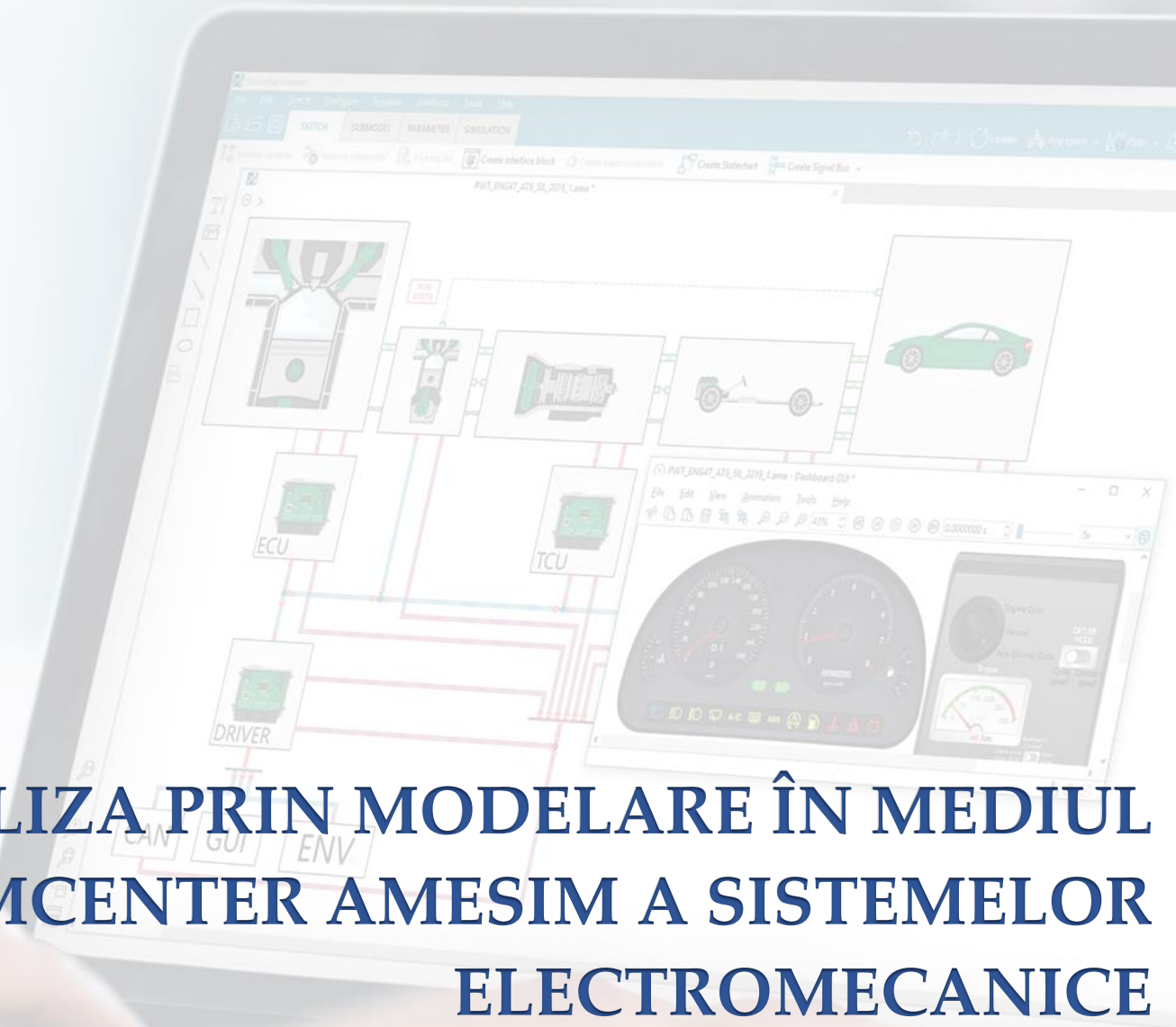

Mircea Ruba, Raul-Octavian Nemeș, Eduard-Edis Răclau, Călin-Ioan Husar



ANALIZA PRIN MODELARE ÎN MEDIUL SIMCENTER AMESIM A SISTEMELOR ELECTROMECHANICE

Îndrumar de laborator

UTPRESS

Cluj-Napoca, 2021

ISBN 978-606-737-513-8

**Analiza prin modelare în mediul Simcenter Amesim
a sistemelor electromecanice
Îndrumar de laborator**

Mircea Ruba, Raul-Octavian Nemeș, Eduard-Edis Răclau, Călin-Ioan Husar



UTPRESS

Cluj-Napoca, 2021

ISBN 978-606-737-513-8

SIEMENS
Ingenuity for life


**UNIVERSITATEA
TEHNICĂ**
Cluj-Napoca



Editura U.T.PRESS
Str. Observatorului nr. 34
400775 Cluj-Napoca
Tel.: 0264-401.999
e-mail: utpress@biblio.utcluj.ro
<http://biblioteca.utcluj.ro/editura>

Director: ing. Călin Câmpean

Recenzia: Conf.dr.ing. Florin Nicolae Jurca
Prof.dr.ing. Claudia Marțiș

Copyright © 2021 Editura U.T.PRESS

Reproducerea integrală sau parțială a textului sau ilustrațiilor din această carte este posibilă numai cu acordul prealabil scris al editurii U.T.PRESS.

ISBN 978-606-737-513-8

Bun de tipar: 20.05.2021

Cuprins

Simcenter Amesim EMR – Descriere generală	1-1
Exercițiul 1 – Instalarea librăriei EMR din Simcenter Amesim	2-1
Exercițiul 2 – Elementele tip bloc EMR	3-1
Exercițiul 3 – Construcția programelor bazate pe regulile EMR	4-1
Exercițiul 4 – Simularea bazată pe organizare EMR în Simcenter Amesim	5-1
Exercițiul 5 – Compararea organizării EMR cu diagramele bloc clasice	6-1
Exercițiul 6 – Conflicte de asociere	7-1
Exercițiul 7 – Dezvoltarea sistemului de control prin inversiunea modelului EMR	8-1
Exercițiul 8 – Acordarea reguletoarelor din structura de control	9-1

Simcenter Amesim EMR – Descriere generală

Această publicație are menirea de a aduce în fața cititorului un pachet de informații legate de utilizarea filozofiei EMR (Energetic Macroscopic Representation) de organizare a programelor de simulare pe calculator. Aplicațiile sunt implementate în programul Simcenter Amesim, program al companiei Siemens cu care Universitatea Tehnică din Cluj Napoca are o colaborare perpetuă de cel puțin 10 ani. Trebuie menționat faptul că acest exercițiu de educare în vederea folosirii organizării EMR în acest program a fost inițial conceput în cadrul proiectului H2020 PANDA (GA#824256, 2018-2021) în care Siemens Brașov, Universitatea Tehnică din Cluj Napoca respectiv Universitatea din Lille, au fost colaboratori. Sistemul de organizare EMR a fost dezvoltat începând cu anul 2000 de laboratorul L2PE ca entitate de cercetare-dezvoltare a Universității din Lille, Franța, condus de Prof. Alain Bouscayrol. Acesta din urmă a fost pionierul organizării EMR pe care materialul prezent îl va detalia.

În acest volumul, subiectele care vor fi atinse și dezbătute ca îndrumar de activitate practică sunt următoarele:

1. Cum se instalează librăria EMR?
2. Elementele librăriei EMR
3. Procesul de implementare a simulărilor EMR
4. Cum construim primul program de simulare EMR?
5. Compararea diagramelor bloc tip EMR
6. Cum soluționăm conflicte de asociere în EMR?
7. Cum construim structuri de control bazate pe inversiuni în filozofia EMR?
8. Cum acordăm regulatoarele unui sistem de control EMR?

Informații legate de formalismul de organizare EMR

EMR (Energetic Macroscopic Representation) este un formalism de organizare a modelelor de simulare pentru sisteme de conversie a energiei. A fost dezvoltat în anul 2000 la laboratorul L2EP a Universității din Lille iar în momentul de față este utilizat pe scară largă în particular pentru analiza prin modelare a sistemelor de electrificare a vehiculelor. EMR are la bază abordarea sistemic cognitivă descrisă din punct de vedere funcțional și cauzal.

Mai multe informații puteți găsi pe pagina web oficială: <http://www.emrwebsite.org/>

Informații legate de Simcenter Amesim

Simcenter Amesim este un software comercial dedicat simulărilor multi-domeniu pentru sisteme complexe. Face parte din ierarhia programelor de simulare pentru ingineria sistemelor focalizat pe arhitecturi mecatronice. Programul Amesim conține o suită de unelte pentru modelare, analiză și predicție de performanțe ale sistemelor electromecanice și mecatronice. Modelele sunt construite pe baza ecuațiilor matematice non-liniare, dependente de timp care descriu comportamentul sistemelor hidraulice, pneumatice, termice sau mecanice. Comparat cu simulările tip 3D CAE, modelarea în Amesim a sistemelor mai sus înșirate este mult mai rapidă și mai la îndemână, astfel încât este o soluție lucrativă chiar în etapele preliminare ale ciclului în V de dezvoltare a sistemelor.

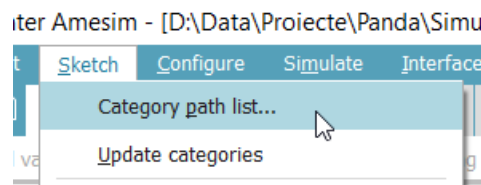
Exercițiul 1 – Instalarea librăriei EMR din Simcenter Amesim

Înainte de a putea realiza modele de simulare în mediul Amesim organizate în filozofia EMR, este necesar ca să se instaleze librăria care conține blocurile necesare.

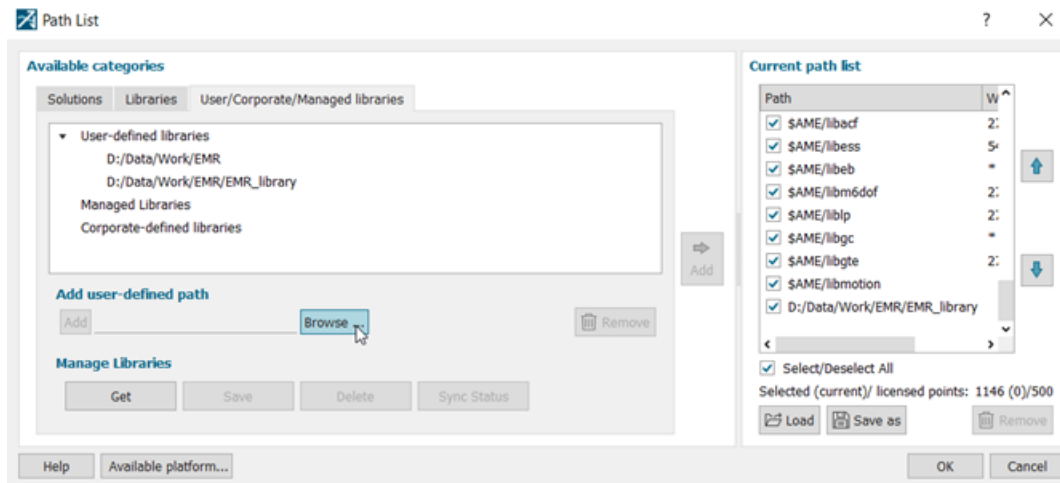
EMR Instalarea librăriei

Inițial, librăria EMR nu face parte din suita de librării cu care vine echipat pachetul de program Simcenter Amesim. Această librărie a fost dezvoltată exclusiv în cadrul proiectului PANDA motiv pentru care are caracter de valoare adăugată la programul Amesim.

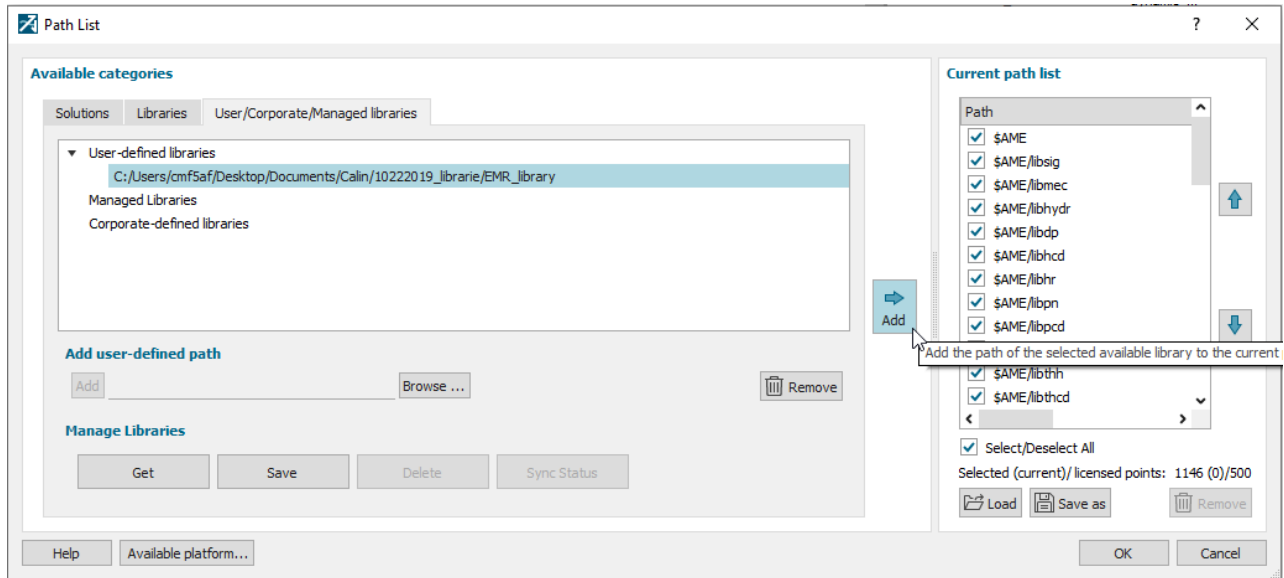
Pentru a instala librăria utilizatorul trebuie să intre la **Sketch > Category path list...**



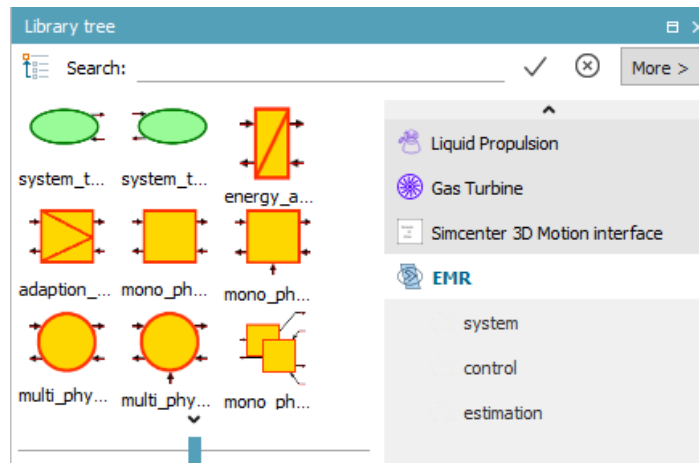
La fereastra care se deschide pe ecran intitulată **Path List** se va selecta **User/Corporate/Managed libraries** iar apoi se va apăsa butonul **Browse**.



Vom direcționa programul spre destinația unde am descărcat librăria și vom apăsa **Select Folder**. În fereastra **Path List** la **User/Corporate/Managed libraries > User defined libraries** să va asigurați că este corectă calea către librăria EMR urmând apoi să apăsați butonul **Add**.



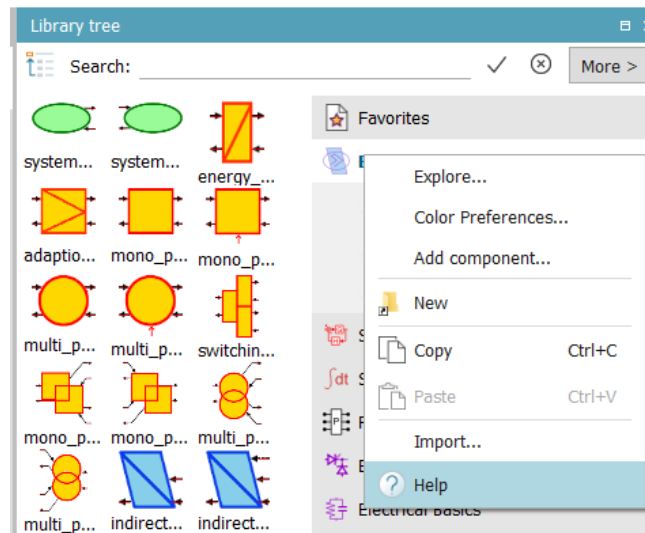
Apoi, apăsați butonul **OK** pentru a ieși din fereastra **Path List**. Astfel, Simcenter Amesim va include în lista librărilor cea care conține elementele EMR. Veți găsi această nouă librărie în zona **Library tree**.



