Kódgenerálás és GUI fejlesztés Matlab-Simulink környezetben

A mérés célja: Grafikus felhasználói felülettel rendelkező alkalmazás létrehozása a Matlab alatt, az App Designer eszköz és elemkönyvtárainak megismerése, kapcsolat a Simulink modellekkel, szimuláció vezérlése.

Felhasznált eszközök: (IK Cloud architektúra – <u>http://cloud.ik.bme.hu</u>), Matlab R2017a vagy 2017b, Simulink, App Designer, Matlab Compiler.

Rendelkezésre álló, letölthető állományok:

- Mérési útmutató (jelen leírás)
- <u>Holtidos szakasz.jpg</u> (szakasz működését szemléltető ábra)

1. feladat – Egyszerű alkalmazás grafikus felülettel

A feladat grafikus felhasználói felület (GUI) létrehozása, amely támogatja, hogy egy holtidős, egytárolós szakaszhoz PI szabályzóméretezést végezzünk¹. A PI szabályzót egy fűtött folyadékot keringető berendezés (például dialízisgép) hőmérséklet szabályozójához kell megtervezni, amelynek sémája az ábrán látható.



1. ábra. A holtidős szakasz működésének vázlata.

A szakasz viselkedése egy adott hőmérséklet, mint munkaponti érték környékén jól jellemezhető egy átviteli függvénnyel: $W(s) = \frac{A}{1+sT} \exp(-sT_h)$. Ehhez az átviteli függvényhez tervezünk egy $W_{PI}(s) = \frac{A_p}{T_i} \frac{1+sT_i}{s}$ átvitelű PI szabályzót.

Megoldás: A fejlesztést a GUIDE vagy az App Designer eszköz segítségével végezhetjük. A GUIDE összetettebb eszköz, ugyanakkor egyszerűbb applikációk (Appok) fejlesztéséhez az egyszerűsített és letisztultabb App Designer javasolható. A feladatban szereplő eszköz esetében az App Designer alkalmazása elegendő lesz, az alkalmazásmodell a GUIDE esetében is hasonló.

A fejlesztő eszköz indításakor egy újabb felület jelenik meg. Az alkalmazás felépítése egyszerű. Külön szerkesztői nézetben helyezhetjük el a felhasználói felület elemeit (Design View). Az programozási modell eseményvezérelt, az egyes kezelőelemek eseményeihez rendelt kódot is egy külön nézetben szerkeszthetjük (Code View). A kód egy részét a fejlesztőkörnyezet generálja, ennek szerkesztése nem tanácsos.

A fejlesztőkörnyezet további elemeinek használata is értelemszerű. Az alkalmazható kezelőszervek könyvtára a bal oldalon található, ezeket *drag & drop* művelettel példányosíthatjuk az alkalmazásunkban. Az egyes elemekhez

¹ PI szabályozó méretezése a Rendszerelmélet (VIHVAB00) tárgyból szerepelt

eseménykezelő függvényeket a jobb oldali menüben rendelhetünk és itt állíthatók a felület elemeinek tulajdonságai is.

📣 MAT	LAB R2	2017a - academic	use	
но	ME	PLOTS		APPS
New Script	New	Open	ïles are	a La
		Script Ctrl-	۰N	VARIABLE CODE
< ⇒			4	iments 🕨 Oktatas 🕨 GyorsPrototipus_valaszthato 🕨 Matlab 🕨
Curren		Live Script		Command Window
	fx	Function		, <i>f</i> x, >>>
	P	Example		
	<u>}</u>	Class		E
	<u>}</u>	System Object	>	
		Figure		
		Арр	>	GUIDE Build figure-based apps with full 2D and 3D graphics support
	7	Command Shortc	ut	
dof1_wrapper.c				Build uifigure-based apps with 2D graphics support

2. ábra. Alkalmazások fejlesztése (GUIDE és App Designer)

Az alkalmazást a Run paranccsal indíthatjuk, a futtatás értelemszerűen a Matlab keretrendszerben zajlik.

