



XXIV SNPTEE  
SEMINÁRIO NACIONAL DE PRODUÇÃO E  
TRANSMISSÃO DE ENERGIA ELÉTRICA

CB/GPC/05

22 a 25 de outubro de 2017  
Curitiba - PR

**GRUPO - V**  
**ESTUDO DE PROTEÇÃO, MEDIÇÃO, CONTROLE E AUTOMAÇÃO EM SISTEMAS DE POTÊNCIA – GPC**

**PROPOSTA DE AUTOMAÇÃO DISTRIBUÍDA DE  
UM BANCO DE TRANSFORMADORES REGULADORES USANDO A NORMA IEC 61499**

**Marcos Fonseca Mendes \***  
ITAIPU BINACIONAL

**Bruna Pletikoszits Andrade Parcianello**  
UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ – UNIOESTE

**RESUMO**

Alguns processos elétricos podem ser automatizados utilizando uma arquitetura distribuída. As tecnologias de automação e de comunicação atuais facilitam esse tipo de solução. A norma IEC 61499 é um marco nesse sentido. Ela permite a execução dos blocos de funções por orientação a eventos. Cada bloco funcional básico pode ter vários algoritmos, além de um gráfico de controle de execução, contendo: estados, transições e ações. O artigo aborda esses conceitos e também os modelos definidos na norma IEC 61499. É apresentada uma proposta para automação distribuída de um banco de transformadores reguladores, usando vários CLP's, com controles automático e manual.

**PALAVRAS-CHAVE**

Automação Distribuída, Controladores Lógicos Programáveis, Norma IEC 61499, Paralelismo de Reguladores de Tensão, Transformadores Reguladores de Tensão.

**1.0 - INTRODUÇÃO**

Alguns processos do setor elétrico envolvem diversos equipamentos eletromecânicos distantes que poderiam ser automatizados de maneira distribuída. Entre eles, pode ser citado o paralelismo de transformadores reguladores de tensão (1). A automação desse processo vem sendo realizada desde a forma convencional, usando lógicas com relés eletromecânicos, até a forma moderna, utilizando dispositivos eletrônicos inteligentes (IEDs, do Inglês: *Intelligent Electronic Devices*). Entretanto, uma abordagem distribuída pode ser explorada usando uma ferramenta específica para isso: a norma IEC 61499 "Function Blocks" (2, 3, 4).

A norma IEC 61499 modela os sistemas de medição e controle de processos industriais como uma coleção de dispositivos físicos comunicando entre si através de redes de comunicação de dados. As funções do sistema de medição e controle são modeladas como aplicações que podem ser hospedadas em um único dispositivo físico ou podem ser distribuídas entre vários dispositivos. Como diferencial, a norma possibilita a execução dos algoritmos (contidos em blocos funcionais) a partir de eventos.

O artigo apresenta uma proposta para automação de um banco de transformadores reguladores utilizando Controladores Lógicos Programáveis (CLP's) em uma arquitetura distribuída. A seguir são apresentados, muito brevemente, os transformadores reguladores, seu funcionamento e controle.

**1.1 Transformadores Reguladores com OLTC e Paralelismo**

(\*) ITAIPU Binacional – Div. de Eng. Eletrônica e Sistemas de Controle / ENES.DT – Av. Pres. Tancredo Neves, 6731 - CEP: 85.866-900 – Foz do Iguaçu / PR – Brasil. Tel: +55 (45) 3520-3650 – E-mail: mendes@ieee.org

As variações na tensão da rede de distribuição de energia, causadas por quedas de tensão ou pela variação das cargas, podem afetar a vida útil dos equipamentos alimentados ou até mesmo desequilibrar o sistema. Uma das maneiras de contornar tal problema é o uso de transformadores reguladores equipados com comutadores de *tap*.

O comutador de *tap* (1, 5, 6) altera a tensão de saída do transformador regulador (através da variação da relação de transformação) com o objetivo de mantê-la próxima de um valor de referência pré-estabelecido. Alterando-se a tensão de saída do transformador é possível controlar a tensão na barra a qual ele está conectado. Quando esse processo é realizado em carga, o comutador é chamado de *On Load Tap Changer* (OLTC). Os OLTC's são utilizados geralmente em sistemas de médias e altas potências como, por exemplo, em subestações.

Para garantir maior disponibilidade e maximizar a potência oferecida ao sistema, as concessionárias têm como prática comum a ligação de transformadores de potência em paralelo (1, 7).

Existem vários métodos para propiciar a operação de transformadores reguladores em paralelo. Um dos métodos mais utilizados é o denominado "mestre/escravo", adotado no estudo de caso deste trabalho. Seu funcionamento baseia-se em eleger um dos OLTC's como mestre e todos os outros OLTC's são escravos, de modo que, quando o mestre muda a posição de seu *tap*, os *taps* dos escravos seguem a posição do mestre, simultaneamente. O posicionamento do *tap* do mestre pode ser feito manualmente ou automaticamente.

O trabalho apresenta uma solução de automação distribuída para atuar nas posições dos *taps* de um banco de transformadores reguladores com OLTC's. O método de controle de tensão do tipo mestre/escravo é utilizado. A solução é baseada na norma IEC 61499 "Function Blocks" introduzida na próxima seção.

## 2.0 - NORMA IEC 61499

A norma IEC 61499 "Function Blocks" foi criada em 2005 e teve sua última edição publicada em 2012. Atualmente ela é dividida em três partes: IEC 61499-1 – "Architecture"; IEC 61499-2 – "Software tool requirements"; IEC 61499-4 – "Rules for compliance profile".

O foco da norma IEC 61499 é estabelecer um padrão de arquitetura para o uso dos chamados *Function Blocks* (FB's). O conceito de FB, já abordado pela norma IEC 61131 – "Programmable controllers" (9), foi ampliado. O diferencial está no fato dos FB's contarem com uma interface de execução orientada a eventos, além da tradicional interface de dados. Além do mais, a norma IEC 61499 pode ser vista como uma ferramenta específica para o desenvolvimento de sistemas de controle distribuído. A norma tem facilidades para a modelagem de sistemas de controle de processos como uma série de dispositivos físicos interconectados através de redes de comunicação.

