

Arduinostage Control (GUI)

```
classdef arduinostage_control_V1_1_5 < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        ArduinoStageControlUIFigure          matlab.ui.Figure
        wichtigegcodeBefehlePanel             matlab.ui.container.Panel
        TextArea                             matlab.ui.control.TextArea
        GltigkeitdesletztenBefehlsPanel       matlab.ui.container.Panel
        letzterBefehlEditField                matlab.ui.control.EditField
        letzterBefehlEditFieldLabel           matlab.ui.control.Label
        GltigkeitLamp                         matlab.ui.control.Lamp
        GltigkeitLampLabel                    matlab.ui.control.Label
        ManuellegcodeEingabeZhlerdesVerfahrwegsistnichtgekoppeltPanel
matlab.ui.container.Panel
        BefehlsendenButton                   matlab.ui.control.Button
        BefehlEditField                       matlab.ui.control.EditField
        BefehlEditFieldLabel                  matlab.ui.control.Label
        VerbindungzurArduinoStagePanel       matlab.ui.container.Panel
        PortDropDown                         matlab.ui.control.DropDown
        PortDropDownLabel                     matlab.ui.control.Label
        VerbindungSwitch                     matlab.ui.control.Switch
        VerbindungSwitchLabel                 matlab.ui.control.Label
        FehlerLamp                           matlab.ui.control.Lamp
        FehlerLampLabel                      matlab.ui.control.Label
        VerbindungsstatusLamp                matlab.ui.control.Lamp
        VerbindungsstatusLampLabel           matlab.ui.control.Label
        SteuerungPanel                       matlab.ui.container.Panel
        SchleifebeginnenButton               matlab.ui.control.Button
        VerfahrwegeDropDown                  matlab.ui.control.DropDown
        VerfahrwegeDropDownLabel              matlab.ui.control.Label
        maxPositionbernehmenButton            matlab.ui.control.Button
        maximalePositionEditField             matlab.ui.control.EditField
        maximalePositionEditFieldLabel        matlab.ui.control.Label
        aktuellePositionEditField             matlab.ui.control.EditField
        aktuellePositionEditFieldLabel        matlab.ui.control.Label
        ResetZhlerButton                     matlab.ui.control.Button
        SchrittweiteDropDown                 matlab.ui.control.DropDown
        SchrittweiteDropDownLabel              matlab.ui.control.Label
        WeiterumSchrittweiteButton            matlab.ui.control.Button
        VerfahrrenzumAnfangButton             matlab.ui.control.Button
    end

    %noetig fuer Zugriff auf Port aus allen Funktionen
    properties (Access = public)
        arduinostage; % fuer serialport Objekt
        stepcount = double(0); % fuer Zaehler des Verfahrwegs
        maxsteps = double(250); % Schritte des gesamten Verfahrwegs
    end

    methods (Access = public)

        function answer = check_grbl_answer(~, device,anzahl_zeichen)

            %noetige Variablen einfuehren
            check = 1;
            counter = 0;
            answer_empty = "";
            answer = "";

            %Schleife bis eine Antwort erhalten wird
            while check

                %serielle Schnittstelle auslesen
```

```

        answer = read(device,anzahl_zeichen,'string');
%(device Objekt,Anzahl der Zeichen,'Datentyp')

        %pruefen ob Antwort leer ist und Bedingung fuer while-Schleife
setzen
        if answer == answer_empty
            check = 1;
        else
            check = 0;
        end

        %Abbruch nach 10000 durchlaufen
        if counter >= 10000
            answer = "answer_fail";
%Antwort fuer weitere Nutzung setzen
            break
        end

        %Kontrollzaehler erhoehen
        counter = counter + 1;

    end

    %Entfernen der Terminatoren der grbl Antwort
    answer = regexprep(answer,'\r\n','');

end

end

% Callbacks that handle component events
methods (Access = private)

    % Value changed function: VerbindungSwitch
    function VerbindungSwitchValueChanged(app, event)
        connect = app.VerbindungSwitch.Value;
        try
            switch connect

                %Verbindung aufbauen, einstellen und Prüfen
                case 'On'

                    %ausgewaehlten Port einlesen
                    port = app.PortDropDown.Value;
                    %Serielle Kommunikation aufbauen
                    app.arduinoStage = serialport(port, 115200);
% Baudrate von grbl-Version 1.1 = 115200
                    pause(0.5);
% muss min 0.3 sein --> 0.5 fuer Stabilitaet

                    %Terminator aendern
                    configureTerminator(app.arduinoStage, "CR/LF");
% Terminator in grbl-Version 1.1: "CR/LF"

                %weitere Einstellungen (Hier gesetzte Einstellungen sind
                %nur gesetzt, um Fehler durch Aenderungen in den Matlab
                %default-Einstellungen zu vermeiden.)
