

## Lekcija 2: *Dizajn mehatroničkih sistema*

Doc.dr.sc. Jasmin Velagić  
Elektrotehnički fakultet Sarajevo

Kolegij: Mehatronika

2007/2008

### 2. Dizajn mehatroničkih sistema

Šta je dobar dizajn?

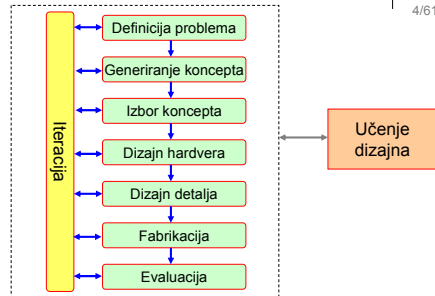


### Dizajn mehatroničkih sistema

- Dizajn je danas dominantno uvjetovan:
  - brzim (rapidnim) promjenama u tehnologiji,
  - svjetskom konkurencijom.
- Stoga dobri inženjeri trebaju:
  - doživotno učenje,
  - iskustvo sa multidisciplinarnim projektima,
  - korištenje vrhunskog dizajna i vještina vođenja projekta koje su koristile vodeće svjetske kompanije.
- Dobri dizajneri uvijek koriste dokazane procese dizajna
  - sa opravdanjem u pogledu izbora,
  - sa prikladnim preporukama u odnosu na izvorni materijal.

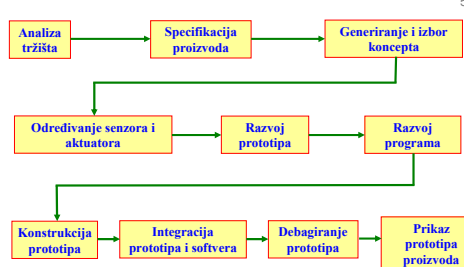
### Dizajn mehatroničkih sistema

- Proces dizajna je jednako važan kao što su ideje i analiza.



### Dizajn mehatroničkih sistema

- Ekonomija i inženjering dizajna



### 2.1. Proces dizajna mehatroničkih sistema

#### Aktivnosti u ciklusu dizajna mehatroničkih sistema

- Prepoznavanje potreba.
- Konceptualni dizajn.
- Matematičko modeliranje.
- Izbor senzora i aktuatora.
- Detaljno matematičko modeliranje.
- Dizajn sistema upravljanja.
- Optimizacija dizajna.
- Hardverski prototip i simulacija.
- Razvoj ugradivog softvera.
- Ciklus cijeloživotne optimizacije.



## Proces dizajna mehatroničkih sistema

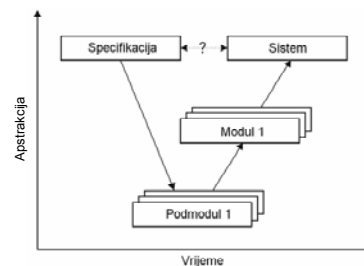
### Proces dizajna “odozdo prema gore”

- Predstavlja klasičnu metodu razvoja elektroničkih i mehaničkih komponenti.
- Početna tačka u ovoj vrsti dizajna je specifikacija, koja se obično iskazuje riječima prirodnog jezika.
- Osnovne komponente, tranzistori, otpornici, kondenzatori, opruge, mase, zglobovi, itd, se sukcesivno dodaju i kombiniraju kako bi se razvili složeniji sistemi, dok se ne kompletira proces dizajna.
- Ovo se obavlja na strukturalnoj razini, gdje se podmoduli kombiniraju u kreiranju modula, pri čemu se posebna pažnja posvećuje povezivanju ovih podmodula.
- Ovaj dizajn se može obaviti korištenjem editora krugova (circuit editor) ili prikladnih alata za višetjelesne (multibody) sisteme.