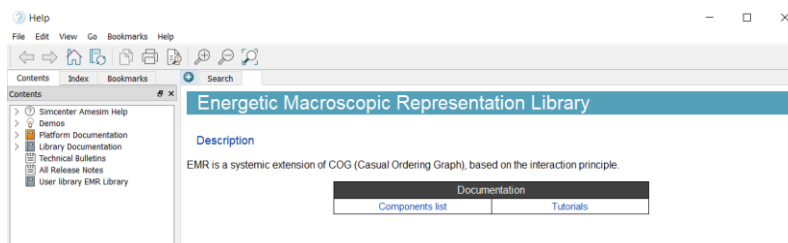
Din acest moment, Simcenter Amesim este pregătit pentru construcția și simularea aplicațiilor EMR.

Pagina de ajutor dedicată librăriei EMR

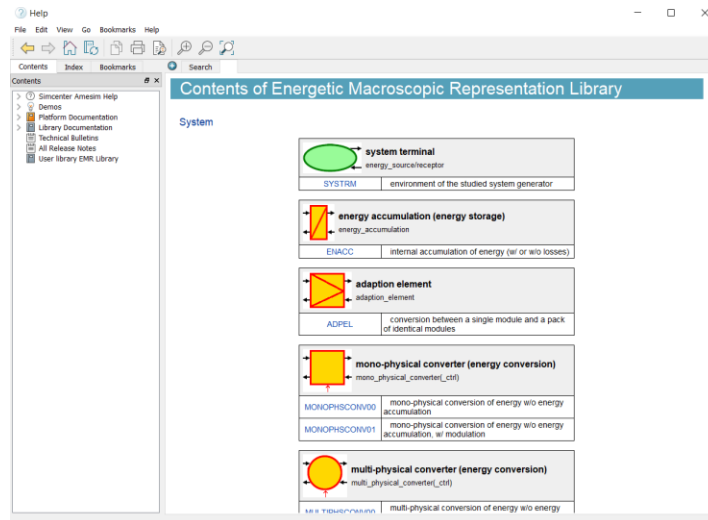
Modulul de ajutor adresat librăriei EMR este accesibil prin următoarea succesiune de comenzi: dați click dreapta pe librăria EMR din **Library tree** iar apoi selectați butonul **Help**.



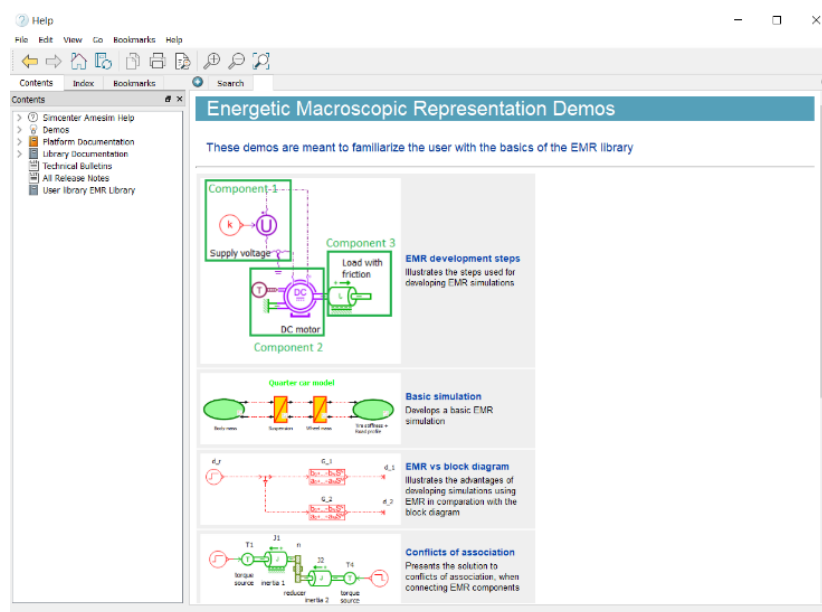
Secțiunea Help conține documentații despre EMR și elementele librăriei precum și exerciții pentru familiarizarea cu Simcenter Amesim în organizare EMR.



Lista de componente conține pictograma aferentă elementelor ERM dar totodată și o succintă descriere pentru fiecare dintre ele.



În secțiunea de exerciții accesată din meniul Help, se vor găsi aplicații tip ghid pentru a familiariza utilizatorii cu mediul Simcenter Amesim și fișierele EMR de simulare.



Exercițiul 2 – Elementele tip bloc EMR

EMR este un formalism grafic pentru organizarea modelelor și controlul sistemelor complexe multi-domeniu. El este compus din elemente derivate din funcțiile energetice (sursă, stocare, conversie și distribuție). Deoarece EMR este o descriere funcțională, nu este legată de componente (descriere structurală): unele componente sunt descrise de mai multe elemente EMR (funcții energetice multiple), iar unele asociații de componente sunt descrise de un element EMR unic (funcție comună). Regulile grafice EMR sunt prezentate în anexă.

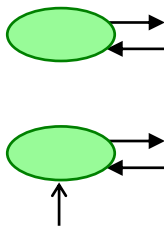
Fiecare element EMR are proprietăți grafice specifice (inclusiv tipul de porturi de interconectare) pentru o identificare ușoară în schema globală. Astfel, pictogramele specifice au fost dezvoltate în biblioteca EMR a Simcenter-Amesim.

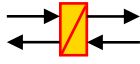
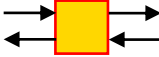
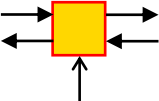
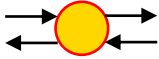
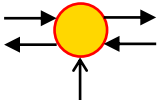
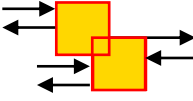
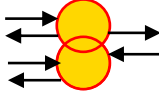
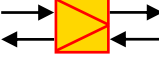

Caracteristicile fiecărui element sunt prezentate în acest exercițiu. Pe baza lor, putem determina circumstanțele care indică momentul utilizării fiecărui element.

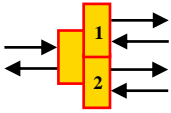
Modelul unui subsistem

Diferit de EMR clasic, unde unii vectori nu sunt obligatorii, în Simcenter Amesim toate componentele posibile sunt descrise astfel încât utilizatorul alege elementul cu numărul corect de intrări/ieșiri (în elementele de control liniile punctate sunt astfel înlocuite cu diferite opțiuni).

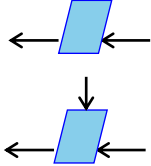
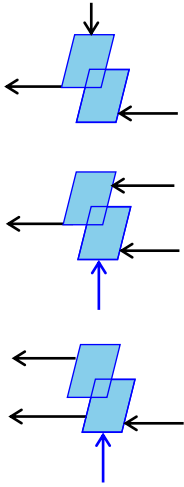
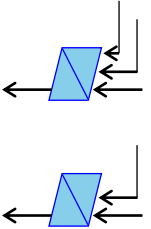
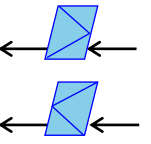
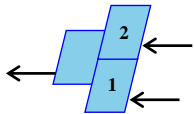
Elemente de model

Nume	Pictograma	Caracteristici
Sursă de energie		Element folosit la extremele sistemului (în afara sistemului) Generator sau receptor de energie 1 vector de intrare (dimensiune n) 1 vector de ieșire (dimensiune n) Sursă cu intrare de reglare: vectorul de intrare de reglare (dimensiune n)
		Element pentru stocarea internă a energiei, cu sau fără pierderi.

Acumulare de energie		<p>Relația dinamică între variabilele I / O (fixă). Vectorii I / O din amonte (dimensiune n) Vectorii I / O din aval (dimensiune n)</p>
Conversie de energie	<p>Mono-domeniu</p>   <p>Multi-domeniu</p>  	<p>Element pentru conversia energiei fără stocare internă, cu sau fără pierderi Transmisie fără întârziere între I/O</p> <p>Mono-domeniu (pătrat): conversia energiei în același domeniu fizic Multi-domeniu (cerc): conversia energiei între două domenii fizice</p> <p>Vectorii I / O din amonte (dimensiune n) Vectorii I / O din aval (dimensiune m)</p> <p>Element de conversie cu intrare de reglare: Vector de intrare de reglaj (dimensiune q)</p>
Distribuție de energie	<p>Mono-domeniu</p>  <p>Multi-domeniu</p> 	<p>Element pentru distribuția energiei fără stocare internă de energie, cu sau fără pierderi de energie Transmisie fără întârziere între I/O</p> <p>n elemente conectate = n-1 forme imbricate</p> <p>Mono-domeniu (pătrate suprapuse): conversie de energie în același domeniu fizic Multi-domeniu (cercuri suprapuse): conversie de energie între mai multe domenii fizice</p> <p>Perechi de vectori I / O de aceeași dimensiune.</p>
Adaptare de energie	 	<p>Element pentru adaptarea puterii între două părți, fără stocare de energie fără pierderi,</p> <p>Semnul > sau < este folosit pentru a indica care putere este mai mare</p> <p>Vectorii I / O din amonte (dimensiune n) Vectorii I / O din aval (dimensiune m)</p>

Elemente de comutare		<p>Element pentru comutarea între 2 modele</p> <p>n dreptunghiuri suprapuse pentru conectarea a n modele</p> <p>Toate perechile I / O sunt identice</p>
----------------------	---	---

Elemente de control

Inversie directă pentru conversia energiei		<p>Control în buclă deschisă fără respingerea perturbării.</p> <p>Control în buclă deschisă cu respingerea perturbării.</p>
Inversie directă pentru distribuția energiei		<p>Control în buclă deschisă cu respingere explicită a perturbării.</p> <p>Control în buclă deschisă cu variabila de control.</p> <p>Control în buclă deschisă cu variabilă de control al distribuției.</p>
Inversie indirectă pentru acumularea de energie		<p>Control cu buclă închisă cu respingere explicită a perturbărilor.</p> <p>Buclă închisă fără respingere explicită a perturbărilor.</p>
Inversie directă pentru adaptare		<p>Control în buclă deschisă pentru ambele tipuri de elemente de adaptare.</p>
Inversie directă pentru comutarea modelului		<p>Control în buclă deschisă pentru ambele tipuri de elemente de comutare.</p>

Elemente de estimare

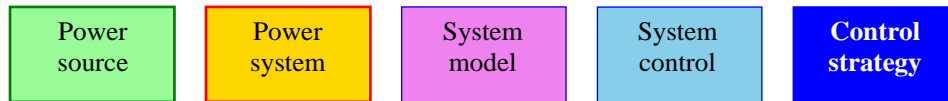
<p>Elemente de estimare (exemple)</p>		<p>Estimează variabilele de sistem pe baza măsurătorilor de sistem (copie a modelelor, dar cu săgeți de semnal).</p> <p>Intrările sunt măsurători ale sistemului. Rezultatele sunt estimări ale variabilelor.</p>
---------------------------------------	--	---

Bloc de strategie (de control)

<p>Element de strategie</p>		<p>Folosit pentru a implementa algoritmi de supervizare, cum ar fi strategiile de gestionare a energiei.</p> <p>Primește intrări de la elementele EMR (strat hardware) și / sau control (strat software).</p> <p>Oferă variabile de control (referințe, variabile de ponderare și distribuție) pentru controlul local.</p>
-----------------------------	--	--

Anexă: reguli grafice

Pictogramele EMR se bazează pe paleta „Web X11” de culori, standard pentru paginile web.



Sursa de alimentare (mediul sistemului):

- fundal verde deschis: RGB = (152,251,152) = # (98, FB, 98) = „Verde pal”
- chenar verde închis (dimensiunea b pt): RGB = (0,128,0) = # (00,80,00) = „Verde”

Sistem de alimentare (sistem energetic studiat):

- fundal portocaliu: RGB = (255,215,0) = # (FF, D7,00) = „Auriu”
- chenar roșu (dimensiunea b pt): RGB = (255,0,0) = # (FF, 00,00) = „Roșu”

Model de sistem (estimare sau model):

- fundal violet: RGB = (238,130,238) = # (EE, 82, EE) = „Violet”
- chenar albastru închis (dimensiunea b / 2 pt): RGB = (0,0,255) = # (00,00, FF) = „Albastru”

Controlul sistemului (controlul inversat, adică nivelul de control local):

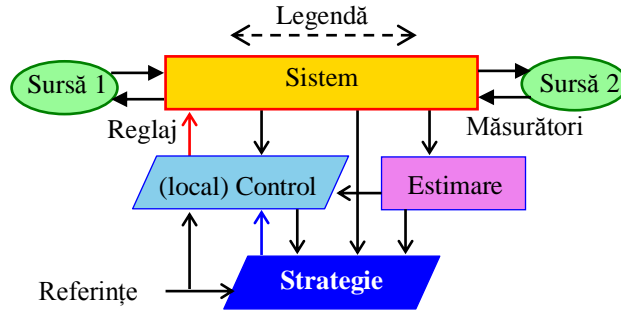
- fundal albastru deschis: RGB (135,206,235) = # (87, CE, EB) = „Albastru deschis”
- chenar albastru închis (dimensiunea b / 2 pt): RGB = (0,0,255) = # (00,00, FF) = „Albastru”

Strategia de control (managementul energiei sau supravegherea sau strategia):

- fundal albastru închis: RGB = (0,0,255) = # (00,00, FF) = „Albastru”
- chenar albastru închis (dimensiunea b / 2 pt): RGB = (0,0,255) = # (00,00, FF) = „Albastru”

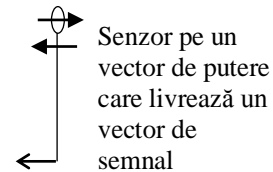
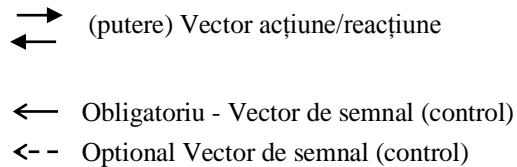
Generalități ale pictogramelor EMR

- pictograme verzi pentru mediul extern sistemului;
- pictograme portocalii pentru sistemul studiat;
- pictograme albastru deschis pentru controlul (local),
- pictograme violete pentru estimare sau modele;
- pictograme albastru închis pentru strategia de control;



Diferite tipuri de vectori

- vector de putere (acțiune / reacție): linie dreaptă, săgeată completă (dimensiune medie),
- vector semnal (reglare, control): linie dreaptă, săgeată goală (dimensiune mică),
- toți vectorii sunt desenați cu săgeți negre, cu excepția următorilor vectori:
 - săgeți roșii pentru intrările de semnal care au fost alese pentru a regla sistemul;
 - săgeți albastru închis pentru vectorul de semnal care oferă un criteriu controlului local (livrat de strategia de control)



Exercițiul 3 – Construcția programelor bazate pe regulile EMR

Obiectiv: În acest exercițiu vom studia pașii utilizați pentru dezvoltarea simulărilor EMR.

Procesul de dezvoltare a simulărilor bazate pe biblioteca EMR ia în considerare următorii pași:

Descrierea sistemului

Pentru a dezvolta EMR-ul unui sistem complex tip multi-domeniu, începem prin alegerea nivelului de abstractizare a sistemului pe baza obiectivelor noastre de simulare. Acest pas implică determinarea elementelor sistemului și a interacțiunilor acestora.

Elementele sunt date de descrierea funcțională a sistemului.

Interacțiunea dintre elemente se face prin transfer de energie caracterizat prin variabile acțiune-reacție. Astfel, trebuie să determinăm limitele elementelor specificând variabile de interconectare consistente.

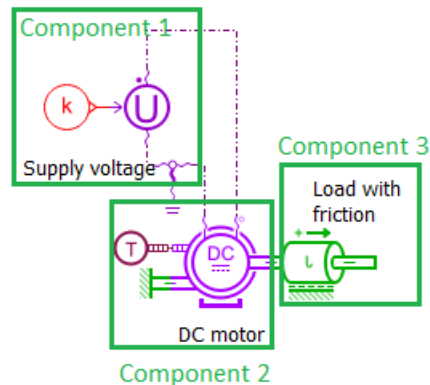


Fig. 1. Descrierea structurală a motorului de curent continuu

În continuare, presupunem că avem la dispoziție modele dinamice pentru sistemul considerat. Astfel, elementele asociate vor organiza practic diferitele arhitecturi de modele în conformitate cu regulile EMR.

Elemente utilizate pentru simulare

Pe baza modelelor componente determinăm tipurile de elemente EMR care trebuie utilizate.