                set(app.arduinoStage,'DataBits',8);
                set(app.arduinoStage,'Parity','none');
                set(app.arduinoStage,'StopBits',1);
                set(app.arduinoStage,'FlowControl','none');

                %Antwort bei Verbindungsaufbau auslesen und pruefen
                ans1_soll = "Grbl 1.1h ['$' for help]";
% wenn korrekt aufgebaut in grbl-Version 1.1 !!aendern Wenn andere Version!!
                ans1 = check_grbl_answer(app,app.arduinoStage,26);
% 26 Zeichen fuer korrekte Antwort abfragen

                %inkrementelle Bewegung festlegen

```

```

        %Befehl senden
        writeline(app.arduinoStage,"G91");
% G91 ist gcode-Befehl fuer Umstellung auf inkrementelle Bewegung
        %Antwort auslesen
        ans2 = check_grbl_answer(app,app.arduinoStage,8);
% 8 Zeichen fuer korrekte Antwort abfragen
        %korrekte Antwort
        ans2_soll = "okok";

        %Status und Fehlerlampen setzen
        if all(ans1 == ans1_soll) && all(ans2 == ans2_soll)
% hier muss all() um die Bedingungen
            app.VerbindungsstatusLamp.Color = 'green';
            app.FehlerLamp.Color = 'white';
        else
            app.FehlerLamp.Color = 'red';
            app.VerbindungsstatusLamp.Color = 'yellow';
            ans_err = check_grbl_answer(app,app.arduinoStage,20);
% alle verbleibenden Zeichen abfragen (ergibt Warnung, da mehr abgefragt wird als
vorhanden) -->Zeichen nicht weiter relevant
        end

        %Verbindung beenden
        case 'Off'
            %zum trennen muss das serialport objekt geloescht werden
            delete(app.arduinoStage);
            app.VerbindungsstatusLamp.Color = 'yellow';
            app.FehlerLamp.Color = 'white';
        end

    catch
        app.FehlerLamp.Color = 'red';
        app.VerbindungsstatusLamp.Color = 'red';
    end
end

% Callback function
function PortDropDownValueChanged(app, event)

end

% Button pushed function: WeiterumSchrittweiteButton
function WeiterumSchrittweiteButtonPushed(app, event)

    %GUI auslesen und Daten umwandeln
    schrittweite_double = str2double(app.SchrittweiteDropDown.Value);
%Umwandlung von char zu int funktioniert nur korrekt ueber Zwischenschritt
str2double oder str2num
    schrittweite = int16(schrittweite_double);

    %Befehl erzeugen
    befehl = "G0 X" + int2str(schrittweite);
%Umwandlung nur so korrekt
    %Befehl in string umwandeln
    str_befehl = string(befehl);
    %Befehl senden
    writeline(app.arduinoStage,str_befehl);
    %Antwort auslesen
    ans3 = check_grbl_answer(app,app.arduinoStage,8);
% 8 Zeichen fuer korrekte Antwort abfragen
    %korrekte Antwort
    ans3_soll = "okok";

    %letzten Befehl und dessen Status ausgeben
    app.letzterBefehlEditField.Value = str_befehl;
    if ans3 == ans3_soll
        app.GltigkeitLamp.Color = 'green';
        app.stepcount = app.stepcount + schrittweite_double;
    end
end

```

```

        app.aktuellePositionEditField.Value = num2str(app.stepcount);
    else
        app.GltigkeitLamp.Color = 'red';
        ans_err = check_grbl_answer(app, app.arduinoStage, 20);
