

# Lekcija 3:

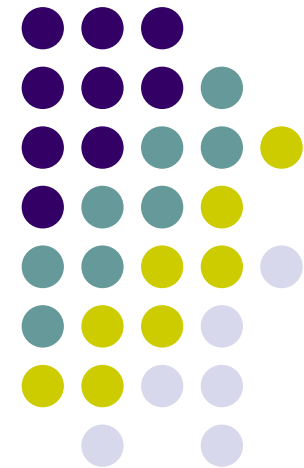
## *Real-time simulacije*

---

Prof.dr.sc. Jasmin Velagić  
Elektrotehnički fakultet Sarajevo

Kolegij: Mehatronika

2012/2013



# 3. Real-time simulacije



**Kontinuirano i diskretno vrijeme.  
Statičko i dinamičko ponašanje.**

**Simulacija bez vremenskog ograničenja**

- osnovno istraživanje ponašanja,
- verifikacija teoretskih modela,
- dizajn procesa,
- dizajn sistema upravljanja.

## Vrste simulacija

**Simulacija brža od realnog vremena**

- modelski sistemi upravljanja
  - prediktivno upravljanje,
  - adaptivno upravljanje.
- on-line optimizacija,
- razvoj strategija, planiranje, predviđanje,
- komponente za real-time simulacije.

**Real-time simulacija**

**Simulacija procesa**

- hardver u simulacijskoj petlji (HIL) ,
- treniranje operatora.

**Simulacija regulatora**

- Testiranje regulatora korištenjem brzog razvoja prototipa (RCP).

**Simulacija procesa i regulatora**

# Real-time simulacije

- Šta je to real-time simulacija?
- **Simuliranje brzine izvršavanja kao u stvarnom svijetu, niti brže niti sporije od realnosti.**
- Temelji se na integraciji fiksnog vremenskog koraka koji se obično mjeri u mikro ili milisekundama.
- **Fizičko vrijeme:** vrijeme u fizičkom sistemu sa kojim sistem radi.
- **Simulacijsko vrijeme:** Simulacijski prikaz stvarnog vremena.
- **Izmjereno simulacijsko vrijeme:** stvarno vrijeme kojim se obavlja simulacija.



# Real-time simulacije

- Skaliranje real-time simulacije

- Imamo linearnu vezu  $\Delta T = S \cdot \Delta W$ , gdje je  $\Delta W$  stvarno trajanje simulacije (izmjereno simulacijsko vrijeme),  $\Delta T$  odgovara simulacijskom vremenu i  $S$  je skalirajući faktor.
- Ako je  $S=1$ , tada imamo **real-time simulaciju**.

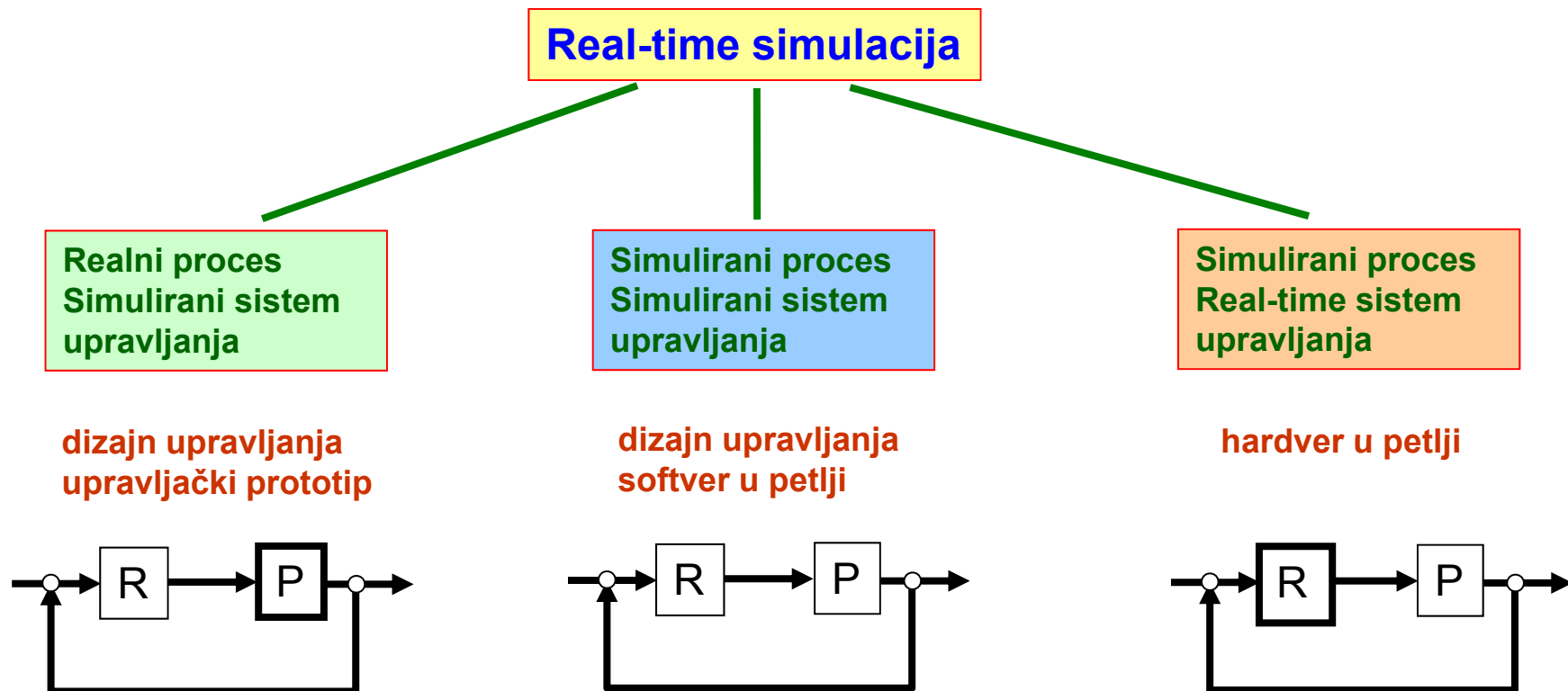
- Što je moguće brže simulacije

- Kada pokušavamo izvršiti (kompletirati) simulaciju čim brže, tada se izvršavanje obavlja na način da nemamo poklapanje sa stvarnim vremenom simulacije.

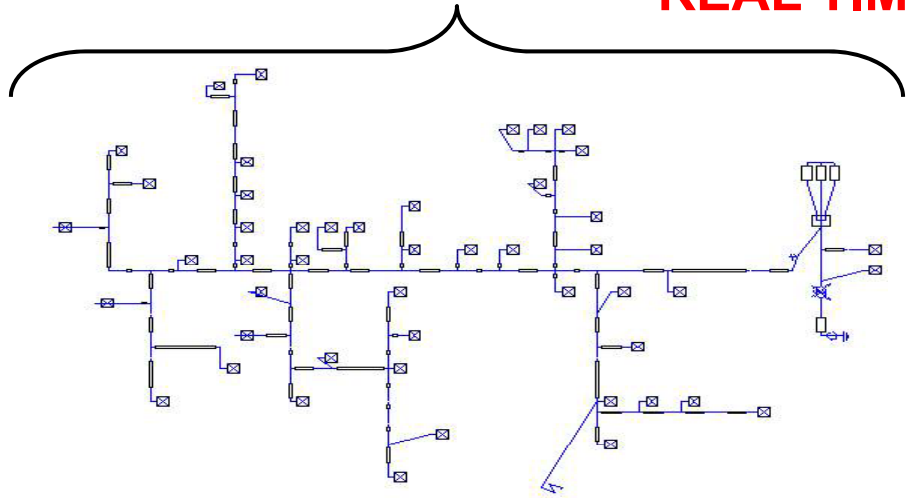
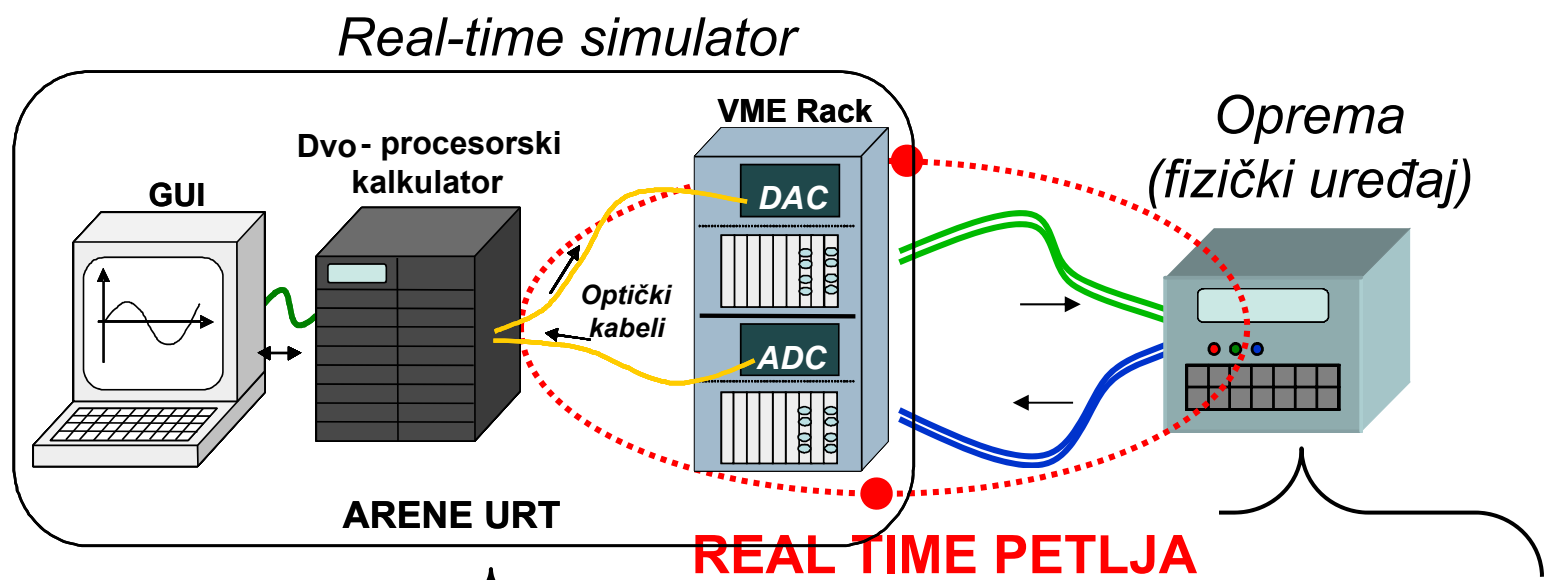


# Real-time simulacije

- Modeli se pokreću na specijalnim procesorski zasnovanim karticama sa *solverima fiksne koraka*.



# Real-time simulacije – zašto?



- Zastitini releji
- Komunikacijske tehnike
- Upravljački uređaji

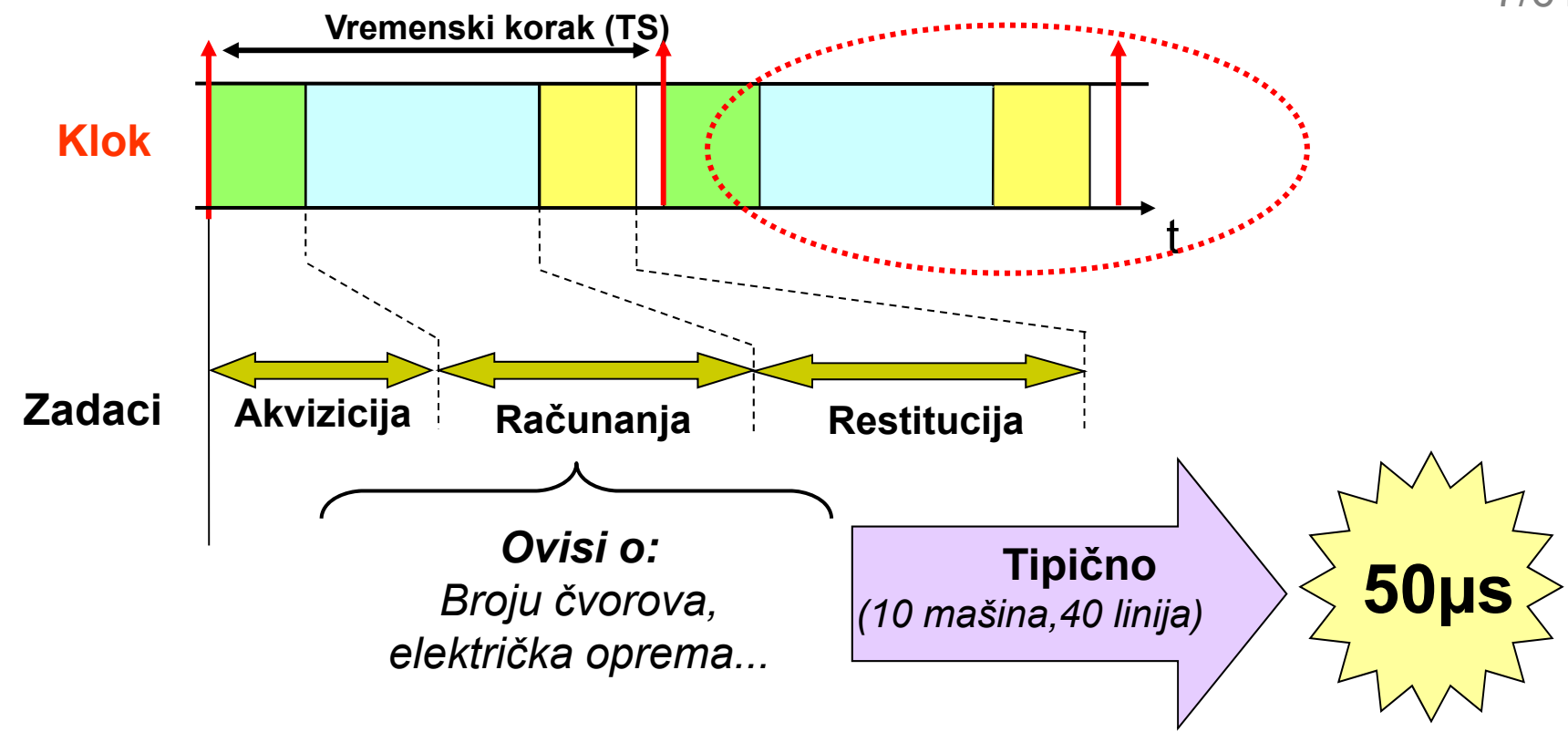


**TEST & VALIDACIJA**

# Real-time simulacija – fiksni vremenski korak



⇒ Sinhronizacija podataka sa vanjskom opremom



⇒ (vrijeme računanja + vrijeme akvizicije) < TS

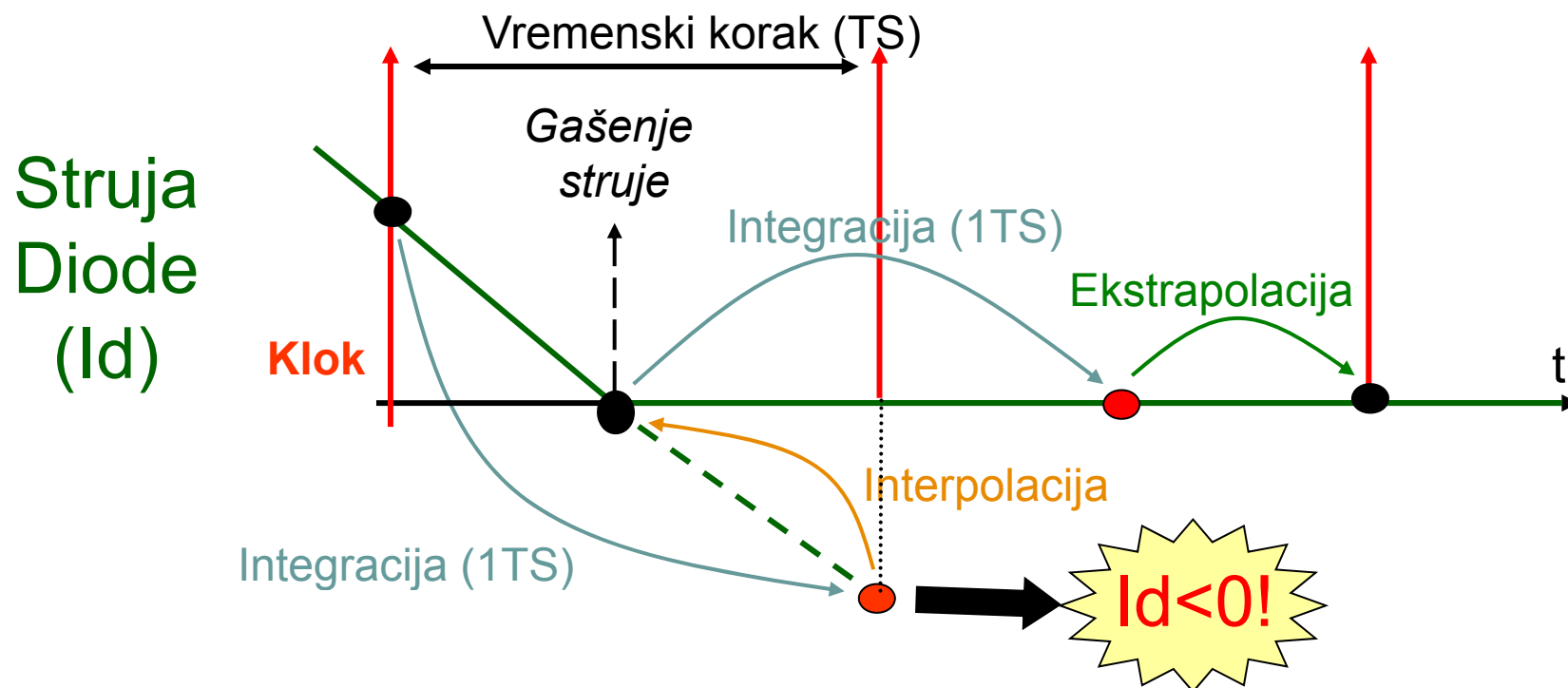
# Real-time simulacija – fiksni vremenski korak



⇒ Simulacija energetske sistema

Događaji koji uzrokuju prekide između dva vremenska koraka

8/61

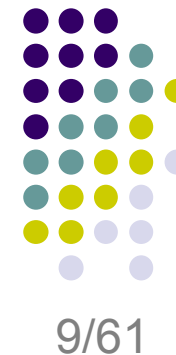


⇒ Algoritmi detekcije

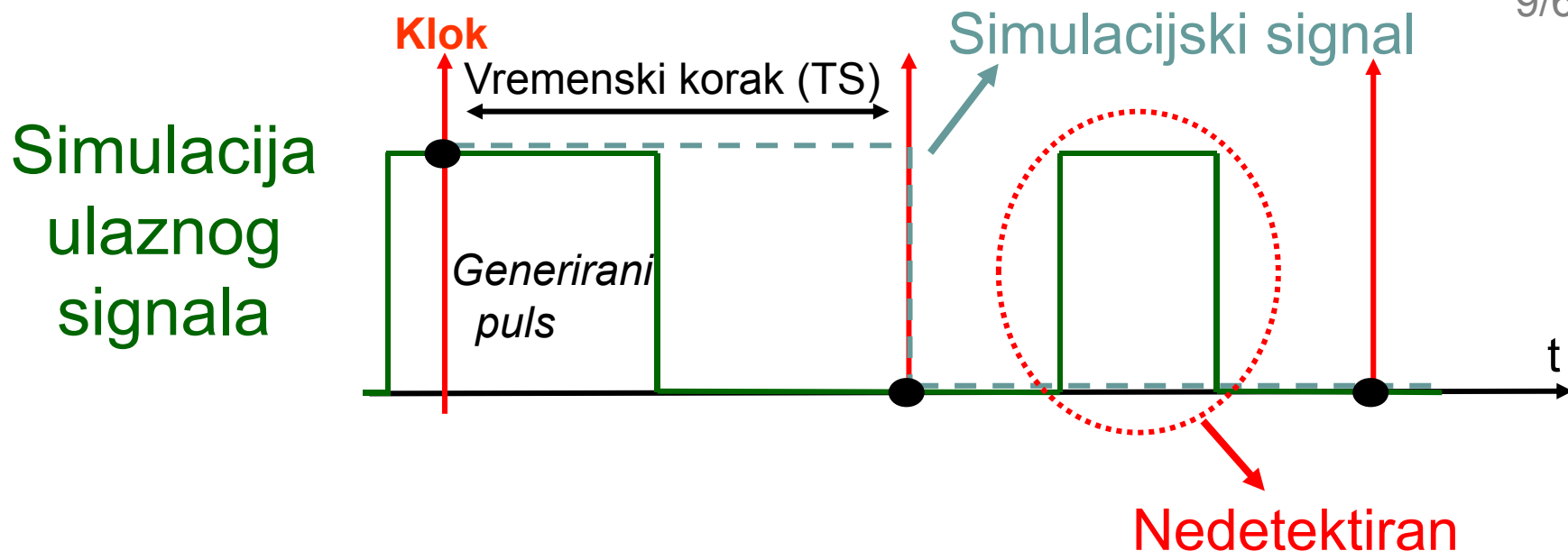
**CSSC** (*Clock Synchronized Status Changing*)



# Real-time simulacija – fiksni vremenski korak



⇒ Akvizicija vanjskog signala



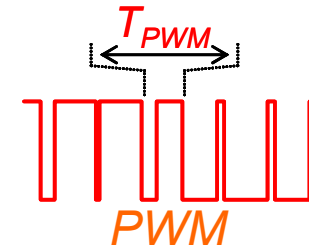
Uzorkovanje realizirano u svakom vremenskom koraku

- Promatrajmo Shannon-ov teorem:  $TS < \frac{T_{signal}}{2}$

**TS > (vrijeme računanja + vrijeme akvizicije) !**

# Real-time simulacija – PWM signal

⇒ Slučaj PWM signala ( $f_{PWM} > 1\text{kHz}$ )



***Više od jednog prekidačkog događaja tokom vremenskog koraka signala***

Razviti odgovarajuće modele:

- smanjiti vrijeme računanja
- vrijeme invarijantno (bez događaja prekidanja).