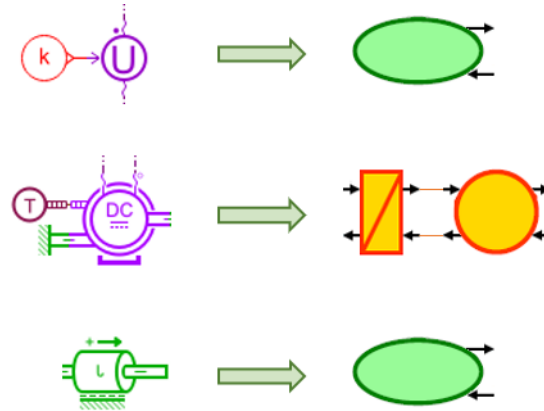


Fig. 2. Elementele structurale și elementele EMR asociate lor

! Notă. Dacă o componentă nu se potrivește cu cerințele oricărui element EMR, atunci modelul de componentă trebuie împărțit astfel încât să poată fi reprezentat folosind o interconectare a mai multor elemente EMR.

Diagrama EMR

Diagrama EMR este configurată utilizând biblioteca EMR, prin interconectarea elementelor EMR ale tuturor componentelor.

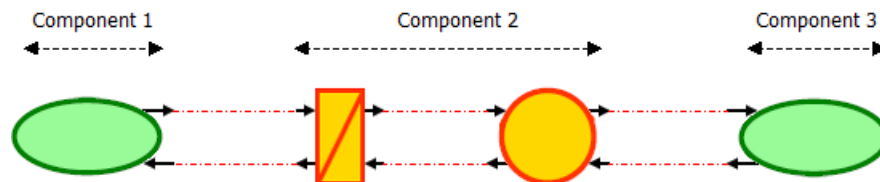


Fig. 3. EMR diagram

Rolurile intrărilor și ieșirilor (I/O)

Primul pas este să determinăm rolurile I / O ale variabilelor de interconectare. În primul rând, atribuim rolurile pentru variabilele de interconectare ale elementelor de acumulare, deoarece acestea au roluri I / O fixe datorită cerinței integrale de reprezentare cauzală a EMR. Apoi alegem rolurile I / O pentru variabilele limită ale celorlalte elemente EMR.

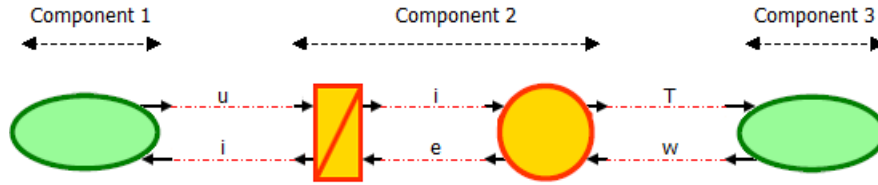


Fig. 4. Diagrama EMR cu roluri I / O

Modelul ecuațiilor

În cele din urmă, populăm blocurile EMR care implementează ecuațiile modelului.

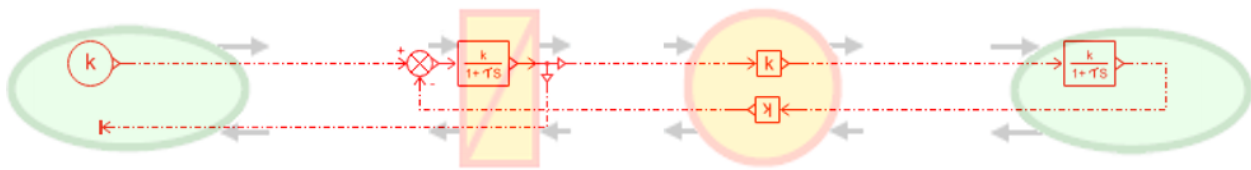


Fig. 5. Modelul bazat pe ecuații

În acest caz, vom folosi biblioteca standard **Semnale și Control** în Simcenter Amesim, pentru a implementa ecuațiile modelului în reprezentarea diagramelor bloc, în interiorul fiecărui element EMR.

Exercițiul 4 – Simularea bazată pe organizare EMR în Simcenter Amesim

Obiectiv: În acest exercițiu vom învăța cum să dezvoltăm o primă simulare EMR, folosind biblioteca EMR din Simcenter Amesim.

Pentru a ilustra modul de dezvoltare a simulărilor EMR în Simcenter Amesim, luăm drept exemplu simularea unei suspensii auto. Obiectivul este de a simula deplasarea caroseriei și a roților peste o treaptă pe drum.

Vom dezvolta simularea EMR, folosind procesul de dezvoltare prezentat în exercițiul nr. 3.

Determinarea componentelor sistemului și a interacțiunilor acestora.

Pentru a face predicții despre deplasarea caroseriei și a roții, modelul suspensiei auto implică: masa mașinii, masa roții, rigiditatea și amortizarea suspensiei, rigiditatea roții și profilul drumului. O descriere structurală a modelului de suspensie este prezentată în figura de mai jos, ilustrând diferitele elemente implicate și interconectările acestora.

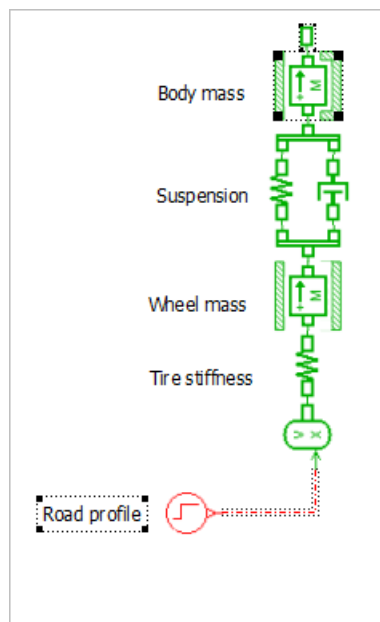


Fig. 1. Descrierea structurală a suspensiei vehiculului

Interacțiunile dintre elemente se realizează prin transfer de energie, caracterizat prin variabile de energie.

Componenta	Conexiune sus		Conexiune jos		Putere	
	Efort	Flux	Efort	Flux	c. sus	c. jos
Greutatea Caroseriei	-	-	F_{M_1} [N]	v_{M_1} [m/s ²]	-	$F_{M_1} v_{M_1}$ [W]
Suspensii	F_{M_1} [N]	v_{M_1} [m/s ²]	F_{M_1} [N]	v_{M_2} [m/s ²]	F_{M_1} [W]	v_{M_1} F_{M_1} v_{M_2} [W]
Masa roții	F_{M_1} [N]	v_{M_2} [m/s ²]	F_{M_2} [N]	v_{M_2} [m/s ²]	F_{M_1} [W]	v_{M_2} $F_{M_2} v_{M_2}$ [W]
Rigiditatea roții	F_{M_2} [N]	v_{M_2} [m/s ²]	-	-	F_{M_2} [W]	v_{M_2} -

Table 1. Variabile de interconectare a elementelor pe baza acțiunii-reației

Modelul de suspensie auto este dat de următoarele ecuații:

$$F_{M_1} = M_1 \frac{dv_1}{dt} \quad (\text{masa vehiculului}) \quad (4.1)$$

$$F_{M_1} = k_s(d_2 - d_1) + b(v_2 - v_1) \quad (\text{suspensia}) \quad (4.2)$$

$$F_{M_2} - F_{M_1} = M_2 \frac{dv_2}{dt} \quad (\text{masa roții}) \quad (4.3)$$

$$F_{M_2} = k_w(d_r - d_2) \quad (\text{rigiditatea anvelopei}) \quad (4.4)$$

$$v_1 = \frac{d}{dt} d_1 \quad (\text{Viteza 1}) \quad (4.5)$$

$$v_2 = \frac{d}{dt} d_2 \quad (\text{Viteza 2}) \quad (4.6)$$

Unde variabilele sunt:

F_{M_1} = forța asupra caroseriei

F_{M_2} = forța la roată

v_1 = viteza verticală a caroseriei

v_2 = viteza verticală a roții

d_1 = deplasarea vehiculului

d_2 = deplasarea roții

d_r = profilul de drum

Iar parametrii sunt:

M_1 = masa vehiculului

M_2 = masa roții

k_s = rigiditatea suspensiei

k_w = rigiditatea anvelopei

b = amortizarea suspensiei

Determinarea tipurilor de elemente EMR care trebuie utilizate

Pe baza ecuației modelului, determinăm elementele EMR pentru fiecare componentă, utilizând tabelul cu elementele EMR din exercițiul 2.

Componenta	Funcția EMR
Caroserie	Element sursă
Suspensia	Element de acumulare
Masa roții	Element de acumulare
Rigiditatea anvelopei	Element sursă

Tabelul 2. Reprezentarea EMR a fiecărei componente

Masa caroseriei și rigiditatea anvelopei sunt elemente terminale, care se conectează la sistem doar printr-un singur port. Ca urmare, ele se potrivesc cu descrierea elementului sursă.

Modelul de suspensie implică variabilele stării de deplasare care pot fi calculate prin integrarea variabilelor de viteză corespunzătoare. Prin urmare, componenta suspensiei se potrivește caracteristicilor elementului de acumulare.

Dinamica roții implică calcularea vitezei roții prin integrarea variabilelor forței. Astfel, componenta de masă a roții va fi reprezentată și cu ajutorul unui element de acumulare.

Diagrama EMR este configurată

Biblioteca EMR este utilizată pentru a crea structura EMR în Simcenter Amesim care se potrivește cu interacțiunile structurale ale componentelor.

Biblioteca EMR poate fi găsită în arborele principal al bibliotecilor, când suntem în modul Sketch.

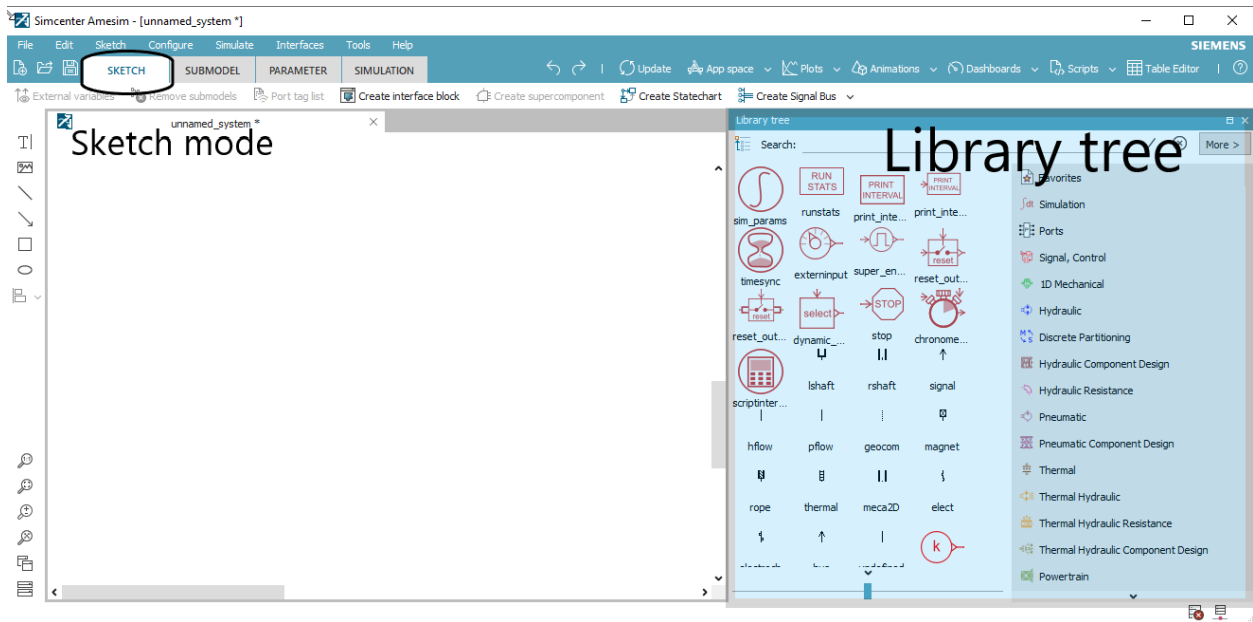


Fig. 2. Aspectul spațiului de lucru Simcenter Amesim

Pentru construirea diagramei EMR, facem clic pe componentele dorite din biblioteca EMR și le plasăm pe spațiul de lucru.

Apoi, interconectăm componentele în funcție de interacțiunile structurale ale sistemului.

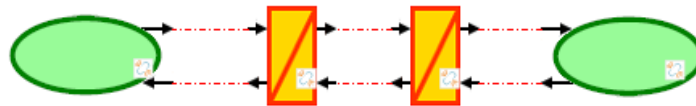



Fig. 3. Elemente EMR de susținere interconectate

Pentru a putea identifica cu ușurință componentele din diagramă, vom adăuga numele componentelor folosind instrumentul Inserare text, găsit în bara de instrumente din stânga și indicat de pictogramă .

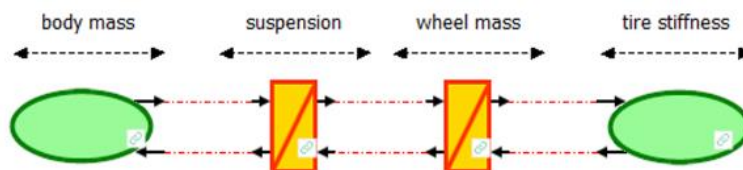


Fig. 4. Numele componentelor evidențiate

Determinați rolurile I / O.

În acest pas, trebuie să determinăm tipul de intrare sau ieșire pentru fiecare variabilă de interconectare din diagrama EMR.

După cum s-a menționat în explicațiile din exercițiul 3, începem cu atribuirea rolurilor pentru variabilele elementelor de acumulare, deoarece acestea sunt fixate. La pasul următor, atribuim și rolurile pentru elementele sursă.

Rolurile I / O sunt selectate astfel încât să respecte cauzalitatea fizică integrală, adică ieșirea depinde doar de propria evoluția din trecut sau / și de semnalul de intrare. Acest fenomen îl vom descrie cu ecuațiile modelului pentru fiecare componentă.

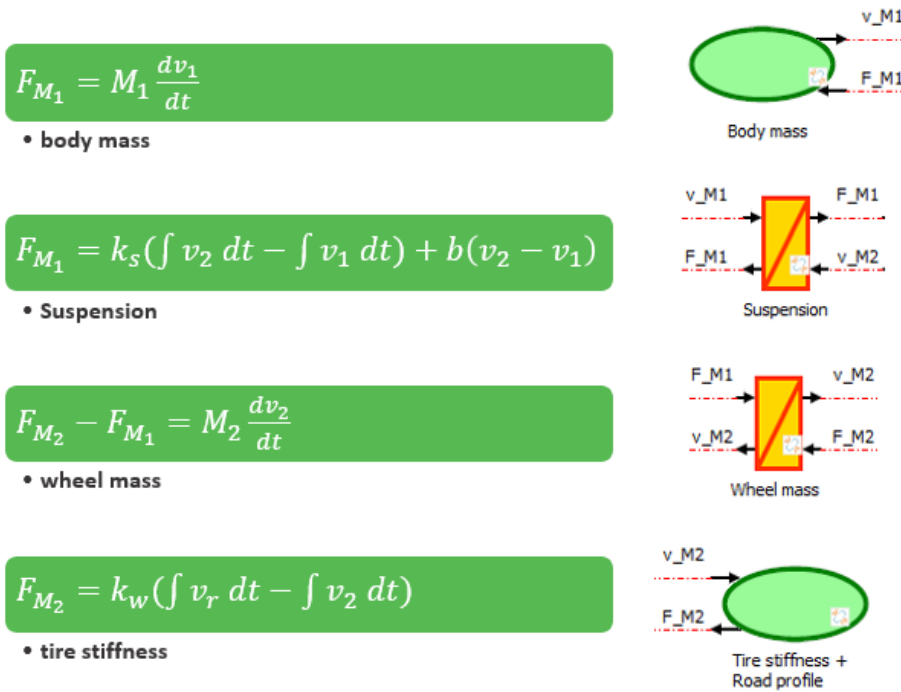


Fig. 5. Alocarea rolului variabilelor

Diagrama EMR rezultată a întregului sistem ar trebui să arate ca mai jos.

