

Projeto de **STAMPS-TR**

Soluções Tecnológicas Aplicáveis às Mídias e Produtos de Saúde em Tempo Real

Instituto Tecnológico de Aeronáutica - ITA

Guia de Desenvolvimento

Integração no SCADE

Sumário

1 Introdução

2 Desenvolvimento

2.1 Passo 1

2.2 Passo 2

2.3 Passo 3

2.4 Passo 4

2.5 Passo 5

2.6 Passo 6

2.7 Passo 7

2.8 Passo 8

2.9 Passo 9

2.10 Passo 10

2.11 Passo 11

2.12 Passo 12

2.13 Passo 13

3 Considerações Finais

1 Introdução

Este documento apresenta um exemplo de integração de projetos do SCADE, como um roteiro para auxiliar os integrantes das disciplinas CES-65, CE-235, CE-230 e CE-237 ministradas no Programa de Pós-Graduação em Engenharia Eletrônica e Computação na Área de Informática (PG/EEC-I) no ITA, no 2º Semestre de 2017.

Ele foi elaborado com o objetivo de auxiliar na integração do *Interdisciplinary Problem Based Learning* (IPBL) denominado Projeto de **Soluções Tecnológicas Aplicáveis às Mídias e Produtos de Saúde em Tempo Real (STAMPS-TR)**.

2 Desenvolvimento

Este exercício foi dividido em vários passos. Para cada um deles, é fornecido um .zip com o conteúdo completo do projeto após realizado cada passo.

2.1 Passo 1

Criar a estrutura de pastas a ser usada:

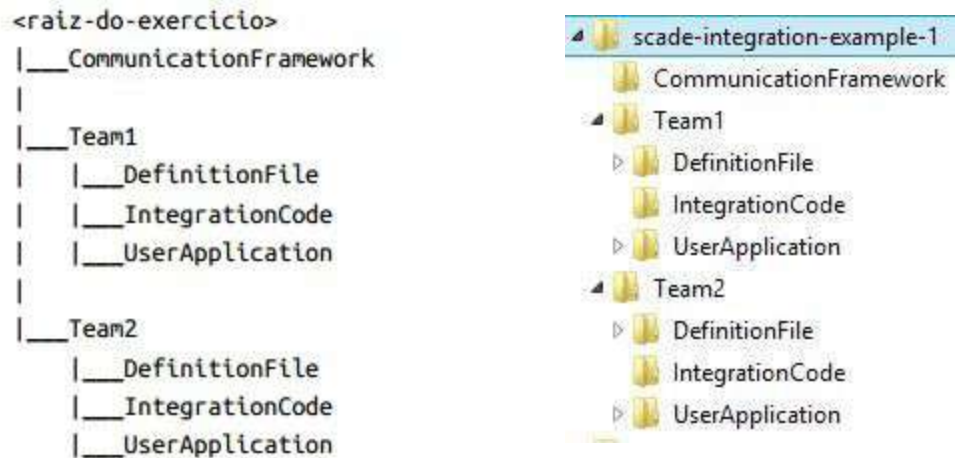


Figura 01 - Representação da Estrutura de Pasta

2.2 Passo 2

Implementar os *Definition Files* (DFs) de cada time. O objetivo é ter dois *Cockpit Display System* (CDS) com as seguintes configurações:



Figura 02 - Representação dos CDSs

```

<raiz-do-exercicio>
|__CommunicationFramework
|
|__Team1
|  |__DefinitionFile
|  |  |__project1.sgfx
|  |  |__project1DF
|  |    |__[arquivos gerados automaticamente pelo SCADE Display (Generate & Test DF)]
|  |__IntegrationCode
|  |__UserApplication
|
|__Team2
|  |__DefinitionFile
|  |  |__project2.sgfx
|  |  |__project2DF
|  |    |__[arquivos gerados automaticamente pelo SCADE Display (Generate & Test DF)]
|  |__IntegrationCode
|  |__UserApplication

```

Detalhamento da implementação:

Passo 1

1. Abrir o SCADE UA Page Creator R17;

1.1 Criar um novo Projeto para o time 1: File > New > Project

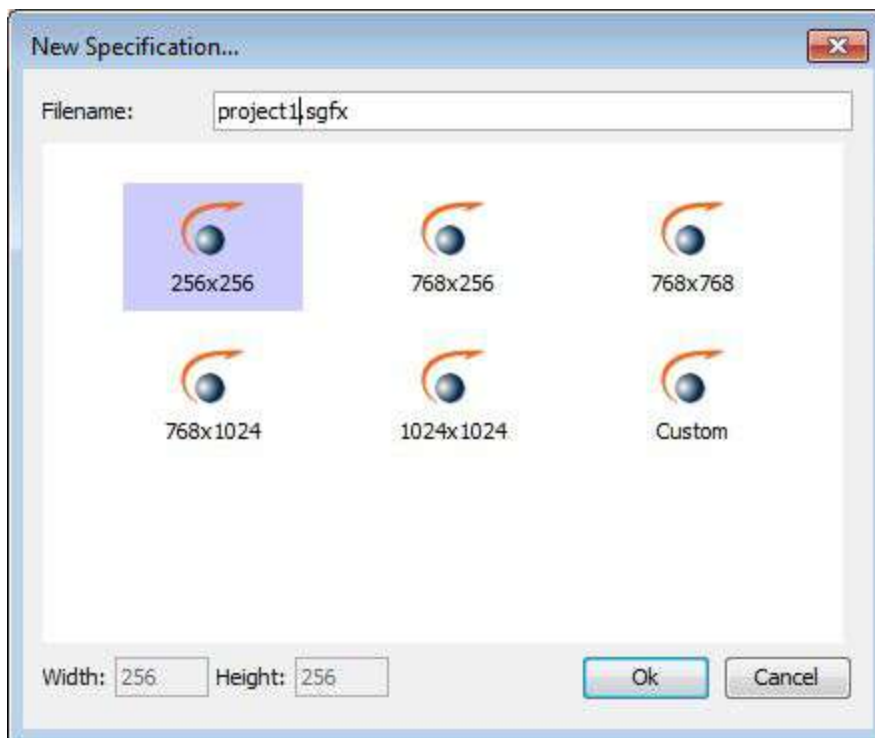
1.2 Escolha a opção **SCADE UA Page Creator**, preencha o Project Name com **project1** e defina o Location conforme abaixo:

Location: <raizdoexercicio>\Team1\DefinitionFile\

E:\Projetos\SCADE_Study\integration\Team1\DefinitionFile

Imagem 1.1: Definindo parâmetros para novo projeto

1.3 Agora crie um Definition File, clicando em File>New>Specification, escolha a opção 256x256 DF dentro do menu Specification, defina o File Name com **project1.sgfx**



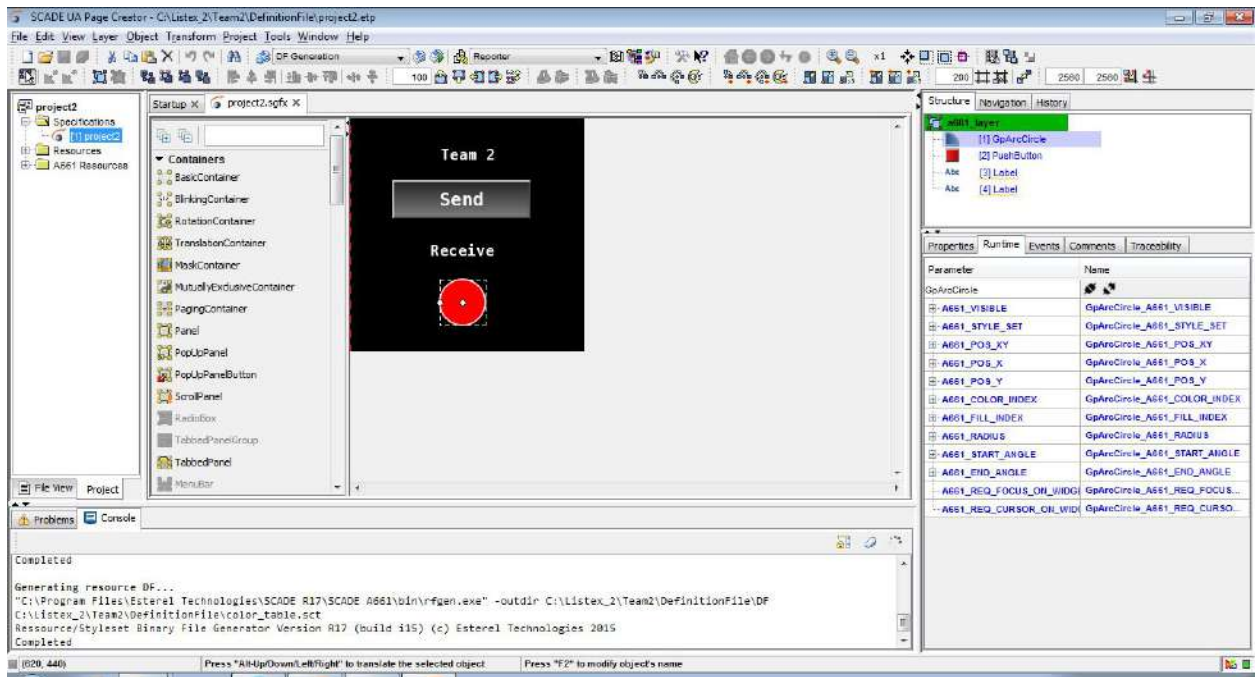
1.4 Criar, pelo menos, o PushButton e o GpArcEllipse (os Labels são opcionais, mas ajudam a identificar cada tela quando elas estiverem em execução), segundo a tabela na página seguinte.