⇒ **Modeli  
usrednjavanja**

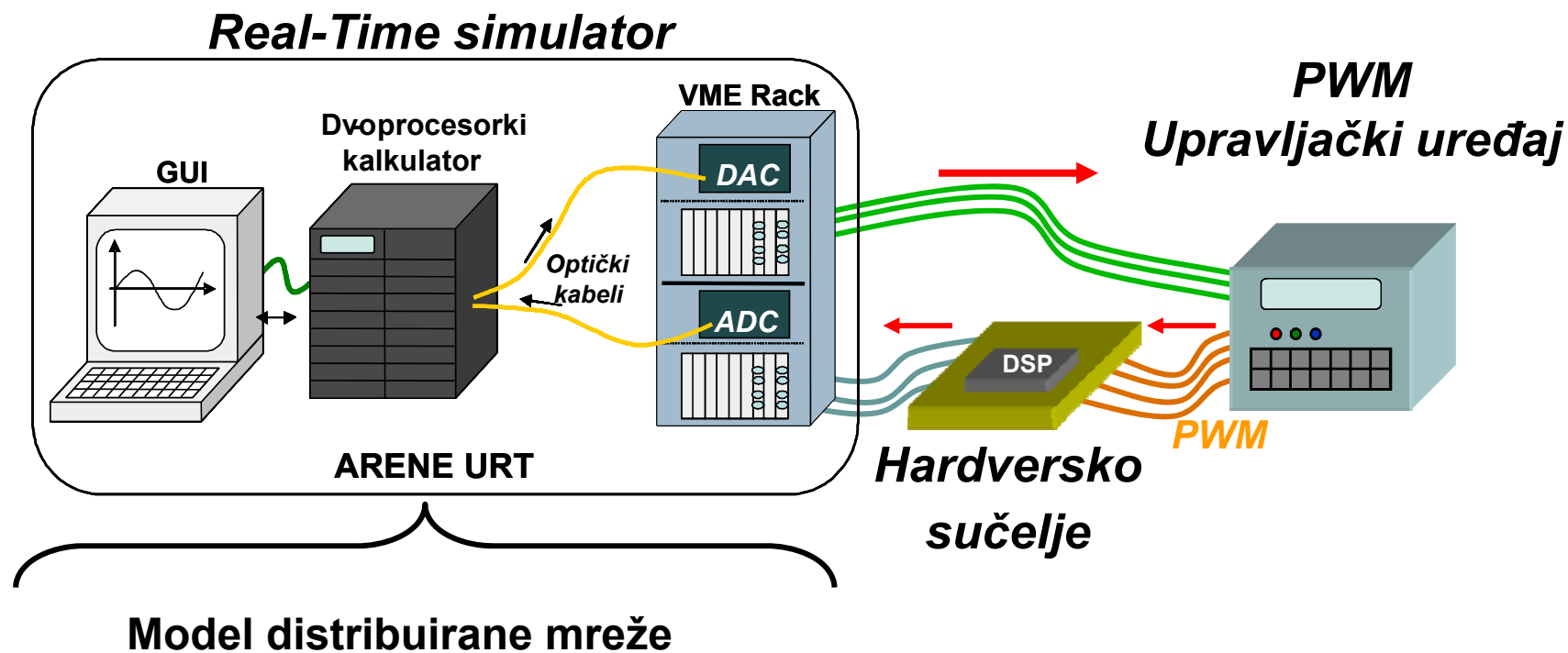
***Problem uzorkovanja...***

Prilagoditi signale simulatora i PWM opreme

⇒ **Razvoj hardverskog sučelja**

# Real-time simulacija

## Hardversko sučelje



# Real-time simulacija

## Hardversko sučelje



12/61

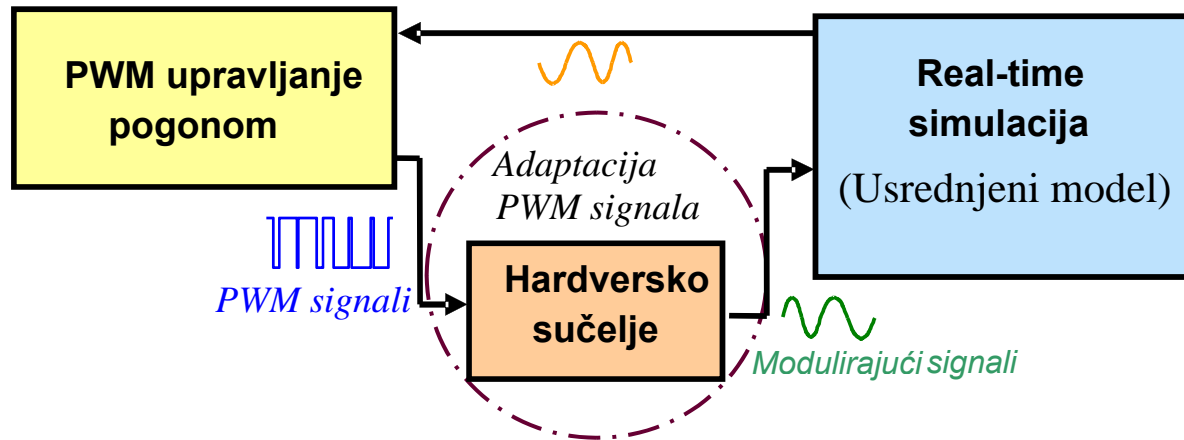
**dSPACE**  
PWM  
upravljajući  
uređaj



**ARENE**  
**URT**  
Usrednjeni  
model  
  
+ mreža

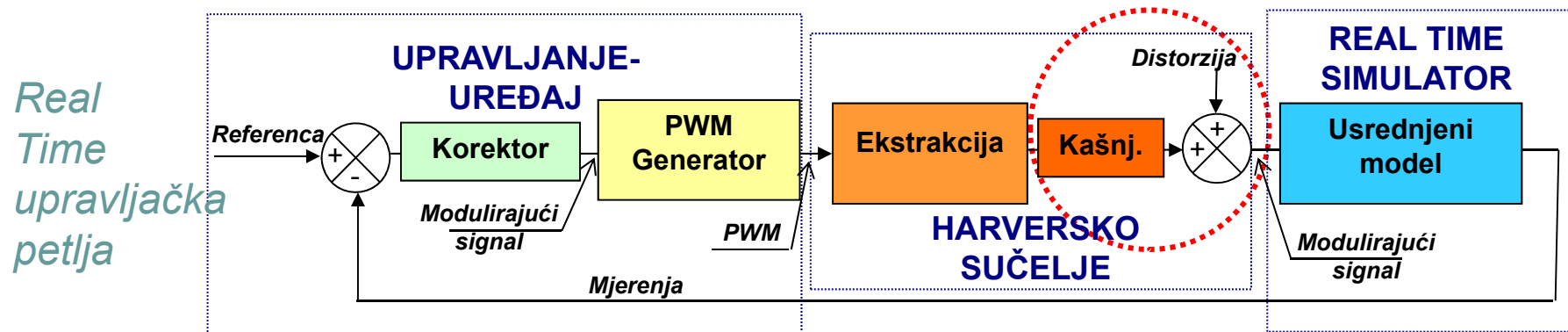
# Real-time simulacija

## Hardversko sučelje



Ekstrakcija modulirajućeg PWM signala

⇒ Minimalno kašnjenje za ekstrakciju i mala distorzija



Real Time upravljačka petlja

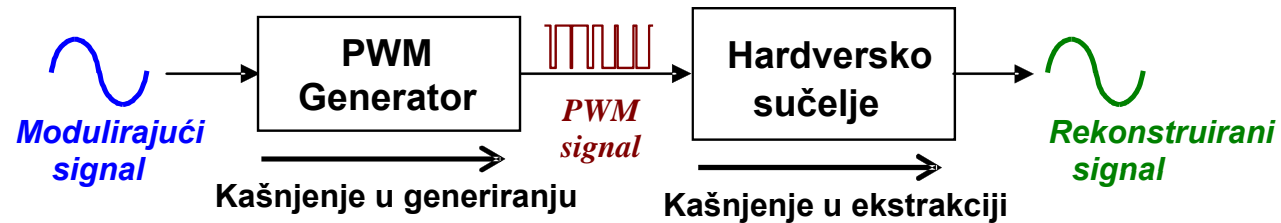
# Real-time simulacija

## Hardversko sučelje - rezultati

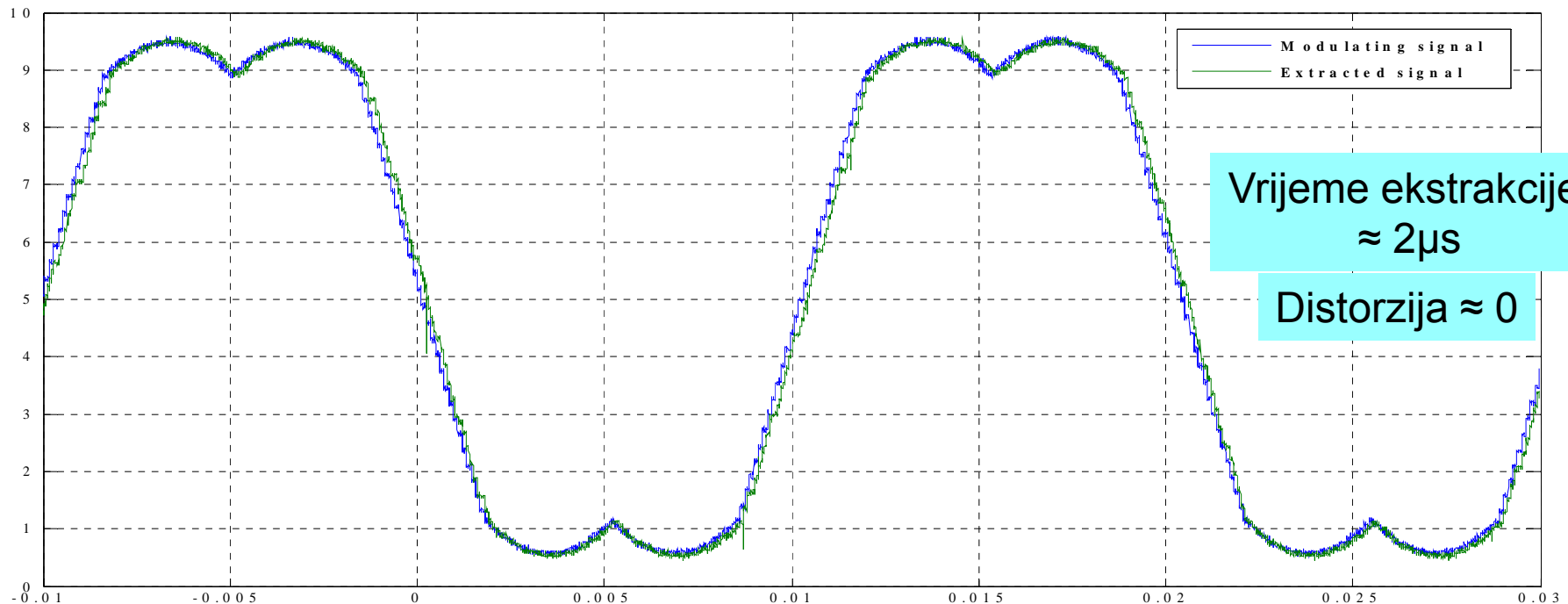
Hardversko sučelje realizirano sa DSP-om



14/61



Rezultati : ekstrakcija PWM vektora ( $f_{\text{PWM}} = 10\text{kHz}$ )



# Real-time simulacije

## Primjena real-time simulacije



15/61

### ***Postavljanje i razvoj prototipa***



Prije mrežne komunikacije  
Validacija algoritama  
Jednostavna i brza realizacija

### ***Testiranje kvarova (nedostataka) na opremi***



Test sigurnosti sistema  
Ponovno kreiranje situacije u mreži : "povrat" pohranjenih podataka  
Testovi reprodukcije (digitalna simulacija)

### ***Testiranje industrijske opreme***



Studije: interakcija, generiranje poremećaja  
Normalne i kritične situacije  
Analiza opreme od različitih proizvođača

## 3.1. Brzi razvoj upravljačkog prototipa (RCP)

- Za dizajn i testiranje složenih upravljačkih sistema i njihovih algoritama unutar real-time ograničenja, može se koristiti simulacijski real-time regulator (kontroler) sa hardverom – [upravljački prototip](#).
- Proces, senzori i aktuatori mogu biti realni (fizički).
- Prednosti brzog razvoja upravljačkog prototipa:
  - Rani razvoj metoda za obradu signala, modela procesa i strukture sistema upravljanja, uključujući algoritme sa visoko razinskim softverom i visoko performansijskim off-the-shelf hardverom.
  - Testiranje upravljačkih sistema i sistema za obradu signala, zajedno sa dizajnima aktuatora, procesnim dijelovima i senzorskim tehnologijama, da bi kreirali sinergijske efekte.
  - Reduciranje modela i algoritama za postizanje zahtjeva za masovnu jeftiniju proizvodnju hardvera.
  - Definiranje specifikacija za finalni hardver i softver.



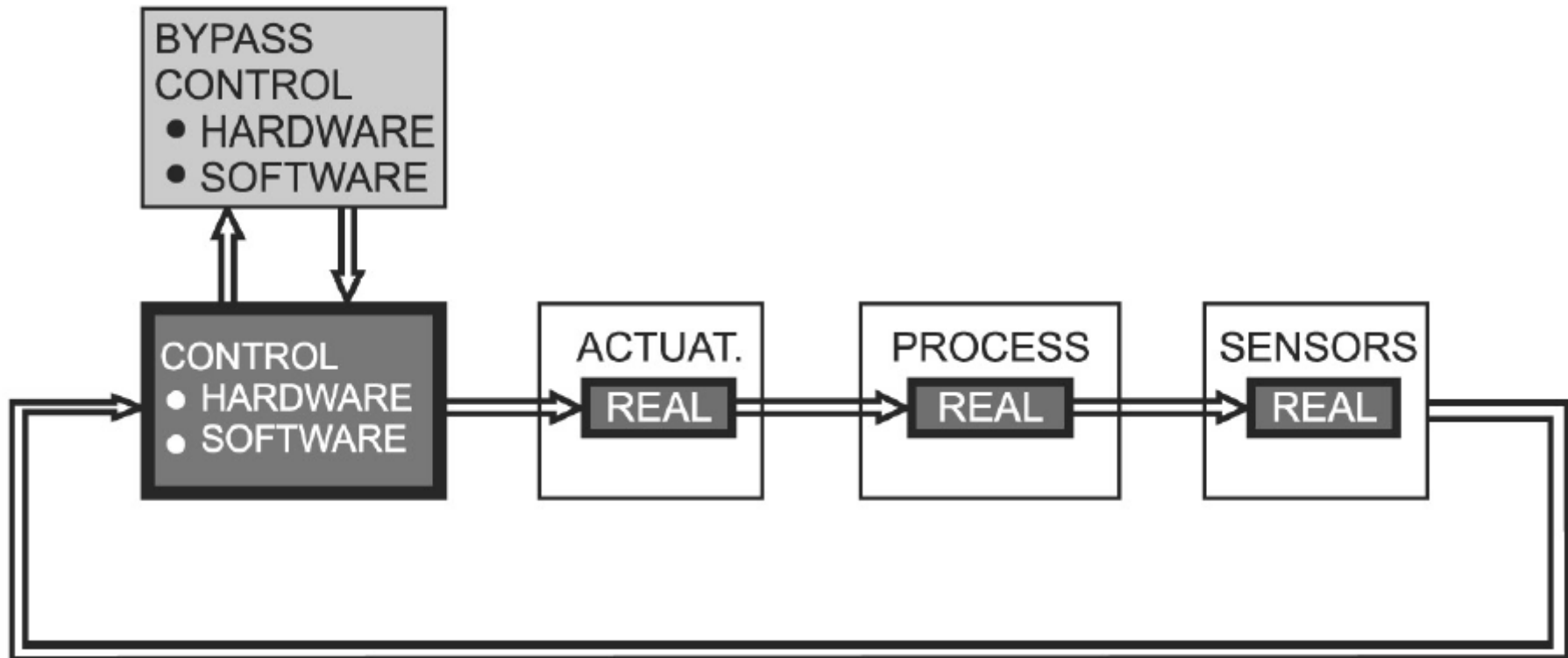


# Brzi razvoj upravljačkog prototipa (RCP)

- Sve komponente sistema su hardverske (realne), izuzev regulatora (kontrolera).