App Designer - app1.mlapp	
DESIGNER CANVAS	🔓 🤊 🔿 🕐
Image: New Open Save Package App Run	
FILE SHARE RUN DI Designer Simulator miano (X) and miano (X)	A
COMPONENT LIBRARY Design View Code View COMPONENT BROWSER	
COMMON A app.UlFigure	
Axes (2D)	
Button Button	
Check Box	
a - Drop Down	
123 Edit Field (Numeric)	
abc Edit Field (Text)	
A Label	
a List Box	
Title a Radio Button Group	
10 Slider	
123 € Spinner Id	►

3. ábra. Az App Designer felülete.

Kezdjük a kép elhelyezésével. Képet a Axes (2D), UIAxes osztályú objektum esetében mutathatunk a felhasználónak. A jpg képet az erre szolgáló imshow metódus meghívásával lehet beilleszteni. A metódus hívását ugyanakkor be kell illeszteni egy eseménykezelő függvénybe, viszont az UIAxes objektumokhoz nem tartozik eseménykezelő. Ezért az alkalmazás ablakának (UIFigure) egyik eseménykezelőét használjuk

(startupFcn), amely annak létrehozásához tartozó esemény nyomán kerül végrehajtásra. Ennek kódját mutatja a 4. ábra.

```
methods (Access = private)
% Code that executes after component creation
function startupFcn(app)
imshow('Holtidos_szakasz.jpg','parent',app.UIAxes);
end
```



A PI szabályozó tervezéséhez a fejlesztői felületen például a 5. ábra szerinti módon helyezhetjük el az elemeket. A szakasz paramétereinek megadásához numerikus szerkesztő mezőket (Edit Field (Numeric)), a számított paraméterek megjelenítéséhez pedig Label elemeket használhatunk.

承 PI méretező	
h u konstans	oltidő
fütőszál	hőmérséklet érzékelő
Stakast paramátorok	PI paramétarak
A0	Ap ismeretlen
Τ0	Ti ismeretlen
Th 0	Méretezés

5. ábra. A PI méretező felülete.

A tervező eljárást egy különálló függvényben is lehet implementálni, hogy a fázistartalék például 60 fok. A méretezés a korábban tanultak alapján mindössze három sor, amit a felület Méretezés nyomógombjának eseményéhez rendelünk.

```
% Button pushed function: DesignButton
function DesignButtonPushed(app, event)
A = app.PlantGain.Value;
T = app.PlantTimeCte.Value;
Th = app.PlantTimeLag.Value;
% controller design calculations for 60 degrees phase margin
Ti = T;
wc = pi/6/Th;
Ap = Ti*wc/A;
app.TiValue.Text = num2str(Ti);
app.ApValue.Text = num2str(Ap);
end
```

6. ábra. A Méretezés gomb megnyomásához tartozó eseménykezelő függvény kódja.

2. feladat - A PI méretező módosítása

Módosítsa az előző feladat nyomán keletkezett alkalmazást, hogy a fázistartalék értékét is a felhasználó adhassa meg!

3. feladat – A PI méretező tesztelése szimulációval

Hozzon létre egy Simulink modellt a zárt szabályozási kör szimulációjához, amelynek munkaterében szerepelnek változóként a szakasz és a szabályzó paraméterei! Egészítse ki a felhasználói felületet, hogy a szabályzó méretezése nyomán a zárt kör szimulációjának eredménye a felületen megjelenjen (egységugrás alapjel esetén).

Megoldás: A Matlabon belül nem csak az 'alap' munkatér (Base Workspace) tartalmazhat változókat, minden Simulink modellnek is van saját munkatere. A szabályozási körünk paramétereit a modell munkaterében hozzuk létre a modellböngésző (Model Explorer) segítségével. Ehhez először egy modellt kell megnyitni.



A modellböngészőben a model munkaterében egyszerűen létrehozhatjuk és inicializálhatjuk a változókat az összes paraméter számára (szakasz és szabályzó). A változókra a modell blokkjainak párbeszéd ablakában hivatkozhatunk.