As subseções a seguir apresentam alguns conceitos e definições fundamentais da norma IEC 61499.

### 2.1 Function Blocks

Os FB's são as entidades básicas da norma IEC 61499. Eles têm "conexões de dados" e "conexões de eventos". A Figura 1 ilustra o modelo de um FB, com suas entradas, saídas e componentes internos.

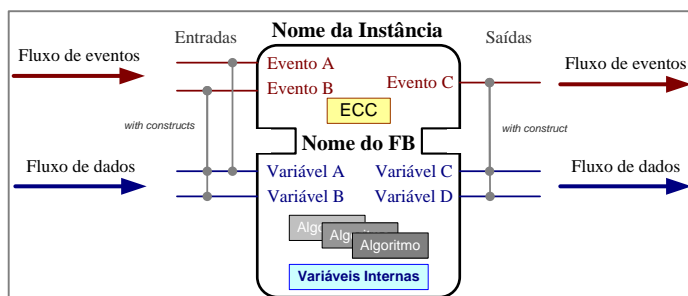


FIGURA 1 – Modelo de FB

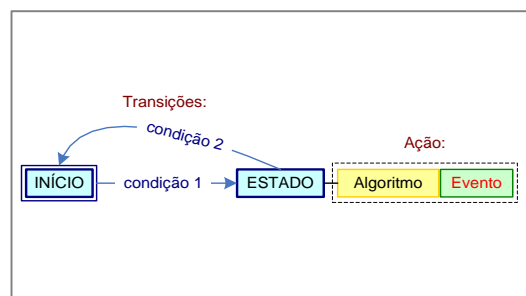


FIGURA 2 – Exemplo de ECC

Cada FB é dividido em duas partes: a parte superior (parte dinâmica) é responsável pela execução de eventos e a parte inferior (parte funcional) representa os dados e algoritmos. O funcionamento do FB associa essas duas partes através de um recurso chamado "with construct" (mostrado na figura), que sincroniza a ocorrência de eventos com a leitura e escrita de variáveis. Basicamente, ao ser detectado um evento de entrada em um FB, dá-se início a leitura de suas variáveis de entrada e a execução de seus algoritmos – que podem ser escritos em várias linguagens (como as definidas na parte 3 "Programming languages" da norma IEC 61131). Em seguida, de acordo com os algoritmos, podem ser gerados eventos de saída do FB, que por sua vez, disponibilizam variáveis na saída.

#### 2.1.1 Tipos de Function Blocks

A norma IEC 61499 define quatro tipos de FB's descritos a seguir (2, 3, 4, 10):

- *Basic Function Block* (BFB): é o tipo mais elementar do gênero. Sua característica principal é ter um ou mais algoritmos, variáveis internas e, obrigatoriamente, um *Execution Control Chart* (ECC). Um ECC é similar a uma máquina de estados que conta com estados, transições e ações que são responsáveis por definir o comportamento de um BFB. A Figura 2 apresenta um exemplo simples de ECC;
- *Composite Function Block* (CFB): é um conjunto de instâncias de FB's. Um CFB não tem variáveis internas e nem ECC. Além disso, sua ordem de execução é definida pelo fluxo de eventos de entradas e saídas dos FB's que o compõe. Sua grande utilidade é o encapsulamento, que simplifica a apresentação das aplicações, de modo a substituir uma rede de FB's por apenas um CFB;
- *Event Function Block* (EFB): é um tipo de FB pré-estabelecido pela norma cuja função é manipular eventos, seja para gerar um evento inicial ou para disparar simultaneamente múltiplas cadeias de FB's;
- *Service Interface Function Block* (SIFB): é utilizado onde há a necessidade de interação entre FB's e o mundo externo. Em sistemas de controle é comum a leitura de valores de entradas físicas, tais como sensores, ou a escrita de valores em saídas, para atuadores, por exemplo. Há também a necessidade de transmitir e receber dados entre dispositivos remotos através de redes de comunicação.

### 2.1.2 Service Interface Function Blocks

Existem dois comportamentos básicos que um SIFB pode assumir: "*responder*" e "*requester*" (2). Um SIFB *requester* é do tipo "*application-initiated*", sendo ativado somente se receber um evento de entrada vindo diretamente do fluxo da aplicação. Já um SIFB *responder* é do tipo "*resource-initiated*", sendo ativado por um recurso externo, independente do fluxo de eventos dentro da aplicação. Esse conceito será exemplificado a seguir.

Como exemplos podem ser citados os SIFB's para acessar as entradas e saídas de dispositivos físicos:

- *IO\_Write*: é usado para atualizar o valor de atuadores. Ele é do tipo *requester*, ou seja, escreve, quando solicitado por um evento vindo de dentro da aplicação, um valor diretamente no endereço físico de saída de um dispositivo;
- *IO\_Read*: é usado para ler valores de sensores. Ele lê constantemente ou por solicitação o valor de uma entrada física de dispositivos aos quais podem estar conectados elementos tais como botoeiras e relés. Assim, ele pode ser do tipo *responder* ou *requester*. Quando é do tipo *responder*, ao ser detectada uma mudança no valor na entrada física, ele é ativado e disponibiliza o valor lido para a aplicação.

Para serviços de comunicação unidirecionais, a norma especifica os SIFB's "*publish*" e "*subscribe*", descritos abaixo (e utilizados neste trabalho). Eles podem utilizar o protocolo UDP para transmissões do tipo *multicast*. Assim, eles podem enviar uma mesma mensagem simultaneamente de um para vários destinatários.