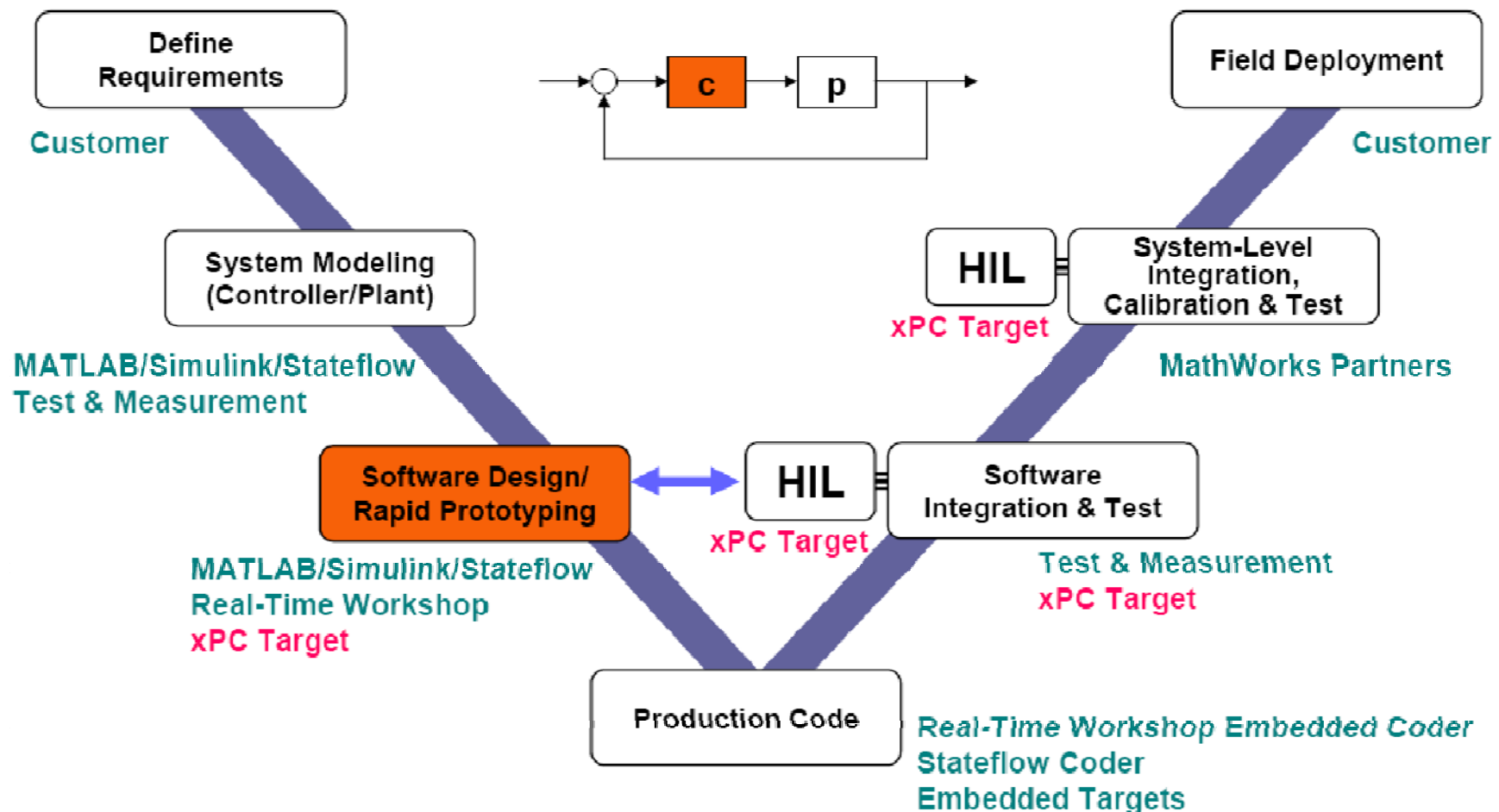
17/61



# Brzi razvoj upravljačkog prototipa (RCP)



## V-dizajn brzog razvoja upravljačkog prototipa

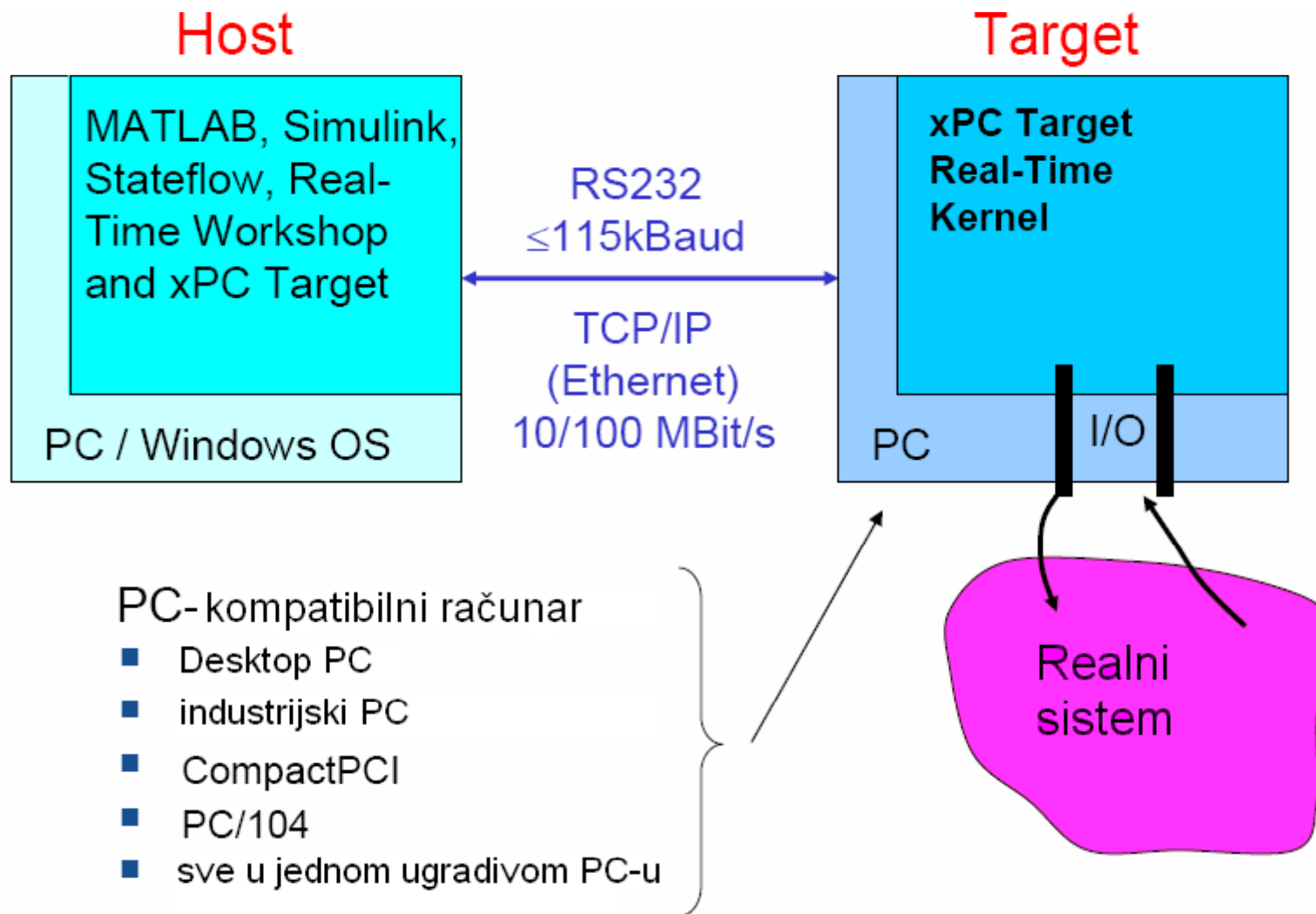


# Brzi razvoj upravljačkog prototipa (RCP)

- MATLAB i Simulink sa svojim dizajnom temeljenim na modelu (model-based design) omogućuju proizvodnju simuliranog prototipa (izvršivi dizajn).
- Cilj je da sistemski inženjeri mogu isporučiti izvršivi model.
- Rješenje za ovo je brzi razvoj prototipa unutar istog alata.
- U MATLAB-u se koristi **xPC Target**: općeniti alat za brzi razvoj prototipa.
- xPC target je u cijelosti integriran unutar Simulink-a.
- **Osnažuje (ojačava) PC kao platformu za izvršavanje real-time aplikacija.**
- Prilagođen potrebama sistemskih inženjera.



# Brzi razvoj upravljačkog prototipa (RCP)

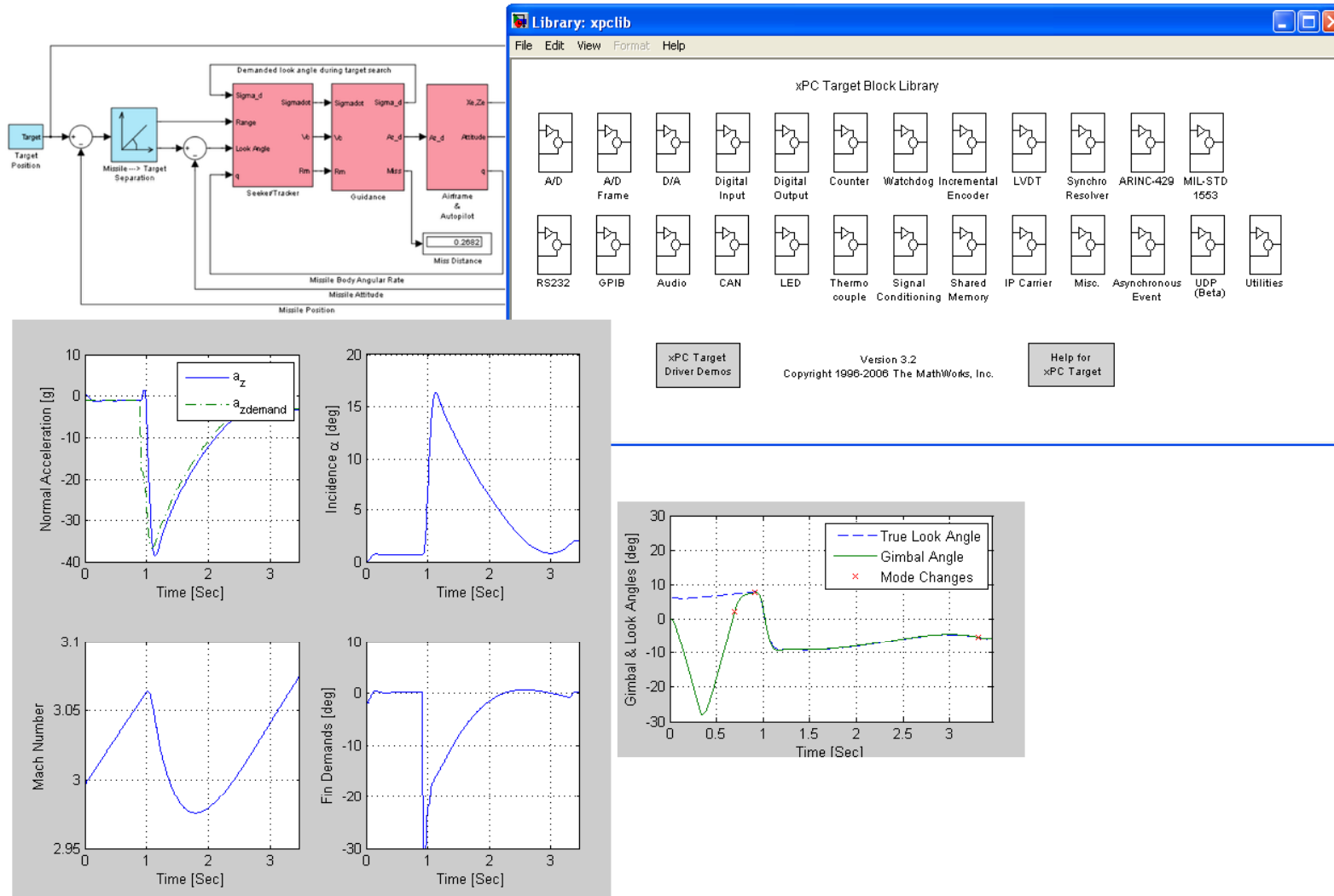


# Brzi razvoj upravljačkog prototipa (RCP)

## Host alati



21/61



# Brzi razvoj upravljačkog prototipa (RCP)

- PC kompatibilni ciljni (target) hardver:
  - Realni COTS standard.
  - Bezbrojni snadbjevači (CPU i I/O hardver).
  - Mnogo različitih form faktora.
  - Specijalni I/O tipovi raspoloživi za PC-ove.
  - Troškovno efikasni.
  - Automatsko poboljšanje performansi.
  - Višekratno korišteni hardver.

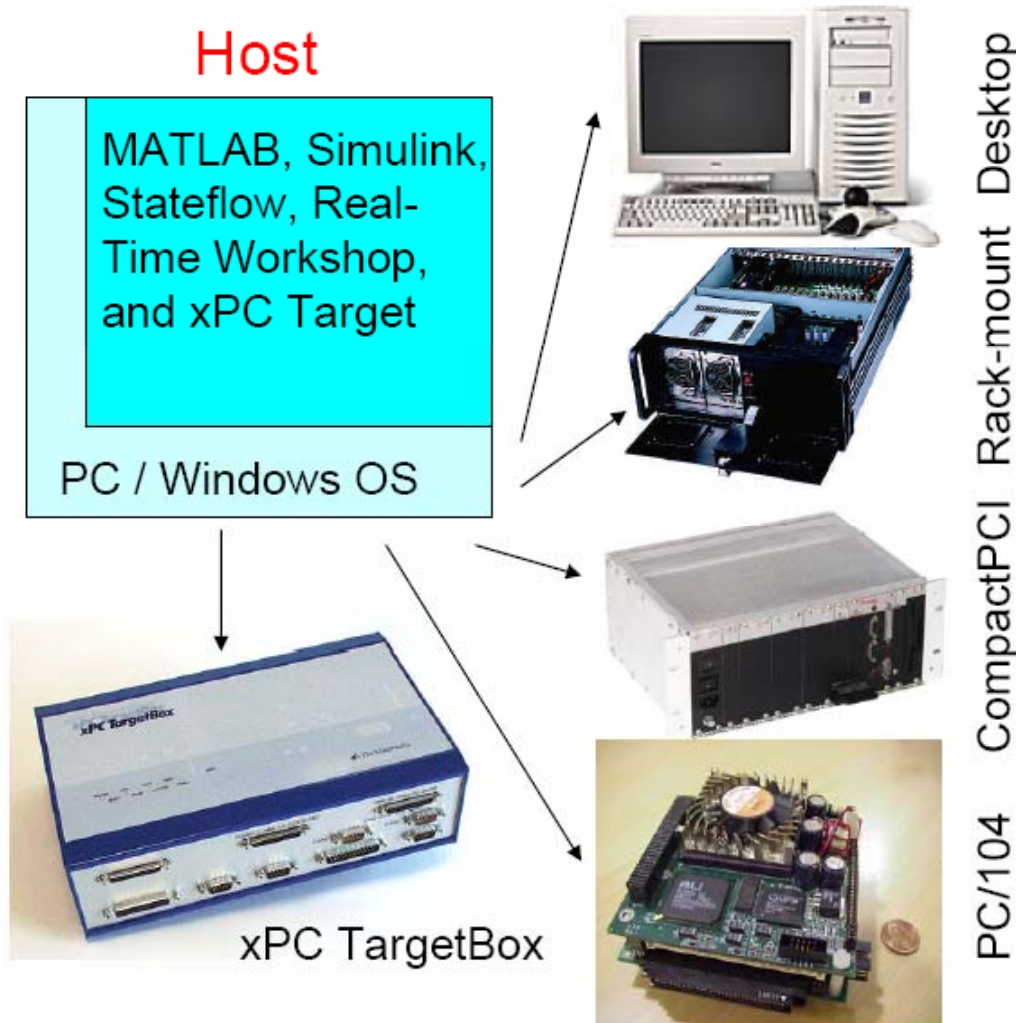


# Brzi razvoj upravljačkog prototipa (RCP)

## PC kompatibilni form faktori



23/61



Primjer: Zračna kompanija koristi stalažno montiranu (rack-mount) sistem za HIL



# Brzi razvoj upravljačkog prototipa (RCP)

- xPC target real-time kernel
  - Neovisan o operacijskim sistemima.
  - Omogućuje minimalno vrijeme uzorkovanja od  $10\mu\text{s}$ .
  - Podržava Interrupt i Polling Scheduling Mode.
  - Optimiziran za Simulink, Stateflow Blockset na modelima temeljene aplikacije.
  - Pokretljiv (bootable) sa diskete i Embedded opcije iz bilo kojeg pokretljivog uređaja (hard disk, FlashRAM, ...).
  - Izvršava u 32bit zaštićenom modu sa linearnim virtualnim adresnim prostorom od 4 GB – nekad ne preklapa nazad na 16bit x86 mod.





# Brzi razvoj upravljačkog prototipa (RCP)

- xPC target box
  - PII 266MHz, PIII 400MHz, PIII 700 MHz procesori malih snaga.
  - ~15W potrošnja energije.
  - Napajanje na ploči sa DC ulazom u intervalu od 8-28V (vanjski AC adapter).
  - Nema rotirajućih dijelova.
  - Temperaturno područje: 0-60°C, -40-75°C.
  - 128MB RAM, 32MB Flash modul.
  - Brzo startovanje BIOS-a: vrijeme učitavanja ~2s.
  - Raspoloživi I/O: A/D, D/A, DIO, brojači, enkoderi, CAN.



25/61



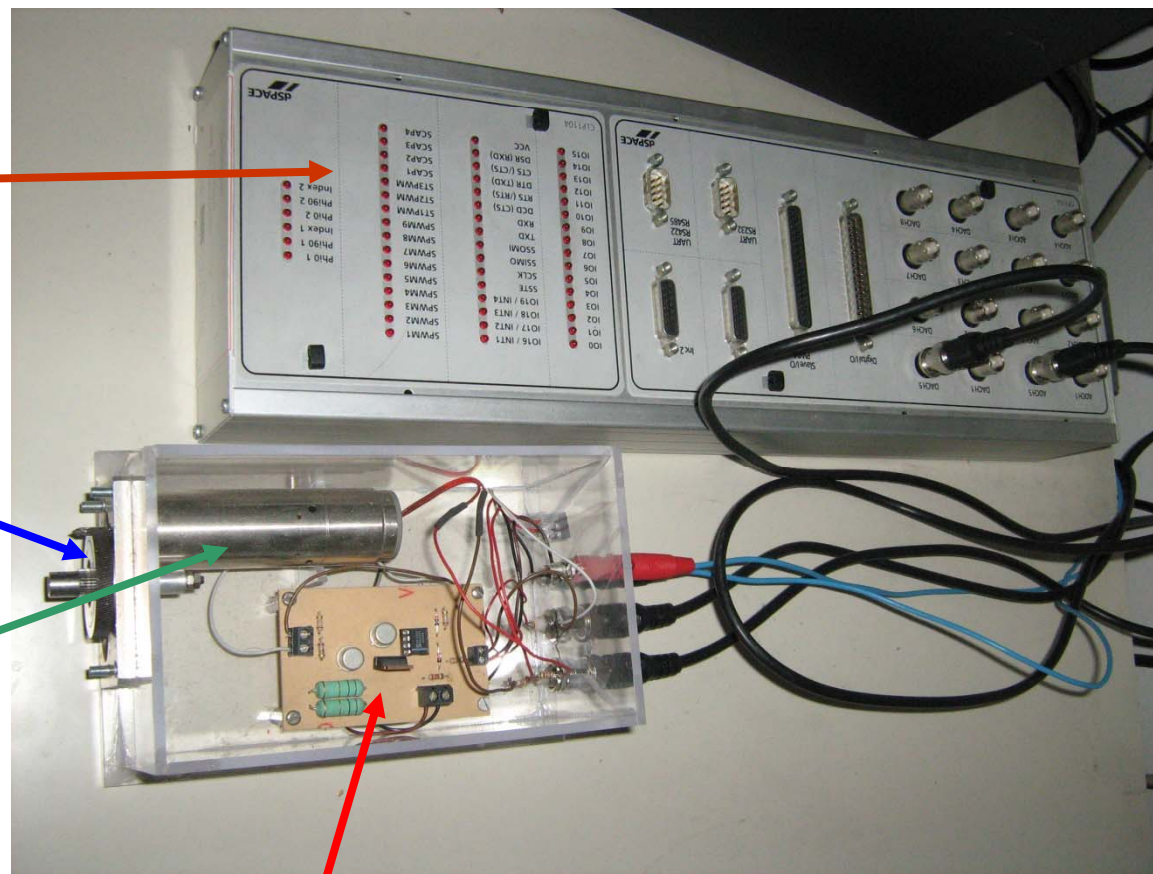
# Brzi razvoj upravljačkog prototipa (RCP)

- Primjer: dizajn procesorski temeljenog regulatora (dSPACE modul) za upravljanje DC motorom

Procesorsko  
sučelje  
(dSPACE 1104)

Senzor  
(tahogenerator)

Proces  
(DC motor)



Aktuator (pojačalo snage)