13/61

## Proces dizajna mehatroničkih sistema

### Proces dizajna “odozdo prema gore”



14/61

## Proces dizajna mehatroničkih sistema

### Proces dizajna “odozdo prema gore”

- Osnovna prednost procesa dizajna “odozdo prema gore” je da se utjecaj “neidealne” implementacije može uzeti u obzir u ranom stadiju procesa dizajna.
- Kod elektroničkih komponenti nezaobilazne su parazitarne otpornosti, kapacitivnosti i induktivnosti.
- U polju mehanike, neizbježni su npr. efekti trenja.
- Međutim, jedan problematičan aspekt se pojavljuje: specifikacije za dizajn, nakon što imamo diverziju (odbacivanje) od strane podmodula i modula sa stajališta apstraktnih opisa funkcionalnosti.
- Ovo je rezultat strukturirano-orijentiranog procesa modeliranja, **sistem može biti simuliran** samo kada je u cijelosti implementiran.
- Zbog toga pogreške i nedostaci u dizajnu sistema nisu primjetni do kasnijih stadija, što može uzrokovati značajne troškove i kašnjenja u razvoju.

15/61

## Proces dizajna mehatroničkih sistema

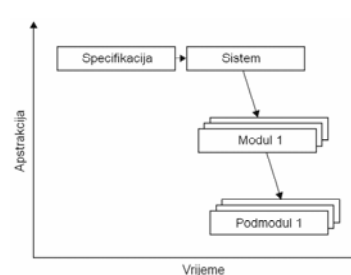
### Proces dizajna “odozgo prema dole”

- Važna karakteristika dizajna “odozgo prema dole” je prevladavajući pravac dizajna od apstrakcije prema detaljnim opisima (pogledati sliku na sljedećem slajdu).
- Početna tačka predstavlja čisti bihevioristički model, čija je funkcija već pokrivena dobrim dijelom sa specifikacijama.
- Model se uspješno dijeli i prečišćava dok se ne postigne implementacija.
- Bitno je opisati sistem ili module na funkcionalan način.
- Ovo je moguće načiniti uvođenjem HDL-a (hardware description language) u polju elektronike.
- Korištenjem ovih jezika dizajn se direktno formulira kao model, tako da većina procesa modeliranja može biti izostavljena.

16/61

## Proces dizajna mehatroničkih sistema

### Proces dizajna “odozgo prema dole”



17/61

## Proces dizajna mehatroničkih sistema

### Proces dizajna “odozgo prema dole”

- **Sekvenca ovog dizajna ima sljedeće prednosti:**
- Pogreške i nedostaci u dizajnu se otkrivaju ranije (u ranim fazama), nasuprot pristupu dizajna “odozdo prema gore”.
- Implementabilni dio specifikacija se može validirati korištenjem simulacija.
- Implementabilni dio specifikacija je raspoloživ, jednako kao i precizno definirane referentne veličine za verifikaciju dizajna.
- Funkcionalni dio specifikacija je nedvosmislen i cjelovit (suprotno specifikacijama prirodnog jezika). U slučaju sumnje, pokreće se simulacija.
- Implementabilne specifikacije i modeli pojedinačnih stadija dizajna znači da je cjelokupna dokumentacija dostupna, koji međutim ostaje da bude nadopunjen sa razumljivim komentarima.

18/61

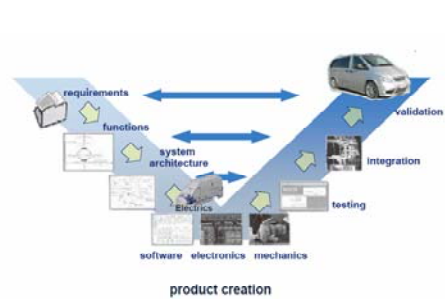
## Proces dizajna mehatroničkih sistema

### Proces dizajna "odozgo prema dole"

- Glavni nedostatak implementabilnih specifikacija je da neki tehnički detalji mogu biti izraženi na jednostavan, više kompaktan i jako nerazumljiv način u prirodnim jezicima u odnosu na jezike formalnog modeliranja.
- Problem je također u formalno korektnom opisu željene semantike, što uzrokuje dodatne troškove u vezi specifikacija.
- I na kraju problemi fizičke realizacije, kao što su prekomjerno vremensko kašnjenje se prepoznaje tek u kasnijim stadijima procesa dizajna.