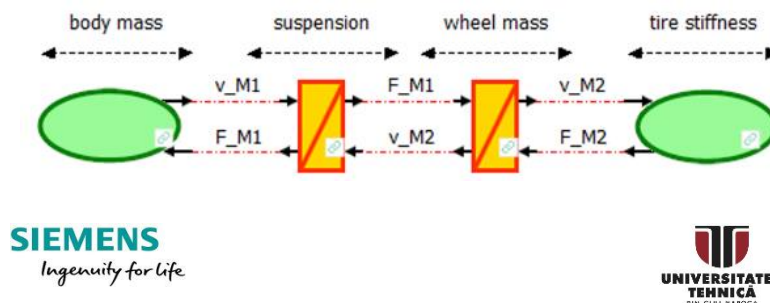


Fig. 6. Diagrama EMR pentru sistemul de suspensie

În figura de mai sus, observăm cum conexiunea directă a componentelor era posibilă, deoarece rolul I / O al fiecărei componente se potrivea la interconectări. Conflictele de asociere vor fi studiate în exercițiul 6.

Popularea cu conținut a blocurilor EMR

Acum, având arhitectura EMR decisă, vom implementa ecuațiile model care determină variabilele de sistem. Pentru fiecare element EMR, putem implementa ecuațiile modelului utilizând biblioteca de semnal și control a Simcenter Amesim.

Pentru a accesa implementarea componentelor, apăsăm clic dreapta pe componentă și selectăm **Open Supercomponent**. Această acțiune ne duce la spațiul de lucru al componenteii. Putem reveni la spațiul de lucru al sistemului apăsând în colțul din stânga sus al spațiului de lucru.

În interiorul unui element EMR găsim terminalele de semnal care se conectează la variabilele de interconectare a elementelor. Terminalele de semnal sunt numerotate, reflectând ordinea lor în afara elementului.

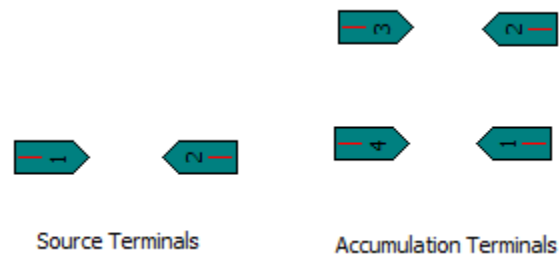


Fig. 7. Secvența (direcția) semnalului în interiorul unui element EMR

Făcând clic stânga pe o componentă EMR din diagrama noastră, provocam afișarea numerotării terminalelor.

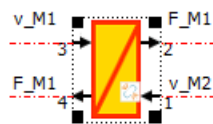


Fig. 8. Afișarea numerotării terminalelor unui element EMR

Recunoașterea variabilelor la terminale este un pas necesar înainte de a implementa ecuațiile modelului.

Putem astfel folosi instrumentul **Insert Text** pentru a scrie numele variabilelor în interiorul componentei lângă terminalele corespunzătoare.

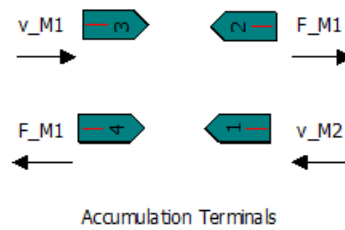


Fig. 9. Exemplu de numire a terminalelor

Acum suntem gata să implementăm ecuațiile modelului folosind biblioteca **Signal and Control**. Implementarea cu ajutorul bibliotecii de semnal și control a ecuației modelului este o implementare tip diagramă bloc. Biblioteca poate fi găsită în arborele bibliotecii din partea dreaptă a spațiului de lucru, când este selectat modul **Sketch**.

Așa cum am menționat mai sus, elementele bibliotecii sunt selectate folosind clic stânga și plasate pe diagramă (drag&drop).

Apoi, interconectarea elementelor se face făcând clic stânga pe un terminal de element și apoi din nou cu clic stânga pe terminalul la care urmează să se facă conexiunea.

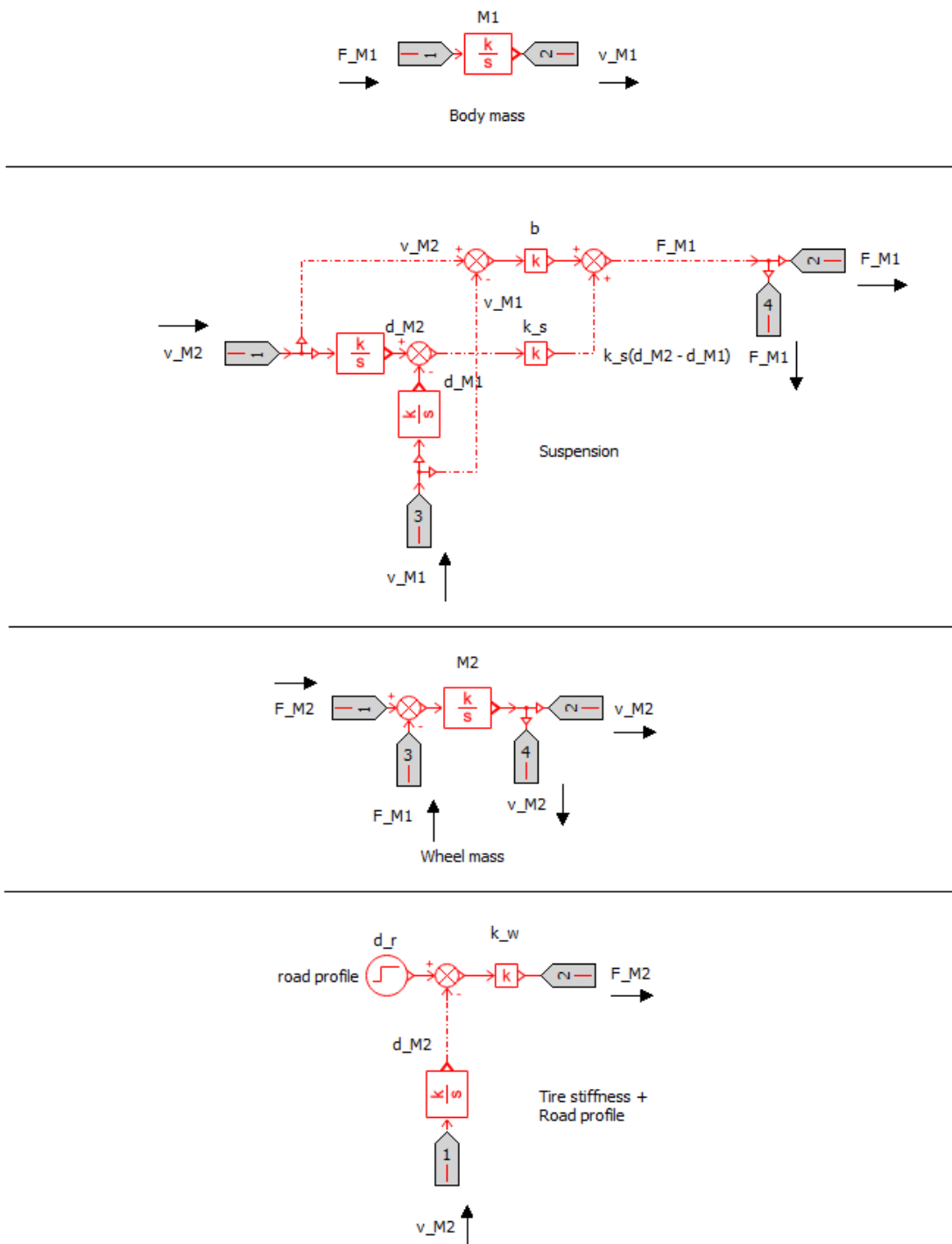








Fig. 10. Modelele tip schemă bloc pentru fiecare element EMR

În figura de mai sus, este prezentată implementarea diagramei bloc pe baza ecuațiilor aferente modelului pentru fiecare element EMR.

Elementele utilizate din biblioteca de semnal și control sunt date mai jos:

Componentă	Nume (în Engleză)	Identificator
	Gain	GA00
	Integrator	INT0
	Splitter junction	SPLT0
	Step function	STEP0
	Subtracting junction	JUN3M
	Comparison junction summing both inputs	JUN3P

Tabloul 3. Elementele utilizate din biblioteca de semnal și control

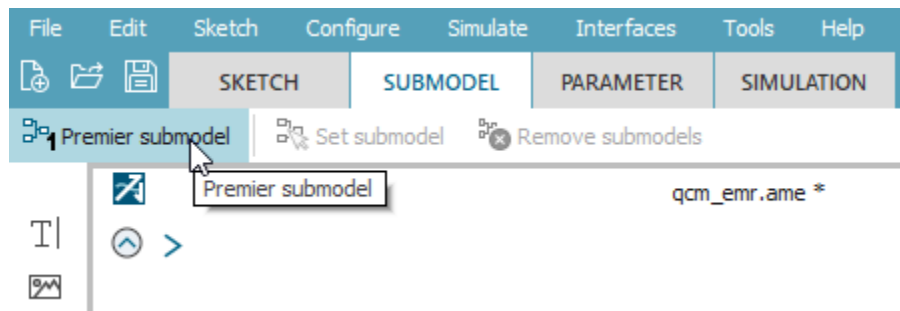
Odată ce modelele dinamice pentru toate elementele EMR ale modelului sunt implementate, completăm diagrama EMR a sistemului nostru.

Pentru a calcula deplasarea caroseriei și a roților pe o treaptă de drum, trebuie să furnizăm parametrii modelului.

Setarea parametrilor modelului

Înainte de a seta parametrii, trebuie să intrăm în modul submodel, făcând clic pe **Submodel**, în filele **Mode**.


În bara de instrumente **Mode-contextual toolbar**, trebuie să selectăm **Premier submodel**.

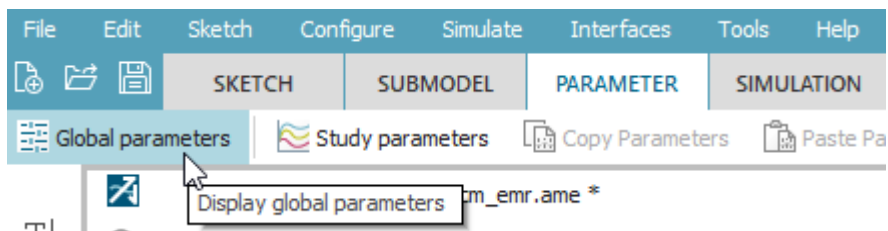


Simcenter Amesim oferă, pentru unele componente, diferite submodele care pot fi alese. Cu submodelul Premier, selectăm pentru toate elementele din spațiul de lucru, cele mai de bază submodele.

Odată ce submodelele au fost configurate, putem trece la modul **Parameter mode**, selectând **Parameter** în filele **Mode**.

Pentru a facilita gestionarea parametrilor, vom folosi instrumentul **Global parameters** a Simcenter Amesim.

1. Accesati sapteul de lucru al modelului de simulare apăsând butonul  >
2. Apăsați **Global parameters**, în bara de unelte **Mode**.



3. In the **Global parameters setup** window, right-click on the workspace and select **New > Real parameter**.
4. În fereastra **Global parameters setup**, apăsați click dreapta și selectați **New > Real parameter**
5. La numele parametrului scrieți "M1", la titlu scrieți "Body mass [kg]" iar la valoare scrieți 400.
6. Pentru restul parametrilor procedați la fel, conform figurii de mai jos.

Acum avem toți parametrii necesari accesibili prin fereastra **Global Parameters**. Orice modificare a parametrilor ar putea fi gestionată în viitor din această fereastră.

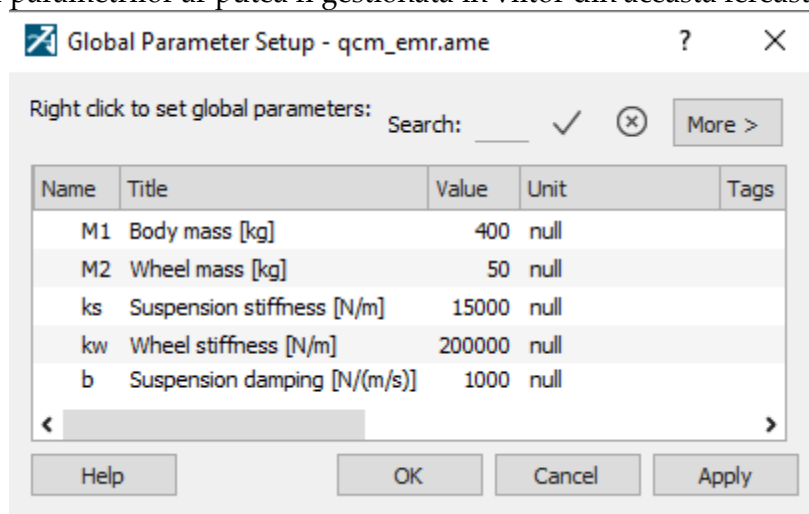
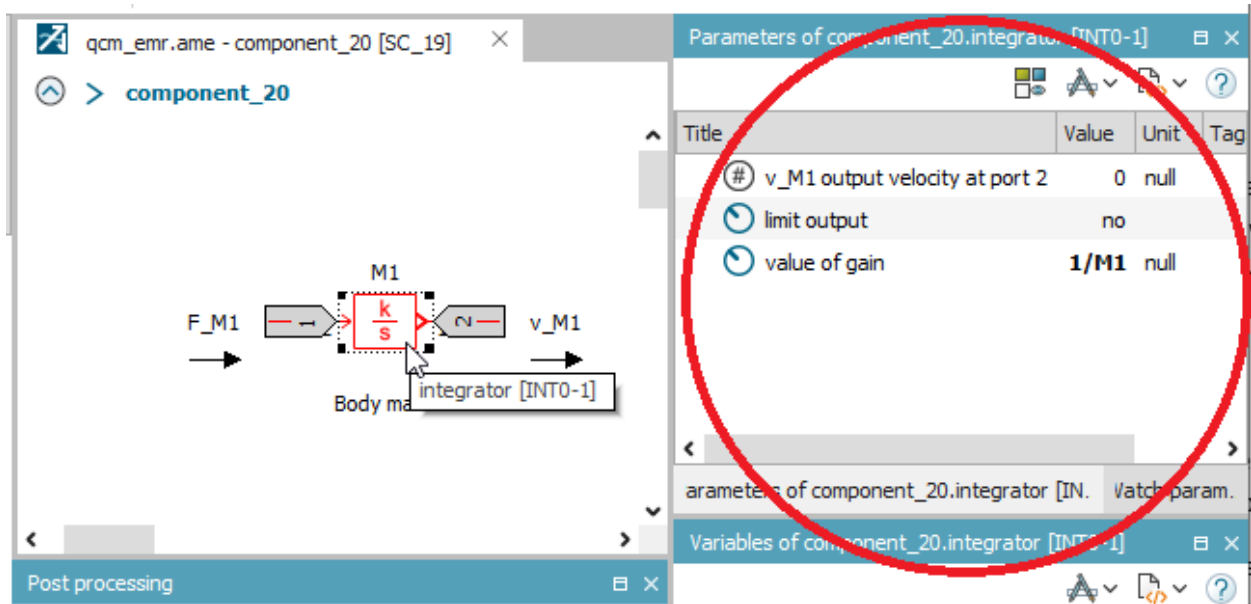


Fig. 11. Fereastra Global Parameters.

Acum vom parametriza schema bloc a sistemului nostru de implementat.

Pentru a modifica sau seta parametrii pentru elementele din biblioteca de semnal și control, apăsați pe pictograma elementului din spațiul de lucru, în modul **Parameter**. Fereastra de parametri, unde aceștia pot fi modificați sau setați este afișată în dreapta.



Setați următorii parametri pentru elementele schemei bloc enumerate mai jos:

Element	Componentă	Parametru	
		Nume (En)	Valoare
Greutate caroserie		value of gain	1/M1
Suspensie		value of gain	b
		value of gain	ks
Greutate roată		value of gain	1/M2
Rigiditate roată		value of gain	kw
		value after step	0.1
		step time	1

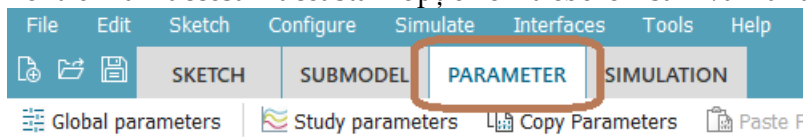
Tabelul 4. Parametrii elementelor schemei bloc.

De fiecare dată când trebuie modificată o valoare a parametrului, aceasta poate fi realizată utilizând caracteristica **Global parameters**. Declarând variabila acolo, se vor modifica automat parametrii din interiorul elementelor schemei bloc, facilitând astfel administrarea parametrilor.

Gestionarea parametrilor.

Pentru a vizualiza parametrii la nivel local, folosim funcționalitatea „Exposed Parameters” în Simcenter Amesim. Cu această funcționalitate putem vizualiza parametrii interni ai elementelor schemei bloc la nivelul elementelor EMR. În acest fel, vom asocia cu fiecare element EMR un set de parametri, care sunt accesibili la nivelul diagramei EMR.