- *Publish*: é um SIFB do tipo *requester*, pois publica um dado na rede de comunicação, no endereço e porta especificados, somente se existir uma solicitação, representada por um evento de entrada;
- *Subscribe*: é um SIFB do tipo *responder*, pois está constantemente monitorando a rede de comunicação. Caso ele detecte, em qualquer momento, o recebimento de dados vindos de um dispositivo remoto, no endereço e porta especificados, ele é ativado e disponibiliza os dados recebidos para a aplicação.

A norma também especifica serviços de comunicação bidirecionais utilizando os SIFB's "*client*" e "*server*" (2).

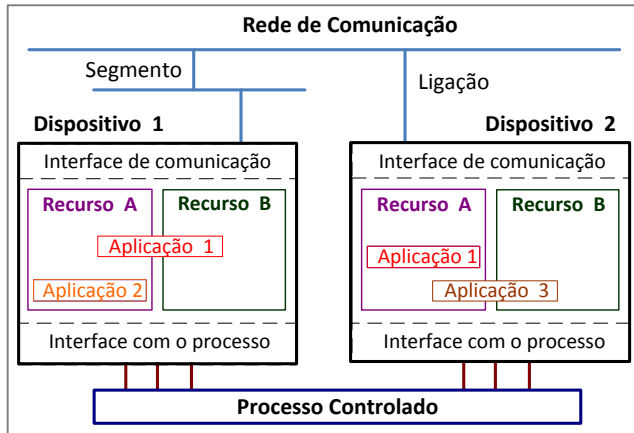
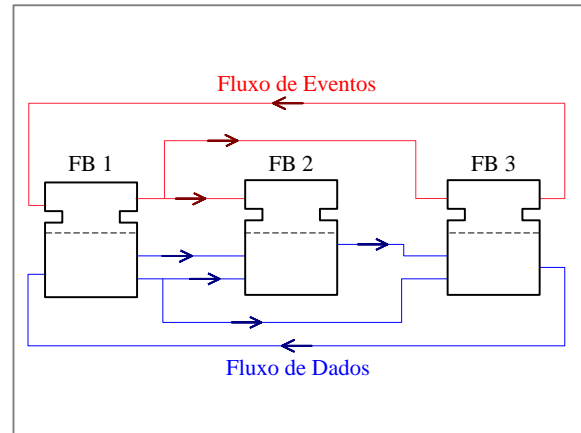
A norma IEC 61499 não define SIFB's específicos, pois cada sistema de controle pode ter diferentes funcionalidades e requisitos. No entanto, ela padroniza entradas (eventos REQ e RSP e dados SD's) e saídas (eventos CNF e IND e dados RD's) para os SIFB's do tipo *requester* e *responder* (2). Existe também uma variável (PARAMS) utilizada para endereçamentos e outra (STATUS) para informações de status (2).

### 2.2 Modelos de Referência

Para definir e modelar os sistemas de medição e controle de processos industriais, a norma IEC 61499 introduz uma arquitetura genérica. Essa arquitetura é composta dos modelos de referência apresentados a seguir (2, 3, 4):

- *Modelo de dispositivo (Device model)*: consiste em um dispositivo físico (como, por exemplo, um CLP) que realiza uma malha de controle e atua sobre um determinado processo. Um dispositivo é o elemento que oferece a interface entre a rede de comunicação e o processo a ser controlado. Além do mais, ele pode conter um ou mais recursos nos quais serão executadas as aplicações;
- *Modelo de recurso (Resource model)*: pode ser entendido como uma unidade funcional que controla independentemente a operação de um dispositivo. Um recurso funciona como um "*container*" para as aplicações, oferecendo ferramentas e serviços de suporte para que elas possam ser executadas;
- *Modelo de sistema (System model)*: consiste em uma série de dispositivos conectados a uma rede de comunicação de dados através de segmentos ou ligações diretas. Um sistema relaciona a comunicação entre esses dispositivos e suas respectivas aplicações, como ilustrado na Figura 3;

- *Modelo de aplicação (Application model)*: é basicamente uma combinação de instâncias de FB's usadas para resolver problemas particulares de medição, automação e controle. A relação entre o fluxo de dados e de eventos requeridos entre as diferentes instâncias de FB's é estabelecida na aplicação. Além disso, uma aplicação pode ser dividida e distribuída entre múltiplos recursos, podendo estar contida em um ou mais dispositivos, como ilustrado na Figura 3. A Figura 4 mostra um exemplo de aplicação e seus fluxos.

FIGURA 3 – Exemplo de *System model*FIGURA 4 – Exemplo de *Application model*

### 3.0 - ESTUDO DE CASO

O processo do estudo de caso consiste em um banco de cinco transformadores abaixadores trifásicos (T1 a T5) de potência nominal de 425 MVA e relação de transformação 500/220 kV conectados a transformadores reguladores trifásicos (R1 a R5) de mesma potência. Esses reguladores são equipados com OLTC's, para manter o nível de tensão desejado nas barras de 220 kV. Cada regulador dispõe de 19 posições de *tap* com tempo de comutação médio de 5 segundos, causando uma variação em torno de 2,0 kV por posição. A Figura 5 ilustra a planta.