19/61

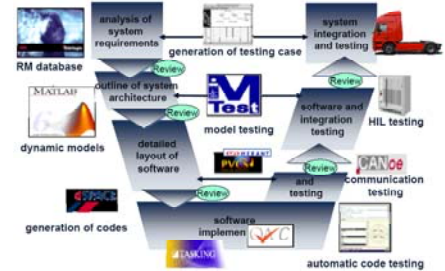
## 2.2. V-proces dizajna mehatroničkih sistema



20/61

## V-proces dizajna mehatroničkih sistema

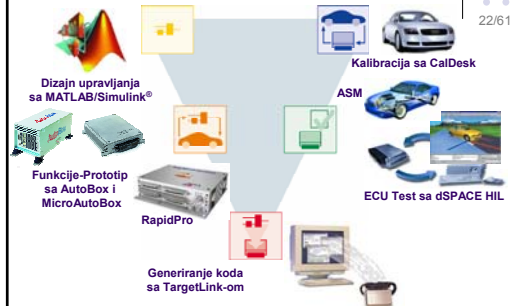
### V proces dizajna softvera



21/61

## V-proces dizajna mehatroničkih sistema

### V proces dizajna – Matlab + dSPACE



22/61

## V-proces dizajna mehatroničkih sistema

### Savjeti za dobar dizajn elektroničkih komponenti

- Verificirati performanse svake komponente neovisno, prije procesa integracije.
- Korištenje specifikacijskih listova za postizanje pouzdanih performansi.
- Opsežno mjerenje veličina el. kruga pomoću voltmetera i osciloskopa.
- Voditi računa o dizajnu u smislu robusnosti, tako da el. krugovi rade dobro, čak i unutar promjena temperature i performansi komponenti.
- Razumijevanje temeljnih fizikalnih osobina svake komponente, kako bi se najbolje razumjele njihove prednosti i ograničenja.

23/61

## Fizički dizajn mehatroničkih sistema

- Fizički dizajn obuhvaća:
  - Modeliranje sistema,
  - Dizajn sistema,
  - Dizajn mehaničkih komponenti,
  - Dizajn elektroničkih komponenti,
  - Dizajn softvera,
  - Integracija (montaža) sistema,
  - Analiza materijalnih troškova.

24/61

## Modeliranje sistema

- Matematičko modeliranje sistema se zahtijeva na nekoliko stadija dizajna mehatroničkog sistema, kao što su simulacija, dizajn upravljanja i rekonstrukcija varijabli.
- Dva načina modeliranja:
  - Teorijsko modeliranje – temelji se na fizikalnim principima.**
  - Eksperimentalno modeliranje (identifikacija) – temelji se na mjerenju ulaznih i izlaznih varijabli**
- Procedura teorijskog modeliranja obuhvaća:**
- 1. Definicija toka
  - Tok energije (električna, mehanička, toplinska).
  - Tok materije i energije (fluidi, prijenos topline, termodinamički, hemijski prijenosi)



25/61

## Modeliranje sistema

- Definicija elemenata procesa: dijagrami toka
  - Izvori, ponori (disipativni).
  - Elementi za pohranu, transformatori, pretvarači.
- Grafički prikaz modela procesa
  - Višeportni dijagrami.
  - Blok dijagrami toka podataka.
  - Bond grafovi za tok energije.
- Postavljanje jednačbi za sve procesne elemente:
  - Jednačbe ravnoteže za pohranu (masa, energija, moment).
  - Konstruktivske jednačbe za procesne elemente (izvori napajanja, transformatori, prtvarači).
  - Fenomenološki zakoni za ireverzibilne procese (disipativni elementi: sinks).



26/61

## Modeliranje sistema

- Jednačbe za međupovezivanje procesnih elemenata
  - Jednačbe kontinuiteta za paralelno povezivanje (zakon povezivanja čvorova).
  - Jednačbe kompatibilnosti za serijsko povezivanje (zakon zatvorenog kruga).
- Računanje cjelokupnog modela procesa
  - Uspostavljanje ulaznih i izlaznih varijabli.
  - Prikaz u prostoru stanja.
  - Ulazno-izlazni modeli (diferencijalne jednačbe, prijenosne funkcije).



27/61

## Dizajn elektroničkih komponenti

### Kategorije elektroničkih komponenti

- Elektroničke komponente koje se koriste u projektiranju mehatroničkih sistema mogu se podijeliti na:
  - Aktivne i pasivne komponente,
  - Analogne i digitalne,
  - Komponente velike i male snage,
  - Standardni (poluvodički) i optički krugovi,
  - Mikroprocesori i U/I moduli.