🔯 Model Explorer			
File Edit View Tools Add	i Help		
- 🚅 X 🖻 🛍 🗙 🗄	🛄 🛨 🐁 張 🗐 🖉 🖡 fo	0 🔲 📣 🕽 😤 🧮	≠ ⁄ ⁄ ⁄ 2
Search: by Name	▼ Name:	Search	
Model Hierarchy 🏾 🦉 😓	Contents of: Model Workspace	Filter Contents	Model Workspace
 Simulink Root Base Workspace 	Column View: Data Obje 🔻 Sho	w Details <u>5 object(s)</u> 💎	Workspace data Data source: MDL-File
Closed_loop	Name Value	DataType Min Max Di	
Model Workspace	🗄 A 5	double (auto)	Model arguments (for referencing this model):
2 Advice for Closed	Ap 0.0785398	double (auto)	
Simulink Design	🗄 Т 1.5	double (auto)	
Configuration (Ac	🗄 Th 2	double (auto)	
	🗄 Ti 1.5	double (auto)	
• III •	Contents Search R	tesults	Revert Help Apply



A Simulink modellben a holtidőt is kezelő LTI blokkot használunk és a szimulált kimenetet egy mat fájlba mentjük.



8. ábra. A holtidőt tartalmazó szabályozási kör Simulink-ben

A felhasználói felületet ki kell egészíteni egy UIAxes típusú objektummal, amelyben a szimulált tranzienst fogjuk megjeleníteni. Természetesen ezt megintcsak a App Designer segítségével tesszük meg.

🚸 App Designer - D:\Documents\Oktatas\RendszertervezesLaboratorium1\IIT meres 2\PI_Designer_Simulator.mlapp	
DESIGNER CANVAS	XXXX 🕨 🖬 🔏 🛍 🔊 c 📀 🛛
PLDesigner_mlapp 🛞 PLDesigner_Simulator.mlapp 🛞	
COMPONENT LIBRARY Design View Orde View	COMPONENT BROWSER
COMMON A	
Axes (2U)	app.PlantGain
	app.PlantTimeCte
Szakasz kimenet	app.PlantTimeLag
Check Box 0.8	 app.PlparamterekPanel
0.8	app.ApLabel
a Drop Down 0.6	app.TiLabel
	app.ApValue
123 Edit Field (Numeric) 0.4	app. I ivalue
abo Edit Elekt (Tevit) 0.2	app.DesignButton
	app.01AXes
A Label 0 0.2 0.4 0.6 0.8 1	ann SimResults
List Box Szakasz paraméterek PI paraméterek	AXES PROPERTIES
Title Qa Radio Button Group	Configuration Callbacks
U Ap ismereten	✓ LABELS
Commission Sider	Title Szakasz kimenet
T 0 Ti ismeretien	X Label Idő (sec)
123 Spinner	Y Label Amplitúdó
Base State Button Th 0 Méretezés	
	 APPEARANCE
Table	Grid and Box
Szimulálás	· · · · · · · · · · · · · · · · · · ·
	× •
	14

9. ábra. Újabb UIAxes objektum hozzáadása a felülethez a szimuláció eredményének megjelenítéséhez

Utolsó lépésként a szimulációhoz tartozó gomb eseménykezelőjének megadására van szükség, amely az alábbi feladatokat végzi el:

- módosítja a modell munkaterében található változókat
- lefuttatja a szimulációt
- betölti az eredményeket tartalmazó állományt
- törli az aktuális tranzienst és a helyére kirajzolja az újat

Ehhez az alábbi utasítások állnak rendelkezésünkre:

- Simulink modellünk megnyitására a load_system utasítás szolgál
- A munkatérhez tartozó handle-t a get_param utasítás adja vissza
- Valamely munkatér változójához az assignin utasítással férhetünk hozzá
- Modellünk szimulációját az sim utasítással kezdeményezetjük

A keletkező kódot a 10. ábra mutatja.

10. ábra. Szimuláció futtatása eseménykezelő függvényből.