T

Nome	Propriedades importantes
PushButton	Alignment: A661_CENTER LabelString: Send
GpArcEllipse	Start angle (float value): -180 End angle (float value): 180 Radius: 500 FillIndex: RED

1.5 Gerar os arquivos de configuração: Project / Generator / Generate. Aguardar o fim da geração.

1.5.1 Ative as mensagens do barramento ARINC 661 para cada componente criado através do menu Runtime, clicando no botão “Connect All”, conforme abaixo:



OBS: Lembre-se de repetir o processo para todos os componentes listados na sessão “*Structure*”.

1.5.2 Configure os eventos dos componentes PushButton, clicando na aba Events e Connect All.

1.6 Salvar, fechar o SCADE e repetir todo o Passo 1, desta vez criando arquivos project2 para o time 2.

2.3 Passo 3

Criar os “User Applications” de cada time nos locais corretos. O objetivo é ter os arquivos do SCADE Suite segundo a estrutura de arquivos:

```

<raiz-do-exercicio>
|__CommunicationFramework
|
|__Team1
|  |__DefinitionFile
|  |  |__[omitido]
|  |__IntegrationCode
|  |__UserApplication
|    |__project1.etp
|    |__project1.l4
|    |__project1.vsw
|    |__project1.xscade
|    |__[outros arquivos do SCADE Suite]
|
|__Team2
  |__DefinitionFile
  |  |__[omitido]
  |__IntegrationCode
  |__UserApplication
    |__project2.etp
    |__project2.l4
    |__project2.vsw
    |__project2.xscade
    |__[outros arquivos do SCADE Suite]

```

Detalhamento da implementação:

1. Abrir o SCADE Suite R17

1.1 Criar um novo User Application para o time 1 no local correto: File / New / SCADE Suite Project:

Project name: project1
 Location: <raizdoexercicio>\Team1\UserApplication
 Next > Project type: New Project
 Language version: Scade 6.4 (compatible...)
 Avançar e concluir.

1.2 Na área de workspace à esquerda, clique com botão direito em Project1 > Insert > Operator, conforme imagem abaixo:

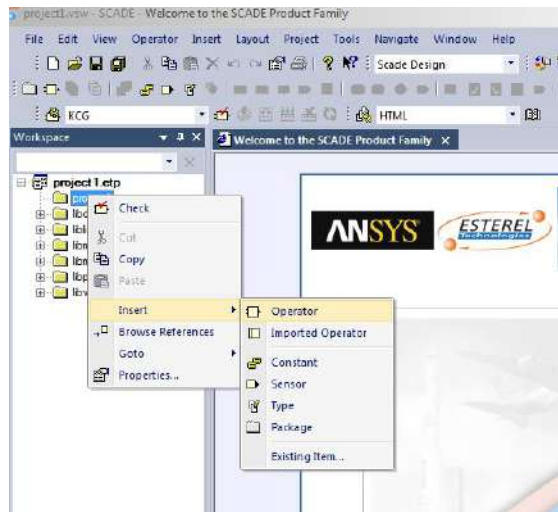


Figura 02 - Criação de Operador

1.3 Altere o nome do operador para “project1”, e configure para que seja salvo no arquivo xscade de mesmo nome (é necessário alterar o local de salvamento do arquivo manualmente):