# Brzi razvoj upravljačkog prototipa (RCP)

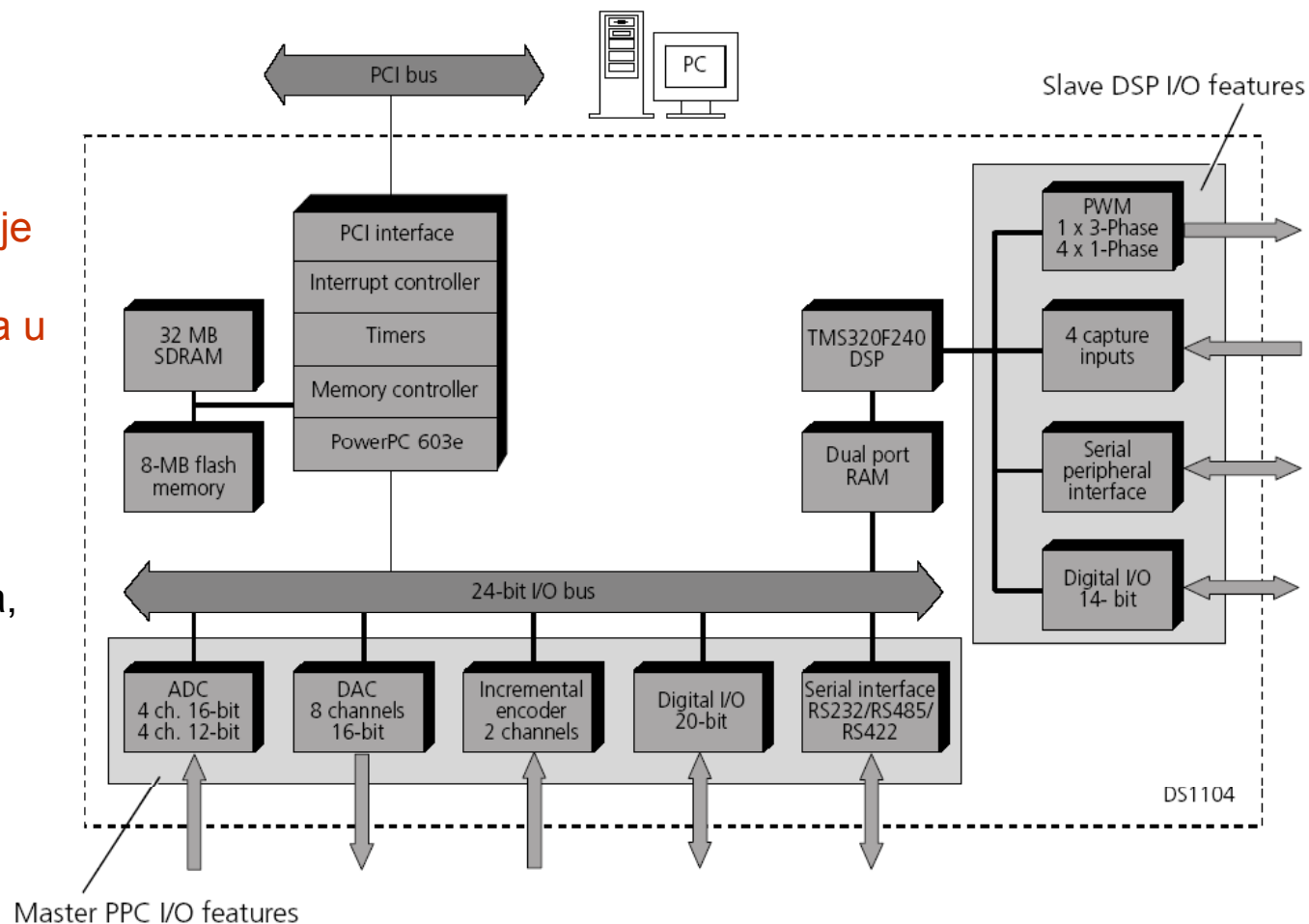
- Arhitektura DS 1104 (CLP 1104) upravljačke kartice
  - Nadograđuje PC za razvoj sistema za brzi razvoj prototipa.
  - Real-time hardver temeljen na PowerPC 603e (250 MHz) mikroprocesoru.



27/61

I/O sučelja čine je idealnom za razvoj kontrolera u raznim poljima industrije i edukacije.

32 MB DRAM-a,  
8 MB Flash-a.



# Brzi razvoj upravljačkog prototipa (RCP)

- Programska podrška
  - **ControlDesk program.**
  - **AutomationDesk program.**
- **ControlDesk** programski paket predstavlja softver za izvođenje eksperimenata, omogućuje izvođenje svih upravljačkih funkcija, praćenje i izvršavanje eksperimenata, što čini razvoj kontrolera mnogo efikasnijim.
- Najvažniji segmenti ovog paketa su:
  - **Experimental Manager.**
  - **Platform Manager.**
- Experiment Manager osigurava dosljedno rukovanje i kontrolu svime što je povezano sa eksperimentom nad relevantnim podacima.
- Platform Manager omogućuje registrovanje real-time ploča i konfiguraciju registrovanih platformi.

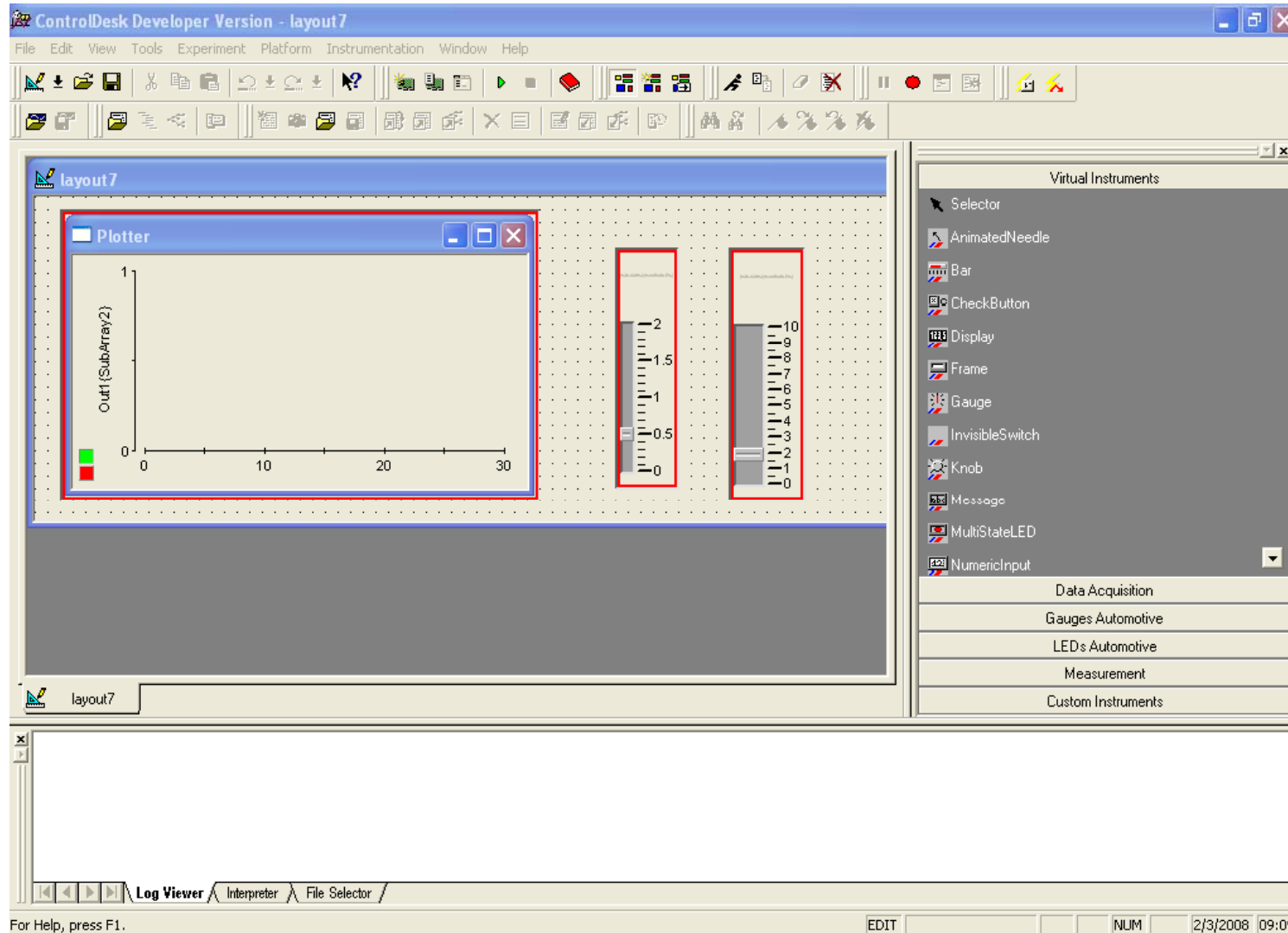


# Brzi razvoj upravljačkog prototipa (RCP)

## ControlDesk – izgled prozora



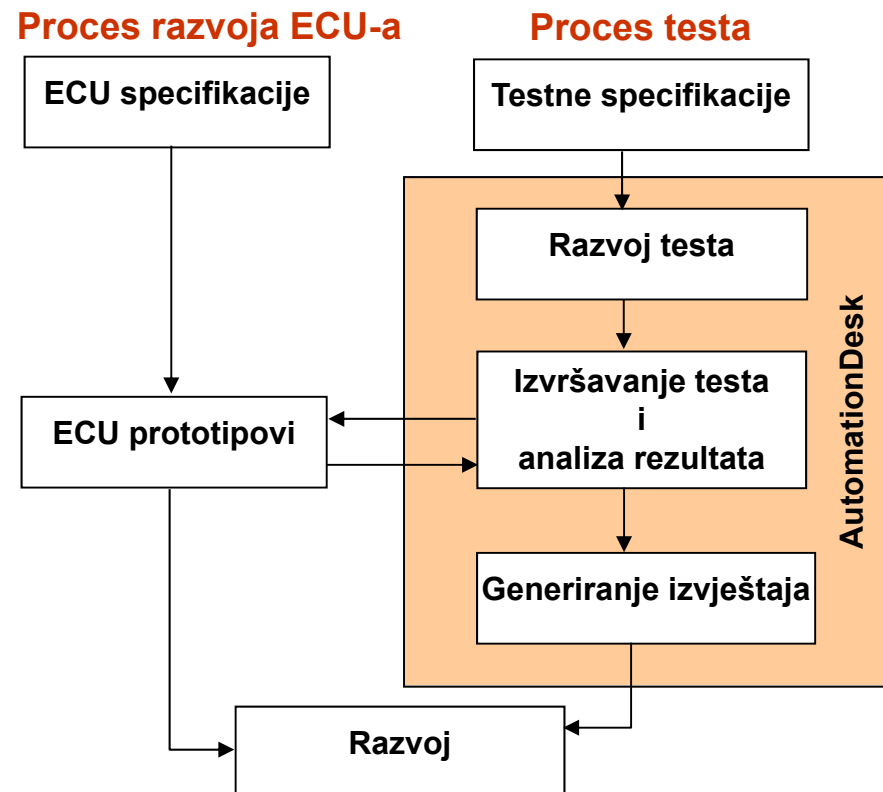
29/61



# Brzi razvoj upravljačkog prototipa (RCP)

## AutomationDesk program

- Univerzalni alat za kreiranje i rukovanje zadacima automatizacije – izvodi automatizirane testove unutar Matlab/Simulink programskog okruženja.
  - **Primjer:** testiranje novih elektronskih upravljačkih jedinica (ECU).



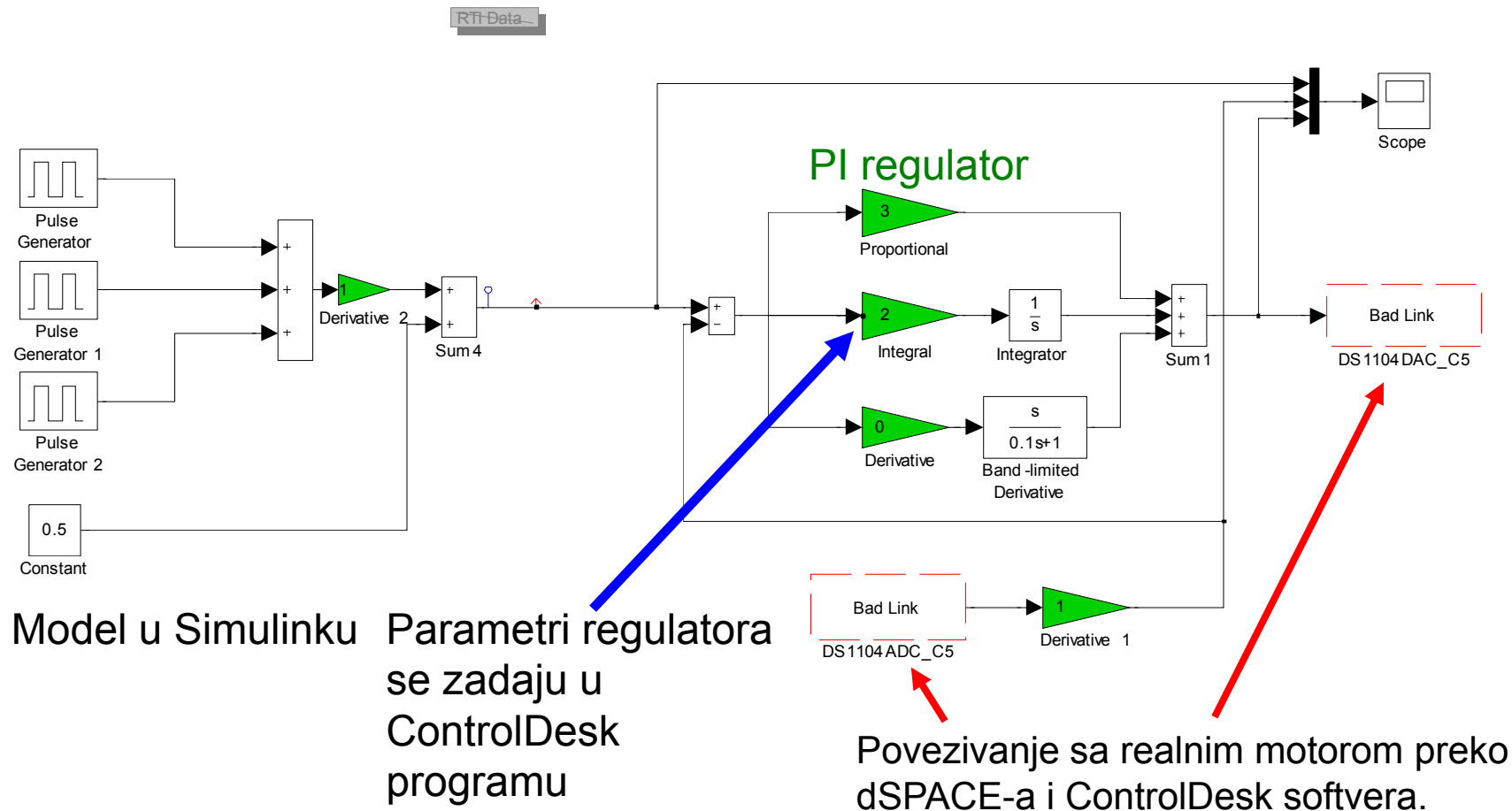
Procesi razvoja ECU-ova i testiranje se odvijaju paralelno.



# Brzi razvoj upravljačkog prototipa (RCP)

## Primjer: PI regulator za DC motor

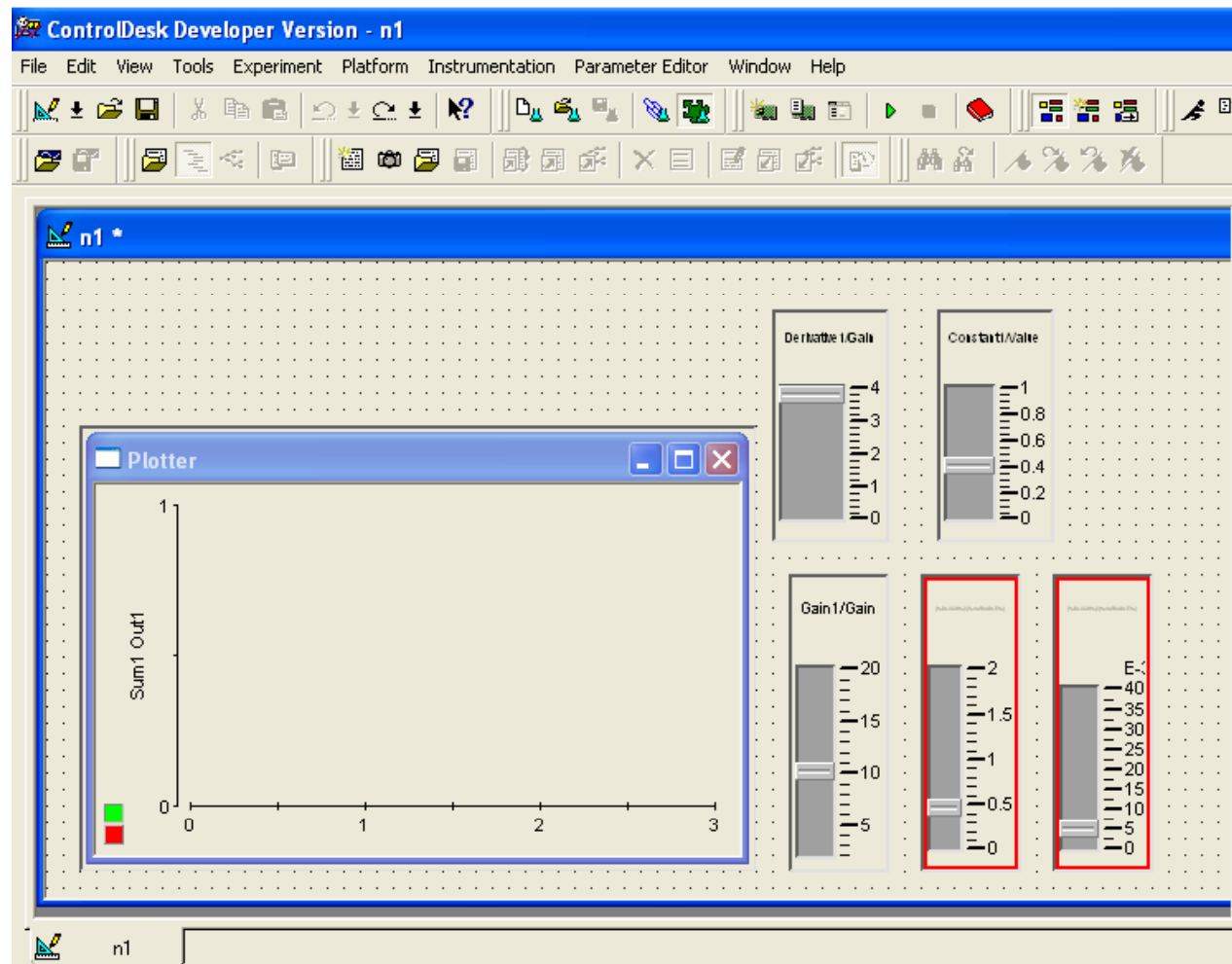
- Testiranje PI regulatora na realnom modelu DC motora korištenjem dSPACE CLP 1104 modula.