% alle verbleibenden Zeichen abfragen (ergibt Warnung, da mehr abgefragt wird als
vorhanden) --> Zeichen nicht weiter relevant
    end
end

% Button pushed function: BefehlsendenButton
function BefehlsendenButtonPushed(app, event)

    %Befehl einlesen
    befehl = app.BefehlEditField.Value;

    %Befehl in string umwandeln
    str_befehl = string(befehl);
    %Befehl senden
    writeline(app.arduinoStage, str_befehl);
    %Antwort auslesen
    ans4 = check_grbl_answer(app, app.arduinoStage, 8);
% 8 Zeichen fuer korrekte Antwort abfragen
    %korrekte Antwort
    ans4_soll = "okok";

    %letzten Befehl und dessen Status ausgeben
    app.letzterBefehlEditField.Value = str_befehl;
    if ans4 == ans4_soll
        app.GltigkeitLamp.Color = 'green';
    else
        app.GltigkeitLamp.Color = 'red';
        ans_err = check_grbl_answer(app, app.arduinoStage, 20);
% alle verbleibenden Zeichen abfragen (ergibt Warnung, da mehr abgefragt wird als
vorhanden) --> Zeichen nicht weiter relevant
    end
end

% Button pushed function: ResetZählerButton
function ResetZählerButtonPushed(app, event)
    app.stepcount = 0;
    app.aktuellePositionEditField.Value = num2str(app.stepcount);
end

% Button pushed function: VerfahrenzumAnfangButton
function VerfahrenzumAnfangButtonPushed(app, event)

    %aktuelle Position auslesen
    position = app.stepcount;

    %Befehl erzeugen um auf 0 zu fahren
    gotozero = "G0 X-" + num2str(position); %Umwandlung nur so korrekt
    %Befehl in string umwandeln
    str_gotozero = string(gotozero);
    %Befehl senden
    writeline(app.arduinoStage, str_gotozero);
    %Antwort auslesen
    ans5 = check_grbl_answer(app, app.arduinoStage, 8);
% 8 Zeichen fuer korrekte Antwort abfragen
    %korrekte Antwort
    ans5_soll = "okok";

    %letzten Befehl und dessen Status ausgeben
    app.letzterBefehlEditField.Value = str_gotozero;
    if ans5 == ans5_soll
        app.GltigkeitLamp.Color = 'green';
        app.stepcount = app.stepcount - position;
        app.aktuellePositionEditField.Value = num2str(app.stepcount);
    else
        app.GltigkeitLamp.Color = 'red';
    end
end

```

```

        ans_err = check_grbl_answer(app,app.arduinoStage,20);
% alle verbleibenden Zeichen abfragen (ergibt Warnung, da mehr abgefragt wird als
vorhanden) --> Zeichen nicht weiter relevant
    end
end

% Button pushed function: maxPositionbernehmenButton
function maxPositionbernehmenButtonPushed(app, event)
    app.maxsteps = app.stepcount;
    app.maximalePositionEditField.Value = num2str(app.maxsteps);
end

% Button pushed function: SchleifebeginnenButton
function SchleifebeginnenButtonPushed(app, event)
    %Dropdown Felder auslesen
    verfahrwege_double = str2double(app.VerfahrwegeDropDown.Value);
%Umwandlung von char zu int funktioniert nur korrekt ueber Zwischenschritt
str2double oder str2num
    verfahrwege = int8(verfahrwege_double);
    schrittweite_double = str2double(app.SchrittweiteDropDown.Value);
%Umwandlung von char zu int funktioniert nur korrekt ueber Zwischenschritt
str2double oder str2num
    schrittweite = int16(schrittweite_double);

    %Befehl mit pos. und neg.Vorzeichen erzeugen