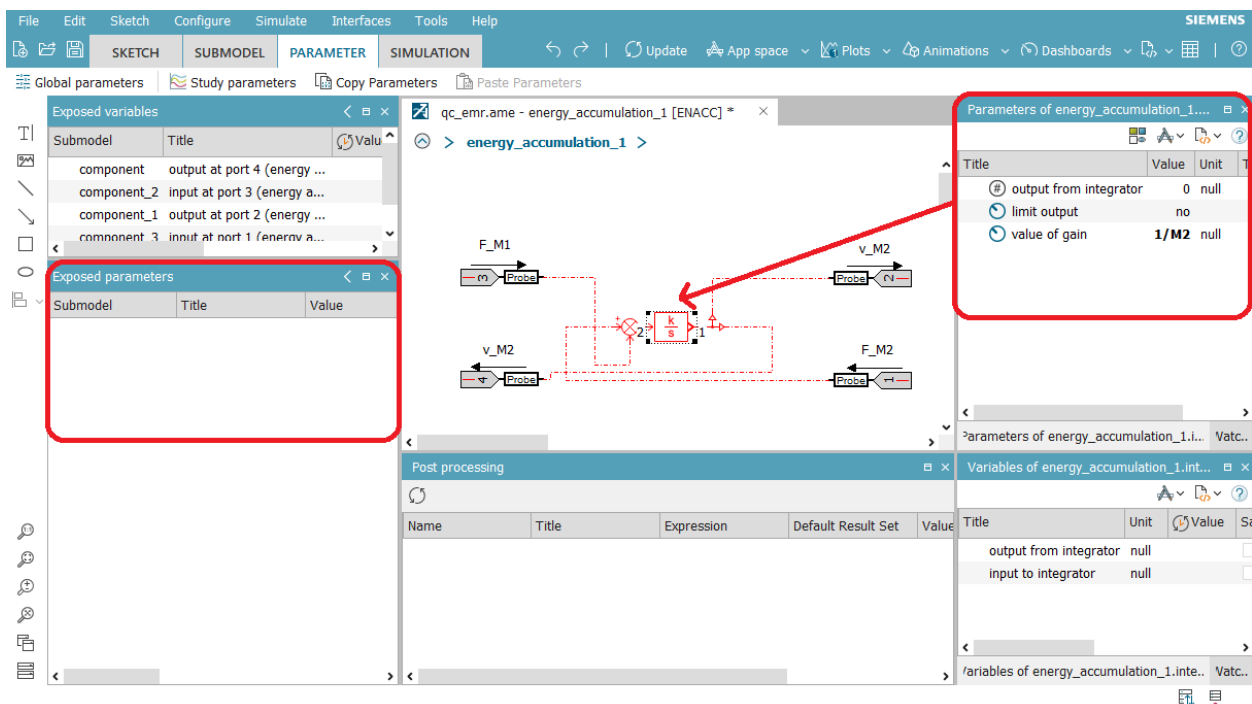
1. Pentru a accesa această opțiune trebuie să vă aflați în fila „Parameter”.



2. Apăsați clic dreapta pe funcția EMR dorită din zona Sketch și selectați „Open supercomponent”.

Puteți trage și plasa parametrii elementelor schemei bloc, de la fereastra „Parameters of ...” a elementului găsit în dreapta, până la fereastra „Exposed parameters” din stânga.

În imaginea de mai jos am selectat elementul integrator și putem vedea în dreapta parametrii săi. Putem trage și plasa, de exemplu, parametrul „value of gain” în fereastra „Exposed parameters” din stânga.









Notă! Dacă fereastra „Exposed parameters” nu este vizibilă, ar putea fi închisă. Mai întâi, asigurați-vă că vă aflați în elementul EMR ales folosind butonul „Open supercomponent”. Pentru a deschide această fereastră accesați **Edit>Show/Hide** și selectați „Supercomponent exposed parameters”.

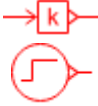
3. Dacă revenim la nivelul diagramei EMR, parametrii interni sunt vizibili în „fereastra Parameters” prin selectarea elementului EMR.

Folosind funcționalitatea variabilă expusă putem gestiona parametrii mai ușor, făcând parametrii fiecărui element EMR accesibil la nivelul diagramei EMR.

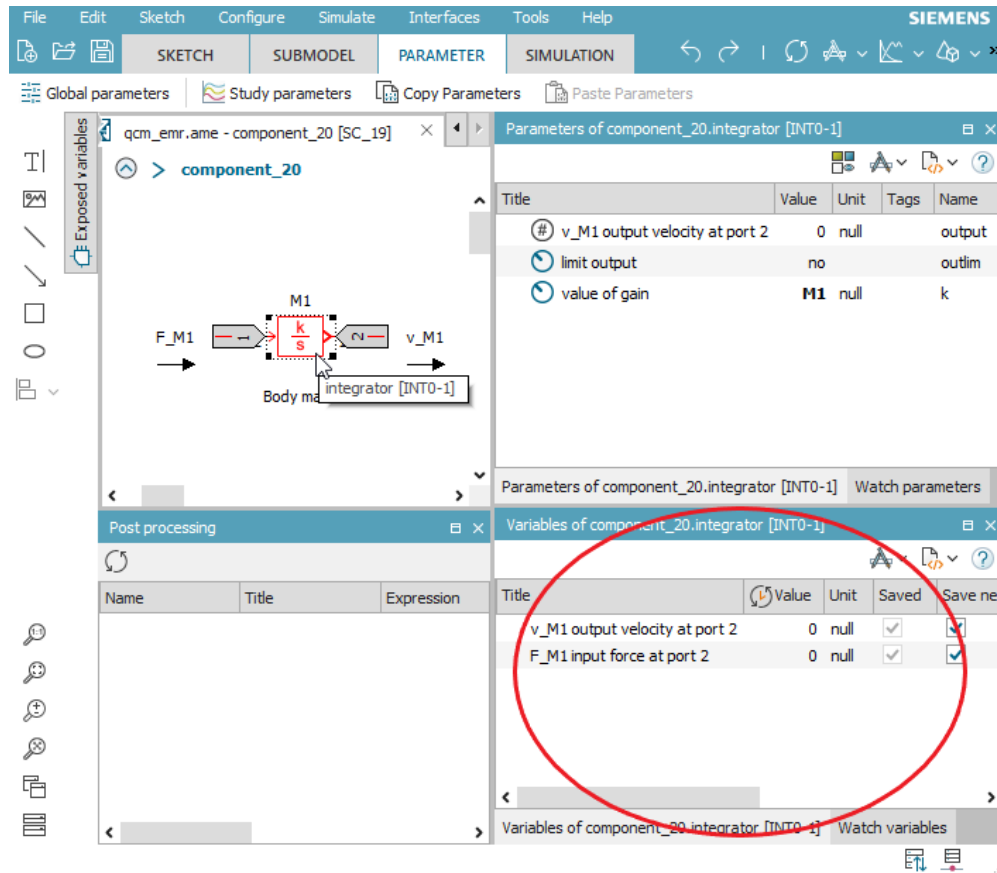
Gestionarea variabilelor

Mai multe variabile de sistem sunt prezente în diagrama noastră de simulare. Este de interes să le putem avea ușor accesibile pentru orice verificare. Putem accesa variabilele de sistem apăsând clic dreapta pe un element de diagramă bloc din diagrama noastră, în timp ce modul **Parameter** este activ. Fereastra **Variables** va fi vizibilă în colțul din dreapta jos. Pentru o mai bună recunoaștere, vom schimba denumirea variabilelor așa cum este indicat în tabelul de mai jos:

Subsistem	Componenta	Variabile	
		Denumire veche (EN)	Denumire nouă (EN)
Greutatea Caroseriei		output from integrator	v_M1 output velocity at port 2
		input to integrator	F_M1 input force at port 2
Suspensia		input at (-ve) port 1	v_M2 input velocity at port 2
		input at (+ve) port 3	v_M1 input velocity at port 1
	output from summing junction	F_M1 output force at port 1	
	input to junction	F_M1 output force at port 2	
		output from integrator	d_M1 displacement of the body
	output from integrator	d_M2 displacement of the wheel	
Greutatea roții		input at (-ve) port 1	F_M2 input force at port 2
		input at (+ve) port 3	F_M1 input force at port 1
	output from integrator	v_M2 output velocity at port 1	
	input to junction	v_M2 output velocity at port 2	
		input to integrator	v_M2 input velocity at port 1

Rigiditatea roții		output from gain step output	F_M2 output force at port 1 d_r road profile
----------------------	---	---------------------------------	---

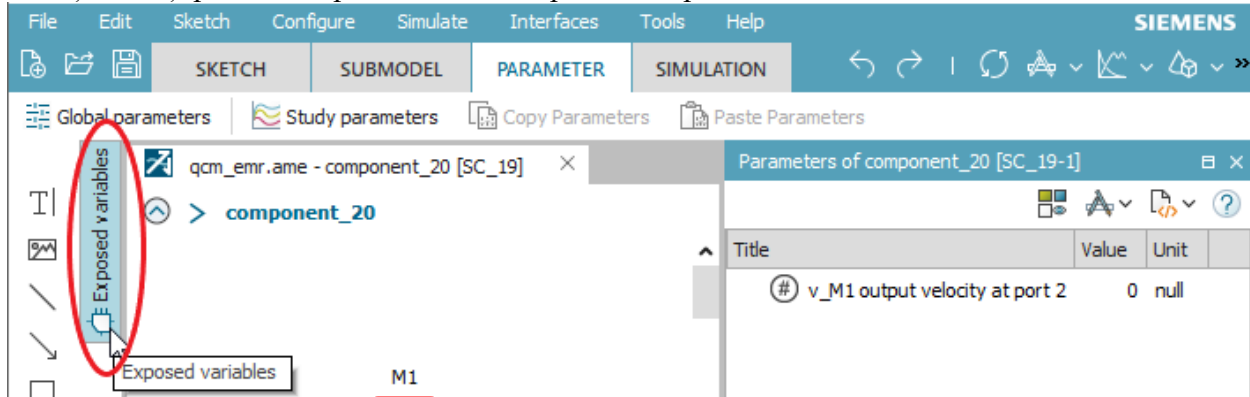
Tabelul 5. Redenumiri de variabile





Vom expune aceste variabile ca variabile supercomponente pentru fiecare element EMR. În acest fel, vom putea accesa aceste variabile din elementele EMR, fără a fi nevoie să parcurgeți implementarea diagramei bloc.

1. În timp va aflați în spațiul de lucru al elementului EMR, când *Parameters mode* este activ, deschideți fereastra **Exposed variables** apăsând butonul *Exposed variables* aflat în stânga spațiului de lucru.
2. Apăsați clic dreapta pe o componentă a diagramei bloc pentru care am redenumit titlul variabilei.
3. Trageți și plasați variabilele componente, cu titlul variabilei redenumite, din fereastra „Variables”, în fereastra „Exposed variables”.
4. Faceți aceeași procedură pentru toate componentele pentru care am redenumit titlurile variabilei.

Faceți aceeași procedură pentru toate componentele pentru care am redenumit titlurile variabilei.



1. Mergeți la nivelul EMR al simulării apăsând  , dacă nu sunteți deja acolo.
2. Click dreapta pe componenta EMR cât timp sunteți în modul *Parameter* activ.

În fereastra **Variables**, în colțul din dreapta jos, variabilele de interes sunt acum vizibile pentru elementele EMR.


S-a obținut astfel vizibilitatea variabilelor care caracterizează interacțiunile EMR la nivelul sistemului, facilitând astfel gestionarea lor. Acum, variabilele sunt accesibile făcând clic dreapta pe elementele EMR și sunt grupate în funcție de organizarea EMR.

Rularea simulării

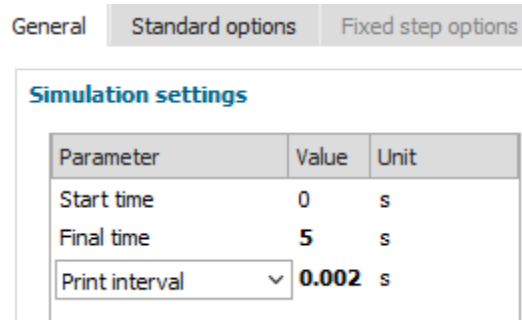
Vom rula o simulare, pentru a prezice deplasarea caroseriei și a roții pe un profil de rulare tip treaptă.


Pentru a rula o simulare:


1. Apăsați clic pe butonul **Simulation**.
2. Va apărea o fereastră numită "System compilation" care va arăta rezultatele simulării.

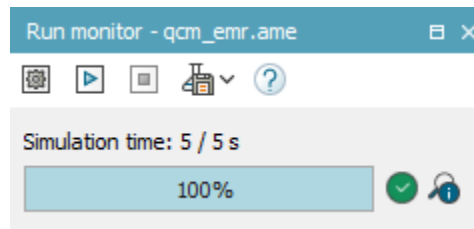
Apăsați butonul **Run parameters**  **Run Parameters**, din bara de unelte în modul Mode-contextual.

Se va deschide fereastra cu parametrii de rulare. Aici putem seta valorile lor pentru simularea dorită, cum ar fi timpul de simulare, algoritmul de calcul și pasul de simulare etc. Setați timpul final la 5 s, iar intervalul de afișare la 0,002 s.



3. Apăsați butonul **Run simulation**  **Run simulation**, din bara de unelte din modul Mode-contextual.

Simularea este rulată. Dacă se finalizează cu succes, va fi indicat prin bifa  din fereastra **Run monitor**, iar variabilele de simulare vor fi actualizate și pregătite de utilizare.



Pentru a afișa rezultatele:

1. Faceți clic pe componenta EMR pentru rigiditatea anvelopelor.
2. Trageți variabila d_r pe planul de lucru, din fereastra Variabile.
3. Faceți clic pe elementul Suspension EMR.
4. Trageți variabilele d_{M1} și d_{M2} pe același grafic ca d_r , din fereastra Variabile.

Rezultatul afisării este cel de mai de jos.

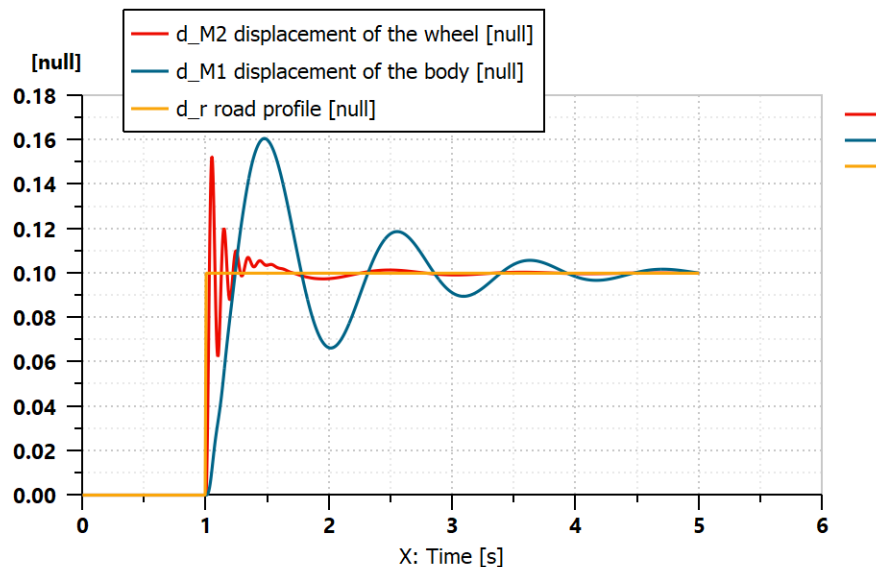


Fig. 12. Deplasarea roții și a caroseriei pentru un anumit profil de drum

Rezultatele arată că o treaptă în profilul de drum produce răspunsuri oscilante în deplasările roții și ale caroseriei. Folosind simularea, putem aprecia amplitudinea și frecvența oscilațiilor. Frecvența oscilației este mai mare pentru deplasarea roții decât pentru caroserie. Atât roata, cât și caroseria revin la starea de echilibru după o deplasare de 0,1 [m], din profilul drumului, în aproximativ 4 secunde.

Concluzii

Biblioteca EMR poate fi utilizată cu succes pentru a dezvolta simulări EMR, utilizând Simcenter Amesim, care afișează descrierea funcțională a sistemului, precum și interacțiunea fizică dintre elementele sistemului. Acest lucru ne permite să evaluăm mai bine proprietățile globale ale sistemului, adică deplasarea caroseriei și a roții, pe baza interacțiunilor elementelor.