# Brzi razvoj upravljačkog prototipa (RCP)

## Primjer: PI regulator za DC motor

- Prozor za zadavanje vrijednosti parametara regulatora i grafički prikaz odziva u realnom vremenu

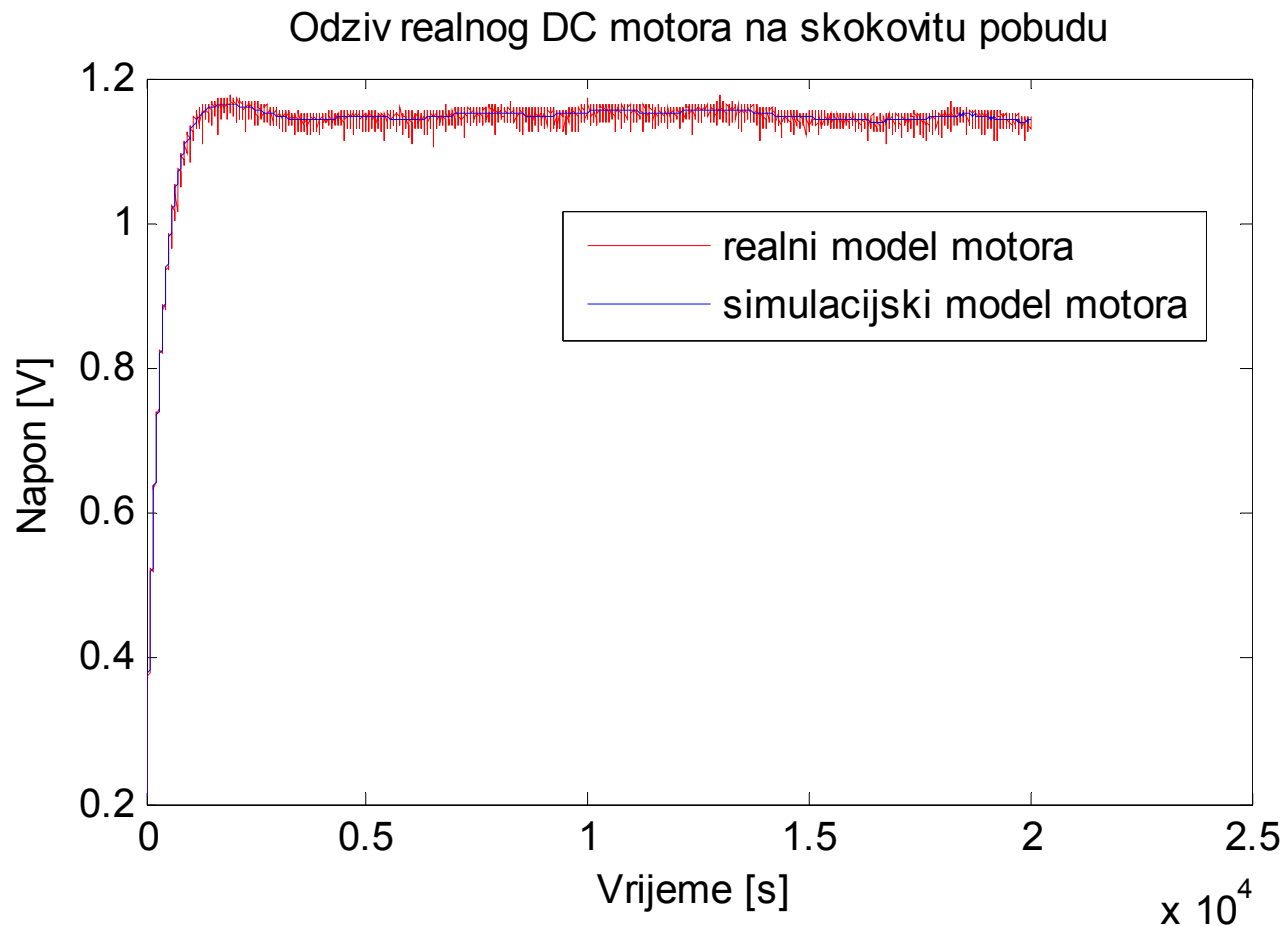




# Brzi razvoj upravljačkog prototipa (RCP)

## Primjer: PI regulator za DC motor

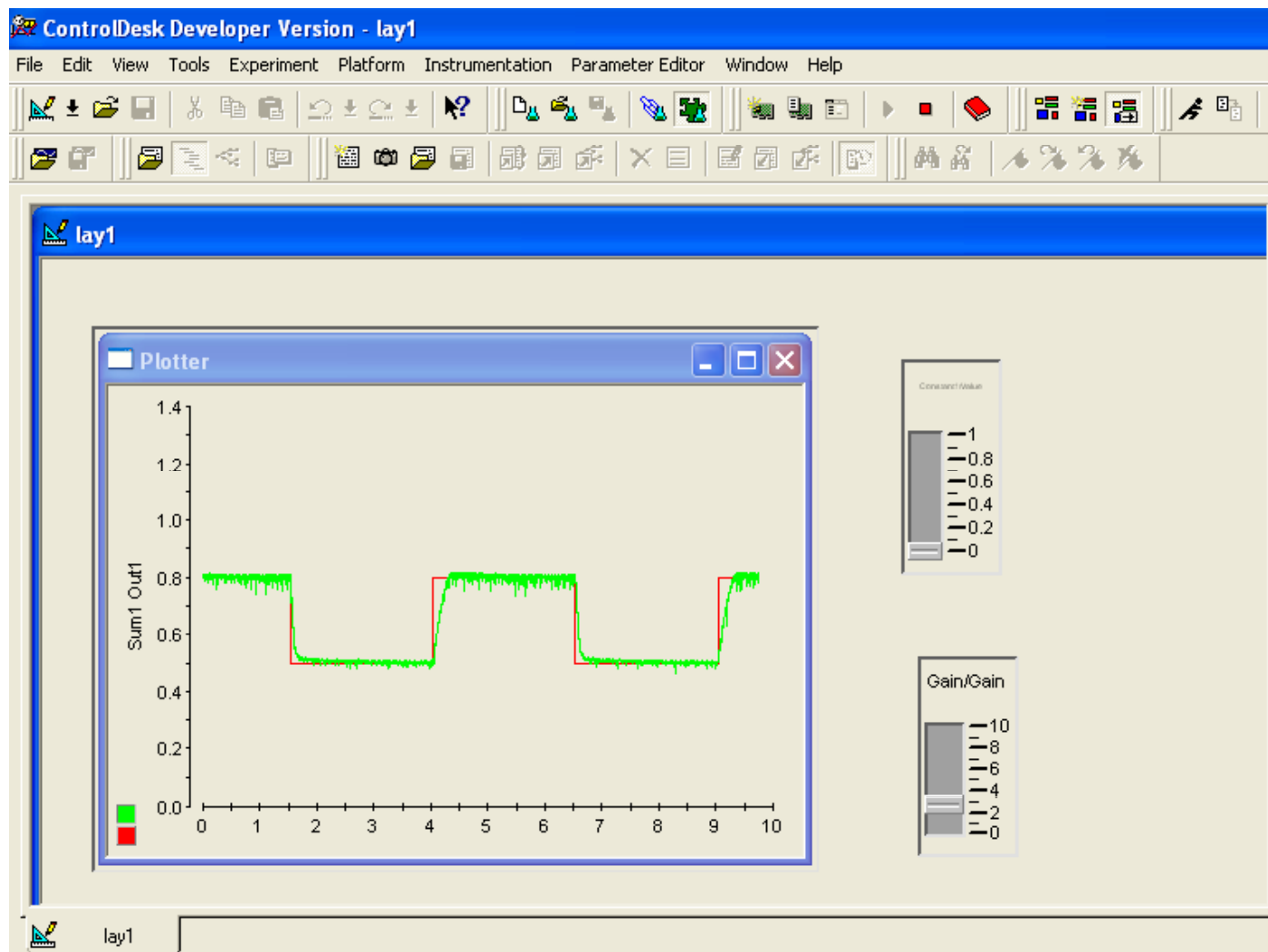
- Odziv na skokovitu pobudu (step signal)



# Brzi razvoj upravljačkog prototipa (RCP)

## Primjer: PI regulator za DC motor

- Odziv na niz četvrtki – ControlDesk prozor



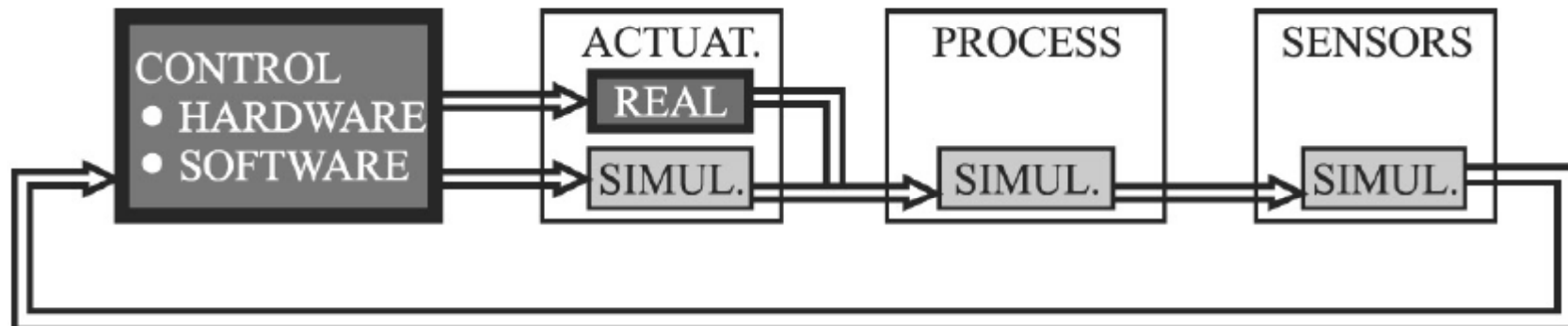
## 3.2. Hardver u simulacijskoj petlji (HIL)

- **Hardver kao dio simulacijske petlje.**
- Zahtijevaju se real-time performanse – fizički hardver neće čekati kašnjenje simulacije.
- HIL se široko koristi u razvoju sistema upravljanja:
  - Dizajn – brzi razvoj upravljačkog prototipa,
  - Test – testiranje u petlji.
- Omogućuje eksperimentiranje sa **fizičkim dijelovima** u **upravljački sintetiziranoj** sredini.
- Eksperimenti se mogu ponavljati i automatizirati.
- Omogućuje paralelni razvoj mehaničkih i upravljačkih sistema
  - Važna tehnika za smanjenje ciklusa dizajn uz istovremeno povećanje kvaliteta proizvoda.



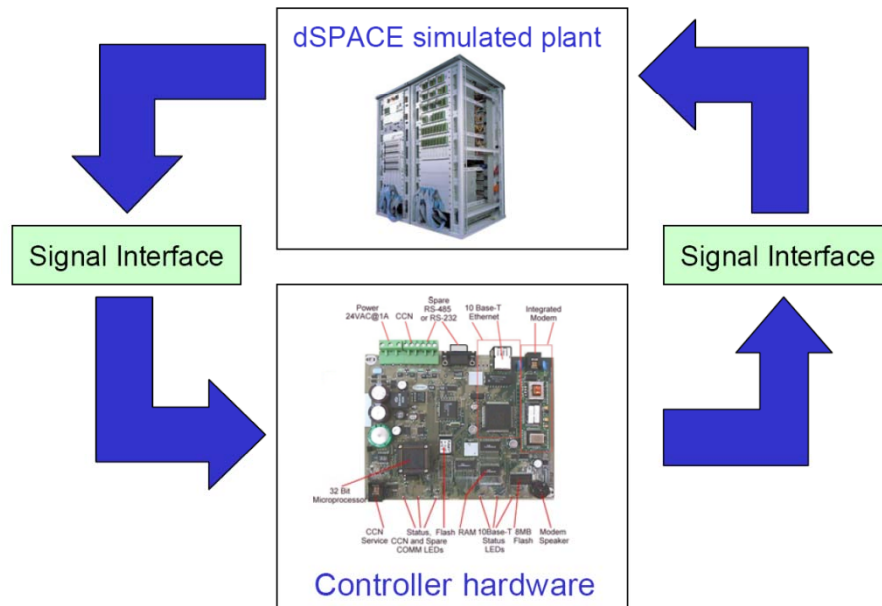
## Hardver u simulacijskoj petlji (HIL)

- HIL je karakteriziran radom realnih komponenti u konekciji sa real-time simuliranim komponentama.
- Upravljački hardver i njemu pridruženi softver su realne komponente, dok ostale komponente u sistemu mogu biti simulacijske, ili uključivati i realne komponente.



# Hardver u simulacijskoj petlji (HIL)

- HIL predstavlja moćan, fleksibilan alat za verifikaciju i validaciju performansi regulatora.
- Također, HIL predstavlja alat i procese za verifikaciju logičke i vremenske korektnosti integriranog upravljačkog sistema hardvera/softvera.
  - Upravljačke petlje se zatvaraju korištenjem real-time simuliranih procesa (pogona, objekata).
  - Sučelja se treniraju da osiguraju ispravnu integraciju sistema.
  - Testiranje se može automatizirati (dizajn za eksperimente).



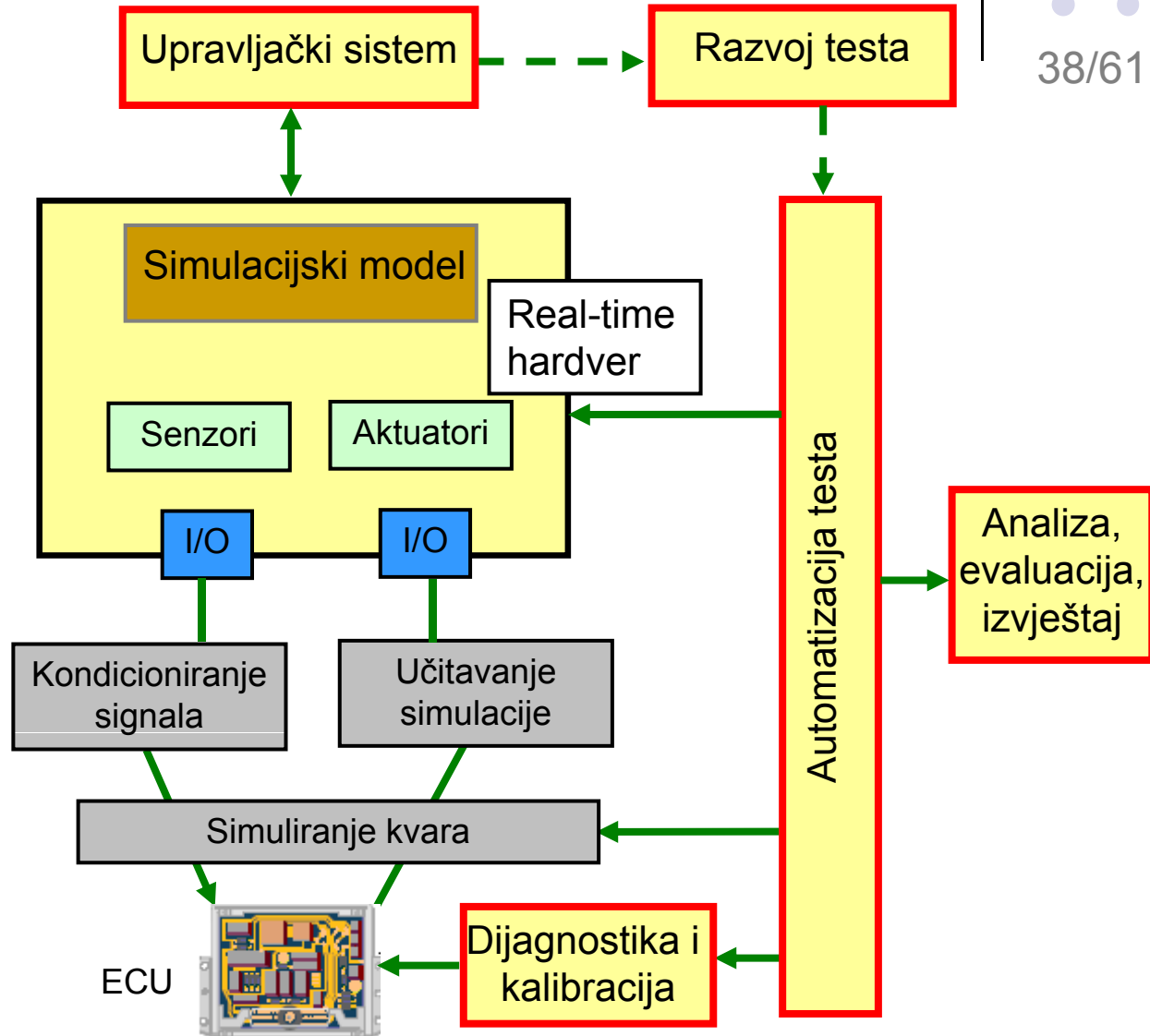
- HIL se također može koristiti za:
- automatizirane proizvodne testove,
  - debugiranje i testiranje usluga,
  - kalibraciju.



# Hardver u simulacijskoj petlji (HIL)

## HIL – Hardware-in-Loop

- Simulacija real-time procesa, fiksni korak solvera.
- Simuliranje senzora, npr. lambda senzor (nelinearni), senzor kotača (digitalni), senzor udara (visoka dinamika).
- Elektronička simulacija kvarova (dijagnostičko testiranje).



# Hardver u simulacijskoj petlji (HIL)

- HIL sistemi su razvijeni zadnjih godina, što je omogućeno
  - **Pojavom najnovijih CPU-ova i paralelnom obradom**
    - Osigurava računarsku moć za složenije i grupne modele.
  - **Razvojem novih hardverskih tehnologija**
    - Smanjenjem potrebe za razvojem vlastitog hardvera.
  - **Razvojem novih tehnologija korisničkog sučelja**
    - Jednostavnije korištenje.
- Povećanje korištenja HIL u automobilskom inženjeringu.
- Međutim, HIL se još ne koristi u svom punom kapacitetu, odnosno ne koriste se svi njegovi raspoloživi potencijali
  - Iako su HIL sistemi razvijeni daleko od očiju korisnika (one-off dizajni), njihova upotreba nije.



# Hardver u simulacijskoj petlji (HIL)

- **Prednosti HIL-a:**

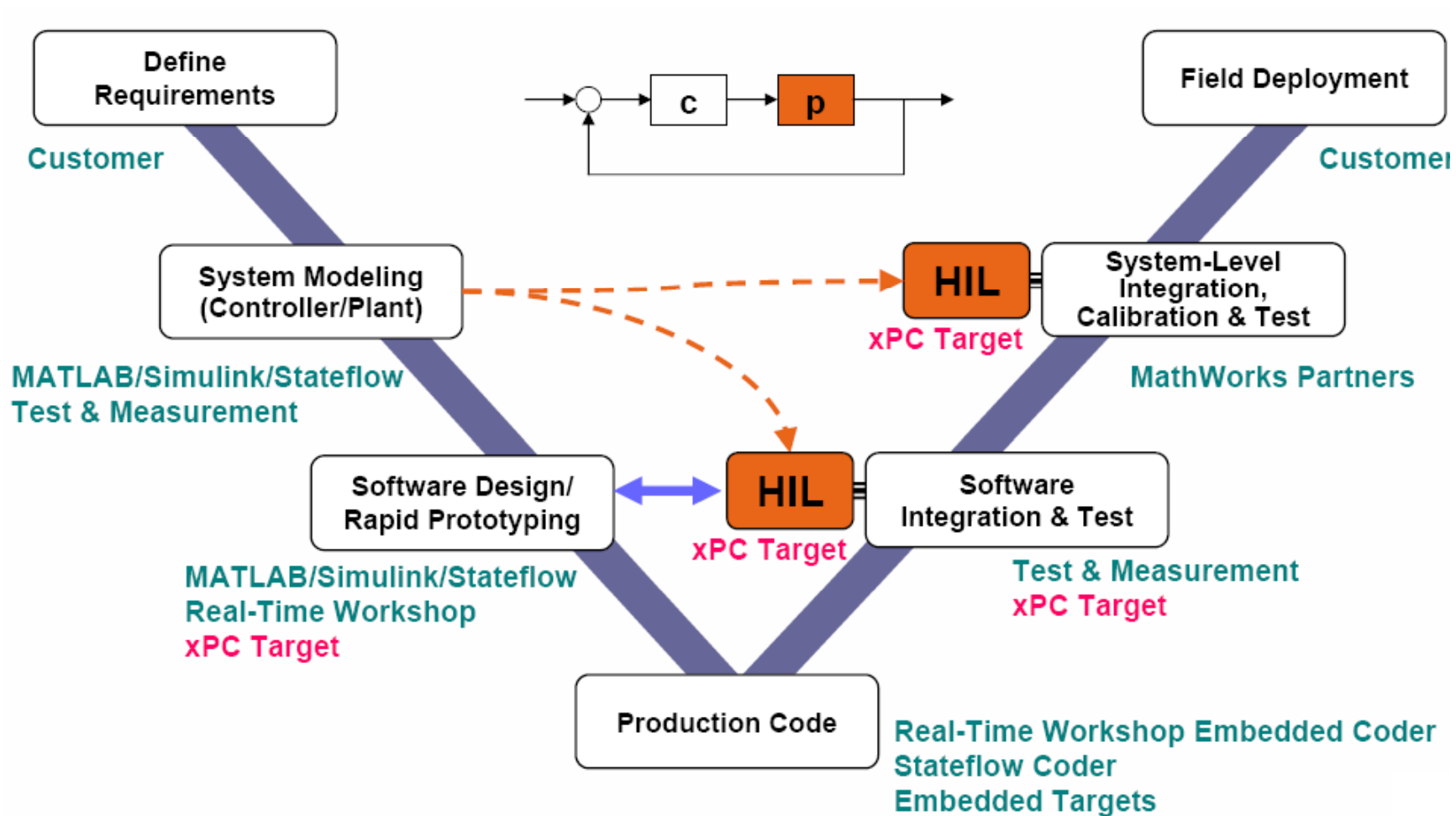
- Dizajniranje i testiranje upravljačkog hardvera i softvera bez operiranja realnog procesa (“premještanje polja procesa u laboratoriju”).
- Testiranje upravljačkog hardvera i softvera unutar ekstremnih uvjeta okoline u laboratoriji (npr. visoka/niska temperatura, veliko ubrzanje i mehanički udari, agresivni mediji, elektromagnetska kompatibilnost).
- Testiranje efekata pogrešaka i kvarova aktuatora, senzora i računara na ponašanje cjelokupnog sistema.
- Rukovanje i testiranje ekstremnih i opasnih radnih uvjeta.
- Ponovljivost eksperimenata.
- Jednostavno rukovanje s različitim MM (man-machine) sučeljima.
- Ušteda troškova i vremena razvoja.