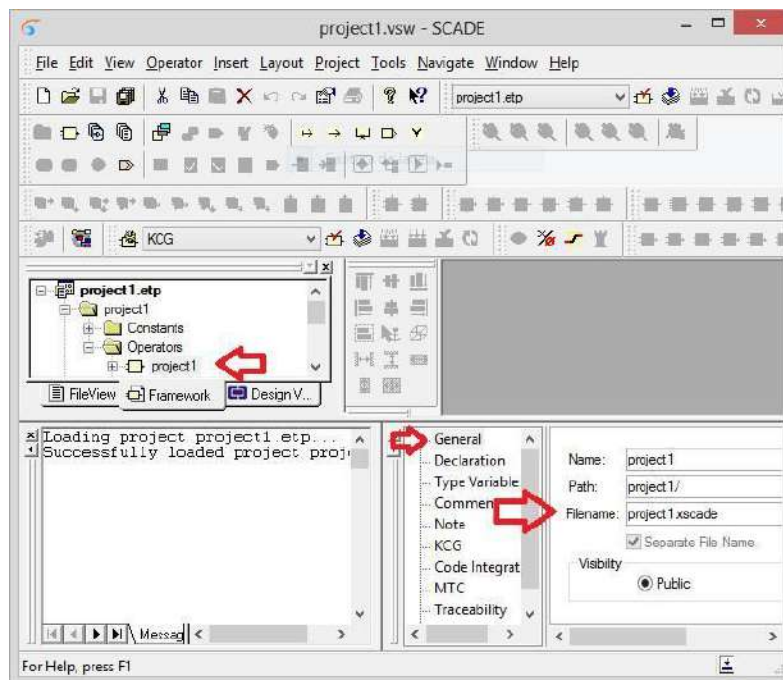


Figura 03 - Criação de Operador

1.3 Conecte o User Application ao Definition File (não se esqueça que o Definition file deve ter a extensão *.etp):

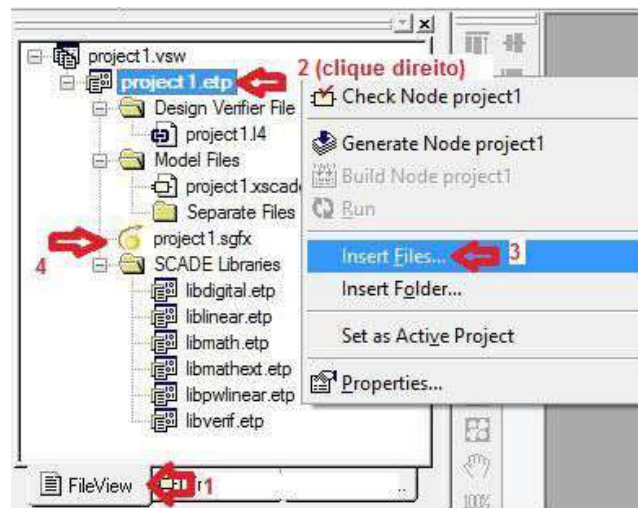


Figura 04 - Conectar US na DF

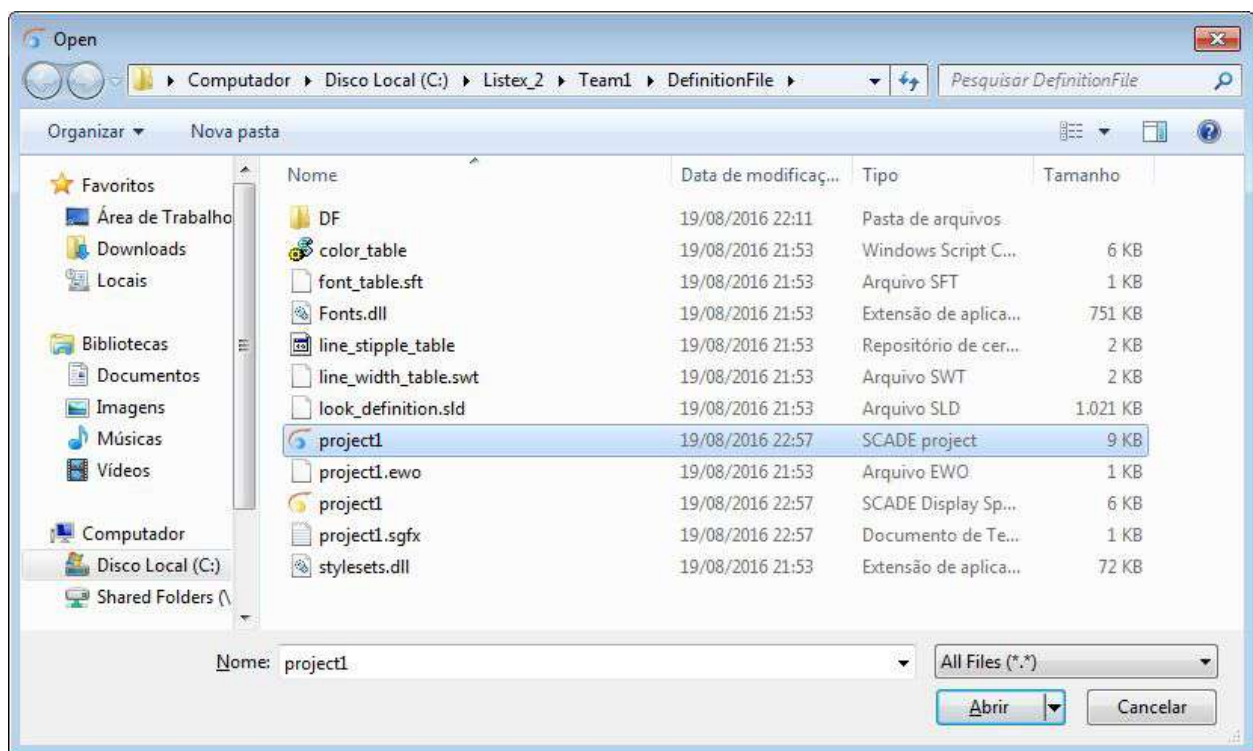


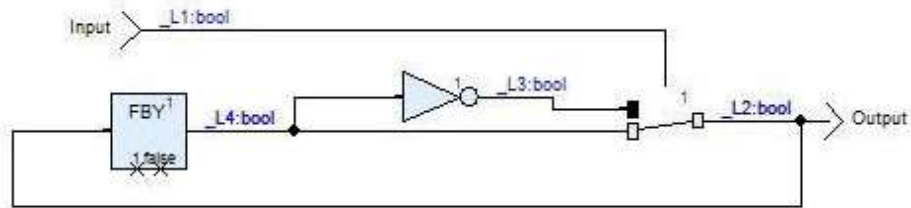
Figura 05 - Conectar US na DF