4. feladat – Az alkalmazás kiegészítése

- 1. Egészítse ki az előző feladatban kapott alkalmazást, hogy a szimuláció időtartalma mindig automatikusan a beállított holtidő ötszöröse legyen! A szimulációs idő a Simulink modell egyik paramétere és közvetlenül beállítható a sim utasítás egy argumentumaként (v.ö. doc sim).
- 2. Egészítse ki az előző feladatban kapott alkalmazást, hogy a felületen a beavatkozó jel is megjelenjen!
- 3. Egészítse ki az alkalmazást, hogy a szabályozó paramétereit egy belső (private láthatósági osztályú) függvény számolja. Ilyen függvény létrehozásához nyújt segítséget a 11. ábra.



11. ábra. Függvények (private vagy public láthatósági osztályú) létrehozása

5. feladat – Önálló alkalmazás létrehozása az elkészített App-ből

Az eddig elkészült Appok futtatásához a Matlab indítása szükséges volt. Ugyanakkor a Matlab fordítóval (Matlab Compiler) eszközének segítéségével lehetőség van olyan önálló (standalone) alkalmazás előállítására is, amelynek futtatásához nincsen szükség a Matlab indítására. Fontos megjegyezni ugyanakkor, hogy a lefordított alkalmazáshoz vagy hozzácsomagoljuk a Matlab egy részét (ez akár 1Gb is lehet) vagy lehetővé tesszük, hogy ehhez az alkalmazásunk a hálózaton keresztül férjen hozzá. A fordító is egy App persze, ezt mutatja a 12. ábra.

A felület használata értelemszerű, esetünkben a szükséges állományok kiválasztása is egyszerű. A kiválasztandó MAIN FILE esetünkben a PI_Designer.mlapp, további függvényekre és szkriptekre nincsen szükség. Az alkalmazás létrehozásához a Package parancsot kell használni. Sajnos a kód analízise nem elég fejlett ahhoz, hogy az imshow utasításnál megadott Holtidos_szakasz.jpg fájlt is a csomagolni kívánt állományok közé illessze a keretrendszer, így azt kézzel kell megtenni a Files installed for your end user mezőnél.

A hosszas fordítás után a végrehajtható állományok a PI_Designer könyvtárban keletkeznek. A Matlab három külnböző változatot is előállít, próbálkozzunk a for_redistribution_files_only könyvtárban található PI_Designer.exe futtatásával.

Rendszertervezés laboratórium

MATLAB R2017a - acad	emic use
HOME PLO	IS APPS
Get More Install Package Apps	Instrument OPC Data Vehicle CAN Control Access Expl Bus Monitor
FILE Current Folder Name A Closed_loop_PI_grt. PI Designer Standal	CODE GENERATION
 slprj -SteLab IIT mérés 2 -WRL3447.tmp Closed_loop_PI.exe Closed_loop_PI.ski Holtidos_szakasz.jpg Holtidos_szakasz.jpg 	2 CODE VERIFICATION VIET VIET Polyspace Bug Finder Code Prover
Augustoffmal.docx output.mat output.mat PI_Designer.mlapp PI_Designer_Simulat PI_Designer_Standal ReteLab IIT mérés 2	APPLICATION DEPLOYMENT Application Compiler Package MATLAB programs for deployment as standalone applications (applicationCompiler) MATLAB Compiler 6.4 Compiler Com

12. ábra. A Matlab fordító felületének indítása

📣 MATLAB Compiler	- Pl_Designer.prj	3
COMPILER		?
New Open Save	Standalone Application PI_Designer.mlapp Runtime included in package MyAppInstaller_mcr 1 GB Settings Package 	
▼ Project ▼	TYPE MAIN FILE PACKAGING OPTIONS SETTINGS PACKAGE	-
		•
	Application information	
	PI_Designer 1.0	
	Hallgató CtrlCV	E
	ВМЕШТ	
	Set as default contact	
	Rendszertervezés laboratórium IIT2 mérés keretében készített alkalmazás	
	Egy holtidős szakaszhoz tervezhető az App segtségével szabályozó.	
	Additional installer options	
		-

13. ábra. A Matlab fordító felülete