fora do *black channel*. O primeiro grupo pode ser implementado usando hardware convencional. Assim, pode-se usar, sem certificação, componentes comuns que implementam os barramentos e seus protocolos correspondentes, PROFIBUS e EtherCat.

O grupo de circuitos fora do *black channel*, que na estação remota corresponde aos módulos seguros de E/S, deve ser construído seguindo as restrições da IEC 61508. Para isso, deve-se escolher uma arquitetura tolerante a falhas com grau de redundância adequado e implementar a cobertura de falhas exigida para alcançar o SIL desejado.

3.2 Comunicação via PROFIsafe

SAIM recebe comandos do CP remotamente via comunicação segura PROFIsafe. Planejamos usar um código fonte certificado de uma pilha PROFIsafe, comercializado pela Siemens, denominado "*PROFIsafe driver for F-Slaves*". O código fonte escolhido está em conformidade com a IEC 61784-3-3. A implementação já certificada do protocolo vai reduzir a sobrecarga de desenvolvimento e de certificação do módulo de entrada.

SAIM opera executando comandos do CP recebidos em um quadro PROFIsafe. Esse quadro passa pelo PROFIBUS e chega a SAIM via a cabeça PROFIBUS e o barramento interno da estação, que opera com EtherCat. Os dados analógicos obtidos dos sensores são convertidos para digital, processados, e montados em um quadro PROFIsafe. Este quadro, através de uma conexão EtherCat com o barramento, é passado para a cabeça PROFIBUS. A cabeça PROFIBUS é o dispositivo que permite comunicação remota com um mestre PROFIBUS residente no CP.

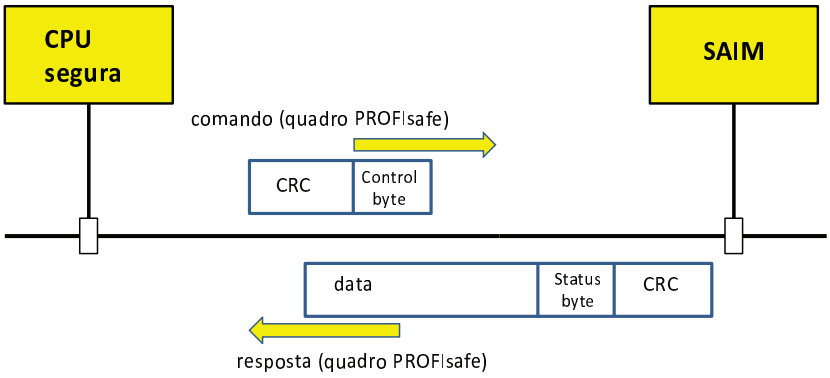

Figura 3 – Comunicação segura entre SAIM e CP

Os seguintes quadros PROFIsafe são manipulados em SAIM (Figura 3):

- Comando: consiste em 4 bytes enviados pela CPU Segura para SAIM, formado por 1 byte de controle e 3 bytes CRC.
- Resposta: consiste de 20 bytes enviados do SAIM e lidos pela CPU Segura. 16 bytes de dados são gerados no SAIM a partir do valor processado das 8 entradas analógicas; adicionalmente são computados um byte de status e 3 bytes CRC.

4 Diagrama de blocos da arquitetura de hardware

A Figura 4 mostra o diagrama de blocos da arquitetura da estação remota. A cabeça PROFIBUS, o barramento e o processamento da interface EtherCat pertencem ao *black channel*. Os dois processadores redundantes, SMCU1 e SMCU2, que processam os comandos vindos do CP e os sinais recebidos dos terminais de entrada de dados, devem garantir o nível de integridade de segurança necessária.

Várias opções de microcontroladores especialmente desenvolvidos para aplicações de segurança estão disponíveis no mercado. Nessa classe podem ser encontradas as famílias Hercules, da Texas, e a PXS da Freescale.