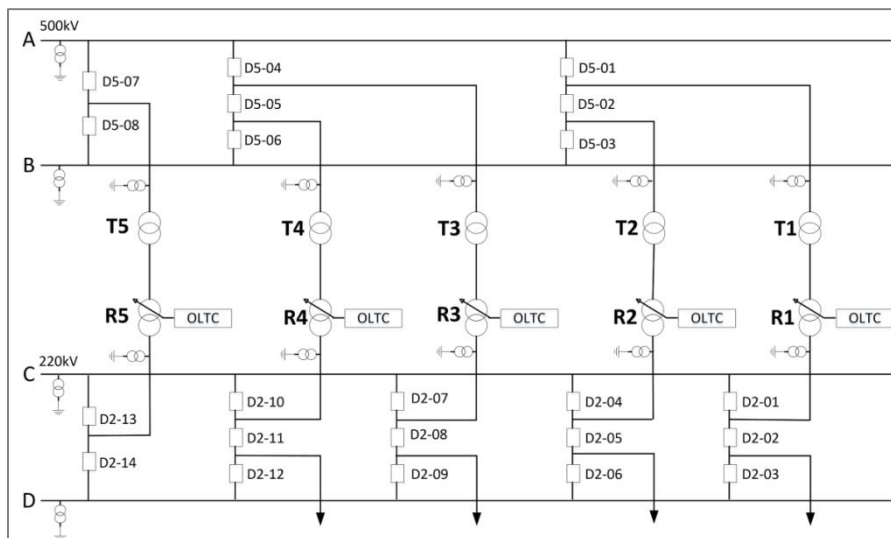


FIGURA 5 – Diagrama unifilar do estudo de caso

#### 3.1 Modos de Operação dos Transformadores Reguladores

Uma vez conectados ao sistema, os transformadores reguladores podem operar em diversos modos. Toda a lógica de controle dos transformadores envolve esses modos de operação.

Primeiramente, a localização do controle deve ser selecionada, por uma chave, em modo local ou remoto (ou ainda desligado). No modo local os comandos são dados diretamente nos OLTC's. No modo remoto os comandos são emitidos a partir de uma IHM da sala de controle.

Com relação ao paralelismo dos transformadores reguladores, cada um deles tem uma chave que os seleciona em: mestre, escravo ou individual. De acordo com a configuração dessas chaves o estado operativo pode ser definido como: paralelo (também chamado conjunto) ou individual<sup>1</sup>.

O controle paralelo baseia-se na configuração que exista um OLTC selecionado como mestre e quatro como escravos. Esse modo de operação permite tanto o controle automático quanto o manual. No modo automático a posição do *tap* do mestre é alterada automaticamente através de uma malha de controle realimentada, usando uma tensão de referência e o valor medido da barra. No controle manual, o *tap* do mestre é posicionado pelo operador.

No controle individual, a posição do *tap* de cada regulador assim selecionado é controlada manualmente por um operador. Na solução adotada neste trabalho, para esse tipo de controle não é necessário que todos os cinco OLTC's estejam selecionados em individual. Note que, o controle automático só é possível no modo paralelo.

Para comutação de *taps* automática, três parâmetros que devem ser considerados (8): a tensão de referência, a largura de faixa do erro e a temporização. A tensão de referência é a tensão desejada na saída do transformador regulador (barramento secundário). A largura de faixa é definida como uma porcentagem da tensão de referência que representa o intervalo de tensão permitido ao transformador operar. A temporização por sua vez, refere-se ao tempo intencional de espera para iniciar a comutação de *taps*. Dessa maneira, é necessário um intervalo de tempo com o valor de tensão medido fora da largura de faixa, para que então se comande uma mudança de *tap*. A existência da largura de faixa e da temporização preserva a vida útil do comutador, evitando comutações desnecessárias causadas por pequenas oscilações de tensão. Neste trabalho a temporização é variável.

### 3.2 Comando e Acionamento do Comutador de *Taps*

A Figura 6 apresenta o diagrama de comando e acionamento do motor trifásico para mudança de *taps*. O comando é realizado por uma série de relés, contatores e chaves fim de curso acionadas por came, e selecionado pela chave LDR (Local / Desligado / Remoto). Além disso, é possível notar os comandos por botoeiras (modo local) e pelo CLP (modo remoto). Note também que, o circuito de acionamento contém alguns elementos para proteção.

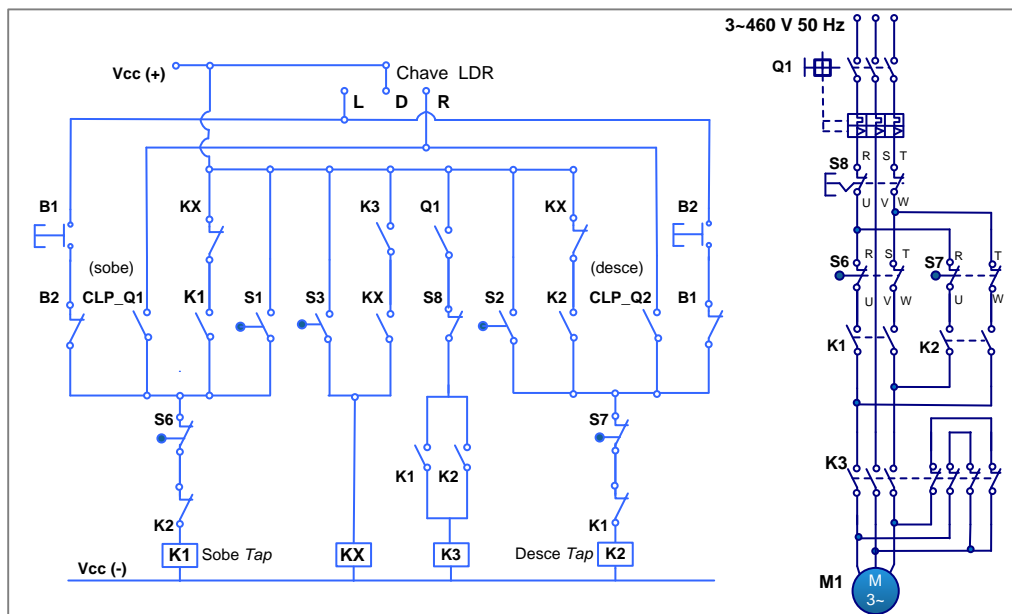


FIGURA 6 – Comando e acionamento do motor do comutador de *taps*

Os contatores K1, K2 e K3, são responsáveis por energizar o motor (K3 é usado para frenagem). As chaves acionadas por came S1 e S2 indicam o término de subida e descida de *tap*, respectivamente. A chave S3 indica que o motor está em rotação (inversor para manobra passo a passo). As chaves S6 e S7 indicam os fins de cursos do comutador de *taps*. A chave S8 é usada para bloqueio do comando, devido seleção para operação à manivela.