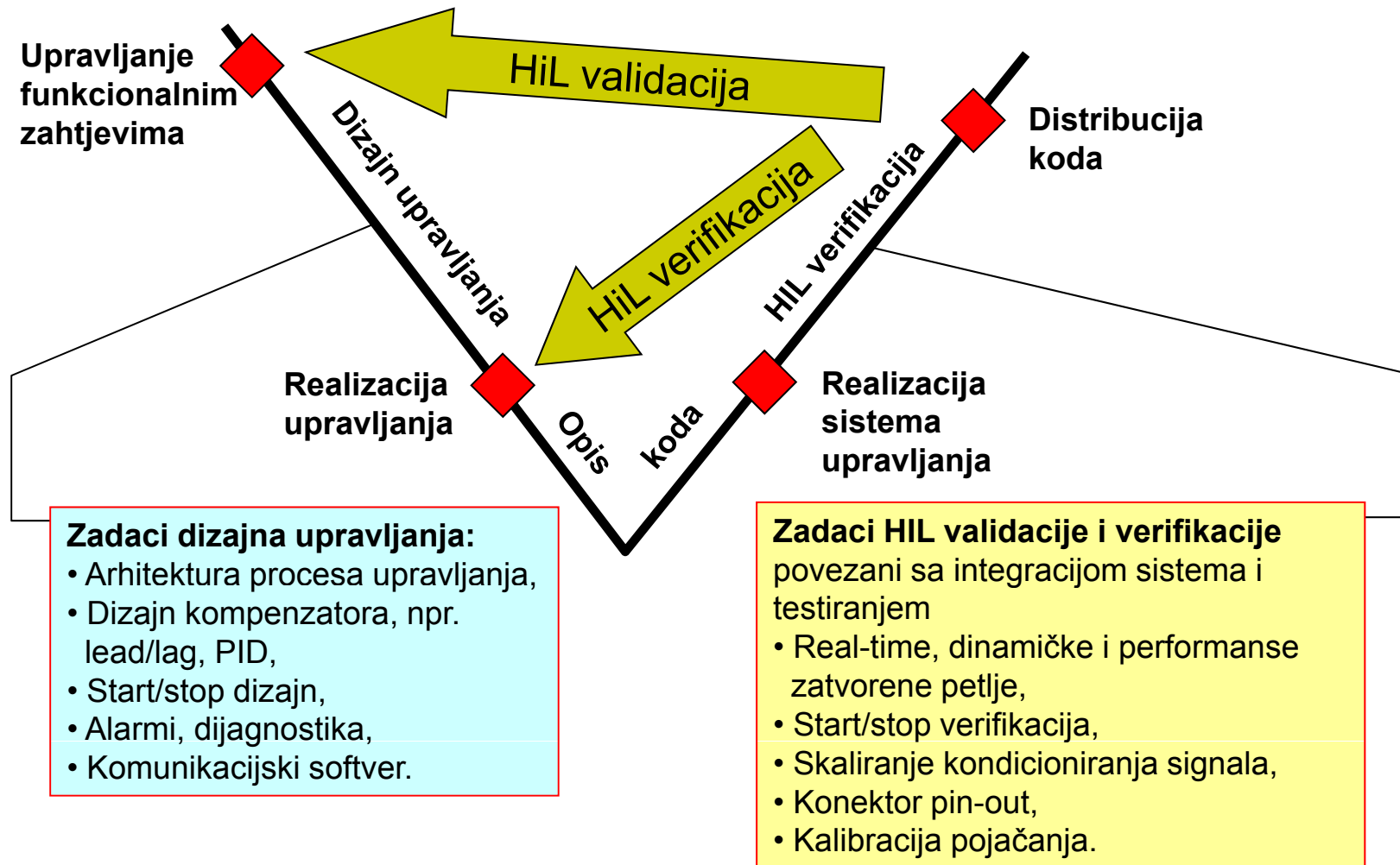
# Hardver u simulacijskoj petlji (HIL)

## V-dizajn HIL-a

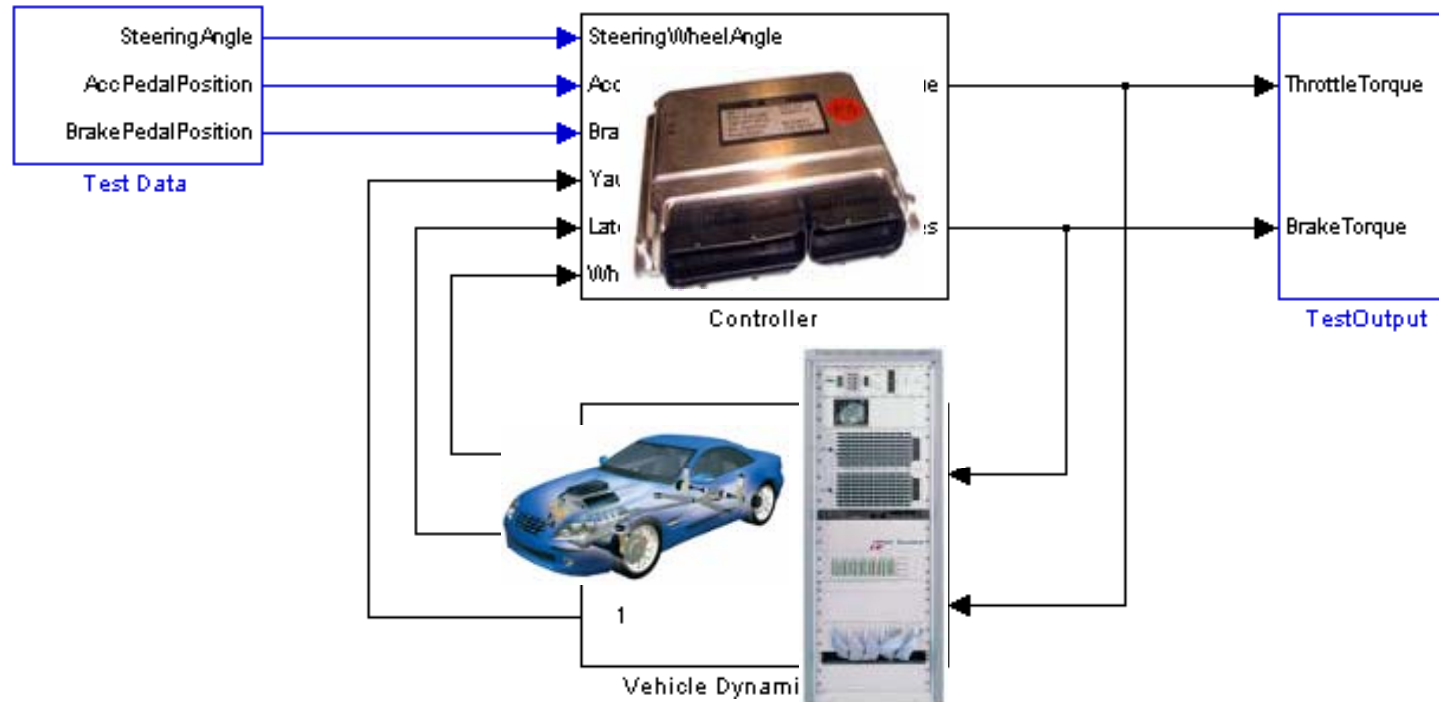


# Hardver u simulacijskoj petlji (HIL)

HIL je ključan u validaciji i verifikaciji dinamičkih performansi regulatora (kontrolera)



# Hardver u simulacijskoj petlji (HIL)

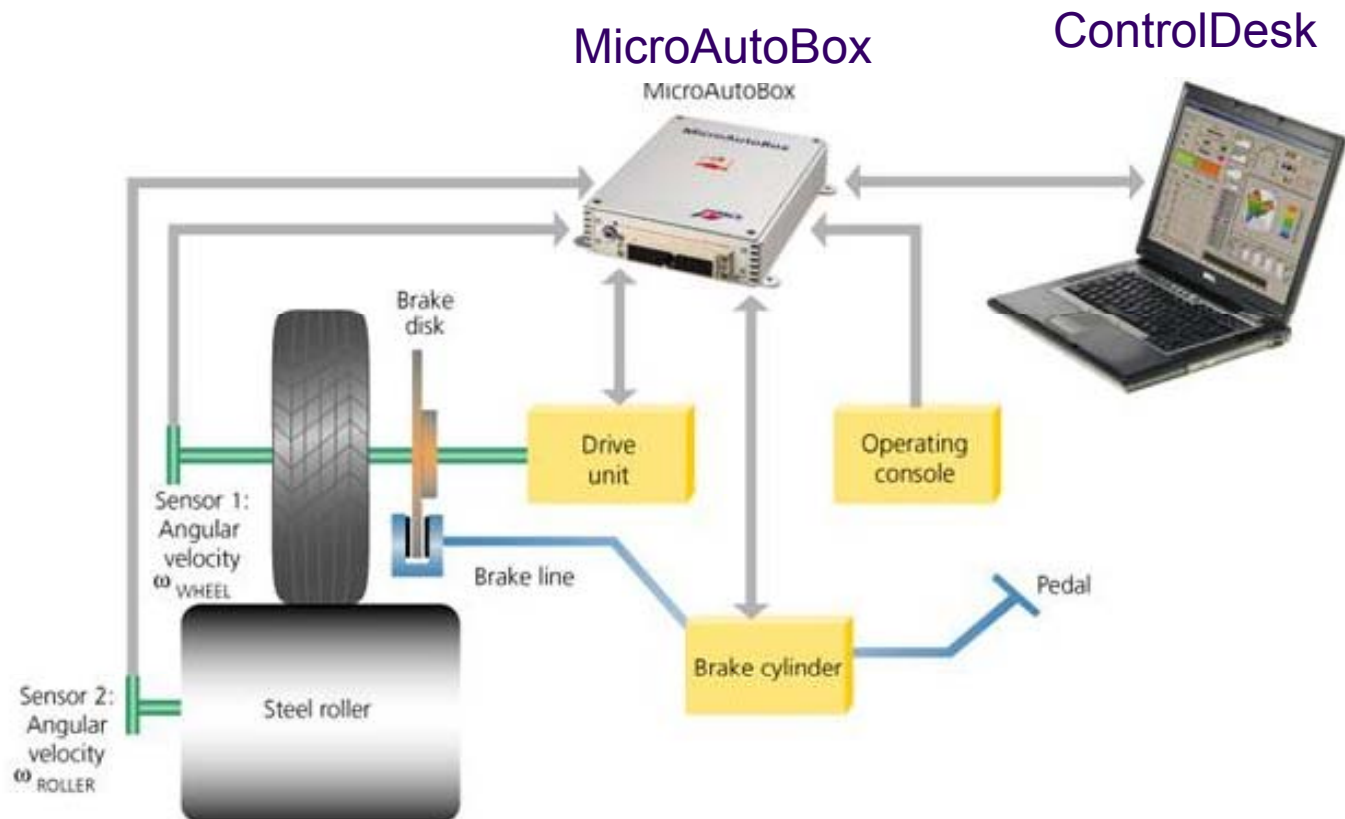


- Sistem se verificira u realističnoj (real-like) sredini.
- Testiranje hardvera i implementacija softvera.
- Testiranje u ranim stadijima bez fizičkih efekata prototipa (kvarovi, loši kontakti, starenje komponenti).
- Testiranje predviđenog ponašanja i mrežnih aspekata.
- Kritični testovi bez rizika.

# Hardver u simulacijskoj petlji (HIL)

## Primjer: ABS testna naprava – Univerzitet Graz

- Testna naprava za testiranje ABS i ASR (anti-slip) algoritama.
- Naprava upravljanja (kontrolirana) sa MicroAutoBox.
- Lagana implementacija kako konvencionalnih, tako i inovativnih ABS i ASR koncepata sa MATLAB®/Simulink®, Stateflow®, i TargetLink.



# Hardver u simulacijskoj petlji (HIL)

## Virtualno vozilo

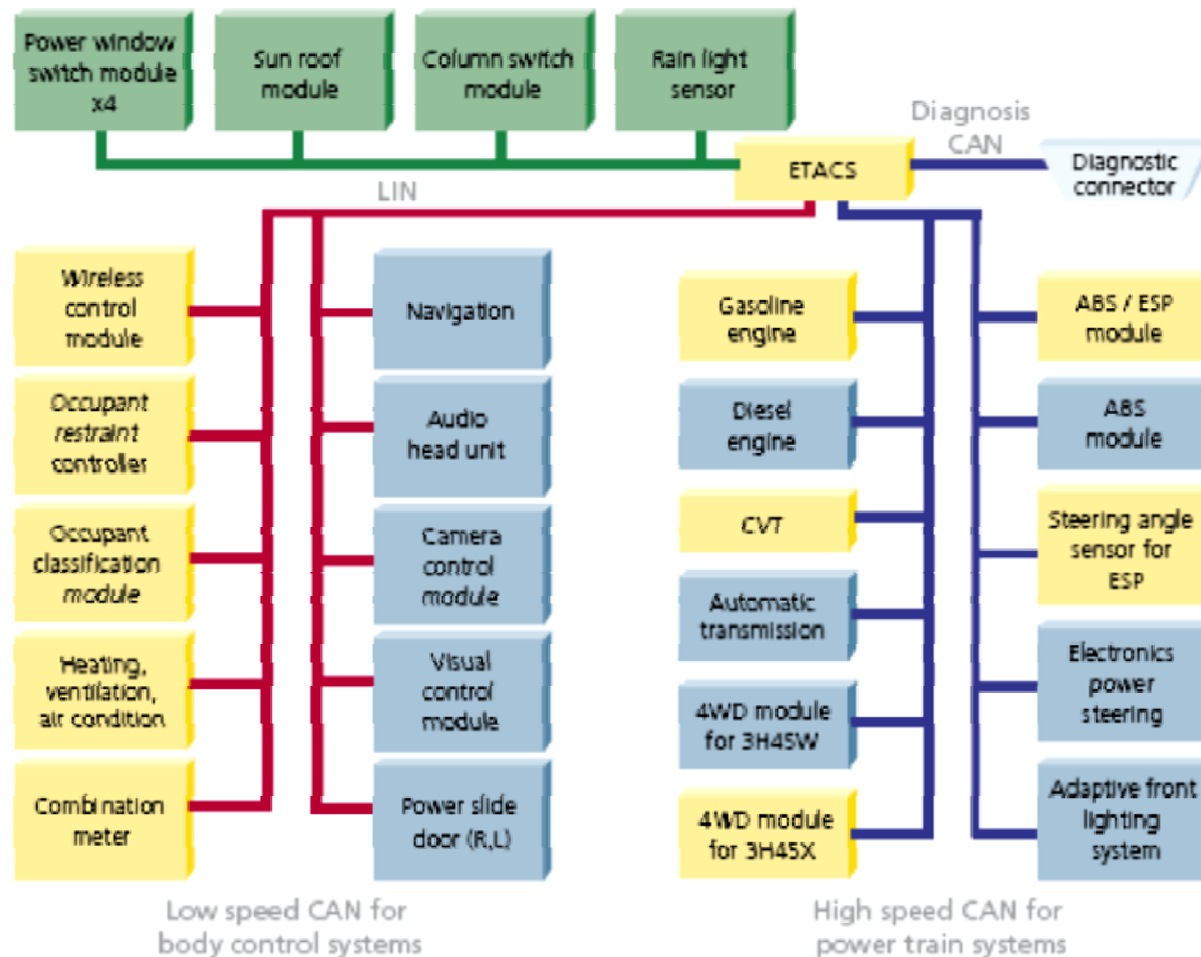
- Virtualno vozilo temeljeno na dSPACE mrežnim simulatorima omogućuje testiranje svih ECU funkcija, uključujući dijagnostičke funkcije, pouzdanost i sistematičnost sa pojedinačnim testnim sistemom.
- Prednosti upotrebe virtualnog vozila:
  - Jednostavani regresivni testovi.
  - Efikasni testovi sudara na ECU softveru.
  - Automatizirani doživotni testovi.
  - Efikasna analiza testa.
- Automobilski simulacijski modeli (ASM) u upotrebi (benzin, dizel, dinamika vozila).
- Posebno je važna fleksibilnost ASM simulacijskih modela, koje se jednostavno proširuju na modele različitih snadbjevača – to je veoma važno za nas.



# Hardver u simulacijskoj petlji (HIL)

## Virtualno vozilo – Mitsubishi Virtual Outlander

- Testiranje preko 20 ECU-ova sa HIL-om

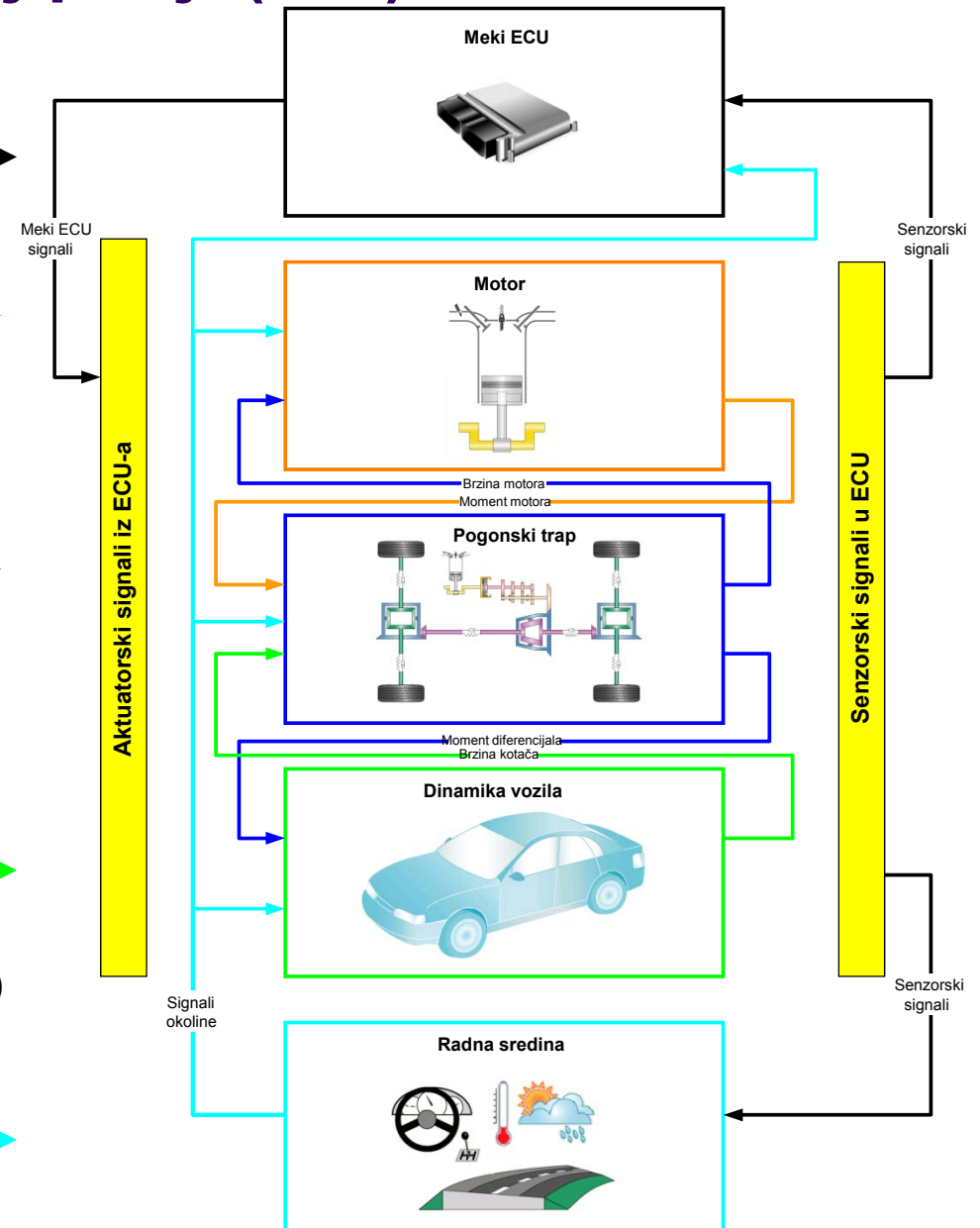


# Hardver u simulacijskoj petlji (HIL)



## ASM struktura modela

- **Meki ECU**
  - Automatski regulator transmisije
- **Model motora**
  - Osnovni motor
- **Model pogonskog trapa**
  - Pogon na zadnje, prednje ili na sve kotače,
  - Automatska i ručni prijenos (transmisija),
  - Elastične pogonske osovine.
- **Dinamika vozila**
  - Kretanje vozila,
  - Kotači,
  - Suspenzija,
  - Slijeđenje,
  - Aerodinamika,
  - Kočioni sistem (ASMBrakeHydraulics)
- **Okolina**
  - Cesta,
  - Vozač,
  - Manevar.