Os processadores da família Hercules, do tipo ARM com dois núcleos, foram escolhidos para SAIM, porque a empresa parceira Altus, já possui experiência na sua

aplicação em produtos industriais. Além de apresentarem alta cobertura de falhas, oferecem toda a documentação necessária para possibilitar um projeto de hardware seguro. O fabricante fornece informações que vão deste as taxas de defeito do componente até os métodos usados para reduzir as falhas sistemáticas de projeto e para detectar falhas aleatórias.

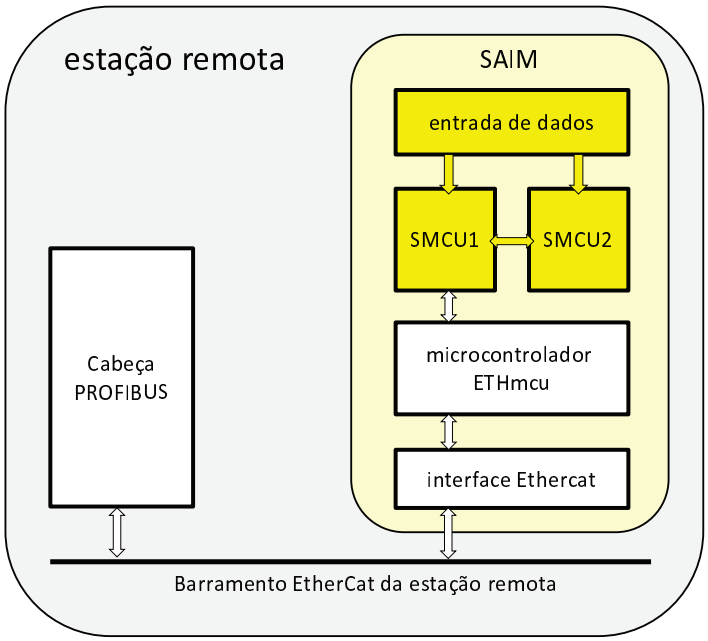

Figura 4 – Diagrama geral da estação remota com módulo SAIM

O processador ETHmcu é um microcontrolador ARM convencional.

4.1 Processamento da pilha de comunicação EtherCat: ETHmcu

A maioria das tarefas de software não relacionadas à segurança são executadas pelo ETHmcu, incluindo algumas tarefas complexas como a execução da pilha de comunicação EtherCAT. O ETHmcu não executa nenhuma tarefa relacionada à segurança, fazendo parte do *black channel.*

As tarefas básicas executadas pelo ETHmcu são:

- Processamento da pilha EtherCAT para comunicação com a cabeça PROFIBUS através do barramento.
- Verificação de endereço de barramento para comunicação EtherCAT com a cabeça PROFIBUS.
- Intermediação da comunicação entre o microcontrolador seguro SMCU1 e a cabeça PROFIBUS.
- Gestão da interface com o usuário.

O protocolo EtherCat, usado no barramento, pode ser trocado por outro protocolo de comunicação baseado em Ethernet industrial, apenas alterando o software do microcontrolador ETHmcu e desenvolvendo um hardware para interface com o novo barramento. Esta alteração pode ser realizada sem necessidade de reprojetar e certificar a parte segura do módulo SAIM.

O ETHmcu intermedia a comunicação PROFIsafe entre a CPU Segura no CP e os processadores SMCU1 e SMCU2. A troca das informações seguras é garantida pelo mestre PROFIsafe, executado na CPU Segura do CP, e o escravo PROFIsafe executado no módulo de entrada pelos processadores SMCU1 e SMCU2. O ETHmcu também intermedia a parametrização enviada da CPU Comum do CP para o par SMCU1/SMCU2, usando a parametrização padrão PROFIBUS. A verificação CRC é executada no par SMCU1/SMCU2 após a recepção dos parâmetros.

Todas as tarefas relacionadas com a segurança são executadas pelo par redundante SMCU1/SMCU2.