    loop_pos = "G0 X" + num2str(schrittweite);
    str_loop_pos = string(loop_pos);
    loop_neg = "G0 X-" + num2str(schrittweite);
    str_loop_neg = string(loop_neg);

    %Zaehlvariable einfuehren
    i = 1;
    %korrekte Antwort einfuehren
    ans6_soll = "okok";

    %Befehle alternierend senden und pruefen
    while i <= verfahrwege

        if rem(i,2) == 1
%wenn i ungerade
            %Befehl senden
            writeline(app.arduinoStage,str_loop_pos);
            %Antwort auslesen
            ans6 = check_grbl_answer(app,app.arduinoStage,8);
% 8 Zeichen fuer korrekte Antwort abfragen
            %letzten Befehl und dessen Status ausgeben
            app.letzterBefehlEditField.Value = str_loop_pos;
            if ans6 == ans6_soll
                app.GltigkeitLamp.Color = 'green';
            else
                app.GltigkeitLamp.Color = 'red';
                ans_err = check_grbl_answer(app,app.arduinoStage,20);
% alle verbleibenden Zeichen abfragen (ergibt Warnung, da mehr abgefragt wird als
vorhanden) --> Zeichen nicht weiter relevant
                app.letzterBefehlEditField.Value = "Schleifenbefehl
abgebrochen";
                break
            end
        end
        if rem(i,2) == 0
%wenn i gerade
            %Befehl senden
            writeline(app.arduinoStage,str_loop_neg);
            %Antwort auslesen
            ans6 = check_grbl_answer(app,app.arduinoStage,8);
% 8 Zeichen fuer korrekte Antwort abfragen
            %letzten Befehl und dessen Status ausgeben
            app.letzterBefehlEditField.Value = str_loop_neg;
            if ans6 == ans6_soll

```

```

        app.GltigkeitLamp.Color = 'green';
    else
        app.GltigkeitLamp.Color = 'red';
        ans_err = check_grbl_answer(app, app.arduinoStage, 20);
% alle verbleibenden Zeichen abfragen (ergibt Warnung, da mehr abgefragt wird als
vorhanden) --> Zeichen nicht weiter relevant
        app.letzterBefehlEditField.Value = "Schleifenbefehl
abgebrochen";
        break
    end
end

        %Zaehlvariable erhoeihen
        i = i + 1;
    end
end
end

% Component initialization
methods (Access = private)

    % Create UIFigure and components
    function createComponents(app)

        % Create ArduinoStageControlUIFigure and hide until all components are
created
        app.ArduinoStageControlUIFigure = uifigure('Visible', 'off');
        app.ArduinoStageControlUIFigure.Position = [100 100 638 668];
        app.ArduinoStageControlUIFigure.Name = 'Arduino-Stage Control';

        % Create SteuerungPanel
        app.SteuerungPanel = uipanel(app.ArduinoStageControlUIFigure);
        app.SteuerungPanel.Title = 'Steuerung';
        app.SteuerungPanel.Position = [14 278 611 224];

        % Create VerfahrenzumAnfangButton
        app.VerfahrenzumAnfangButton = uibutton(app.SteuerungPanel, 'push');
        app.VerfahrenzumAnfangButton.ButtonPushedFcn = createCallbackFcn(app,
@VerfahrenzumAnfangButtonPushed, true);
        app.VerfahrenzumAnfangButton.Position = [425 68 156 22];
        app.VerfahrenzumAnfangButton.Text = 'Verfahren zum Anfang';

        % Create WeiterumSchrittweiteButton