28/61

## Dizajn elektroničkih komponenti

- Dizajn mehatroničkih sistema korištenjem BS2 (Basic Stamp 2) tehnologije gradnje mikroprocesorskih sistema.
- BS2 je ugradivi sistem koji uključuje:**
  - Mikrokontroler (PIC16C57) – "mozak" sistema, osigurava osnovni (BASIC) interpreter, serijsku komunikaciju i U/I.
  - Memoriju (EEPROM) – za pohranu korisničkih programa i dugo čuvanje podataka.
  - Naponski regulator – generira 5 VDC iz izvora napajanja od 5.5 VDC do 15 VDC.
  - Klok (sat) – 20MHz rezonator.
  - Raznorsne komponente – tranzistori, otpornici,...

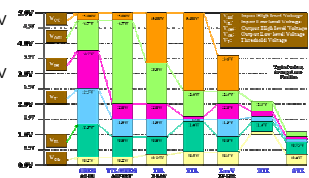


29/61

## Dizajn elektroničkih komponenti

BS2 digitalni ulazi koriste TTL pragove

- Digitalni ulaz ima visoko stanje (logička 1) ako je  $V > 2$
- Digitalni ulaz ima nisko stanje (logička 0) ako je  $V < 0.8$ .
- Ako je digitalni ulaz između 0.8 i 2, on može biti dvoznačan.



30/61

## Dizajn elektroničkih komponenti

BS2 digitalni izlazi

- Visoka razina za digitalni izlaz iznosi 4.7V ili više.
- Niska razina za digitalni izlaz iznosi 0.2V ili manje.

Specifikacije digitalnog izlaza:

Pin set – visoka razina

- Svaki pin može prosljediti maksimalnu struju iznosa 20mA.
- Svaka grupa pinova (P0-P7 i P8-P15) može prosljediti maksimalnu struju iznosa 40 mA.

Pin set – niska razina

- Svaki pin može primiti maksimalnu struju iznosa 25mA.
- Svaka grupa pinova (P0-P7 and P8-P15) može primiti maksimalnu struju 50 mA.

Napomena: ostali mikroprocesori tipično se snabdjevaju strujom, ali zahtijevaju pull-up otpornike.

31/61

## Dizajn elektroničkih komponenti

Kondicioniranje signala (serijski)

Kondicioniranje naponskih signala između PC serijske konekcije (+/-12V) i BS-a (5V)

5V Regulator

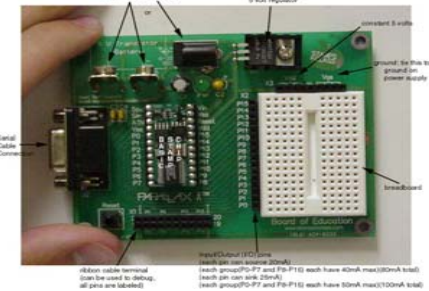
Regulira napon na 5V sa izvora napajanja od 5.5VDC do 15VDC

EEPROM  
Pohranjuje označeni PBASIC program.

Interpreter Chip  
Čita osnovni program iz EEPROM i izvršava instrukcije.

32/61

## Dizajn elektroničkih komponenti



33/61

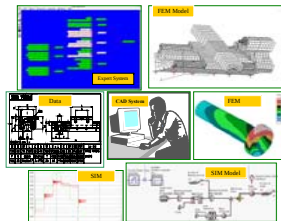
## Dizajn mehaničkih komponenti

Koristi se kompjuterski podržani alati (CAD/CAE)

- Koraci u dizajnu mehaničkih komponenti:
  - Specifikacije konstrukcija u inženjerskom razvoju korištenjem CAD i CAE alata.
  - Gradnja modela za postizanje statičkih i dinamičkih procesnih modela.
  - Transformacija u kompjuterski kod za potrebe simulacije sistema.
  - Programiranje i implementacija finalnog mehatroničkog softvera.
- Primjena CAD alata (npr. Auto CAD-a) za 2D i 3D modeliranje mehaničkih sistema i njegova direktna veza sa CAM (computer-aided manufacturing) alatima.