4.2 Processamento relacionado à segurança: SMCU1 e SMCU2

Os microcontroladores SMCU1 e SMCU2 (Figura 5) estão configurados de forma redundante para obter tolerância a falhas de hardware. Eles são interligados através de um canal serial, para que possam trocar dados e realizar detecção de erros cruzada. Estão configurados como um par rodando em *lock step* (ou seja, com sincronização de relógios).

SMCU1 e SMCU2 são micros Hercules, da Texas. Cada Hercules tem dois núcleos. São usados dois microcontroladores, totalizando assim quatro núcleos para processamento seguro. Como os núcleos dentro de um MCU operam de forma redundante, é possível diagnosticar erros por comparação dos resultados. Adicionalmente, realizando verificação cruzada entre os dois MCUs, é possível alcançar o grau de tolerância a falhas e de cobertura de falhas exigido pela norma para SIL 3.

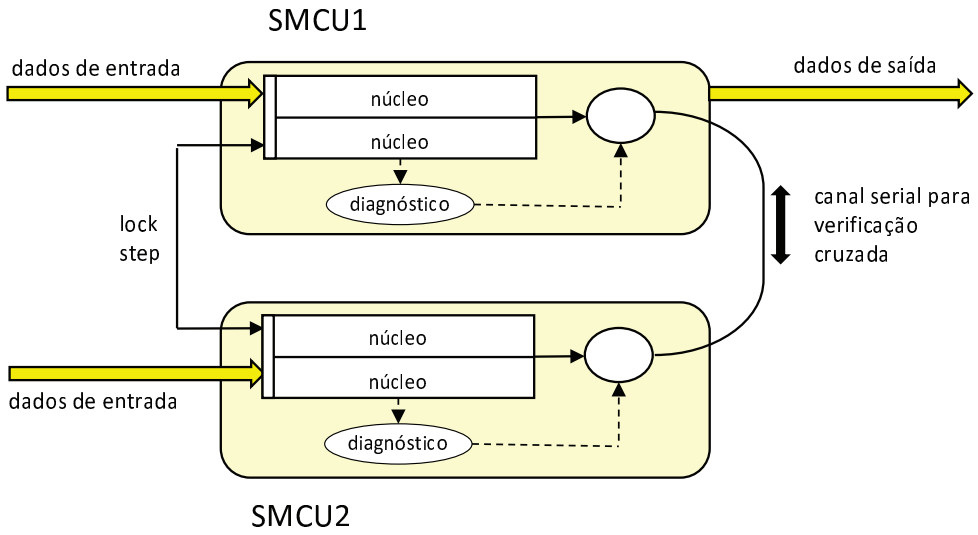

Figura 5 – Microcontroladores seguros em arquitetura redundante

O fato de termos quatro núcleos executando a mesma tarefa de forma redundante pode parecer um desperdício de capacidade de processamento. Entretanto para o nível de integridade de segurança almejada, SIL 3, tal redundância de hardware é plenamente justificada. Esta arquitetura tolerante a falhas simplifica as tarefas de detecção de erros e aumenta a cobertura de falhas sem a necessidade de exaustivos testes de diagnóstico por software sobre todos os componentes de hardware internos aos microcontroladores.

O software nestes microcontroladores redundantes é dividido em duas fases: inicialização e operacional. Durante a fase de inicialização, o software executa auto-teste em SMCU1 e SMCU2 e configura todos os registradores internos para executar as funcionalidades necessárias. A fase operacional é cíclica, com período de ciclo compatível com os tempos de resposta esperados na área de aplicação (1ms), nos quais são executadas as seguintes tarefas principais:

- Interface com o barramento através do ETHmcu (somente no SMCU1, que está conectado ao ETHmcu).
- Gestão da comunicação PROFIsafe com a CPU Segura no CP.
- Gestão da comunicação com a CPU Comum do CP (para diagnóstico e parametrização).
- Leitura e processamento de todas as 8 entradas analógicas do SAIM.
- Detecção, recuperação e sinalização de falhas.
- Diagnóstico.
- Sincronização e verificação cruzada entre os microcontroladores redundantes SMCU1 e SMCU2.