        app.WeiterumSchrittweiteButton = uibutton(app.SteuerungPanel, 'push');
        app.WeiterumSchrittweiteButton.ButtonPushedFcn = createCallbackFcn(app,
@WeiterumSchrittweiteButtonPushed, true);
        app.WeiterumSchrittweiteButton.Position = [235 68 136 22];
        app.WeiterumSchrittweiteButton.Text = 'Weiter um Schrittweite';

        % Create SchrittweiteDropDownLabel
        app.SchrittweiteDropDownLabel = uilabel(app.SteuerungPanel);
        app.SchrittweiteDropDownLabel.HorizontalAlignment = 'right';
        app.SchrittweiteDropDownLabel.Position = [24 68 68 22];
        app.SchrittweiteDropDownLabel.Text = 'Schrittweite';

        % Create SchrittweiteDropDown
        app.SchrittweiteDropDown = uidropdown(app.SteuerungPanel);
        app.SchrittweiteDropDown.Items = {'5', '10', '20', '50', '100', '500',
'1000', '2000'};
        app.SchrittweiteDropDown.Position = [107 68 100 22];
        app.SchrittweiteDropDown.Value = '5';

        % Create ResetZhlerButton
        app.ResetZhlerButton = uibutton(app.SteuerungPanel, 'push');
        app.ResetZhlerButton.ButtonPushedFcn = createCallbackFcn(app,
@ResetZhlerButtonPushed, true);
        app.ResetZhlerButton.Position = [425 126 156 22];
        app.ResetZhlerButton.Text = 'Reset Zähler';

        % Create aktuellePositionEditFieldLabel

```

```

app.aktuellePositionEditFieldLabel = uilabel(app.SteuerungPanel);
app.aktuellePositionEditFieldLabel.HorizontalAlignment = 'right';
app.aktuellePositionEditFieldLabel.Position = [83 126 93 22];
app.aktuellePositionEditFieldLabel.Text = 'aktuelle Position';

% Create aktuellePositionEditField
app.aktuellePositionEditField = uieditfield(app.SteuerungPanel,
'text');

app.aktuellePositionEditField.Position = [191 126 182 20];
app.aktuellePositionEditField.Value = '0';

% Create maximalePositionEditFieldLabel
app.maximalePositionEditFieldLabel = uilabel(app.SteuerungPanel);
app.maximalePositionEditFieldLabel.HorizontalAlignment = 'right';
app.maximalePositionEditFieldLabel.Position = [73 165 103 22];
app.maximalePositionEditFieldLabel.Text = 'maximale Position';

% Create maximalePositionEditField
app.maximalePositionEditField = uieditfield(app.SteuerungPanel,
'text');

app.maximalePositionEditField.Position = [191 165 181 21];
app.maximalePositionEditField.Value = '2300';

% Create maxPositionbernehmenButton
app.maxPositionbernehmenButton = uibutton(app.SteuerungPanel, 'push');
app.maxPositionbernehmenButton.ButtonPushedFcn = createCallbackFcn(app,
@maxPositionbernehmenButtonPushed, true);
app.maxPositionbernehmenButton.Position = [426 163 155 22];
app.maxPositionbernehmenButton.Text = 'max Position übernehmen';

% Create VerfahrswegeDropDownLabel
app.VerfahrswegeDropDownLabel = uilabel(app.SteuerungPanel);
app.VerfahrswegeDropDownLabel.HorizontalAlignment = 'right';
app.VerfahrswegeDropDownLabel.Position = [19 35 74 22];
app.VerfahrswegeDropDownLabel.Text = 'Verfahrswege';

% Create VerfahrswegeDropDown
app.VerfahrswegeDropDown = uidropdown(app.SteuerungPanel);
app.VerfahrswegeDropDown.Items = {'2', '4', '6', '8', '10', '12', '14'};
app.VerfahrswegeDropDown.Position = [107 34 101 25];