Exercițiul 5 – Compararea organizării EMR cu diagramele bloc clasice

Obiectiv: În acest exercițiu vom afla despre avantajele dezvoltării simulărilor folosind EMR, comparându-l cu diagrama bloc.

Sistemul studiat și diferite descrieri funcționale

Vom lua în considerare același sistem de suspensie ca în exercițiul 4, care are structura prezentată în diagrama de mai jos:

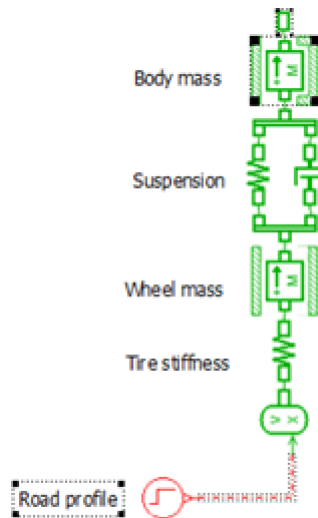


Fig. 1. Descrierea structurală a suspensiei vehiculului.

EMR-ul acestui sistem, dezvoltat în exercițiul 4, are structura dată în Fig. 2 (descriere funcțională).

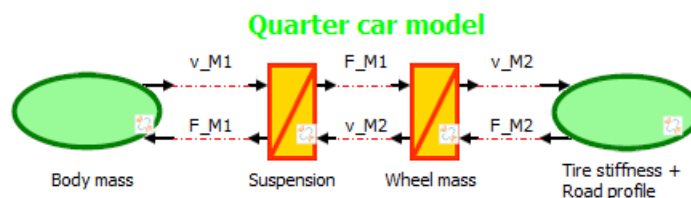


Fig. 2. EMR al suspensiei vehiculului

În Fig.3 este detaliată o diagramă bloc a sistemului, fiind o descriere funcțională mai clasică.

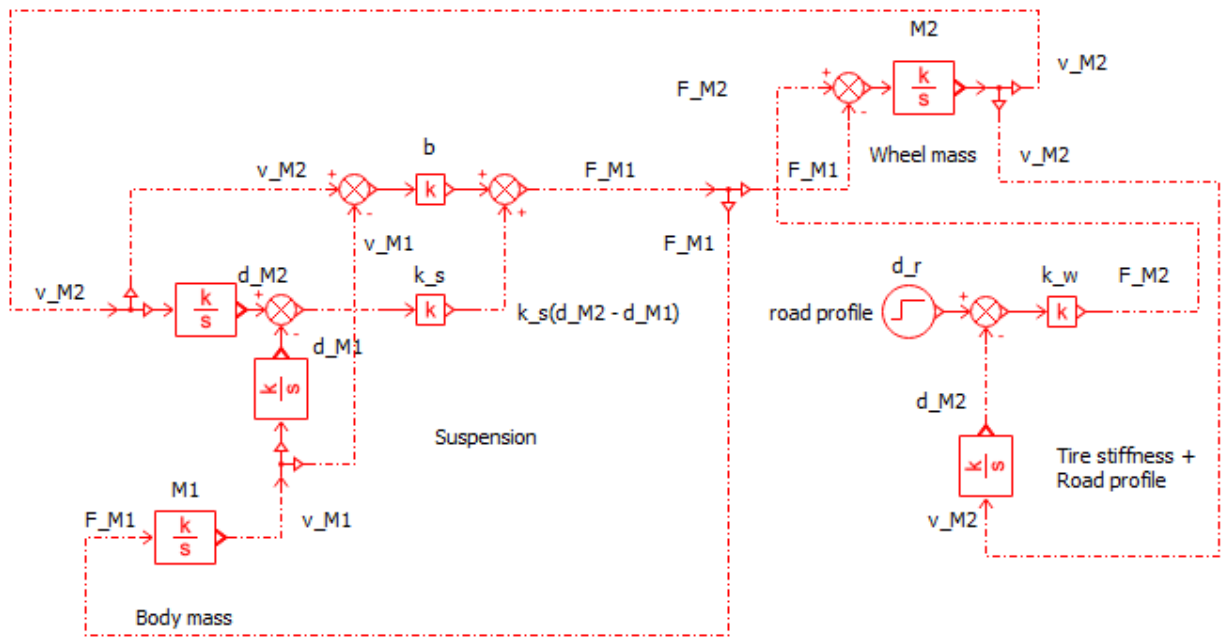


Fig. 3. Schema bloc a suspensiei vehiculului.

Această diagramă bloc a fost dezvoltată utilizând conținutul elementelor EMR ținând cont de interconectările corecte ale componentelor în conformitate cu diagrama EMR.

Compararea descrierilor funcționale

Comparând cele două tipuri de organizări, EMR prezintă:

- subcomponente mai bine transparente de distins,
- ilustrează mai bine interacțiunile dintre componente,
- simplifică diagrama *ascunzând* implementarea efectivă în blocurile funcționale energetice.

Acest lucru permite o gestionare mai ușoară a componentelor sistemului.

Există multe modalități diferite de a organiza diagramele bloc. Să luăm în considerare un alt context. O altă diagramă bloc ar putea fi derivată luând în considerare ecuațiile sistemului (componentelor) și eliminând variabilele de interconectare.

Modelul de suspensie auto este reprezentat prin următoarele ecuații:

$$F_{M_1} = M_1 \frac{dv_1}{dt} \quad (\text{masa caroseriei}) \quad (4.1)$$

$$F_{M_1} = k_s(d_2 - d_1) + b(v_2 - v_1) \quad (\text{suspensia}) \quad (4.2)$$

$$F_{M_2} - F_{M_1} = M_2 \frac{dv_2}{dt} \quad (\text{masa roții}) \quad (4.3)$$

$$F_{M_2} = k_w(d_r - d_2) \quad (\text{rigiditatea roții}) \quad (4.4)$$

Variabilele sunt:

F_{M_1} = forța asupra caroseriei

F_{M_2} = forța la roată

v_1 = viteza verticală a caroseriei

v_2 = viteza verticală a roții

d_1 = deplasarea caroseriei

d_2 = deplasarea roții

d_r = profilul de drum

Iar parametrii sunt:

M_1 = masa caroseriei

M_2 = masa roții

k_s = rigiditatea suspensiei

k_w = rigiditatea roții

b = atenuarea suspensiei

Prin eliminarea forțelor din ecuațiile sistemului, am obține un model care conține doar deplasări și viteze. Ecuațiile model rezultate sunt

$$M_1 \frac{dv_1}{dt} = k_s(d_2 - d_1) + b(v_2 - v_1),$$

$$M_2 \frac{dv_2}{dt} = k_w(d_r - d_2) - k_s(d_2 - d_1) - b(v_2 - v_1).$$

Exprimarea acestor ecuații numai în variabilele d_1, d_2, d_r și aplicarea transformatei Laplace duce la o reprezentare a funcției de transfer a relației dintre variabile.

$$d_1(s) = G_1(s)d_r(s)$$

$$d_2(s) = G_2(s)d_r(s)$$

$$G_1(s) = \frac{B(s)}{A(s)}$$

$$G_2(s) = \frac{C(s)}{A(s)}$$

Aici, $B(s), C(s), A(s)$ sunt polinoame în variabila complexă s :

$$B(s) = \sum_{i=0}^1 b_i s^i$$

$$C(s) = \sum_{i=0}^2 c_i s^i$$

$$A(s) = \sum_{i=0}^4 a_i s^i$$

$$b_0 = k_w k_s$$

$$b_1 = k_w b$$

$$c_0 = k_w k_s$$

$$c_1 = k_w b$$

$$c_2 = k_w M_1$$

$$a_0 = k_w k_s$$

$$a_1 = k_w b$$

$$a_2 = (k_w + k_s)M_1 + k_s M_2$$

$$a_3 = b(M_1 + M_2)$$

$$a_4 = M_1 M_2$$

O reprezentare în diagramă bloc a ecuațiilor este dată în Fig. 4, care este strict echivalentă cu cea din Fig. 3, dar descrisă într-un mod mai compact.

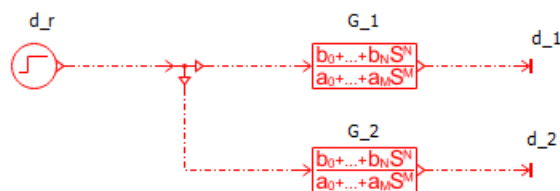


Fig. 4. Schema bloc a suspensiei vehiculului fără forțele luate în considerare.

Noua diagramă bloc nu modelează interacțiunea fizică dintre componente.

Cu toate acestea, această reprezentare arată că, funcțiile de transfer $d_r \rightarrow d_1$, $d_r \rightarrow d_2$ au aceiași poli. În plus, această reprezentare a schemei bloc este, de asemenea, mai compactă decât EMR.

Aceleași observații în favoarea EMR se mențin, așa cum s-a menționat pentru prima diagramă bloc introdusă, făcând EMR mai lucrativ pentru gestionarea modelelor pentru sisteme multi-domeniu complexe.

Concluzii

Prin furnizarea unor scheme de simulare cu componente distincte care interacționează fizic prin transferul de energie, EMR este potrivit pentru modelarea și analiza sistemelor complexe multi-domeniu și evidențierea mai bună a interacțiunilor subsistemelor.

Exercițiul 6 – Conflicte de asociere

Obiectiv: În acest tutorial vom învăța cum să rezolvăm conflictele de asociere, atunci când conectăm componente EMR.

Noțiunea de conflict de asociere

Să reamintim că EMR se ocupă de cauzalitatea fizică (integrală) exclusivă. Toate I / O sunt astfel impuse de elemente de acumulare, definite în funcție de această cauzalitate naturală. Cu toate acestea, unele asociații de componente duc la I / O incompatibile într-o descriere a funcției. Această situație se numește **conflict de asociere**.

În EMR, interconectarea elementelor se face prin conectarea variabilelor de element adecvate pentru a forma o legătură de putere. Interconectarea directă între două subsisteme S1 și S2 se realizează atunci când relațiile:

$$\text{In (S1)} = \text{Out(S2)}$$

$$\text{Out(S1)} = \text{In(S1)}$$

sunt valide.

Există situații în care interconectarea directă a componentelor eșuează, având în vedere tabelul de intrare-ieșire al fiecărui element. În EMR, tabelul de intrare-ieșire al elementelor de acumulare este fix și, ca rezultat, prezintă o provocare atunci când apare un conflict de asociere. Metoda EMR evidențiază conflictul de asociere (I / O incompatibile) și trebuie definite proceduri specifice pentru rezolvarea acestei probleme.

Example in Mechanics

În acest exercițiu, demonstrăm soluțiile EMR la conflictele de asociere, considerând un exemplu de sistem mecanic.

Avem sistemul format din doi arbori conectați printr-o cutie de viteze ca în figura de mai jos.

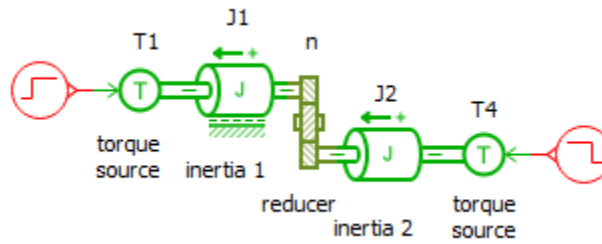


Fig. 1. Descrierea structurală a unei linii mecanice de acționare.

Acest model nu poate fi construit folosind schița Simcenter Amesim obișnuită. Reprezentarea EMR a fiecărei componente separat și a ecuațiilor acestora este cea prezentată mai jos.

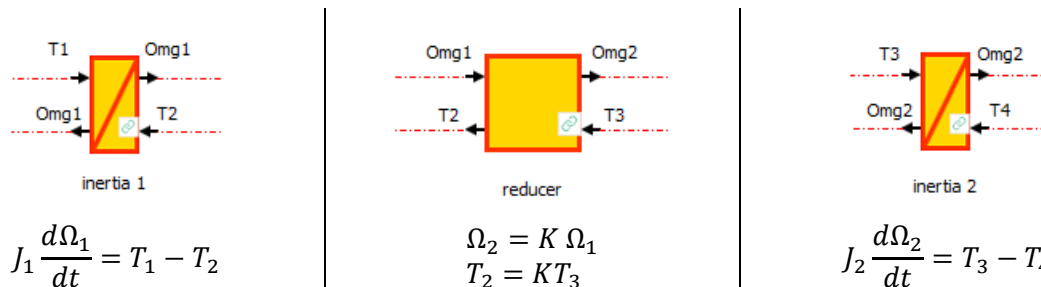


Fig. 2. Blocurile EMR și ecuațiile componentelor.

După cum se poate vedea, intrările și ieșirile nu se potrivesc la legătura reductorului cu inerția 2.

Există diferite soluții pentru rezolvarea acestui conflict:

- Exprimați inerția 2 în forma sa derivată (nu este permisă în EMR din cauza lipsei de cauzalitate fizică)
- Adăugați o rigiditate între cutia de viteze și inerția 2 (dar nu fizică atât timp cât nu există o rigiditate reală, însă asta va duce la creșterea timpului de calcul).
- înlocuiți descrierea bazată pe structură prin funcție echivalentă fără conflict de asociere (aceasta va duce la păstrarea cauzalității fizice reale; **soluție utilizată în filosofia EMR**).

Astfel, pentru a rezolva acest conflict, EMR are reguli de asociere specifice. În acest exemplu, trei transformări ale interconectărilor subsistemelor sunt utilizate pentru a rezolva conflictele de asociere

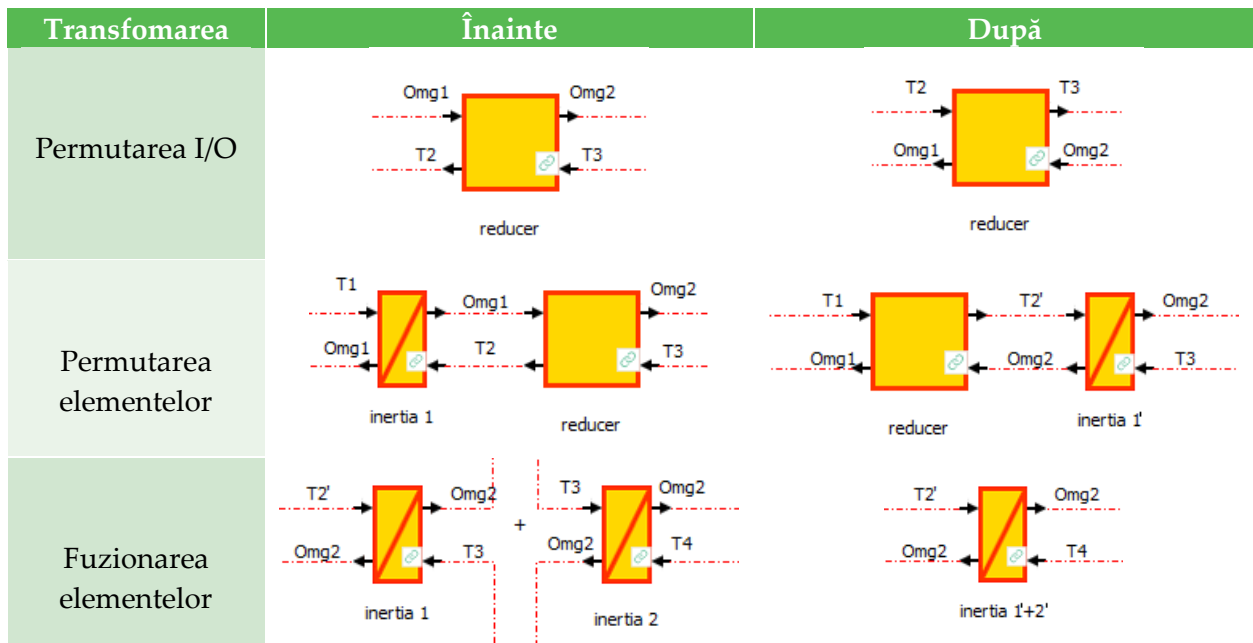
Acestea sunt:

- Permutări intrare-ieșire pentru elemente care nu se acumulează. Elementele care nu sunt de acumulare au I / O flotante și astfel variabilele pot permuta rolul de intrare / ieșire pentru a obține aceeași reprezentare.

- Permutări ale elementelor. Unele elemente pot fi pozițional permute, menținând în același timp fixă interconectarea externă, redefinind noi relații între variabile.

- Fuzionarea elementelor. Unele elemente de acumulare adiacente ar putea fi îmbinate pentru a forma un singur element echivalent (funcție globală echivalentă).