34/61

## Dizajn mehaničkih komponenti



• Ekspertni sistemi u dizajnerskoj praksi

- Integracija računarskih alata za dizajn proizvodnih strojeva.

• Primjena integriranih CAE alata za dizajn proizvoda

- Simulacijski sistemi u dizajnu strojarskih alata.

• Računarska analiza struktura strojarskih alata.

35/61

## Dizajn mehaničkih komponenti



• Modeliranje proizvodnih procesa

Modeliranje proizvodne infrastrukture kompanija za optimizaciju proizvodnih procesa - lay out, stock and technological plans optimization.

Softver: iGrafx, ProModel, ProPLAN, Factory Suite,...

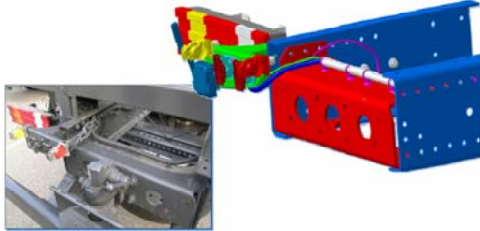
• Menadžment podataka proizvodnje

Modeliranje strukture proizvoda, kako sa stajališta pohrane, tako i sa stajališta integracije njegovih podataka za uvođenje metodologije integriranog razvoja proizvoda, organizacije rada i toka rada u skladu sa koncepcijom konkurentnog inženjeringa i PDM/TDM sistema.

36/61

## Dizajn mehaničkih komponenti

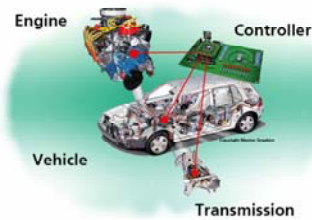
Dizajn mehaničkih komponenti korištenjem CAD alata – primjer Catia



37/61

## 2.3. Primjer dizajna automobila

- Mala složenost.
- Omogućuje jasan fokus na tok dizajna.
- Korisno za edukaciju u dizajnu digitalnih sistema i simulacijama.



Fraunhofer Institut  
Integrierte Schaltungen

38/61

## Primjer dizajna automobila

### Dizajn sistema

- Koncept razvoja proizvoda i verifikacija.
- Kreiranje modela procesa

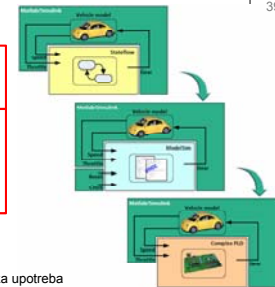
### Implementacija i verifikacija

- Dizajn u skladu sa implementacijskim targetom.
- Funkcionalna verifikacija.

### Razvoj prototipa

- Kompilacija.
- Verifikacija prototipa.

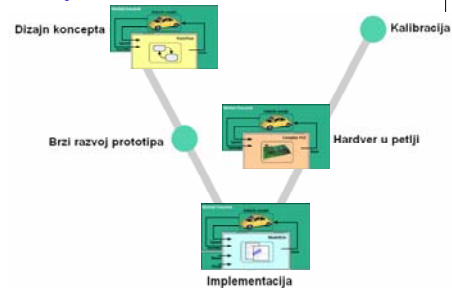
● - višestruka upotreba



39/61

## Primjer dizajna automobila

### V-dizajn



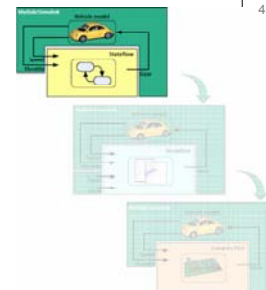
40/61

## Primjer dizajna automobila

### Dizajn sistema

### U fazi dizajna sistema ključni su:

- Definicija zahtjeva proizvoda.
- Razvoj tehničkog koncepta.



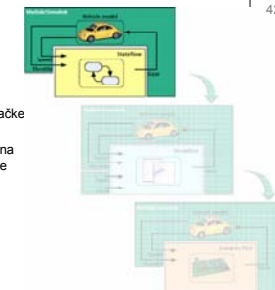
41/61

## Primjer dizajna automobila

### Dizajn sistema

### Zahtjevi na proizvod

- Razvoj automatizirane upravljačke jedinice za transmisiju.
- Određivanje tačnog prijenosa na temelju brzine vozila i papučice gasa.