Note que, parte da lógica para o controle remoto está no circuito de comando, uma vez que de toda forma essa lógica é necessária para o controle local (usando as botoeiras B1 e B2, sem passar pelo CLP). Se desejado, toda a lógica para o controle remoto poderia estar nos CLP's. De toda forma, a solução apresentada acima é mais segura.

### 3.3 Solução Proposta

<sup>1</sup> Pode ser estabelecido um terceiro modo: indefinido (não está no modo paralelo e não estão todos em individual).

É proposto o uso de seis pequenos CLP's, um para cada transformador regulador mais um CLP para o controle remoto. Os CLP's são os "Dispositivos" representados na Figura 3. Além disso, existe um dispositivo usado como Interface Humano-Máquina (IHM). A transmissão de dados entre os CLP's é feita através de uma rede local Ethernet gerenciada por um *switch*. A Figura 7 ilustra a arquitetura de *hardware* proposta.

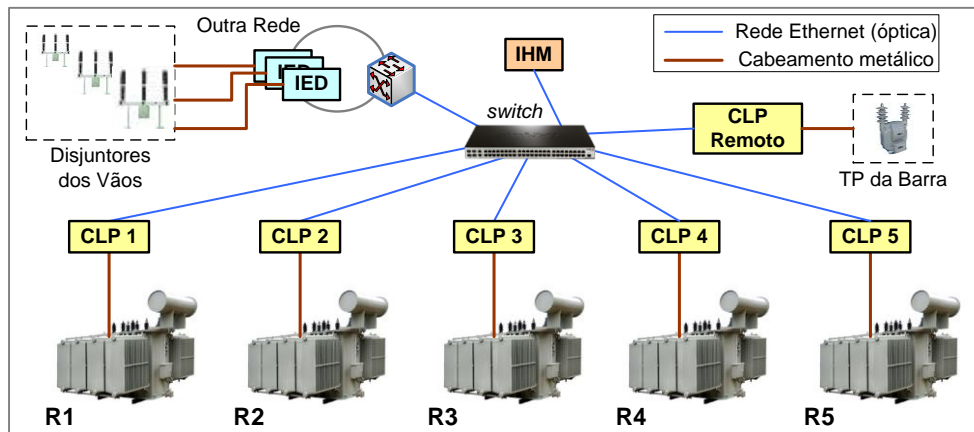


FIGURA 7 – Arquitetura de *hardware* da solução proposta

Note que, a figura também inclui os IEDs (que também poderiam ser CLP's) que fornecem os dados das posições dos disjuntores (e chaves seccionadoras). Esses dados são utilizados no CLP remoto para saber se cada um dos transformadores reguladores está conectado ou não às barras (FB "ESTADO\_TRAFOS", ver Figura 11).

A Figura 8 apresenta resumidamente a aplicação dos CLP's dos transformadores reguladores (o CLP de cada regulador contém essa aplicação). As conexões em vermelho representam os fluxos de eventos e as conexões em azul representam os fluxos de dados. Note que, foi utilizada a notação "\_X" para se referir ao regulador número "X", onde há necessidade de identificá-lo. Note também que, os endereços e parâmetros apresentados são simbólicos.

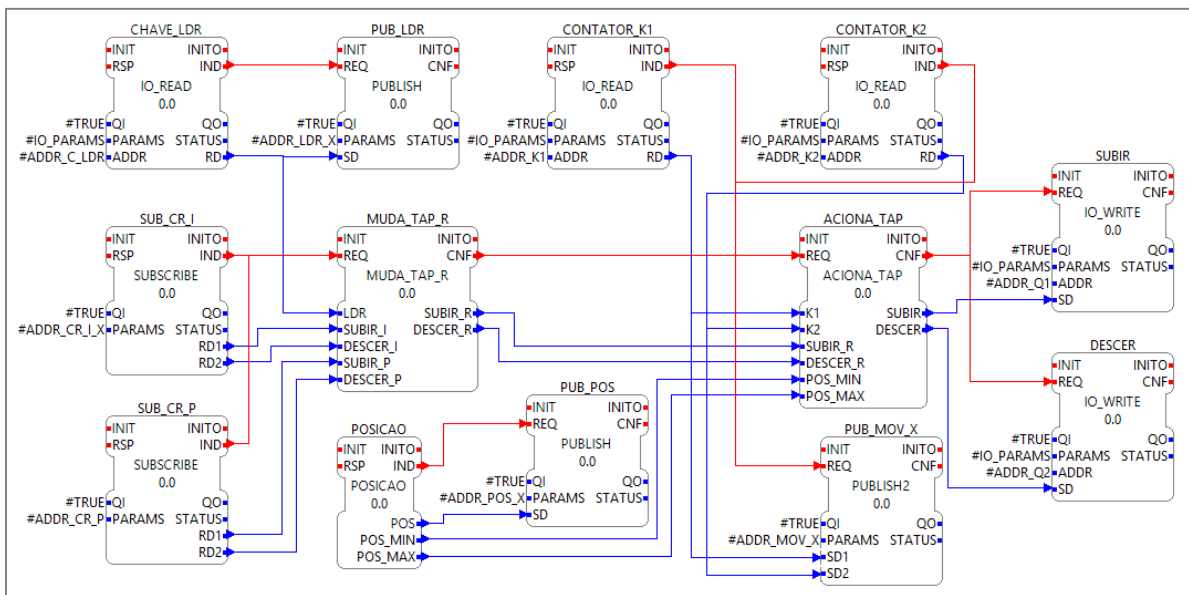


FIGURA 8 – Aplicação do CLP do transformador regulador

Para simplificar, os FB's "E\_MERGE" e "E\_SPLIT" (2) não são apresentados nas figuras. Também para simplificação, as conexões dos eventos INIT e INITO não são mostradas, tampouco as ligações "with construct".