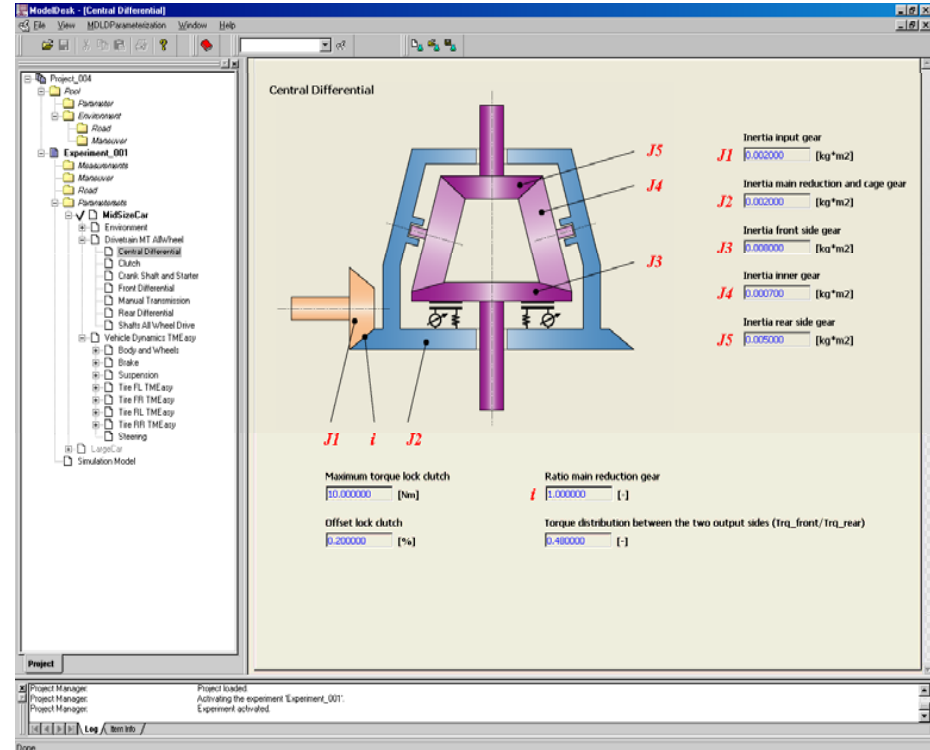
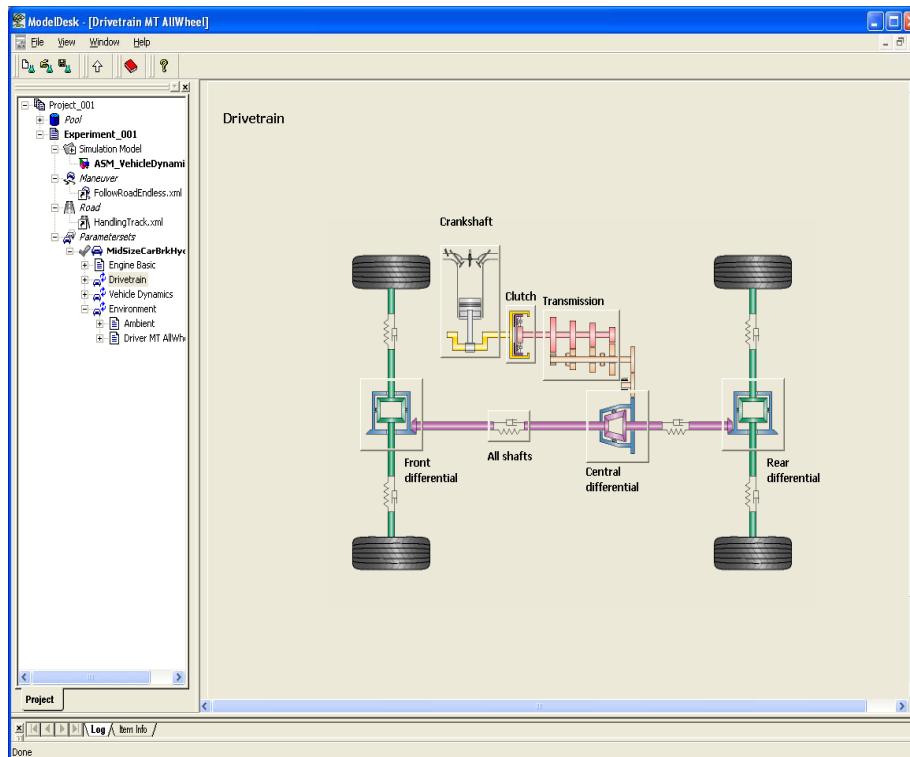




# Hardver u simulacijskoj petlji (HIL)

## Model Desk program za ASM

- Grafičko korisničko sučelje (GUI) za parametrizaciju dSPACE ASM dinamike vozila.
- Definicija i izbor ceste i manevara.
- Konfiguracija komponenti modela vozila (npr. prednji ili svi kotači)





# Hardver u simulacijskoj petlji (HIL)

## Rezultati HIL ASM testa za Audi A5



49/61

Uprkos povećanju složenosti softvera i elektronike u periodu 2001-2006:

- Broj nezgoda  **smanjen za 61%**.
- Troškovi garancije:  **smanjeni za 55%**.
- Vozilo se ne pokreće:  **smanjeni za 75%**.

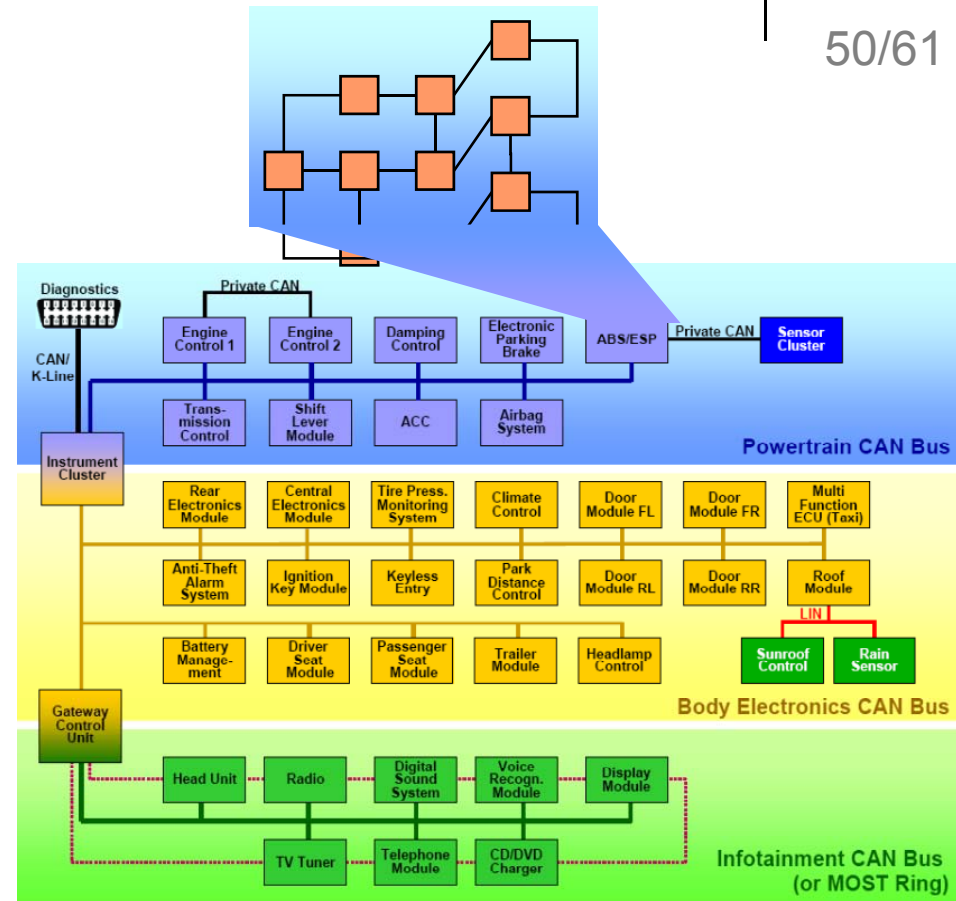


Zadnja serija Audi A5: softverski test urađen 17 sedmica prije puštanja u pogon.

# Hardver u simulacijskoj petlji (HIL)

## Složenost elektroničkih sistema

- **Složenost arhitekture ECU-a**
  - Ukupno 40 do 80 ECU-ova.
  - Nekoliko ECU podсистema.
  - ECU-ovi imaju potrebu za razmjenom podataka.
- **Složenost umrežavanja**
  - 100s do 1000s signala.
  - Različiti mrežni protokoli.
  - Dijagnostički i mrežni menadžment na sabirnici.
- **Složenost pojedinačnog ECU-a**
  - 50 do 500 softverskih modula.
  - Aplikacijski softver i osnovni softver platforme.



# Hardver u simulacijskoj petlji (HIL)

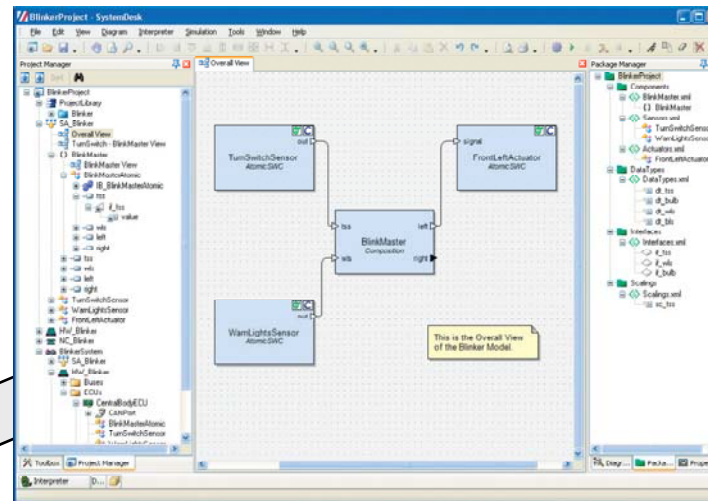
## Novi alati – System Desk

- SystemDesk proširuje postojeće alate iz dSPACE-a
  - Dizajn pojedinačnih funkcija ⇒ Dizajn elektroničkih sistema vozila
  - Generiranje koda pojedinačne funkcije ⇒ Integracija kompletnog koda ECU-a
  - Kasnije HIL testiranje ⇒ Ranija analiza sistema
  - Raznovrsne direktive modeliranja ⇒ Podrška standardima tipa AUTOSAR



51/61

Na modelu zasnovan dizajn za upravljačke funkcije



SystemDesk

Na modelu zasnovan dizajn na sistemskoj razini

# Hardver u simulacijskoj petlji (HIL)

- Faktori koji ograničavaju učinkovitu upotrebu HIL-a:
  - Zahtjevi za tačnošću modela postrojenja (procesa),
  - Intergracija alata,
  - Integracija metoda dizajna,
  - Integracija procesa.
- Rješenja se pojavljuju u off-line simulaciji/CAE svijetu
  - Ovo je motivacija za studije izvodivosti sa inženjerskim softverom korištenjem iSIGHT-a.
  - Ostaje da se vidi eksperimentalna verzija.



### 3.3. HIL (PIL) kosimulacija

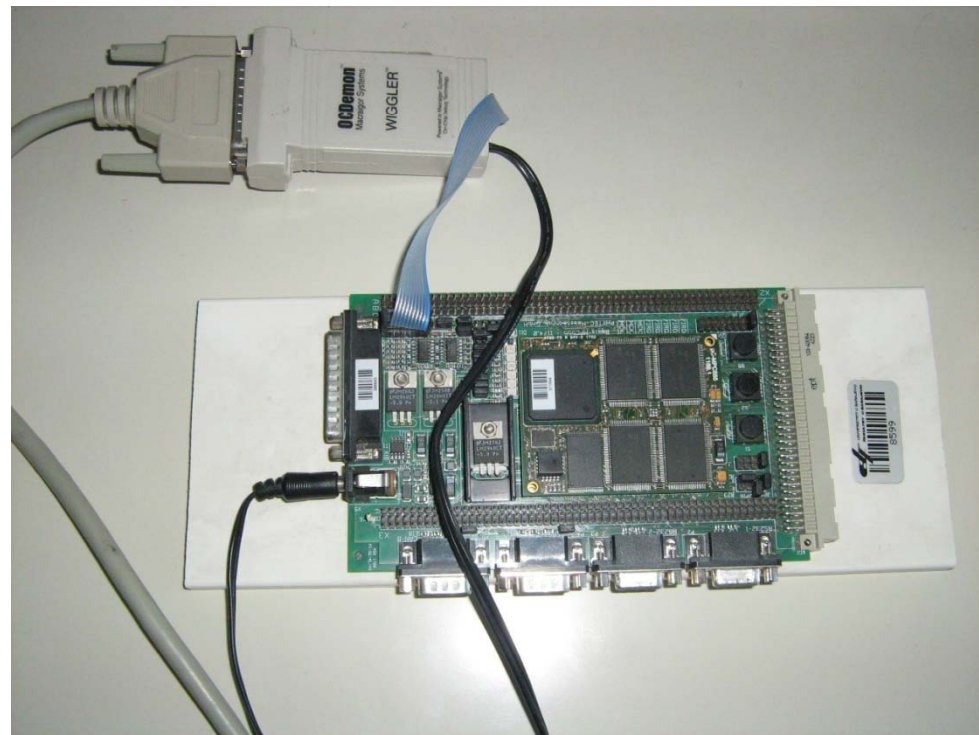
- Procesor u petlji (Processor in the loop, PIL) **kosimulacija** je tehnika koja pomaže u evaluaciji kvalitete upravljačkog sistema kojim se želi operirati preko ciljnog procesora (target processor).
- Kompanija Freescale je razvila MPC5xx ciljne procesore koji predstavljaju proširenu verziju ugradive (embedded) real-time ciljne konfiguracije.
- PIL kosimulacija je izuzetno korisna za simuliranje, testiranje i validaciju upravljačkih algoritama u sistemu koji sadrži regulator (kontroler) i objekt upravljanja (proces, postrojenje).
- U klasičnoj simulaciji u zatvorenoj petlji ovaj sistem je predstavljen sa dva podsistema u Simulink ili State-flow



# PIL kosimulacija

## Motorola MPC 555 razvojni sistem + WIGGLER sučelje

- Wiggler sučelje omogućuje učitavanje koda sa računara, preko paralelnog porta, u RAM ili FLASH memorije MPC 555 kartice.
- Aplikacija iz Simulinka se pomoću RealTime Workshop-a prevodi u C kod, a dalje preko Code Warrior programa (za Motorola procesore Power PC) učitava u memoriju kartice.
- Nakon učitavanja programa u karticu, komunikacija MPC 555 kartice sa okolinom se ostvaruje preko serijskog porta.



Wiggler sučelje služi samo za učitavanje koda u memoriju kartice.