42/61



### Primjer dizajna automobila

#### Dizajn sistema

**Koncept**

- Vremenski kontinuirani model vozila.
- Gear shift algorithm implementiran sa dva FSM-a (Finite State Machines).
- Tačke prebacivanja u drugu brzinu su uzete sa grafa i postavljene unutar dvije look-up tabele.
- Verifikacija koncepta korištenjem kombinacije Simulink-a i Stateflow-a.

43/61

### Primjer dizajna automobila

#### Implementacija i verifikacija

- Donošenje odluke o izboru platforme.
- Transfer dizajna u odgovarajuću implementaciju ili opisni jezik.
- Verifikacija funkcionalnog dizajna (kosimulacija).

44/61

### Primjer dizajna automobila

#### Implementacija i verifikacija

**Hardver ili softver?**

- Softverski temeljen target (DSP, Microcontroller,...):
  - ANSI C/C++,
  - ASSEMBLER,
  - ADA,
  - ...
- Hardverski temeljen target (FPGA, CPLD,...):
  - VHDL,
  - Verilog,
  - Abel,
  - ...

45/61

### Primjer dizajna automobila

#### Implementacija i verifikacija

**Softver**

- Ručno pisani C kod za niskotroškovne implementacije.
- Real-Time Workshop – Mathworks.
- SystemBuild/Autocode– National Instruments

46/61

### Primjer dizajna automobila

#### Implementacija i verifikacija

**Hardver**

- Specijalni blokovski skupovi su raspoloživi, koji uključuju DSP funkcije.
  - Xilinx,
  - Lattice,
  - Altera,
  - AccelChip,
  - ...

47/61

### Primjer dizajna automobila

#### Implementacija i verifikacija

**Implementacija**

- Prevođenje Stateflow modela u VHDL kod.
- Prevođenje može biti izvršeno korištenjem alata (ograničeno) ili ručno.
- VHDL kod može biti verificiran korištenjem Model Technologie's ModelSim programa.

**Problem**

- Teško je predstaviti dinamiku vozila unutar VHDL testne jedinice.

48/61



### Primjer dizajna automobila

#### Implementacija i verifikacija

49/61

#### Kosimulacija

- Omogućuje korištenje specijaliziranih simulatora
- Vremenski kontinuirani model može biti simuliran sa Simulinkom.
- VHDL opis se simulira sa ModelSim.
- Moguće je razviti dvosmjerni link između Simulink-a and ModelSim-a.

### Primjer dizajna automobila

#### Implementacija i verifikacija

#### Uparivanje Simulinka i ModelSim-a

50/61

- Primjer razvijen na Fraunhofer institutu.
- Proširenje Matlab/Simulink i ModelSim programa sa dodatnim sučeljskim blokom.
- Komunikacija preko TCP/IP kontakata.
- Sinhronizacija preko jednakih prostorno raspoređenih tačaka uzorkovanja ili događaja.
- Slobodan izbor frekvencije uzorkovanja.
- Podrška solverima fiksnog i promjenjivog koraka.

### Primjer dizajna automobila

#### Implementacija i verifikacija

51/61

#### Rezultati kosimulacije

- Male razlike u rezultatima zbog interpolacije signala pritiska na papučicu gasa (throttle) unutar Stateflow-a (Matlab), pri čemu VHDL implementacija koristi fiksiranu look-up tabelu.
- Stateflow model ima veću tačnost (precizniji je).
- Zbog male veličine CPLD-a, veličina lookup tabele je minimizirana.

### Primjer dizajna automobila

#### Razvoj prototipa

52/61

Bitne funkcije ove faze su:

- Preslikavanje funkcija dizajna na željenu hardversku platformu.
- Verifikacija korištenjem hardvera u simulacijskoj petlji (HIL – hardware in the loop).

### Primjer dizajna automobila

#### Razvoj prototipa

53/61

#### Preslikavanje na hardver

- Koristi ranije razvijeni VHDL kod za mjenjačku kutiju (prikazanu sa FSM-ovima).
- Dodani extra moduli za
  - Simulink sučelje,
  - Vizualizaciju (LED diode).
- Sintetiziranje VHDL za PLD (Programmable Logic Device) uređaje.
- Učitavanje koda u hardver.

