classdef app1 < matlab.apps.AppBase

% Properties that correspond to app components

properties (Access = public)

TangoExample1 matlab.ui.Figure

ConnectButton matlab.ui.control.Button

DisconnectButton matlab.ui.control.Button

ListBoxLabel matlab.ui.control.Label

ListBox matlab.ui.control.ListBox

COMportDropDownLabel matlab.ui.control.Label

COMportDropDown matlab.ui.control.DropDown

senddatatocontrollerPanel matlab.ui.container.Panel

MoveRelativeButton matlab.ui.control.Button

XEditFieldLabel matlab.ui.control.Label

XEditField matlab.ui.control.NumericEditField

YEditFieldLabel matlab.ui.control.Label

YEditField matlab.ui.control.NumericEditField

MoveAbsoluteButton matlab.ui.control.Button

readdatafromcontrollerPanel matlab.ui.container.Panel

PositionButton matlab.ui.control.Button

MotorCurrentButton matlab.ui.control.Button

SpindlePitchButton matlab.ui.control.Button

end

properties (Access = private)

dll = 'Tango\_DLL';

LSID = int32(0); % Tango connection ID given when calling CreateLSID()

status = int32(0); % general return value of every DLL function (0 indicates no error)

cstrPtr = libpointer('cstring', blanks(64)); % general pupose cstring poiner

dxPtr = libpointer('doublePtr',0); % general purpose double pointer for axes (x,y,z,a)

dyPtr = libpointer('doublePtr',0);

dzPtr = libpointer('doublePtr',0);

daPtr = libpointer('doublePtr',0);

ixPtr = libpointer('int32Ptr',0); % general purpose integer pointer for axes (x,y,z,a)

iyPtr = libpointer('int32Ptr',0);

izPtr = libpointer('int32Ptr',0);

iaPtr = libpointer('int32Ptr',0);

end

% Callbacks that handle component events

methods (Access = private)

% Button pushed function: ConnectButton

function ConnectButtonPushed(app, event)

% 64bit Matlab version needs 64bit Tango DLL in application path

% make sure 64bit Tango DLL is inside application path

if libisloaded('Tango\_DLL')

calllib(app.dll,'LSX\_Disconnect',1);

calllib(app.dll,'LSX\_FreeLSID',1);

unloadlibrary('Tango\_DLL')

end

if ~libisloaded('Tango\_DLL')

loadlibrary('Tango\_DLL', 'Tango\_DLL.h')

% display all available DLL functions in MATLAB Command Window

libfunctions Tango\_DLL -full

end

% The Tango DLL allows access up to 8 different Tango controllers

% command DLL to return unique ID for any Tango connection

% make sure to disconnect and free the given ID before leaving the App

idPtr = libpointer('int32Ptr',0);

str = "LSX\_CreateLSID() => ";

app.status = calllib(app.dll,'LSX\_CreateLSID',idPtr);

if (app.status > 0)

str = str + "Error=" + app.status;

else

app.LSID = idPtr.Value;

str = str + "ID=" + app.LSID;

end

app.ListBox.Items = str;

% get DLL version

str = "GetDLLVersionString => ";

[app.status,app.cstrPtr] = calllib(app.dll,'LSX\_GetDLLVersionString',app.LSID,app.cstrPtr,50);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + app.cstrPtr;

end

app.ListBox.Items = [app.ListBox.Items str]; % add next line

% autodetect connect to the first available USB Tango

% in this case there ist no need to specify correct baud rate

% this example is using ifctyp=-1 to autodetect any USB Tango connection

if (app.COMportDropDown.Value == "autodetect")

ifctyp = -1;

port = '';

else

ifctyp = 1;

port = app.COMportDropDown.Value;

end

% connect to a communication port

str = "ConnectSimple(" + port + ") => ";

app.status = calllib(app.dll,'LSX\_ConnectSimple',app.LSID, ifctyp, port, 57600, 0);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + "OK";

end

app.ListBox.Items = [app.ListBox.Items str];

% get Tango version

str = "GetTangoVersion => ";

[app.status,app.cstrPtr] = calllib(app.dll,'LSX\_GetTangoVersion',app.LSID,app.cstrPtr,50);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + app.cstrPtr;

end

app.ListBox.Items = [app.ListBox.Items str];

% get Tango Serial Number

str = "GetSerialNr => ";

[app.status,app.cstrPtr] = calllib(app.dll,'LSX\_GetSerialNr',app.LSID,app.cstrPtr,50);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + app.cstrPtr;

end

app.ListBox.Items = [app.ListBox.Items str];

% read Tango error (if any. zero = OK = no error)

errPtr = libpointer('int32Ptr',0);

str = "GetError() => ";

app.status = calllib(app.dll,'LSX\_GetError',app.LSID,errPtr);

if (app.status > 0)

str = str + "DLL Error=" + app.status;

else

TangoError = errPtr.Value;

if (TangoError == 0)

str = str + "no error";

else

str = str + "err=" + TangoError;

end

end

app.ListBox.Items = [app.ListBox.Items str];

end

% Callback function: DisconnectButton, TangoExample1

function DisconnectButtonPushed(app, event)