2. Repetir os passos anteriores, desta vez para o time 2.

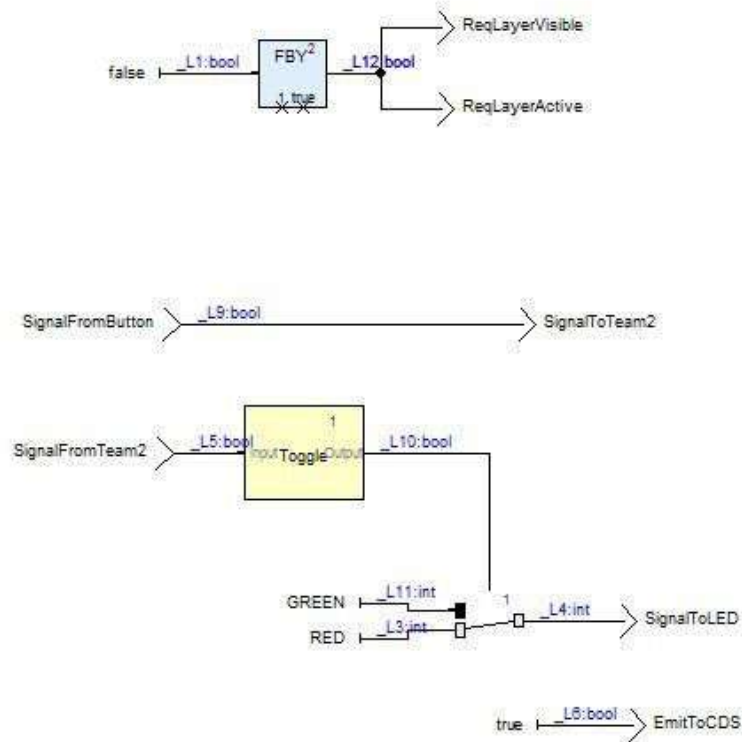
2.4 Passo 4

Implementar a lógica do User Application do time 1. O objetivo é ter a seguinte lógica implementada:

Toggle.xscade (criar novo operador):



project1.xscade (operador principal):



Detalhes:

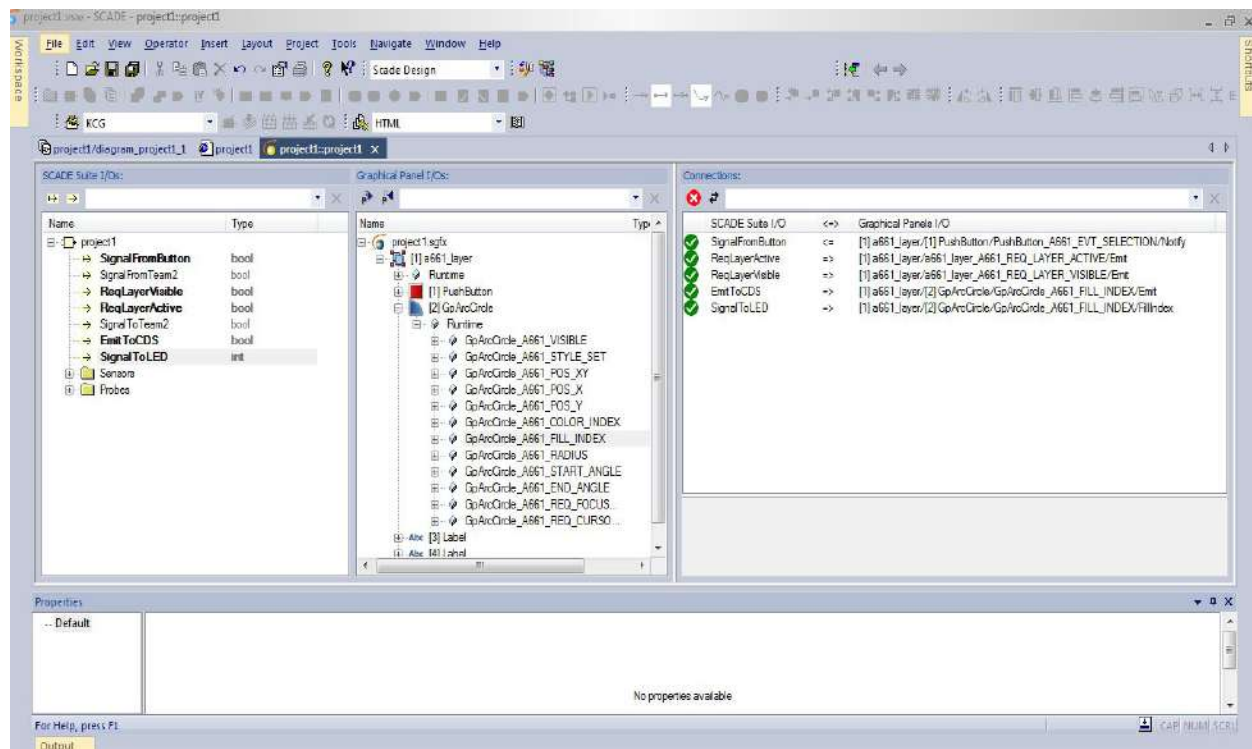
As seguintes constantes devem ser criadas:

Nome	Tipo	Valor
RED	int	21
GREEN	int	51

Os tipos de cada input/output podem ser visualizados nas figuras. Atente para os nomes dos inputs e outputs que serão utilizados na integração com o outro time: “SignalToTeam2” e “SignalFromTeam2”. Você pode escolher outros nomes, mas lembre-se deles nos passos posteriores.

2.5 Passo 5

Conectar os inputs e outputs do User Application do time 1 nos respectivos inputs e outputs do Definition File do time 1. O objetivo é ter as conexões como na figura abaixo:



SCADE Suite I/O	<=>	Graphical Panels I/O
SignalFromButton	<=>	[1] a661_layer/[1] PushButton/PushButton_A661_EVT_SELECTION/Notify
ReqLayerActive	=>	[1] a661_layer/a661_layer_A661_REQ_LAYER_ACTIVE/Emit
ReqLayerVisible	=>	[1] a661_layer/a661_layer_A661_REQ_LAYER_VISIBLE/Emit
EmitToCDS	=>	[1] a661_layer/[2] GpArcCircle/GpArcCircle_A661_FILL_INDEX/Emit
SignalToLED	=>	[1] a661_layer/[2] GpArcCircle/GpArcCircle_A661_FILL_INDEX/FillIndex

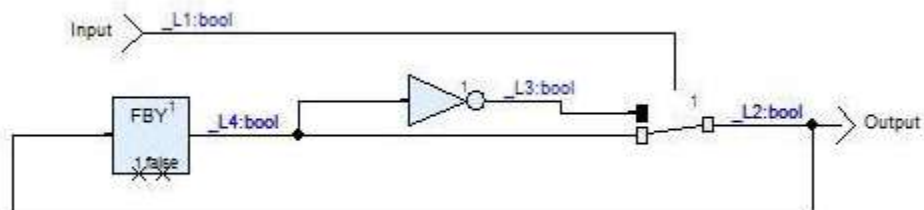
Figura 05 - Conexões

Os sinais a serem conectados estão todos representados na figura acima.

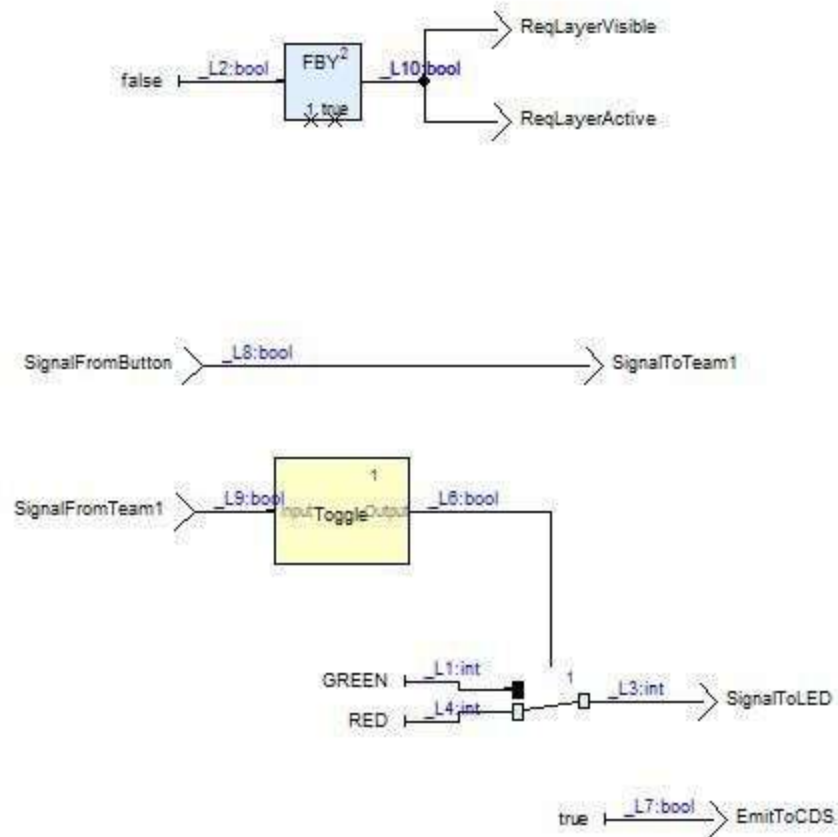
2.6 Passo 6

Implementar a lógica do User Application do time 2. O objetivo é ter a seguinte lógica implementada:

Toggle.xscade (criar novo operador):



project2.xscade (operador principal):



Detalhes:

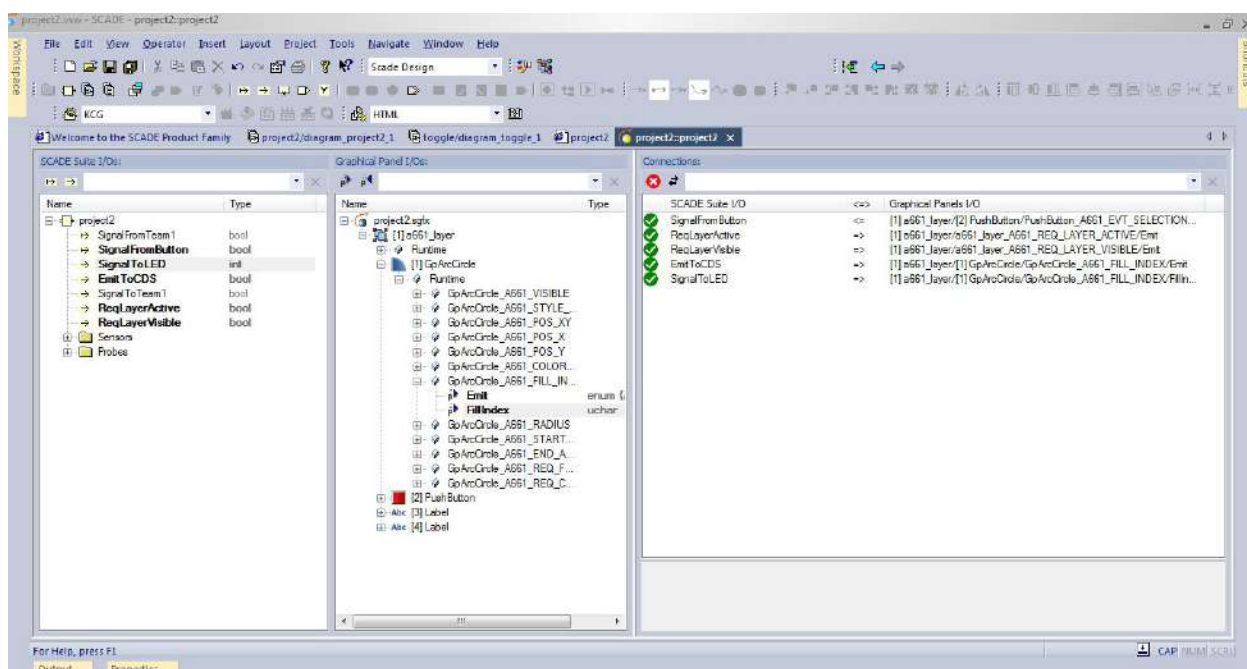
As seguintes constantes devem ser criadas:

Nome	Tipo	Valor
RED	int	21
GREEN	int	51

Os tipos de cada input/output podem ser visualizados nas figuras. Atente para os nomes dos inputs e outputs que serão utilizados na integração com o outro time: “SignalToTeam1” e “SignalFromTeam1”. Você pode escolher outros nomes, mas lembre-se deles nos passos posteriores.

2.7 Passo 7

Conectar os inputs e outputs do User Application do time 2 nos respectivos inputs e outputs do Definition File do time 2. O objetivo é ter as conexões como na Figura 07.



SCADE Suite I/O	<=>	Graphical Panels I/O
SignalFromButton	<=	[1] a661_layer/[2] PushButton/PushButton_A661_EVT_SELECTION...
ReqLayerActive	=>	[1] a661_layer/a661_layer_A661_REQ_LAYER_ACTIVE/Emit
ReqLayerVisible	=>	[1] a661_layer/a661_layer_A661_REQ_LAYER_VISIBLE/Emit
EmitToCDS	=>	[1] a661_layer/[1] GpArcCircle/GpArcCircle_A661_FILL_INDEX/Emit
SignalToLED	=>	[1] a661_layer/[1] GpArcCircle/GpArcCircle_A661_FILL_INDEX/FillIn...

Figura 07 - Conexões

Os sinais a serem conectados estão todos representados na Figura 07.

2.8 Passo 8

Executar a simulação dos projetos. Este passo não possui instruções, pois pressupõe-se que os alunos já tenham feito os labs da listEx1 e já saibam como configurar e executar o projeto. Caso você tenha baixado os arquivos a partir dos commits, será necessário alterar o ID do Definition File do time 2 para conseguir simular no SCADE Suite, verifique a seção 2.10 para instruções de como fazê-lo.

Faz-se necessário apontar um ponto importante: o projeto precisa tanto ser simulado (executar Run com a configuração ativada para Simulation) quanto gerado para o KCG (executar Generate com a configuração ativada para KCG).