Basicamente a aplicação recebe os comandos para subir ou descer os *taps* provenientes do CLP remoto, tanto no modo paralelo (FB "SUB\_CR\_P") como no modo individual (FB "SUB\_CR\_I"). Em seguida é verificada se a localização do controle é remota (FB "MUDA\_TAP\_R") e é feito o acionamento do motor do OLTC para mudança de *tap* (FB "ACIONA\_TAP"), de acordo com outros intertravamentos (os *taps* não estão movimentando e não estão nas posições limite). O estado da chave LDR e os estados dos contadores K1 e K2, que indicam que o *tap* está se movimentando, são transmitidos para o CLP remoto.

Note que, poderiam ter sido definidos eventos distintos para subir e descer *tap*. Além disso, alguns intertravamentos já realizados no CLP remoto (Figura 11) também são realizados aqui para aumentar a segurança.

Note também que, na aplicação existem alguns CFB's. Para ter uma ideia da construção dos CFB's, a Figura 9 apresenta, como exemplo, o CFB "POSICAO" (não estão representados todos os blocos "IO\_READ").

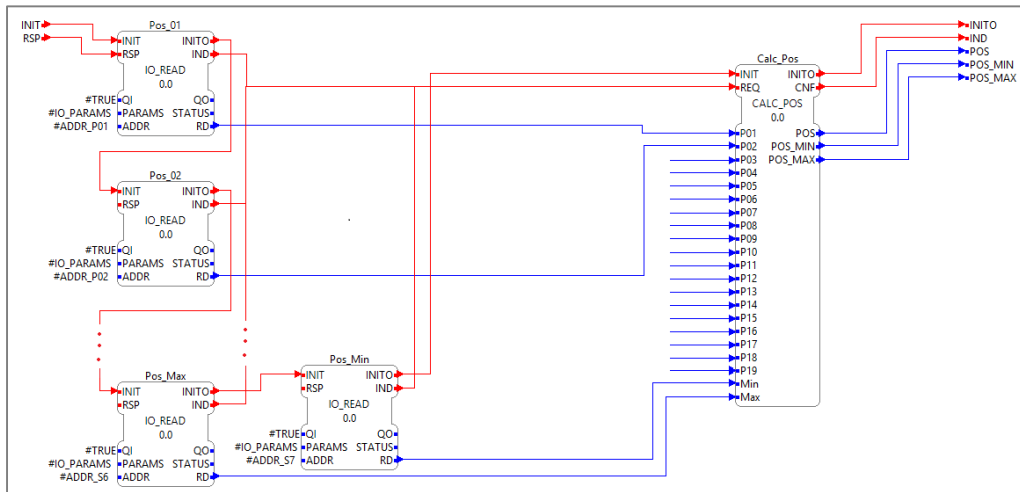


FIGURA 9 – CFB "POSICAO"

O CFB "POSICAO" recebe os estados de todas as chaves indicadoras de posições do *tap*, bem como as indicações de posições mínima e máxima através das chaves S6 e S7 (FB's "IO\_READ"). Em seguida, o FB "CALC\_POS" fornece a posição em que se encontra o *tap* (1 a 19) em uma variável do tipo *byte* (POS), além de identificar se o comutador está em seus limites máximo ou mínimo (variáveis booleanas POS\_MIN e POS\_MAX).

Conforme estabelecido na Seção 2, cada BFB tem algoritmos e um ECC. Como exemplo, a Figura 10 apresenta a interface, o ECC e o algoritmo principal (Alg\_ACIONA) do BFB "ACIONA\_TAP", que gera os comandos que vão para o circuito de acionamento do motor do OLTC. Todos os algoritmos dos BFB's deste trabalho foram desenvolvidos na linguagem *Structured Text* (ST), especificada na norma IEC 61131. Os comandos para mudar o *tap*, fornecidos pelo FB "ACIONA\_TAP" (variáveis SUBIR e DESCER), são enviados para as saídas do CLP através dos FB's "IO\_WRITE", uma vez gerado o evento CNF do FB "ACIONA\_TAP".

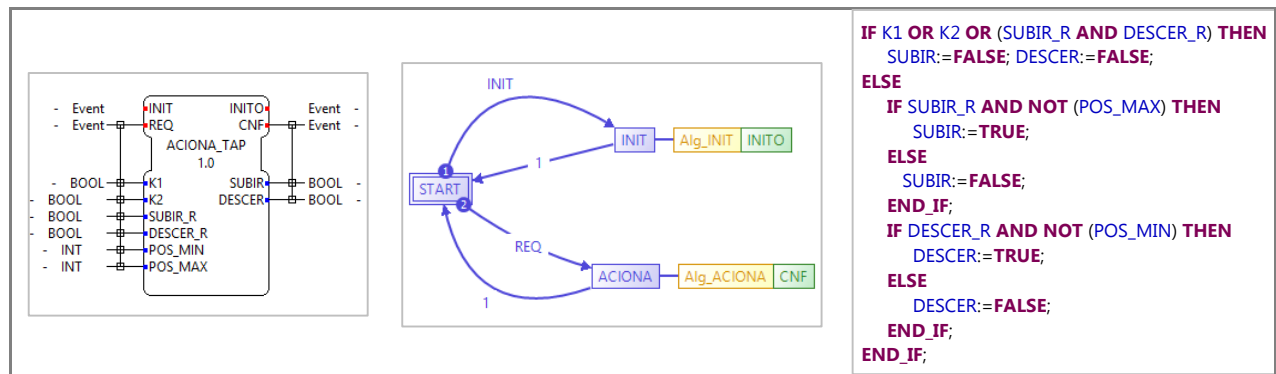


FIGURA 10 – Interface, ECC e Algoritmo "Alg\_ACIONA" do BFB "ACIONA\_TAP"

A Figura 11 apresenta resumidamente a aplicação do CLP remoto referente ao controle paralelo (automático e manual). As aplicações de paralelismo e de comandos remotos individuais são realizadas nesse CLP.