Aceste operații sunt ilustrate mai jos pentru exemplul nostru.



Tabelul 1. Operațiuni pentru rezolvarea conflictului de asociere

Pentru a rezolva conflictul de asociere în exemplul de inerție cu două zone de rotație:

1. Efectuați o permutare a elementelor între inerția 1 și cutia de viteze.
2. Efectuați fuziunea dintre cele două, acum adiacente, inerții.

Sistemul EMR rezultat este prezentat în Fig. 3.

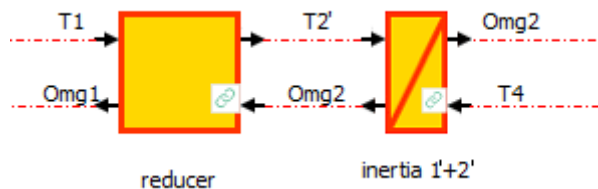


Fig. 3. EMR complet al sistemului de inerție cu două rotații.

În acest caz, trebuie definită o nouă inerție echivalentă (inerție 1 + 2):

$$J_{eq} \frac{d\Omega_2}{dt} = T_{2'} - T_4 \text{ with } J_{eq} = \frac{J_1}{K^2} + J_2$$

EMR-ul nou, complet, al celor două sisteme de inerție de rotație, reprezintă funcțiile energetice ale sistemului original și permite interconectarea acestuia ca parte a unui sistem mai complex (mai mare), cu interacțiuni fizice modelate pe baza schimbului de energie.

Puncte de vedere generale

Astfel, acest sistem structural cu 3 componente ar trebui descris prin 2 funcții echivalente. Conflictele de asociere sunt principala diferență între descrierea structurală și cea funcțională.

În software-urile structurale, conflictele de asociere vor fi rezolvate de către programul de integrare numerică (solver). Aceasta duce la o simulare prietenoasă, dar la o lipsă de înțelegere fizică.

În simularea bazată pe EMR, conflictele de asociere vor fi rezolvate de utilizatori înainte de implementarea în software-ul de simulare. Astfel se atinge o bună înțelegere a comportamentului fizic, dar necesită timp suplimentar de dezvoltare. Mai mult, va permite o simulare mai rapidă din cauza rezolvării prealabile a problemei.

Conclusions

EMR utilizează transformări de interconectare a elementelor pentru a rezolva conflictul de asocieri și pentru a obține modele de sistem în care semnalele sunt ordonate cauzal. Această caracteristică a EMR ne permite să modelăm o gamă mai largă de sisteme fizice și să producem simulări care influențează proiectarea sistemului nostru.

Exercițiul 7 – Dezvoltarea controlului prin inversiunea modelului EMR

Obiectiv: În acest exercițiu vom învăța cum să dezvoltăm structura de control a sistemelor complexe multi-domeniu folosind EMR și controlul bazat pe inversare în Simcenter Amesim.

Principiul controlului bazat pe inversiune

Structura sistemului de control bazată pe inversiune are la bază inversarea modelului în forma EMR a sistemului considerat. O astfel de schemă de control poate fi dedusă doar din organizarea modelului, dacă aceasta este dezvoltată în funcție de cauzalitatea integrală.

Controlul sistemelor de propulsie este un factor important în dezvoltarea vehiculelor electrice eficiente. Complexitatea modelului poate fi o povară pentru inginer, totuși descrierea funcțională cauzală a lui ar putea fi utilizată pentru a simplifica problema controlului. EMR oferă o organizare adecvată a sistemului prin control bazat pe inversare. Controlul bazat pe inversiune este o metodă sistematică de organizare a sistemelor complexe de control utilizând o structură distribuită. Structura de control rezultată este transparentă și ușor de gestionat, deoarece se bazează pe construcția și deci automat reglajul progresiv a ei.

Vom ilustra metoda de control bazată pe inversiune pentru un vehicul electric cu baterie (BEV) propulsat de un motor DC. Structura sistemului de propulsie realizat în EMR este dat în Fig. 1.

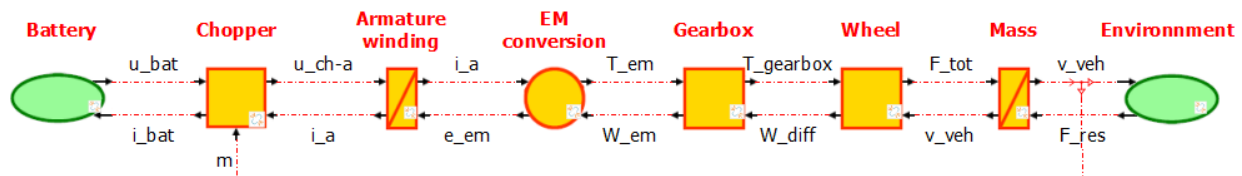


Fig. 1. Structura EMR a propulsiei unui vehicul electric cu baterie

Obiectivul controlului este urmărirea profilului de referință al vitezei vehiculului $V_{veh-ref}$. Viteza vehiculului putea fi controlată printr-un variator de tensiune continuă. Prin modificarea variabilei de modulație m a acestuia (numită variabilă de reglare a sistemului), ajustăm puterea livrată de la baterie spre motorul de propulsie.

Schema de control bazată pe inversiune rezultă din organizarea EMR a modelului de sistem. Urmărind schema model a sistemului, arhitectura de control este implementată prin inversiunea acțiunii fiecărui bloc EMR.

Mai precis, având în vedere că dorim ca V_{veh} să urmărească un anumit profil de referință $V_{veh-ref}$, construim un regulator „Mass controller” pentru a da comanda forța F_{tot} corespunzătoare elementului „Masă”. După cum vedem în diagrama din Fig. 1, nu putem furniza variabila de control F_{tot} elementului „Masă” direct de la controler. Ca rezultat, definim variabila de control pentru F_{tot} ca un profil de referință $F_{tot-ref}$, pentru intrarea F_{tot} a elementului „Roată”. Apoi, continuăm să construim un regulator „Wheel controller” pentru elementul „Wheel”, în mod similar. Continuăm să construim regulatoare în acest mod până când variabila de reglare poate fi aplicată direct la convertorul electronic care controlează tensiunea aplicată motorului de propulsie.

Regulatorul bazat pe inversiune are reguli clare cu privire corespondența blocurilor de control vis a vis de cele din modelul sistemului. Fie că discutăm de control în buclă deschisă sau închisă, regulile rămân valide și automat se menționează și care sunt cantitățile măsurate (în cazul regulatorului în buclă închisă)

Aceste reguli sunt rezumate în figura de mai jos.

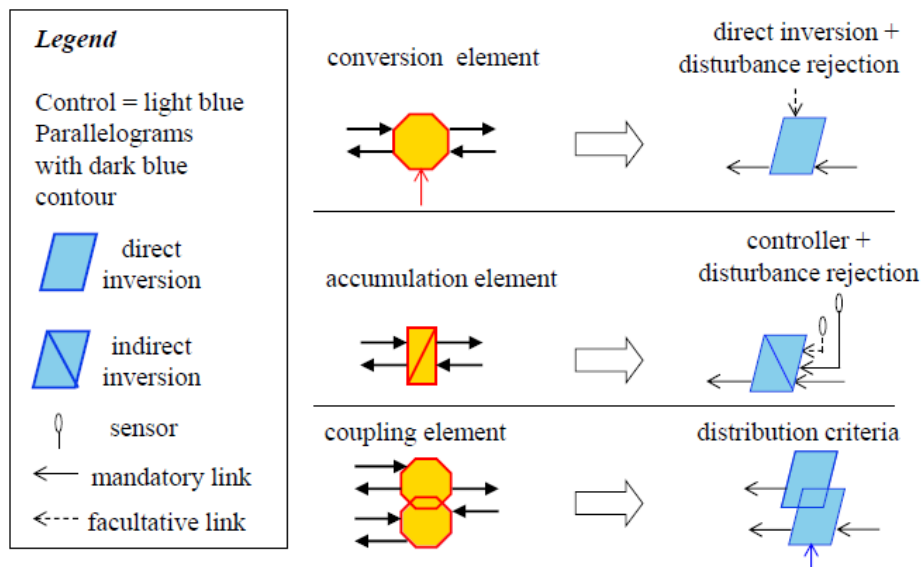


Fig. 2. Reguli de control bazate pe inversiune (explicativă în termeni originali în Engleză)

Diferite tipuri de operațiuni de control

Legăturile punctate (opționale), în Fig. 2, sunt semnale măsurate utilizate pentru respingerea perturbării directe în interconectările interne, așa cum este indicat în Fig. 3. Tot Fig. 3, indică, cum este implementat controlul intern pentru elementele de inversare indirectă.

Elementele de inversiune indirectă utilizează un control în buclă închisă descris:

$$u_{ref}(t) = u_{dist_meas}(t) + C(t)(y_{ref}(t) - y_{meas}(t))$$

unde

$u_{ref}(t)$ – variabila de reglare,

$u_{dist_meas}(t)$ – perturbația măsurată (legătura opțională),

$y_{ref}(t)$ – referința de ieșire,

$y_{meas}(t)$ – ieșirea măsurată/reglată,

$C(t)$ – bucla de control închisă.

Controlerul $C(t)$ ar putea fi implementat folosind structuri de control locale precum regulatorul PID (Proportional-Integral-Derivat). Structura în buclă închisă este ilustrată în Fig. 4

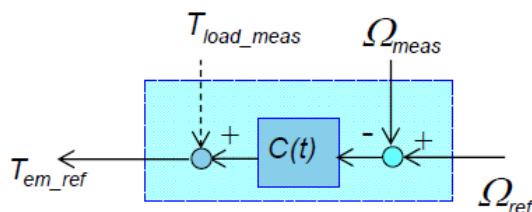


Fig. 3. Utilizarea internă a legăturii opționale în inversare indirectă

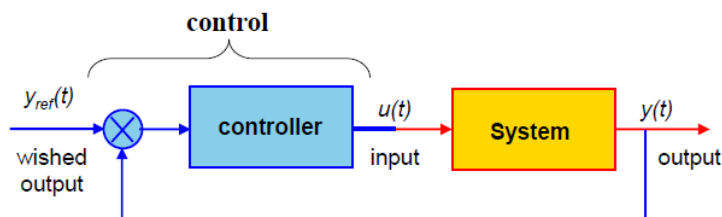


Fig 4. Buclă de control închisă

Pe de altă parte, elementele de inversare directă (vezi Fig. 2) sunt implementate folosind controlul în buclă deschisă. Aceste blocuri inversează acțiunea modelului și sunt aplicate numai pentru elementele EMR indicate în Fig. 2 și când relația I / O pe calea de reglare poate fi inversată. Organizarea generală a controlului în buclă deschisă este prezentată în Fig. 5.

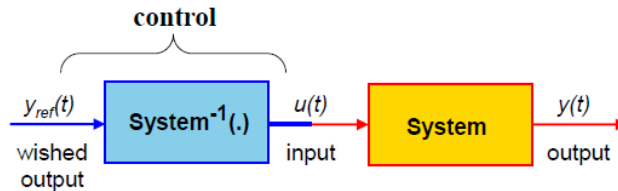


Fig. 5. Buclă de control deschisă

Librăria de control

Folosind regulile de inversiune EMR, schema de control este ușor de definit. Acordarea reguletoarelor ar trebui făcută într-o etapă diferită. Elementele prezentate fac parte din subsecțiunea Control-EMR, în Simcenter Amesim (Fig. 6).

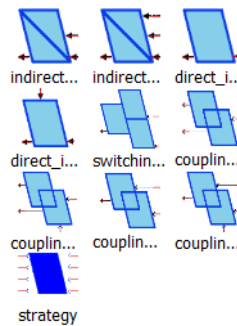


Fig. 6. Librăria EMR cu elemente de control

Exemplu de BEV

Procesul de dezvoltare a controlului bazat pe inversiune este după cum urmează:

1. Se elaborează schema EMR a sistemului, Fig. 1.
2. Se determină calea de reglare (acțiunea din model), Fig. 7.

3. Se realizează inversiunea pas cu pas, Fig. 8.

Calea de reglare este calea cauzală de la variabila de reglare la variabila controlată. Determinarea căii de reglare este posibilă datorită ordonării cauzale a semnalelor din EMR.

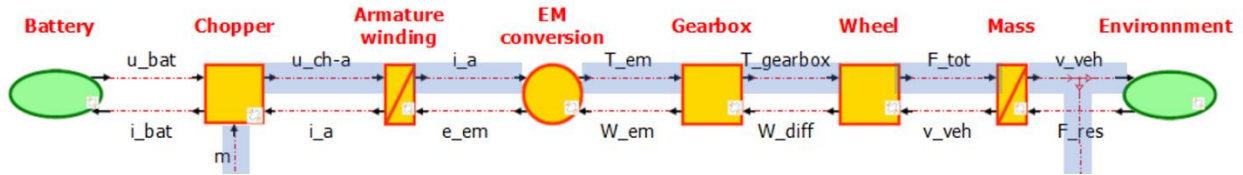


Fig. 7. Calea de reglaj (acțiunea)

Fiecare element este apoi inversat conform regulilor de inversare din Fig. 2.

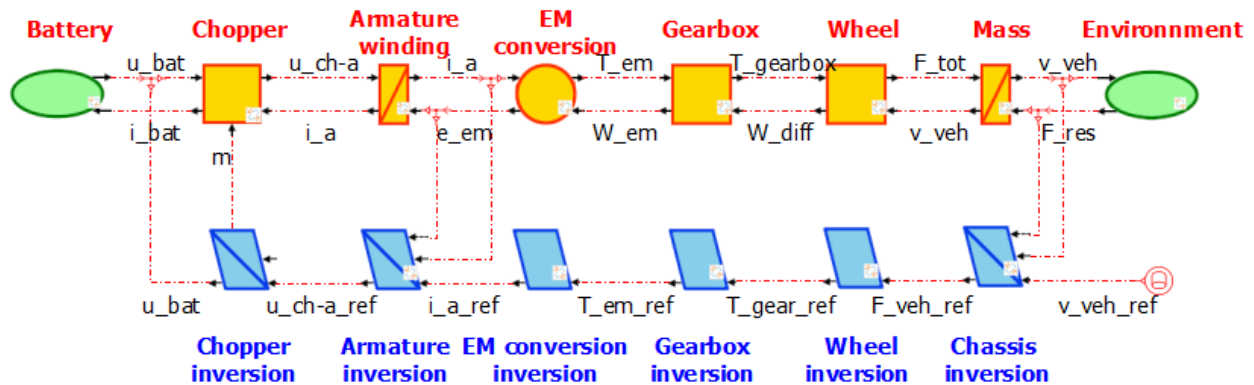


Fig. 8. BEV în EMR cu bucla de control

EMR asigură o structură de simulare în care modelul (ansamblele) și elementele de control sunt ușor de identificat. Rolul fiecărui regulator și interacțiunea sa cu procesul fizic sunt transparente.

Concluzii

EMR oferă organizarea sistemului care ar putea fi utilizată eficient pentru o definiție sistematică a schemei de control pentru dezvoltarea unei structuri de control distribuite ușor de întreținut pentru complexe multi-fizică

Exercițiul 8 – Acordarea reguletoarelor din structura de control

Obiectiv: În acest exercițiu vom învăța cum să acordăm reguletoarele pentru controlul bazat pe metoda inversării.

Reguletoarele tip PID

Arhitectura bazată pe inversiune este o metodă sistematică de control al sistemelor complexe. Structura rezultată în forma ei distribuită folosește reguletoare locale implementate fie în configurații cu buclă deschisă, fie închisă. În timp ce controlul în buclă deschisă utilizează modelul de sistem inversat drept regulator, există mai multe modalități de implementare a controlului local în buclă închisă, de ex. PID (Proportional-Integral-Derivat de control), LQG (liniar quadratic gaussian control) și există diferite proceduri de reglare a parametrilor reglatorului, de ex. Ziegler-Nichols, alocarea polilor, etc.

În acest exercițiu, abordăm o modalitate de acordare a reguletoarelor PI pentru utilizare în structuri de control bazate pe inversare, folosind alocarea polilor.

Regulatorul PI dă comanda de control pentru a conduce sistemul, astfel:

$$u(t) = K_p \cdot e(t) + K_i \int_{t_0}^t e(t) dt,$$

Unde:

$u(t)$ – variabila de reglare,

$e(t)$ – eroarea măsurată,

K_p – factorul proporțional,

K_i – factorul integral.