Decidimos que para SMCU1 e SMCU2 não seria usado qualquer tipo de sistema operacional de tempo real, tanto para reduzir a complexidade do software embarcado como para reduzir os custos de produção. Isto significa que todo o software de baixo nível necessário será desenvolvido no escopo do projeto e submetido à certificação.

4.3 Separação entre hardware de segurança e não relacionado à segurança

O hardware não relacionado à segurança é separado do hardware relacionada à segurança por circuitos para proteção contra sobretensão e isolamento na comunicação entre ETHmcu e SMCU1.

As tarefas não relacionadas à segurança executadas pelo ETHmcu poderiam ser executadas pelo SMCU1, mas a nossa escolha no projeto foi alocar um microcontrolador adicional (o ETHmcu) para essas tarefas. Esta escolha apresenta algumas vantagens importantes: diminui a complexidade do software no SMCU1 e, portanto, simplifica o cumprimento dos requisitos de segurança do software; e reduz o número de sinais de interface entre o SMCU1 e o barramento, reduzindo assim a complexidade da proteção contra sobretensão.

5 Tratamento dos sinais de entrada analógicos

Para SAIM poder ser considerado seguro, a entrada dos sinais analógicos também deve ser segura. Assim, cada uma das 8 entradas é conectada a um circuito de condicionamento de sinal diferente e a um conversor digital para analógico (ADC) diferente, formando uma arquitetura de canal duplo (Figura 6).

Cada circuito de condicionamento de sinal tem uma relação de tensão diferente. Além disso, cada conversor (ADC1 e ADC2) tem uma tensão de referência diferente, com a mesma relação de tensão. Isso resultará em ambos os ADCs retornando o mesmo valor binário. Podem ocorrer discrepâncias devido a ruído ou diferenças nos componentes. Portanto, uma pequena diferença pode ser tolerada. Se a diferença for maior, um erro deve ser sinalizado e o módulo deve ir para o estado seguro.

O valor de saída é sempre o valor médio entre as leituras dos dois ADCs. SMCU1 só pode ler ADC1, e SMCU2 só pode ler ADC2, portanto, os valores devem ser trocados e o valor médio é calculado por ambos os microcontroladores. Como cada microcontrolador cria todo o pacote PROFIsafe e o CRC é calculado em cada microcontrolador, qualquer erro durante o cálculo do valor médio é detectado na verificação do CRC do pacote.