Este passo gera uma série de arquivos automaticamente. Os arquivos gerados para o KCG serão utilizados na integração. Abaixo, visualiza-se a estrutura de pastas após a execução desse passo:

```
<raiz-do-exercicio>
|__CommunicationFramework
|
|__Team1
|  |__DefinitionFile
|  |  |__[omitido]
|  |__IntegrationCode
|  |__UserApplication
|    |__[arquivos do SCADE Suite do project1]
|    |__KCG
|    |  |__[arquivos gerados para o KCG]
|    |__Simulation
|    |  |__[arquivos gerados para a simulação]
|
|__Team2
|  |__DefinitionFile
|  |  |__[omitido]
|  |__IntegrationCode
|  |__UserApplication
|    |__[arquivos do SCADE Suite do project2]
|    |__KCG
|    |  |__[arquivos gerados para o KCG]
|    |__Simulation
|    |  |__[arquivos gerados para a simulação]
```

Como os arquivos gerados por este passo são criados automaticamente, eles não são incluídos no versionamento, ou seja, caso baixem um arquivo de um passo posterior a este, será necessário executar este passo para que os arquivos das pastas KCG sejam criados.

2.9 Passo 9

Considerando que atingiram este ponto, desenvolvendo os displays e respectivas lógicas de aplicação, resta somente testar o projeto. Prossigam a atividade a partir dos arquivos disponíveis em algum dos links abaixo (de acordo com a sua versão do windows):

- [link 1 \(para computadores com Windows 32Bit\)](#)
- [link 2 \(para computadores com Windows 64Bit\)](#)

Esses arquivos contêm o framework de comunicação para a integração dos projetos. Para facilitar este passo, todos os arquivos são fornecidos prontos, inclusive os displays e lógicas SCADE, organizados já segundo a estrutura sugerida.

2.10 Passo 10

Configurar os IDs do Definition File do time 2

Instruções:

1. Usando o SCADE Display R17, abra o projeto project2.etp
2. Altere o ID do display: File / Properties / A661 Properties
Application ID: 2
3. Execute novamente: Project / Generator / Generate e salve.
4. Abra o SCADE Suite R17, carregue o arquivo project2.vsw e refaça o build em KCG e Simulation.

A alteração acima é necessária para que o servidor Arinc execute o serviço em uma porta diferente da 1231. Nesse caso, como Application ID = 2, o serviço será executado na porta 1232.

Note que, ao fazer essa alteração, o SCADE Suite errará a conexão ao Display (pois, por padrão, ele executa na porta 1231). Sempre que os alunos quiserem

executar a simulação no próprio SCADE Suite, precisarão trocar o Application ID de volta para 1.

Caso alguém deseje implementar um script para criar automaticamente uma cópia do “server_conf.xml” (encontrado nas pastas DF) e substituir automaticamente o parâmetro app_id=”1” pelo ID respectivo a cada time, por favor entrar em contato para que possamos reescrever este tutorial omitindo este passo.

O script atual que invoca o servidor e o arquivo “server_conf.xml” é o Launch_teamXDF.bat, onde X = 1 ou 2. E o script Launch_teamXUA.bat utiliza a porta 123X.

Adicionalmente, caso as DFs fossem ser integradas em uma única DF, seria necessário mudar também o ID do Layer. Para o SIG-TRAC 2016, isso não será feito.

2.11 Passo 11

Executar os projetos de ambos os times e verificar que eles estão realmente se comunicando.

Observações:

- a) Pare este passo, é necessário antes instalar o Cygwin com os pacotes “make” e “gcc-core” selecionados para instalação.
- b) Baixe e instale o cygwin através do link <https://cygwin.com/install.html>, **escolhendo a versão 32Bit.**
- c) Após a instalação, o gerenciador de pacotes dará opção de filtrar e escolher os pacotes citados acima. Clique em avançar, e as dependências serão resolvidas, baixadas e instaladas automaticamente.

2.12 Passo 12

Executar os projetos de ambos os times e verificar que eles estão efetivamente integrados.

Instruções:

1. Na pasta “<raizdoprojeto>\Team1\IntegrationCode”, execute na sequência a seguir:

Compile_team1UA_step1.bat

Compile_team1UA_step2.bat
Launch_team1DF.bat
Launch_team1UA.bat

2. Na pasta "<raizdoprojeto>\Team2\IntegrationCode", execute na sequência a seguir:

Compile_team2UA_step1.bat
Compile_team2UA_step2.bat
Launch_team2DF.bat
Launch_team2UA.bat

3. Aperte os botões de cada display e observe que os LEDs do outro display têm sua cor alternada.

Se tudo correr sem problemas, os projetos dos times estarão trocando mensagens.

3 Considerações Finais

Este documento apresentou, passo a passo, como cada integrante envolvido no **Projeto de STAMPS-TR** pode integrar o desenvolvimento com SCADE.