% disconnect communication

str = "Disconnect(" + app.LSID + ") => ";

app.status = calllib(app.dll,'LSX\_Disconnect',app.LSID);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + "OK";

end

app.ListBox.Items = [app.ListBox.Items str];

% free LSID

str = "FreeLSID(" + app.LSID + ") => ";

app.status = calllib(app.dll,'LSX\_FreeLSID',app.LSID);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + "OK";

end

app.ListBox.Items = [app.ListBox.Items str];

end

% Button pushed function: PositionButton

function PositionButtonPushed(app, event)

% read actual axes position

str = "GetPos [dim] => ";

[app.status, app.dxPtr, app.dyPtr, app.dzPtr, app.daPtr] = calllib(app.dll,'LSX\_GetPos',app.LSID, app.dxPtr, app.dyPtr, app.dzPtr, app.daPtr);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + app.dxPtr + " " + app.dyPtr + " " + app.dzPtr + " " + app.daPtr;

end

app.ListBox.Items = [app.ListBox.Items str];

end

% Button pushed function: MoveRelativeButton

function MoveRelativeButtonPushed(app, event)

% move relative

str = "MoveRel(" + app.LSID + ") => ";

xx = app.XEditField.Value;

yy = app.YEditField.Value;

app.status = calllib(app.dll,'LSX\_MoveRel',app.LSID, xx, yy, 0, 0, 1);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + "OK";

end

app.ListBox.Items = [app.ListBox.Items str];

end

% Button pushed function: MoveAbsoluteButton

function MoveAbsoluteButtonPushed(app, event)

% move relative

str = "MoveAbs(" + app.LSID + ") => ";

xx = app.XEditField.Value;

yy = app.YEditField.Value;

app.status = calllib(app.dll,'LSX\_MoveAbs',app.LSID, xx, yy, 0, 0, 1);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + "OK";

end

app.ListBox.Items = [app.ListBox.Items str];

end

% Button pushed function: MotorCurrentButton

function MotorCurrentButtonPushed(app, event)

% read axes motor current

str = "GetMotorCurrent [A] => ";

[app.status, app.dxPtr, app.dyPtr, app.dzPtr, app.daPtr] = calllib(app.dll,'LSX\_GetMotorCurrent',app.LSID, app.dxPtr, app.dyPtr, app.dzPtr, app.daPtr);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + app.dxPtr + " " + app.dyPtr + " " + app.dzPtr + " " + app.daPtr;

end

app.ListBox.Items = [app.ListBox.Items str];

end

% Button pushed function: SpindlePitchButton

function SpindlePitchButtonPushed(app, event)

% read axes spindle pitch

str = "GetPitch [mm] => ";

[app.status, app.dxPtr, app.dyPtr, app.dzPtr, app.daPtr] = calllib(app.dll,'LSX\_GetPitch',app.LSID, app.dxPtr, app.dyPtr, app.dzPtr, app.daPtr);

if (app.status > 0)

str = str + "Error=" + app.status;

else

str = str + app.dxPtr + " " + app.dyPtr + " " + app.dzPtr + " " + app.daPtr;

end

app.ListBox.Items = [app.ListBox.Items str];

end

end

% Component initialization

methods (Access = private)

% Create UIFigure and components

function createComponents(app)

% Create TangoExample1 and hide until all components are created

app.TangoExample1 = uifigure('Visible', 'off');

app.TangoExample1.Position = [100 100 926 539];

app.TangoExample1.Name = 'TANGO Example1 (created with MATHLAB R2019a)';

app.TangoExample1.ButtonDownFcn = createCallbackFcn(app, @DisconnectButtonPushed, true);

% Create ConnectButton

app.ConnectButton = uibutton(app.TangoExample1, 'push');

app.ConnectButton.ButtonPushedFcn = createCallbackFcn(app, @ConnectButtonPushed, true);

app.ConnectButton.Tooltip = {'establish connection to Tango controller and get some information'; ''};

app.ConnectButton.Position = [101 446 100 22];

app.ConnectButton.Text = 'Connect';

% Create DisconnectButton

app.DisconnectButton = uibutton(app.TangoExample1, 'push');

app.DisconnectButton.ButtonPushedFcn = createCallbackFcn(app, @DisconnectButtonPushed, true);

app.DisconnectButton.Tooltip = {'disconnect UI from TANGO DLL'};

app.DisconnectButton.Position = [101 411 100 22];

app.DisconnectButton.Text = 'Disconnect';

% Create ListBoxLabel

app.ListBoxLabel = uilabel(app.TangoExample1);

app.ListBoxLabel.HorizontalAlignment = 'right';

app.ListBoxLabel.Position = [220 479 48 22];

app.ListBoxLabel.Text = 'List Box';

% Create ListBox

app.ListBox = uilistbox(app.TangoExample1);

app.ListBox.Items = {};

app.ListBox.FontName = 'Courier New';

app.ListBox.Position = [283 28 624 475];

app.ListBox.Value = {};