app.VerfahrswegeDropDown.Value = '2';

% Create SchleifebeginnenButton
app.SchleifebeginnenButton = uibutton(app.SteuerungPanel, 'push');
app.SchleifebeginnenButton.ButtonPushedFcn = createCallbackFcn(app,
@SchleifebeginnenButtonPushed, true);
app.SchleifebeginnenButton.Position = [236 35 136 23];
app.SchleifebeginnenButton.Text = 'Schleife beginnen';

% Create VerbindungzurArduinoStagePanel
app.VerbindungzurArduinoStagePanel =
uipanel(app.ArduinoStageControlUIFigure);
app.VerbindungzurArduinoStagePanel.Title = 'Verbindung zur Arduino-
Stage';

app.VerbindungzurArduinoStagePanel.Position = [15 513 610 131];

% Create VerbindungsstatusLampLabel
app.VerbindungsstatusLampLabel =
uilabel(app.VerbindungzurArduinoStagePanel);
app.VerbindungsstatusLampLabel.HorizontalAlignment = 'right';
app.VerbindungsstatusLampLabel.Position = [369 75 105 22];
app.VerbindungsstatusLampLabel.Text = 'Verbindungsstatus';

% Create VerbindungsstatusLamp
app.VerbindungsstatusLamp = uilamp(app.VerbindungzurArduinoStagePanel);
app.VerbindungsstatusLamp.Position = [338 75 20 20];
app.VerbindungsstatusLamp.Color = [1 1 0.0667];

% Create FehlerLampLabel
app.FehlerLampLabel = uilabel(app.VerbindungzurArduinoStagePanel);

```

```

app.FehlerLampLabel.HorizontalAlignment = 'right';
app.FehlerLampLabel.Position = [369 28 42 22];
app.FehlerLampLabel.Text = 'Fehler';

% Create FehlerLamp
app.FehlerLamp = uilamp(app.VerbindungzurArduinoStagePanel);
app.FehlerLamp.Position = [338 29 20 20];
app.FehlerLamp.Color = [1 1 1];

% Create VerbindungSwitchLabel
app.VerbindungSwitchLabel =
uilabel(app.VerbindungzurArduinoStagePanel);
app.VerbindungSwitchLabel.HorizontalAlignment = 'center';
app.VerbindungSwitchLabel.Position = [205 28 70 22];
app.VerbindungSwitchLabel.Text = 'Verbindung';

% Create VerbindungSwitch
app.VerbindungSwitch = uiswitch(app.VerbindungzurArduinoStagePanel,
'slider');
app.VerbindungSwitch.ValueChangedFcn = createCallbackFcn(app,
@VerbindungSwitchValueChanged, true);
app.VerbindungSwitch.Position = [215 65 45 20];

% Create PortDropDownLabel
app.PortDropDownLabel = uilabel(app.VerbindungzurArduinoStagePanel);
app.PortDropDownLabel.HorizontalAlignment = 'right';
app.PortDropDownLabel.Position = [81 28 28 22];
app.PortDropDownLabel.Text = 'Port';

% Create PortDropDown
app.PortDropDown = uidropdown(app.VerbindungzurArduinoStagePanel);
app.PortDropDown.Items = {'COM1', 'COM2', 'COM3', 'COM4', 'COM5',
'COM6', 'COM7', 'COM8', 'COM9', 'COM10', 'COM11', 'COM12'};
app.PortDropDown.Position = [45 64 100 22];
app.PortDropDown.Value = 'COM1';

% Create ManuellegcodeEingabeZehlerdesVerfahrwegsistnichtgekoppeltPanel
app.ManuellegcodeEingabeZehlerdesVerfahrwegsistnichtgekoppeltPanel =
uipanel(app.ArduinoStageControlUIFigure);
app.ManuellegcodeEingabeZehlerdesVerfahrwegsistnichtgekoppeltPanel.Title
= 'Manuelle gcode Eingabe (Zähler des Verfahrwegs ist nicht gekoppelt)';

app.ManuellegcodeEingabeZehlerdesVerfahrwegsistnichtgekoppeltPanel.Position = [14
199 611 66];

% Create BefehlEditFieldLabel
app.BefehlEditFieldLabel =
uilabel(app.ManuellegcodeEingabeZehlerdesVerfahrwegsistnichtgekoppeltPanel);