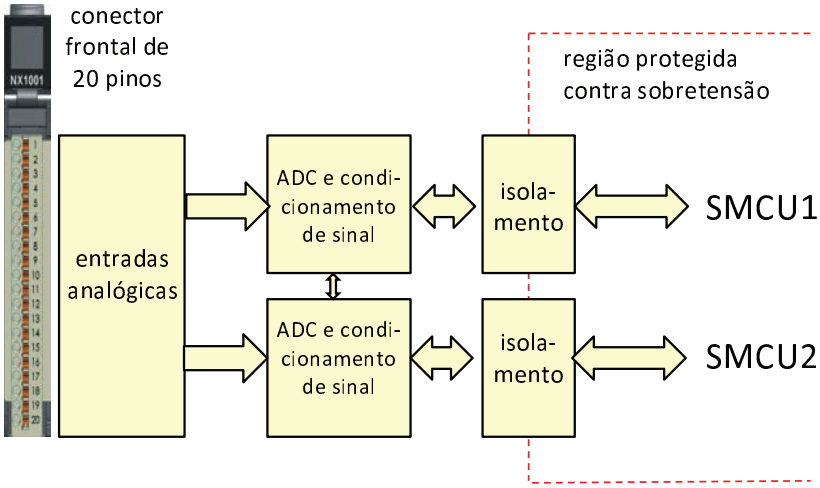

Figura 6 – Entrada de sinais analógicos

O ADC tem resolução de 24 bits. Para reduzir o ruído lido pelo ADC, apenas os 16 bits mais significativos são utilizados na leitura. Além disso, cada referência pode ser lida pelo outro ADC. Cada ciclo de leitura executa 9 leituras, isto é, todas as 8 entradas analógicas, e um teste de referência numa entrada alternada. Assim, todas as entradas são testadas após 16 ciclos de leitura. Depois disso, um interruptor analógico alterna entre o valor de referência e 0 Vdc, para o teste de valor seguro. Após mais 16 ciclos de leitura, o teste está completo.

6 Conclusões

Lloyd e Reeve (Lloyd and Reeve 2009) listam várias dificuldades que as empresas enfrentam para se adaptar a norma IEC 61508 e obter a certificação do primeiro produto. As dificuldades passam pela falta de experiência com segurança funcional da equipe de desenvolvimento e validação, pouca familiaridade com ferramentas e técnicas de tolerância a falhas e engenharia de software embarcado, documentação incompleta ou inconsistente, práticas usuais de desenvolvimento muito diferentes das impostas ou sugeridas pela norma, impossibilidade de rastrear os requisitos de segurança até a implementação, uso de código legado, adoção de ciclo de vida inadequado. Uma vez, entretanto, que consigam a primeira certificação, os demais produtos são mais facilmente certificados devido à mudança na cultura técnica introduzida na empresa.

O mercado para equipamentos certificados com SIL é excelente e talvez seja essa a melhor motivação para que tolerância a falhas e engenharia de software sejam bem recebidas no ambiente industrial de desenvolvimento e produção de equipamentos para sistemas relacionados à segurança. Da perspectiva do fabricante, a certificação de um equipamento aumenta enormemente a credibilidade do seu produto (Bard and Petprayoon 2013).

O projeto de SAIM é um movimento no sentido de dominar a tecnologia de desenvolvimento de equipamentos para sistemas de segurança. A função do nosso grupo é estudar os problemas relacionados e ajudar no processo de projeto e certificação (Vidal et al. 2014) (Bandeira et al. 2013). A empresa parceira, Altus SA, já recebeu e aprovou o resultado da primeira etapa do projeto de SAIM que foi realizada no nosso grupo, nas dependências da Universidade. O resultado desta etapa consiste na arquitetura de hardware do sistema, relatada aqui, e nas especificações de segurança correspondentes a esta arquitetura.

O documento que contém as especificações de segurança detalha não apenas a estrutura de blocos dos componentes, mas também a especificação elétrica e mecânica do módulo de entrada analógica. Todos os requisitos de projeto de hardware, tanto os relacionados à segurança como os requisitos não relacionados à segurança, fazem parte deste documento, que na versão atual tem a dimensão de 80 páginas. O documento precisa agora ser aprovado pela agência certificadora para a prototipação em hardware e o desenvolvimento do software possam ser iniciados. O encaminhamento deve ser pela empresa interessada na produção, que será também avaliada para fins de certificação de produto.

Como o processo de certificação envolve a empresa e também o seu processo de produção do produto, o desenvolvimento precisa ser conduzido com forte cooperação entre a empresa e o grupo de pesquisa. O aprendizado para ambas as partes tem sido o melhor resultado do projeto até o momento.

Agradecimentos

O trabalho foi financiado com recursos BNDES no âmbito do projeto SDCD.

Referências

Bandeira, Diego, Taisy S. Weber, Sergio L. Cechin, Rodrigo Dobler, and João C. Netto. 2013. “Assistente Para Desenvolvimento de Software Crítico Segundo a IEC 61508.” In *Workshop de Tolerância a Falhas*, 17–30. Brasília, DF: SBC.

Bard, Irv, and Patchanee Petprayoon. 2013. “Successful Compliance with IEC 61508 Safety Standards.” CT316. *IBM developerWorks*. May 21. http://www.ibm.com/developerworks/rational/library/compliance-IEC-61508-safety-standards/.