% Create COMportDropDownLabel

app.COMportDropDownLabel = uilabel(app.TangoExample1);

app.COMportDropDownLabel.HorizontalAlignment = 'right';

app.COMportDropDownLabel.Position = [28 481 58 22];

app.COMportDropDownLabel.Text = 'COM port';

% Create COMportDropDown

app.COMportDropDown = uidropdown(app.TangoExample1);

app.COMportDropDown.Items = {'autodetect', 'COM1', 'COM2', 'COM3', 'COM4', 'COM5', 'COM6', 'COM7', 'COM8', 'COM9', ''};

app.COMportDropDown.Tooltip = {'select on which COM port the TANGO controller is connected'};

app.COMportDropDown.Position = [101 481 100 22];

app.COMportDropDown.Value = 'autodetect';

% Create senddatatocontrollerPanel

app.senddatatocontrollerPanel = uipanel(app.TangoExample1);

app.senddatatocontrollerPanel.Title = 'send data to controller';

app.senddatatocontrollerPanel.Position = [8 28 260 141];

% Create MoveRelativeButton

app.MoveRelativeButton = uibutton(app.senddatatocontrollerPanel, 'push');

app.MoveRelativeButton.ButtonPushedFcn = createCallbackFcn(app, @MoveRelativeButtonPushed, true);

app.MoveRelativeButton.Tooltip = {'move XY relative'};

app.MoveRelativeButton.Position = [148 59 100 22];

app.MoveRelativeButton.Text = 'MoveRelative';

% Create XEditFieldLabel

app.XEditFieldLabel = uilabel(app.senddatatocontrollerPanel);

app.XEditFieldLabel.HorizontalAlignment = 'right';

app.XEditFieldLabel.Position = [12 88 25 22];

app.XEditFieldLabel.Text = 'X';

% Create XEditField

app.XEditField = uieditfield(app.senddatatocontrollerPanel, 'numeric');

app.XEditField.ValueDisplayFormat = '%11.5g';

app.XEditField.Position = [52 88 76 22];

app.XEditField.Value = 1.2345;

% Create YEditFieldLabel

app.YEditFieldLabel = uilabel(app.senddatatocontrollerPanel);

app.YEditFieldLabel.HorizontalAlignment = 'right';

app.YEditFieldLabel.Position = [12 59 25 22];

app.YEditFieldLabel.Text = 'Y';

% Create YEditField

app.YEditField = uieditfield(app.senddatatocontrollerPanel, 'numeric');

app.YEditField.ValueDisplayFormat = '%11.5g';

app.YEditField.Position = [52 59 76 22];

app.YEditField.Value = 2.3456;

% Create MoveAbsoluteButton

app.MoveAbsoluteButton = uibutton(app.senddatatocontrollerPanel, 'push');

app.MoveAbsoluteButton.ButtonPushedFcn = createCallbackFcn(app, @MoveAbsoluteButtonPushed, true);

app.MoveAbsoluteButton.Position = [148 88 100 22];

app.MoveAbsoluteButton.Text = 'Move Absolute';

% Create readdatafromcontrollerPanel

app.readdatafromcontrollerPanel = uipanel(app.TangoExample1);

app.readdatafromcontrollerPanel.Tooltip = {'read out actual parameter from TANGO controller'};

app.readdatafromcontrollerPanel.Title = 'read data from controller';

app.readdatafromcontrollerPanel.Position = [8 178 260 221];

% Create PositionButton

app.PositionButton = uibutton(app.readdatafromcontrollerPanel, 'push');

app.PositionButton.ButtonPushedFcn = createCallbackFcn(app, @PositionButtonPushed, true);

app.PositionButton.Tooltip = {'read actual axes position from TANGO controller'};

app.PositionButton.Position = [93 14 100 22];

app.PositionButton.Text = 'Position';

% Create MotorCurrentButton

app.MotorCurrentButton = uibutton(app.readdatafromcontrollerPanel, 'push');

app.MotorCurrentButton.ButtonPushedFcn = createCallbackFcn(app, @MotorCurrentButtonPushed, true);

app.MotorCurrentButton.Tooltip = {'get actual axes motor current in [A]'};

app.MotorCurrentButton.Position = [93 168 100 22];

app.MotorCurrentButton.Text = 'Motor Current';

% Create SpindlePitchButton

app.SpindlePitchButton = uibutton(app.readdatafromcontrollerPanel, 'push');

app.SpindlePitchButton.ButtonPushedFcn = createCallbackFcn(app, @SpindlePitchButtonPushed, true);

app.SpindlePitchButton.Position = [93 136 100 22];

app.SpindlePitchButton.Text = 'Spindle Pitch';

% Show the figure after all components are created

app.TangoExample1.Visible = 'on';

end

end

% App creation and deletion

methods (Access = public)

% Construct app

function app = app1

% Create UIFigure and components

createComponents(app)

% Register the app with App Designer

registerApp(app, app.TangoExample1)

if nargout == 0

clear app

end

end

% Code that executes before app deletion

function delete(app)

% Delete UIFigure when app is deleted

delete(app.TangoExample1)

end

end

end