app.BefehlEditFieldLabel.HorizontalAlignment = 'right';
app.BefehlEditFieldLabel.Position = [31 11 55 22];
app.BefehlEditFieldLabel.Text = 'Befehl';

% Create BefehlEditField
app.BefehlEditField =
uieditfield(app.ManuellegcodeEingabeZehlerdesVerfahrwegsistnichtgekoppeltPanel,
'text');
app.BefehlEditField.Position = [101 9 183 25];

% Create BefehlsendenButton
app.BefehlsendenButton =
uibutton(app.ManuellegcodeEingabeZehlerdesVerfahrwegsistnichtgekoppeltPanel,
'push');
app.BefehlsendenButton.ButtonPushedFcn = createCallbackFcn(app,
@BefehlsendenButtonPushed, true);
app.BefehlsendenButton.Position = [358 10 100 22];
app.BefehlsendenButton.Text = 'Befehl senden';

% Create GltigkeitdesletztenBefehlsPanel
app.GltigkeitdesletztenBefehlsPanel =
uipanel(app.ArduinoStageControlUIFigure);

```



```

    app.GltigkeitdesletztenBefehlsPanel.Title = 'Gültigkeit des letzten
Befehls';
    app.GltigkeitdesletztenBefehlsPanel.Position = [14 127 611 61];

    % Create GltigkeitLampLabel
    app.GltigkeitLampLabel = uilabel(app.GltigkeitdesletztenBefehlsPanel);
    app.GltigkeitLampLabel.HorizontalAlignment = 'right';
    app.GltigkeitLampLabel.Position = [11 10 56 22];
    app.GltigkeitLampLabel.Text = 'Gültigkeit';

    % Create GltigkeitLamp
    app.GltigkeitLamp = uilamp(app.GltigkeitdesletztenBefehlsPanel);
    app.GltigkeitLamp.Position = [82 10 20 20];
    app.GltigkeitLamp.Color = [1 1 1];

    % Create letzterBefehlEditFieldLabel
    app.letzterBefehlEditFieldLabel =
uilabel(app.GltigkeitdesletztenBefehlsPanel);
    app.letzterBefehlEditFieldLabel.HorizontalAlignment = 'right';
    app.letzterBefehlEditFieldLabel.Position = [160 9 76 22];
    app.letzterBefehlEditFieldLabel.Text = 'letzter Befehl';

    % Create letzterBefehlEditField
    app.letzterBefehlEditField =
uieditfield(app.GltigkeitdesletztenBefehlsPanel, 'text');
    app.letzterBefehlEditField.Position = [251 11 234 19];

    % Create wichtigegcodeBefehlePanel
    app.wichtigegcodeBefehlePanel =
uipanel(app.ArduinoStageControlUIFigure);
    app.wichtigegcodeBefehlePanel.Title = 'wichtige gcode-Befehle';
    app.wichtigegcodeBefehlePanel.Position = [14 20 611 94];

    % Create TextArea
    app.TextArea = uitextarea(app.wichtigegcodeBefehlePanel);
    app.TextArea.Position = [31 15 549 43];
    app.TextArea.Value = {'G91          --> Bewegungsart auf
inkrementelle Bewegungen setzen'; 'G0 Xwert    --> Verfahren um "wert" ("wert" kann
positiv oder negativ sein)'};

    % Show the figure after all components are created
    app.ArduinoStageControlUIFigure.Visible = 'on';
end
end
% App creation and deletion
methods (Access = public)

    % Construct app
    function app = arduinostage_control_V1_1_5

        % Create UIFigure and components
        createComponents(app)

        % Register the app with App Designer
        registerApp(app, app.ArduinoStageControlUIFigure)

        if nargin == 0
            clear app
        end
    end

    % Code that executes before app deletion
    function delete(app)

        % Delete UIFigure when app is deleted
        delete(app.ArduinoStageControlUIFigure)
    end
end
end

```