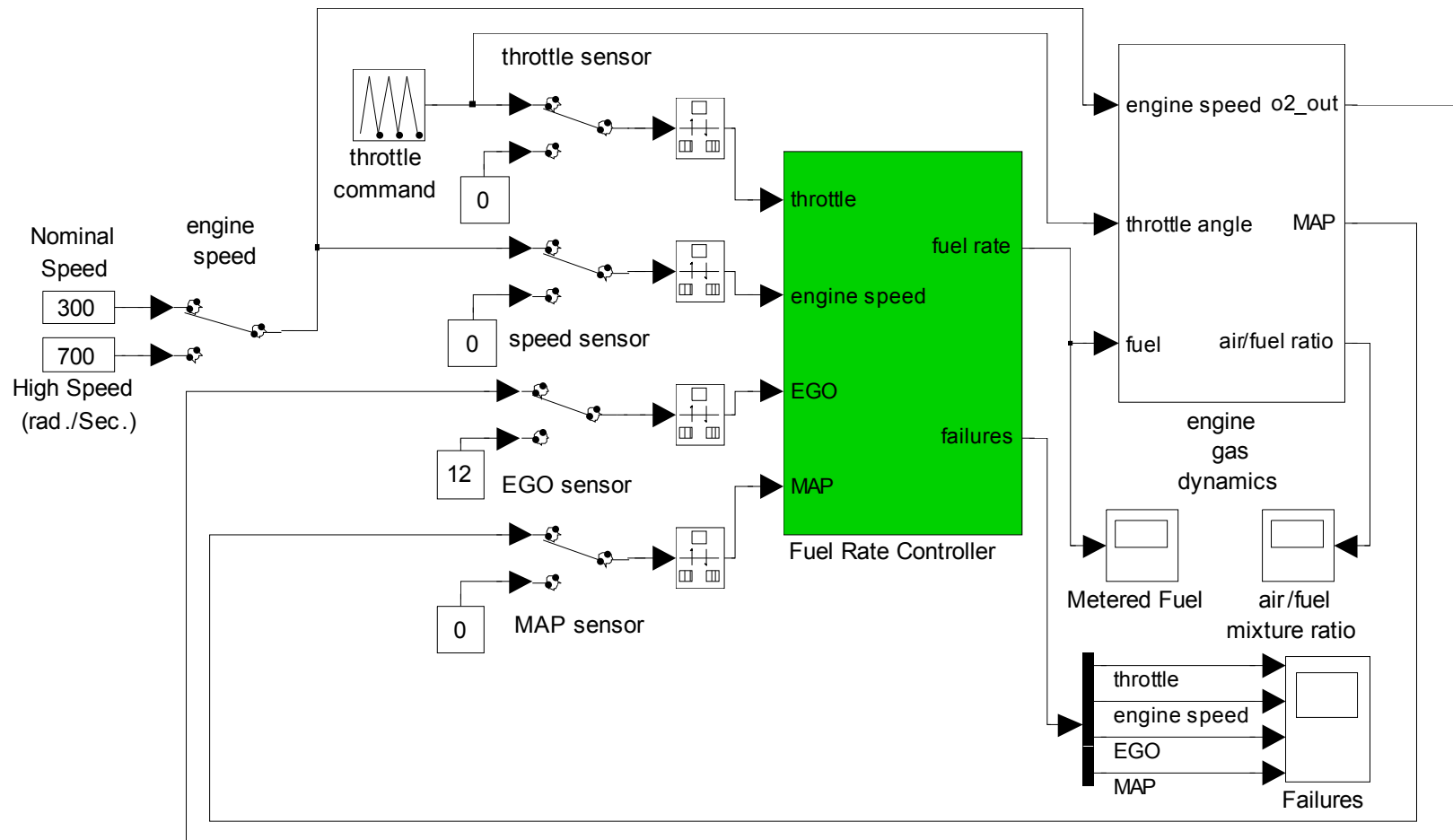


# Primjer PIL kosimulacije

Na kvarove tolerantan sistem upravljanja ubrizgavanja goriva u automobilu



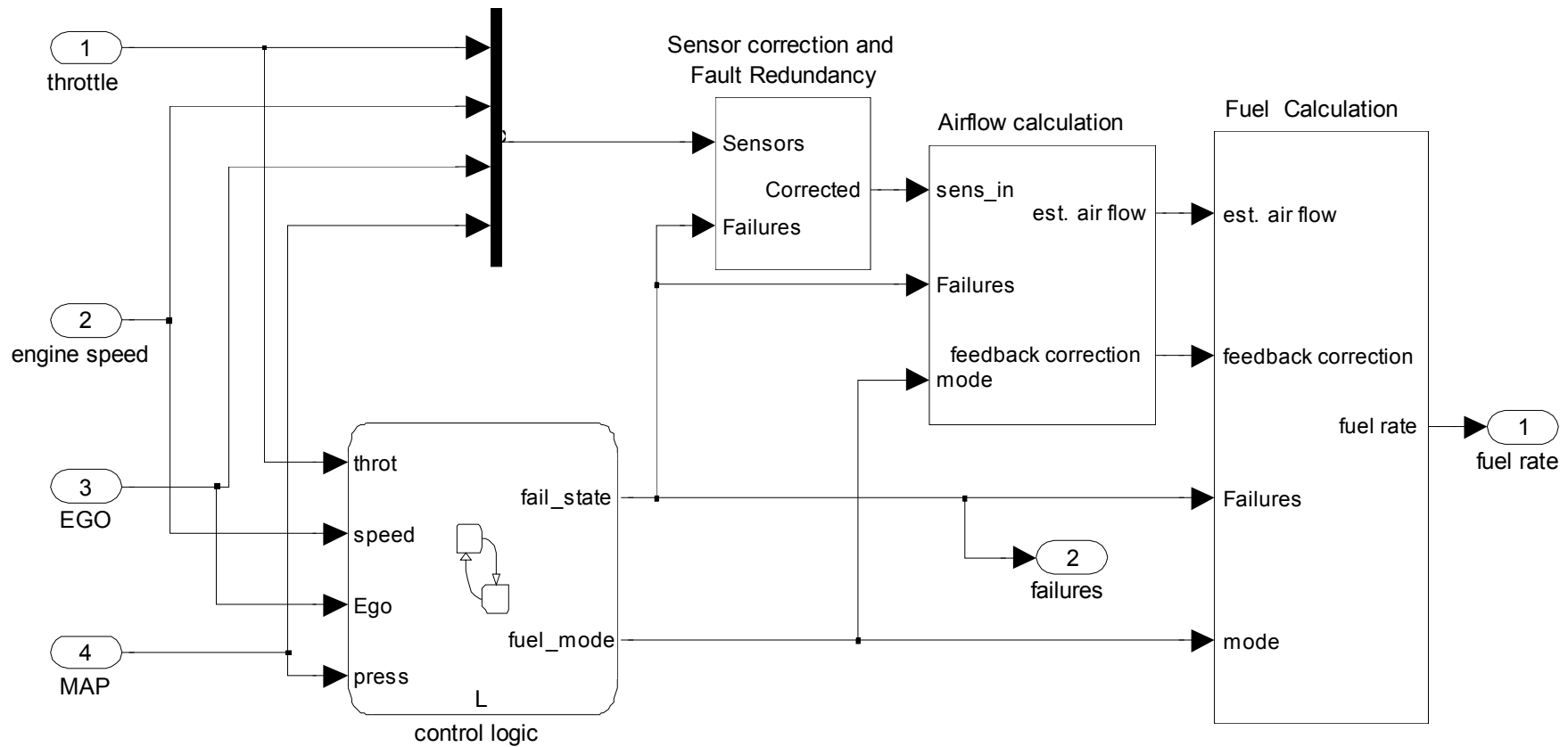
Fault-Tolerant Fuel Control System





# Primjer PIL kosimulacije

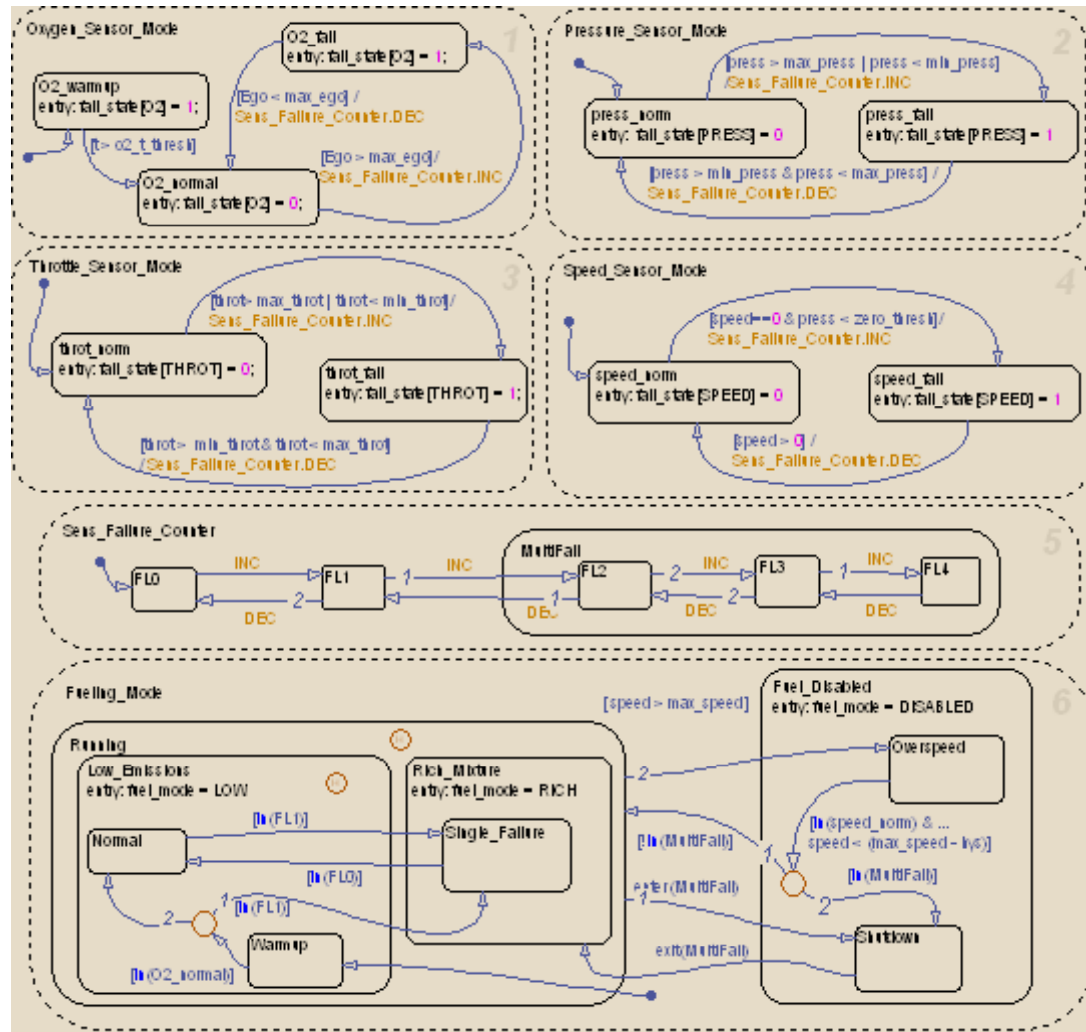
## Regulator ubrizgavanja goriva u Simulinku





# Primjer PIL kosimulacije

## Stateflow upravljačka logika regulatora

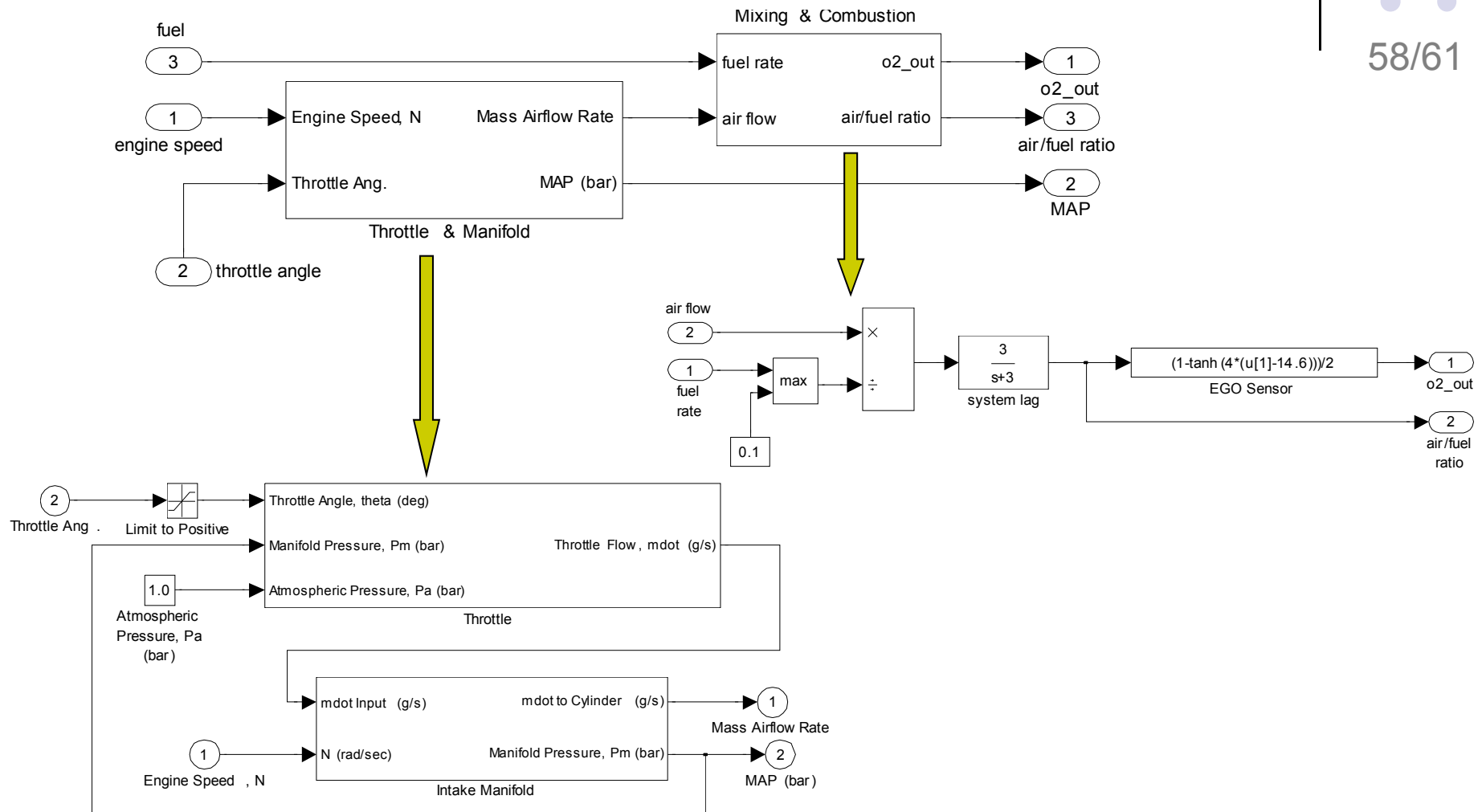


# Primjer PIL kosimulacije

## Proces – sistem ubrizgavanja goriva i zraka



58/61



**Air-Fuel Intake Dynamics**

# Primjer PIL kosimulacije

## Proces – sistem ubrizgavanja goriva i zraka



**Fault-Tolerant Fuel Control System**

Build code for Subsystem: fuel rate controller

Pick tunable parameters

Variable Name	Class	StorageClass
A1	double	Inlined
A2	double	Inlined
B1	double	Inlined
B2	double	Inlined
Ki	double	Inlined

Blocks using selected variable

Block	Parent

Build Cancel Help

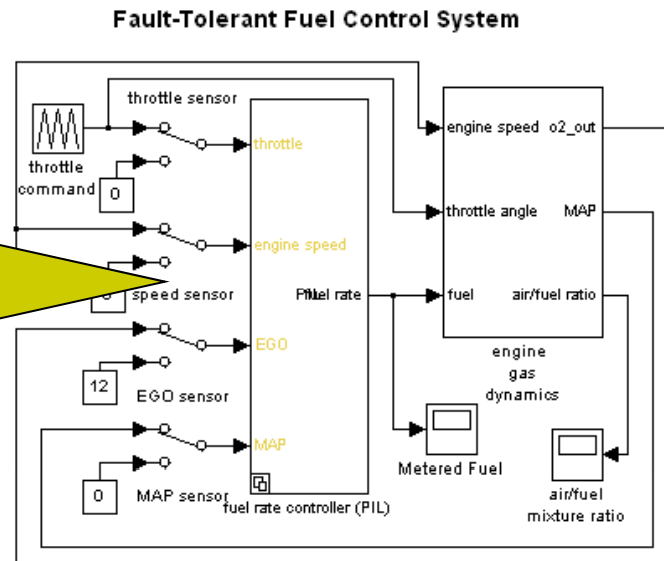
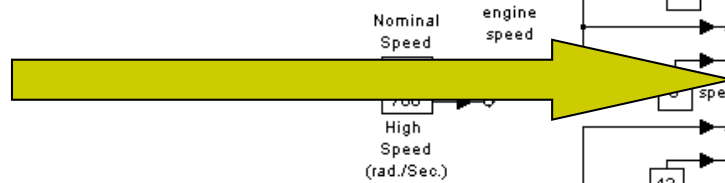
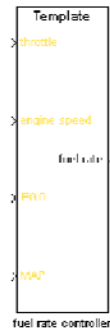
Status  
Select tunable parameters and click Build

# Primjer PIL kosimulacije

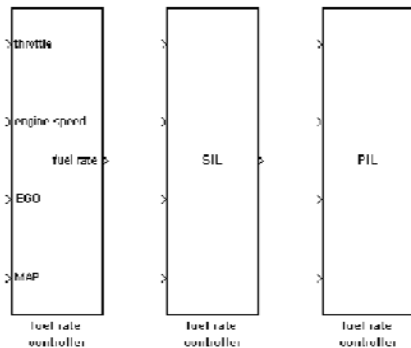
## Proces – sistem ubrizgavanja goriva i zraka



\*\*\* Simulink library generated for MPC555PIL \*\*\*  
 RealTime Workshop 5.0  
 Embedded Target for Motorola MPC555 1.0.1  
 Generated on: 2008-02-09 18:17:29  
 Original subsystem:  
 mp0665pil\_fuelrate\_controller

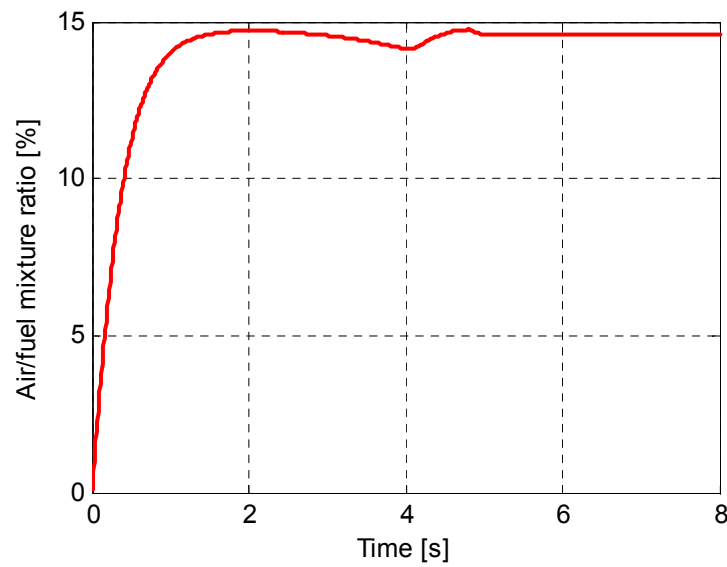
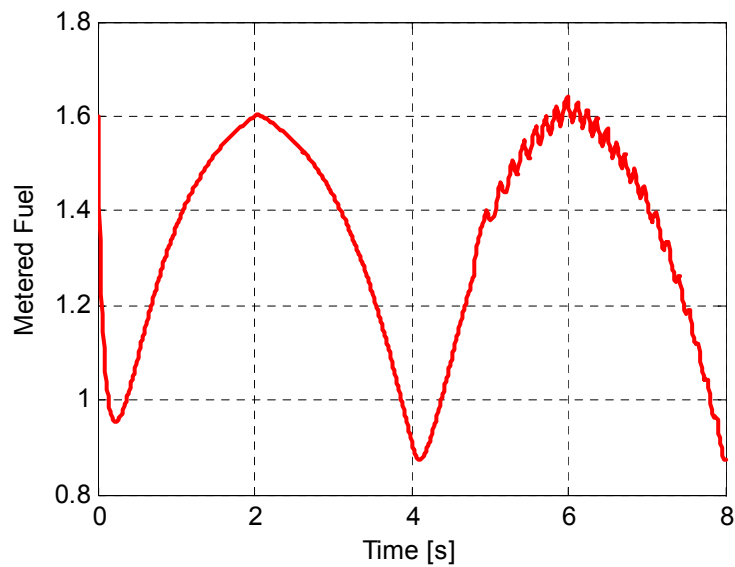
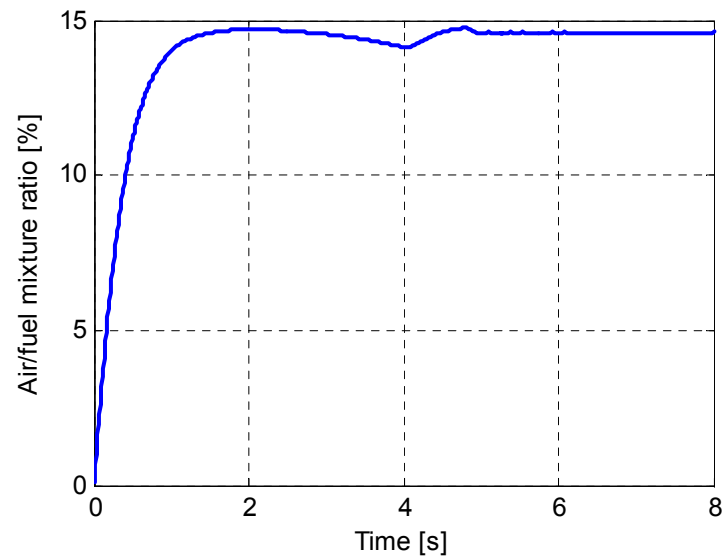
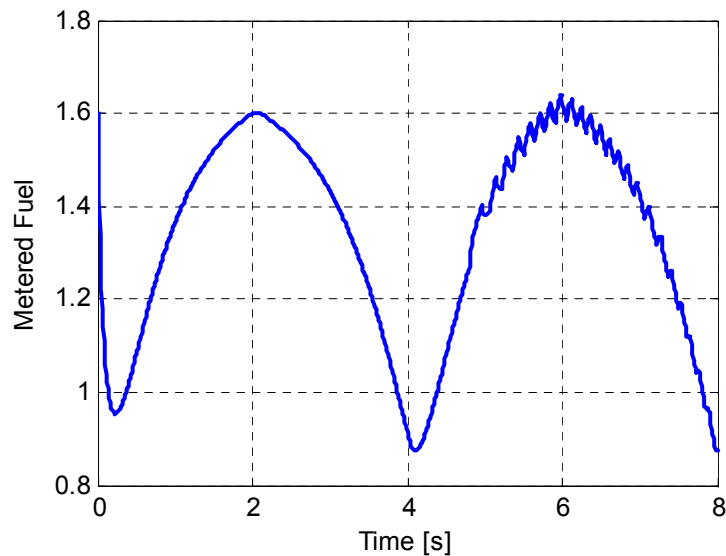
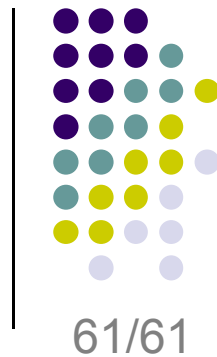


- Double click for help with modifying the original model.
- Replace the original subsystem in the model with the configurable subsystem from this library.
- Replace the configurable subsystem in the model with the copy of the original subsystem in this library.



# Primjer PIL kosimulacije

## Rezultati PIL i SIL kosimulacija



PIL

SIL