Os três FB's à esquerda fornecem os dados de estados e seleções dos CLP's dos transformadores reguladores. O FB "MODO\_PARALELO" determina se o sistema está no modo paralelo e, caso positivo, inicia os ciclos de controle usando o EFB "E\_CYCLE". O FB "MOVENDO\_TRAFOS" é utilizado para verificar se todos os comutadores de *taps* estão parados. Em cada ciclo é verificada a tensão da barra (FB "V\_MED"), calculado o erro (FB "CALC\_ERROR"), verificado se o erro está fora da zona morta (FB "ZONA\_MORTA") e esperado o tempo atraso (FB "ATRASO") calculado de acordo com o erro (FB "CALC\_ATRASO"). Uma vez determinado que devam ser gerados comandos para mudar os *taps* (FB "MESTRE"), eles são enviados para os CLP's dos transformadores reguladores (FB "PUB\_CR\_P"). Note que, o FB "MESTRE" também é responsável por processar os comandos manuais no modo paralelo, provenientes do FB "MESTRE\_MANUAL" (um CFB que recebe comandos da IHM).

As posições dos *taps* dos transformadores reguladores fornecidas pelo FB “POSICOES” poderiam ser utilizadas, por exemplo, para supervisão, confirmação de mudanças de *taps* ou para outros intertravamentos.

A Figura 12 apresenta resumidamente a aplicação do CLP remoto referente ao controle individual (manual), para cada transformador regulador (diferenciados por “\_X”).

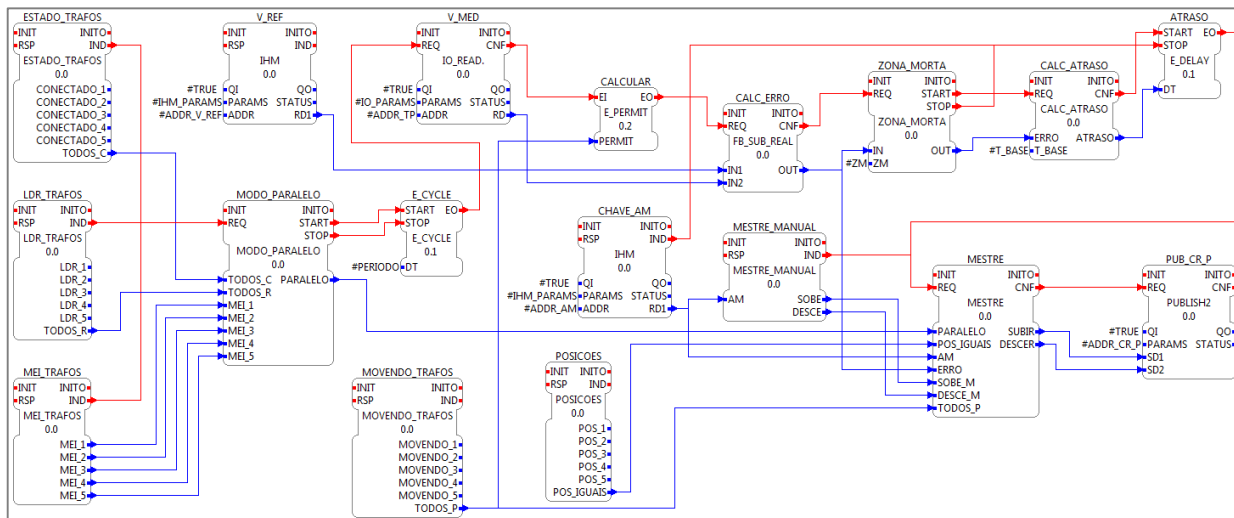


FIGURA 11 – Aplicação do CLP remoto (controle de paralelismo – automático e manual)

Os comandos para mudar individualmente o *tap* de cada regulador são recebidos da IHM (FB “TR\_X\_SD\_M”). As condições de intertravamentos são verificadas (FB “VERIFICA\_I\_X”) e os possíveis comandos são transmitidos para a aplicação do CLP do transformador regulador X (FB “PUB\_CR\_I\_X”). Note que, essas lógicas poderiam ser realizadas nos próprios CLP’s dos reguladores (enviando alguns dados do CLP remoto para esses CLP’s).

Naturalmente outras realizações são possíveis para resolver o problema. Por exemplo, o comando no modo paralelo poderia usar os mesmos FB’s “publish” e “subscribe” do comando no modo individual. Outro exemplo é o uso dos intertravamentos redundantes, para aumentar a segurança, que poderiam ser diferentes.

Para simplificação, alguns intertravamentos foram omitidos. As aplicações e os algoritmos também podem ser melhorados para incluir validação de dados e checagem de transmissões de mensagens entre CLP’s. Isso aumentaria a confiabilidade do sistema, e infelizmente também a complexidade.

Além disso, outras realizações são possíveis. Por exemplo, um (ou vários) CLP dos transformadores reguladores poderia hospedar a aplicação do CLP remoto, de modo que o último não seria necessário. Realizações mais distribuídas também são possíveis. Foi considerado que cada aplicação está contida em um único recurso.

Alguns ajustes de ordem prática podem ser necessários para colocar o sistema de controle em operação. A simulação do sistema completo (controle e processo) também pode ajudar a fazer esses ajustes e encontrar qualquer erro. Otimizações também podem ser feitas com ajuda de ferramentas de simulação.

As aplicações e todos os FB’s foram desenhados usando a ferramenta 4DIAC-IDE (11). Trata-se de uma ferramenta *open-source* que permite o desenvolvimento de sistemas automação e controle em conformidade com a norma IEC 61499. Essa ferramenta permite a edição e criação de novos FB’s, incluindo os ECC’s e algoritmos.