### Primjer dizajna automobila

#### Razvoj prototipa

54/61

#### HIL

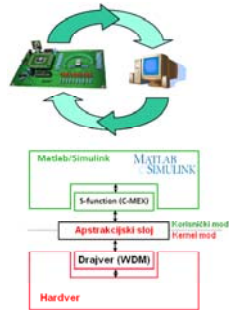
- Omogućuje integraciju realnog hardvera unutar simulacije.
- Vremenski kontinuiran model vozila može se simulirati sa Simulinkom.
- VHDL opis FSM-a se sintetiziraju za zahtijevanu hardversku platformu.
- Fraunhofer institut je razvio dvosmjerni link između Simulinka i korištenog hardvera.

## Primjer dizajna automobila

### Razvoj prototipa

#### Link za Hardver

- Matlab/Simulink ije proširen sa blokom (S-function) koji omogućuje pristup vanjskom hardveru.
- Drajver (WDM) implementira stvarni pristup hardveru i real-time ponašanje.
- Interakcije se odvija preko apstrakcijskog sloja (API).

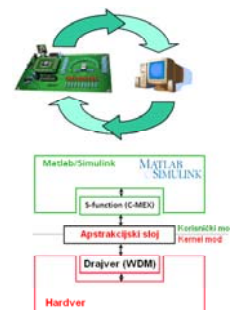


55/61

## Primjer dizajna automobila

### Razvoj prototipa Karakteristike

- Frekvencija uzorkovanja na hardveru se može slobodno podešavati.
- Minimalno hardversko vrijeme uzorkovanja je 10 ms (Windows OS ograničenja).
- Interni Simulink model može se pokretati i sa manjim vremenima uzorkovanja.
- Raspoloživa više hardverskih sučelja:
  - SPI (Synchronous Peripheral Interface)
  - Paralelni port, RS-232 (Bluetooth).
  - Niski troškovi.



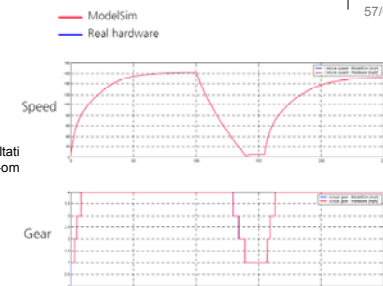
56/61

## Primjer dizajna automobila

### Razvoj prototipa

#### Rezultati

- Identični rezultati sa ModelSim-om i realnim hardverom.



57/61

## 2.4. Dizajn ECU za upravljanje vozilom

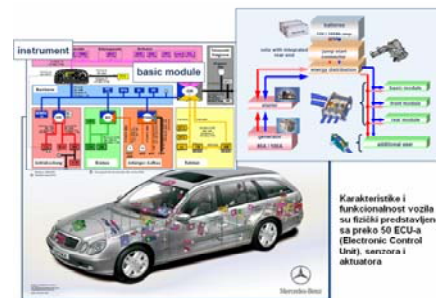
Primjer: Adaptivno upravljanje vozilom (ACC)  
Senzorska i upravljačka jedinica ECU

Elektroničke komponente

58/61



## Elektronika za sigurnost i komfor u vožnji

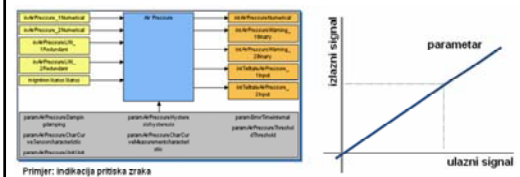


59/61

## Signali i parametri ECU-ova

Signali opisuju vremenski promjenjivu komunikaciju ulaza i izlaza ECU-a, statičke parametre ECU-a i karakteristike vozila.

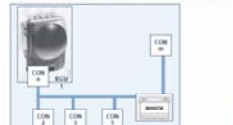
Primjer: indikacija pritiska zraka



60/61

## Električne komponente u automobilu

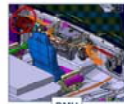
Example: Adaptive Cruise Control Sensor and Central Unit ECU in network



wiring diagrams and topology plans



(connector) mechanics



DMU



wiring harness

61/61