Bell, Ron. 2006. “Introduction to IEC 61508.” In *Proceedings of the 10th Australian Workshop on Safety Critical Systems and Software-Volume 55*, 3–12. Australian Computer Society, Inc.

———. 2011. “Introduction and Revision of IEC 61508.” In *Advances in Systems Safety*, edited by Chris Dale and Tom Anderson, 273–91. Springer London.

Bilich, Carlos, and Zaijun Hu. 2009. “Experiences with the Certification of a Generic Functional Safety Management Structure According to IEC 61508.” In *Computer Safety, Reliability, and Security*, edited by Bettina Buth, Gerd Rabe, and Till Seyfarth, 5775:103–17. Lecture Notes in Computer Science. Springer Berlin / Heidelberg.

Gall, Heinz, and Joachim Wen. 2010. “Functional Safety IEC 61508 and Sector Standards for Machinery and Process Industry the Impact to Certification and Users Including IEC 61508 2nd Edition.” In *23rd International Congress on Condition Monitoring and Diagnostic Engineering Management, COMADEM 2010, June 28, 2010 - July 2, 2010*, 73–81. Nara, Japan: Sunrise Publishing Limited.

Jin, H., M. A. Lundteigen, and M. Rausand. 2011. “Reliability Performance of Safety Instrumented Systems: A Common Approach for Both Low-and High-Demand Mode of Operation.” *Reliability Engineering & System Safety* 96 (3): 365–373.

Joannou, Paul, and Alan Wassyng. 2014. “Understanding Integrity Level Concepts.” *Computer*, no. 11: 99–101.

Lin, Zhihong, and Stephanie Pearson. 2013. “An inside Look at Industrial Ethernet Communication Protocols.” *White Paper Texas Instruments*..

Lloyd, M. H., and P. J. Reeve. 2009. “IEC 61508 and IEC 61511 Assessments-Some Lessons Learned.” *4th IET International Conference on Systems Safety*, 2A1.

Lundteigen, Mary Ann, and Marvin Rausand. 2009. “Architectural Constraints in IEC 61508: Do They Have the Intended Effect?” *Reliability Engineering & System Safety* 94 (2): 520–25.

Lundteigen, Mary Ann, Marvin Rausand, and Ingrid Bouwer Utne. 2009. “Integrating RAMS Engineering and Management with the Safety Life Cycle of IEC 61508.” *Reliability Engineering & System Safety* 94 (12): 1894–1903.

Malik, Robi, and Reinhard Mühlfeld. 2003. “A Case Study in Verification of UML Statecharts: The PROFIsafe Protocol.” *J. UCS* 9 (2): 138–51.

Neumann, Peter. 2007. “Communication in Industrial automation—What Is Going On?” *Control Engineering Practice* 15 (11): 1332–47.

Panesar-Walawege, Rajwinder Kaur, Mehrdad Sabetzadeh, Lionel Briand, and Thierry Coq. 2010. “Characterizing the Chain of Evidence for Software Safety Cases: A Conceptual Model Based on the IEC 61508 Standard.” In *3rd International Conference on Software Testing, Verification and Validation, ICST 2010, April 7, 2010 - April 9, 2010*, 335–44. ICST 2010 - 3rd International Conference on Software Testing, Verification and Validation. Paris, France: IEEE Computer Society.

Rostan, Martin, Joseph E. Stubbs, and Dmitry Dzilno. 2010. “EtherCAT Enabled Advanced Control Architecture.” In *2010 IEEE/SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*, 39–44. IEEE.

Vidal, William, Rodrigo Dobler, Sérgio Cechin, Taisy Weber, and João Netto. 2014. “Aplicação Da IEC 61508 Na Prototipação de Protocolos Seguros de Comunicação.” In *Workshop de Testes E Tolerância a Falhas*, 147–59. Florianópolis: Sociedade Brasileira de Computação.