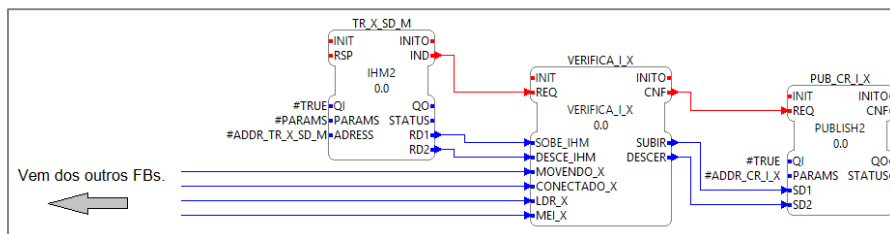


FIGURA 12 – Aplicação do CLP remoto (controle individual – manual).

#### 4.0 - CONCLUSÃO

A solução proposta é funcionalmente completa. Vários FB’s foram criados para essa solução, com ECC’s e algoritmos específicos, mostrando o poder e a versatilidade da norma IEC 61499. Existem outras diversas formas



de realizar o controle discutido no trabalho. Entretanto, a ideia principal é apresentar a norma IEC 61499 como ferramenta para a solução do tipo de controle abordado, mostrando um novo enfoque para o problema.

Além disso, outros métodos para realizar o paralelismo (como, por exemplo, o método da corrente circulante) podem ser realizados utilizando a norma IEC 61499. A mesma arquitetura de *hardware* proposta pode servir como base para outras soluções. A arquitetura proposta tem os recursos necessários para realizar qualquer controle distribuído, bastando criar os FB's, ECC's, algoritmos e eventos adequados. Em todos os casos, os CLP's também devem ter as capacidades adequadas (entradas e saídas, etc.).

A solução proposta tem vantagens sobre as soluções usuais. Entre elas, pode ser citado o comportamento mais dinâmico devido à execução orientada a eventos. Essa abordagem permite criar uma sequência lógica para interação e execução dos FB's. As aplicações de cada CLP dos transformadores reguladores são praticamente idênticas, facilitando a realização e a manutenção. Por fim, a solução proposta tem as vantagens geralmente encontradas em sistemas de automação digitais, como a flexibilidade e interface amigável, por exemplo.

As aplicações apresentadas (FB's e algoritmos) podem ser refinadas e aperfeiçoadas para realização prática.

## 5.0 - REFERÊNCIAS BIBLIOGRÁFICAS

- (1) HARLOW, J. H. **Electric Power Transformer Engineering**. 3<sup>rd</sup> Ed., CRC Press, 2012.
- (2) International Electrotechnical Commission – IEC. **IEC 61499, Function Blocks**. 2<sup>nd</sup> Ed. Geneva, 2012.
- (3) ZOITL, A.; LEWIS, R. **Modeling Control Systems Using IEC 61499**. Londres: The Institution of Engineering and Technology, 2014.
- (4) ZOITL, A.; STRASSER, T. **Distributed Control Applications: Guidelines, Design Patterns, and Application Examples with the IEC 61499**. CRC Press, 2015.
- (5) CARVALHO, E. B.; SIQUEIRA, A.; M.; NOGUEIRA, L. R. Monitoramento do *tap* dos transformadores de consumidores de média tensão. In: **XIV Encontro Latino Americano de Iniciação Científica e X Encontro Latino Americano de Pós-Graduação**. Universidade do Vale do Paraíba, p. 1-4, 2011.
- (6) VIEIRA, M. F. **Automatismo de Regulação de Tensão em Unidades Remota de Teleação**. Dissertação (Mestrado em Engenharia Electrotécnica e de Computadores) - Universidade Técnica de Lisboa, Lisboa, 2010.
- (7) RAMBO, M. V. H.; MARUYAMA, N.Y. Arquitetura de um sistema digital para comutação em paralelo do *tape* de bancos de transformadores de uma subestação compartilhada. In: **XXI SNPTEE Seminário Nacional de Produção e Transmissão de Energia Elétrica**. Florianópolis, 2011.
- (8) PARCIANELLO, B. P. A. **Estudo e Aplicação da Norma IEC 61499 no Controle de Paralelismo de Transformadores Reguladores**. Trabalho de Conclusão de Curso de Graduação (Engenharia Elétrica) - Universidade Estadual do Oeste do Paraná – UNIOESTE, Foz do Iguaçu, 2016.
- (9) INTERNATIONAL ELECTROTECHNICAL COMMISSION – IEC. **IEC 61131-3, Programmable Controllers - Part 3: Programming languages**. 3<sup>rd</sup> Ed., Geneva, 2013.
- (10) DAMASO, E. G. P. **Plataforma Configurável para Gestão de Edifícios baseada em IEC 61499**. Dissertação (Mestrado em Engenharia Eletrotécnica e de Computadores) - Faculdade de Engenharia da Universidade do Porto (EUP), Portugal, 2011.
- (11) Web Site Eclipse **4diac - The Open Source Environment for Distributed Industrial Automation and Control Systems**. Disponível em: <<https://eclipse.org/4diac/>>. Acesso em: 12 out. 2016.

## 6.0 - DADOS BIOGRÁFICOS



Marcos Fonseca Mendes é graduado em Engenharia Elétrica (ênfase em Eletrônica) pela UFMG (1996), mestre em Engenharia Elétrica (Automação, Controle e Informática Industrial) pela UFSC (1999), tem licenciatura plena pelo CEFET-MG (2000) e é doutor em Ciências (Engenharia Elétrica - Sistemas de Potência) pela USP (2011). Em 2015 realizou um pós-doutorado em Moscou / Rússia.

Atualmente Marcos é engenheiro sênior da ITAIPU Binacional (2000), na Superintendência de Engenharia (área de Controle e Automação) e professor da Universidade Estadual do Oeste do Paraná – UNIOESTE (2001), no curso de Engenharia Elétrica (área de Controle e Automação).



Bruna Pletikoszits Andrade Parcianello é graduada em Engenharia Elétrica (ênfase em Sistemas de Controle e Automação) pela Universidade Estadual do Oeste do Paraná – UNIOESTE (2017) e atualmente é mestranda no Programa de Engenharia Elétrica e Computação (PGEEC) também na UNIOESTE.