În simulare, PI-ul poate fi implementat folosind diagrama bloc, ca în Fig. 1, unde $y(t)$ este variabila de controlat măsurată, iar $r(t)$ este semnalul de referință pentru $y(t)$. Eroarea de determinată este:

$$e(t) = r(t) - y(t).$$

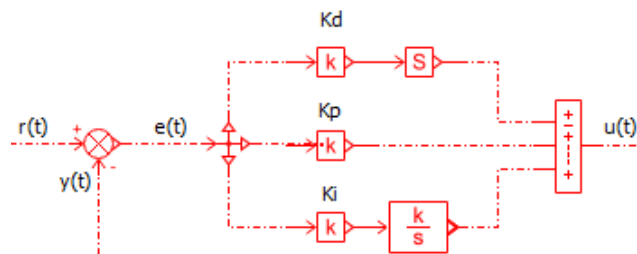


Fig. 1. Implementarea schemei bloc a controlerului PID.

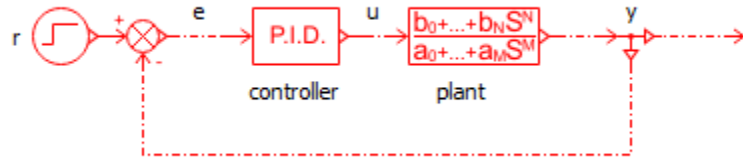


Fig. 2. Schema bloc a structurii de control în buclă închisă.

Controlerul PI este implicit implementat utilizând configurația în buclă închisă, așa cum este indicat în Fig.2.

Calitatea sistemului de control, adică răspunsul regăsit la ieșirea din Fig.2, este definită prin specificații de performanță. O modalitate de exprimare a performanțelor unui sistem de control este folosirea răspunsului la un semnal de tip treaptă pentru sistemul studiat (în buclă închisă).

Măsurile standard de performanță pentru răspunsul la semnalul de tip treapta ale unui sistem de ordinul doi includ:

- Timpul de creștere T_r ,
- Perioada de vârf T_p ,
- Suprareglajul σ ,
- Valoarea de vârf M_p ,
- Timpul de stabilizare T_s , (timpul de răspuns)
- Eroarea staționară e_{ss} (eroarea rezilientă).

Cele de mai sus înșirate pot fi urmărite pe oscilograma din Fig. 2.

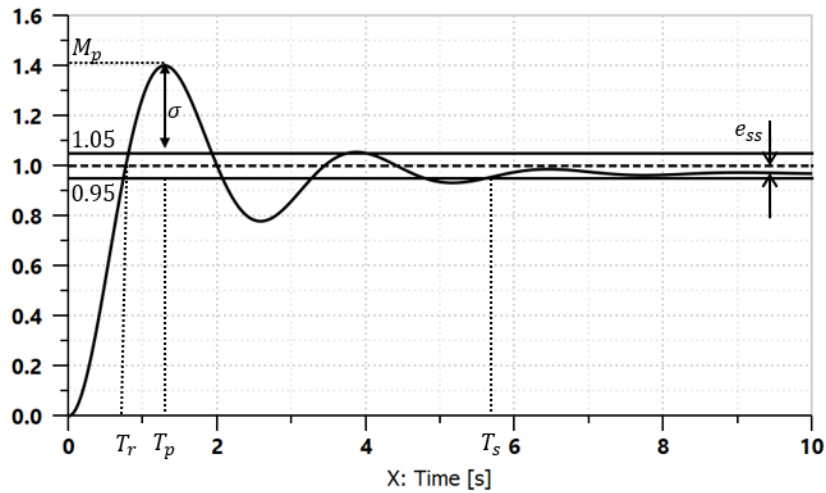


Fig. 2. Răspunsul unui sistem de gradul II la un semnal treaptă

Funcția de transfer a unui sistem de gradul doi răspunde la un semnal tip treaptă asemenea unde din figura 2.

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2},$$

unde

ζ – factor de atenuare,

ω_n – pulsația naturală,

s – variabila complexă Laplace.

Răspunsul funcției la semnalul tip treaptă depinde de parametrii sistemului.

Suprareglajul este

$$\sigma = 100 e^{-\zeta\pi/\sqrt{1-\zeta^2}},$$

Iar timpul de stabilizare

$$T_s \approx \frac{3}{\zeta\omega_n},$$

pentru ca răspunsul să rămână într-un interval de 5% în jurul valorii stării de echilibru.

Acesta este un aspect important, deoarece putem converti specificațiile din domeniul timp într-o funcție de transfer de ordinul doi pentru sistemul analizat.

Specificațiile de performanță ale sistemului de control impun ca răspunsul sistemului în buclă închisă controlată la un semnal tip treaptă să se potrivească cu răspunsul unui sistem de ordinul doi.

Este necesară deci construirea unei funcții de transfer, care să atingă specificațiile noastre de performanță. Metoda abordată este cea de alocare a polilor, pentru a determina parametrii regulatorului PI K_p, K_i .

Funcția de transfer în buclă închisă, cu referința $r(t)$ și ieșirea măsurată $y(t)$, este dată de

$$G_{cl}(s) = \frac{Y(s)}{R(s)} = \frac{K(s)P(s)}{1 + K(s)P(s)},$$

unde $P(s)$ și $K(s)$ reprezintă sistemul și funcția de transfer a regulatorului.

Având în vedere specificațiile de performanță ale sistemului în ceea ce privește o funcție de transfer de ordinul doi

$$G_{des}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2},$$

Dorim să alocăm polii $G_{cl}(s)$ ai sistemului în buclă închisă la locațiile polilor funcției de transfer dorite $G_{des}(s)$.

Considerând funcția de transfer de gradul I

$$P(s) = \frac{K_{plant}}{T_{plant}s + 1},$$

Și funcția de transfer a regulatorului PI

$$K(s) = \frac{K_p s + K_i}{s},$$

Polii sistemului în buclă închisă G_{cl} sunt dați de rădăcinile ecuației caracteristice,

$$T_{plant}s^2 + (1 + K_{plant}K_p)s + K_{plant}K_i = 0.$$

Prin reglarea parametrilor controlerului K_p și K_i putem atribui locația polilor, la cea a funcției de transfer dorite $G_{des}(s)$.

În acest fel, folosind alocarea polilor, putem alege parametrii controlerului PI astfel încât specificațiile de performanță ale sistemului să fie îndeplinite.

Exemplu pentru vehicul electric cu baterii

Vom ilustra în continuare dezvoltarea structurii de control local pentru organizarea EMR bazată pe inversare folosită la vehiculul electric cu baterii și motor de curent continuu.

Avem un model simplu EMR al unui vehicul electric cu baterie și motor de c.c. (Fig. 3), inclusiv schema de control bazată pe inversare. Vom acorda regulatoarele locale pentru această structură de control, pentru a realiza urmărirea vitezei de referință a vehiculului.

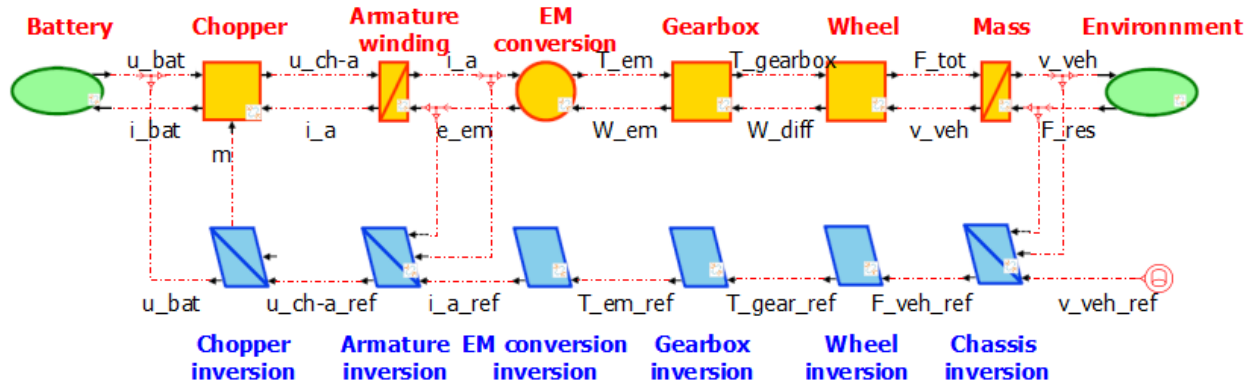


Fig. 3. Controlul EMR și inversarea unui vehicul electric cu baterie cu motor DC.

Controlul în buclă deschisă

Începem prin proiectarea regulatorului în buclă deschisă pentru elementele de inversiune directă. Aceste blocuri implementează direct inversul modelului matematic al subsistemului EMR (vezi Tabelul 1). Parametrii de simulare corespunzători sunt dați în Tabelul 2 și pot fi găsiți în meniul Global Parameters, în modul Parameters, în Simcenter Amesim.

Element	Ecuția de model	Ecuția de control
Convertor DCDC	$u_{ch-a} = m u_{bat}$	$m = u_{ch-a-ref} / u_{bat}$
Conversie EM	$T_{em} = K_{em} i_a$	$i_{a-ref} = \frac{1}{K_{em}} T_{em-ref}$
Cutie de viteze	$T_{gb} = K_{gb} T_{em}$	$T_{em-ref} = \frac{1}{K_{gb}} T_{gb-ref}$
Roată	$F_{veh} = \frac{1}{R_w} T_{gb}$	$T_{gb-ref} = R_w F_{veh}$

Table 1. Controlul inversiunii directe.

Parametru	Nume
K_{em}	EMpK_em
K_{gb}	MTpK_gear
R_w	MTpR_wheel

Table 2. Numele parametrilor în simulare.

Odată ce blocurile de inversare directă sunt la locul lor, rămâne să proiectăm cele două regulatoare necesare la buclă închisă: pentru înfășurarea indusului mașinii și elementul de masă.

Reglarea în buclă închisă

Controlul vitezei vehiculului

Considerăm mai întâi controlul vitezei și dezvoltăm un regulator local PI utilizând metoda de plasare a polilor prezentată în secțiunea a doua.

Primul pas este de a defini specificațiile de performanță privind răspunsul la viteza vehiculului.

Pentru teste, un profil de viteză a vehiculului (cum ar fi New European Driving Cycle) poate fi utilizat pentru a defini specificațiile minime de performanță în ceea ce privește viteza și accelerația vehiculului (vezi Fig. 4). Testul este dat ca un profil de viteză de referință, care include profilul de accelerație dorit. Acesta este determinat luând derivata în timp a funcției vitezei.

Deoarece profilul vitezei vehiculului este trapezoidal, performanța vehiculului poate fi reformulată din punct de vedere al profilului de accelerație. Un detaliu mărit al semnalului de accelerație dorit este dat în Fig. 5, care prezintă formă dreptunghiulară pentru referința de accelerație. Prima măsură de performanță pe care dorim să o realizăm este un răspuns al sistemului în viteză fără depășire $\sigma = 0\%$. Ne dorim ca viteza vehiculului să atingă starea de echilibru fără oscilații amortizate în jurul acestei stări. Depășirea referinței ar implica ca vehiculul să accelereze și frâneze într-o perioadă scurtă de timp, ceea ce creează disconfort.

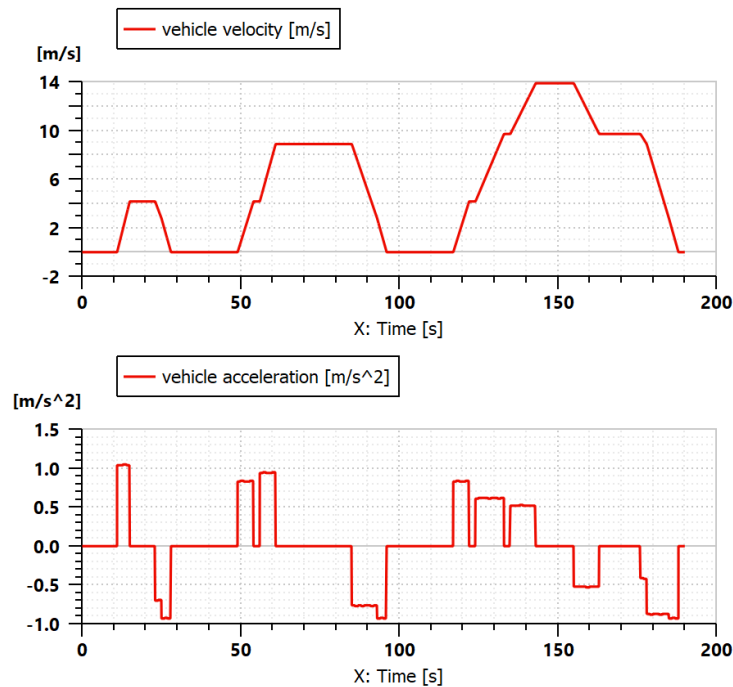


Fig. 4. Viteza de referință și profilurile de accelerație ale vehiculului.

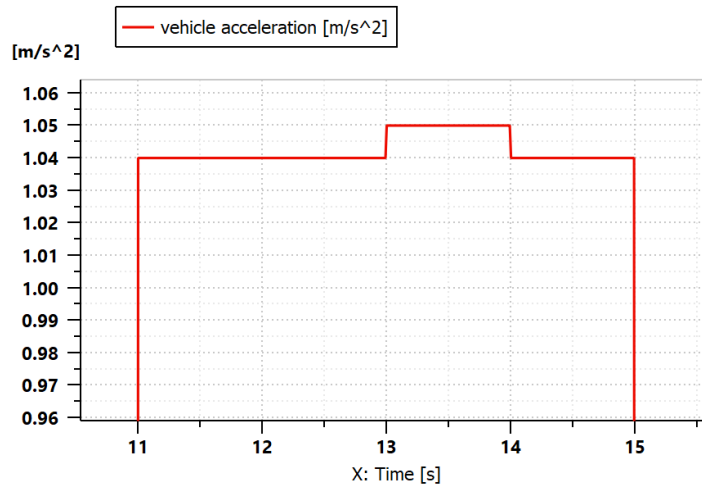


Fig. 5. Detaliu la accelerația de referință a vehiculului.

O a doua măsură de performanță este derivată din profilul de viteză NEDC, în ceea ce privește răspunsul la timpul de accelerație. În Fig. 5, putem vedea variațiile accelerației în profilul de referință. Răspunsul la acestea ar trebui, astfel, să fie suficient de rapid pentru a atinge starea de echilibru înainte de a se produce o altă schimbare. Pe baza profilului de accelerație dorit din Fig. 4, am putea necesita un timp de stabilire a răspunsului pasului de $T_s = 0.1$ sec

Măsurile de performanță dorite pentru răspunsul în viteză a vehiculului sunt:

- $\sigma = 0 \%$,
- $T_s = 0.1$ [s] (la accelerație).

Un model în buclă închisă care atinge aceste performanțe este dat de:

$$G_{des}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2},$$

cu

$$\zeta = 1, \text{ rezultând } \sigma = 100 e^{-\zeta\pi/\sqrt{1-\zeta^2}} = 0 \%,$$

$$\omega_n = 30, \text{ rezultând } T_s \approx \frac{3}{\zeta\omega_n} = 0.1.$$

Considerând sistemul

$$P_M(s) = \frac{1}{M_{eq}s},$$

cu $M_{eq} = CHApK_{eq}$, masa echivalentă a vehiculului și funcția de transfer a regulatorului PI, polii sistemului în buclă închisă sunt date de ecuația caracteristică

$$s^2 + \frac{1}{M_{eq}} K_p s + \frac{1}{M_{eq}} K_i = 0.$$

Polii sistemului în buclă închisă pot fi fixați în locația polilor funcției de transfer dorite $G_{des}(s)$, alegând parametrii regulatorului PI

$$K_p = 2\zeta\omega_n M_{eq},$$

$$K_i = \omega_n^2 M_{eq}.$$

Pentru a testa și acorda meticulos regulatorul, putem dezvolta o simulare separată în care testăm local funcționarea lui pe bucla de inversare fără influențe din sistemul complet. Diagrama de simulare este prezentată în Fig. 6. Acest test preliminar este util pentru a determina dacă performanțele sistemului de control local sunt îndeplinite, deoarece simularea întregului vehicul necesită toate regulatoarele să fie funcționale.

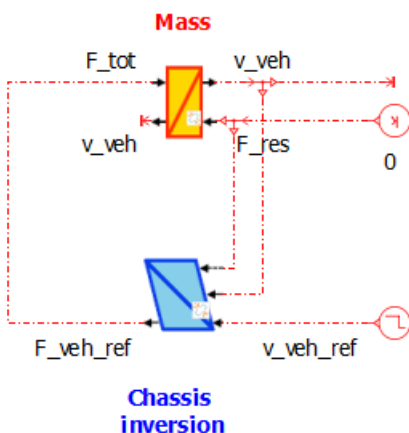


Fig. 6. Test local pentru regulatorul de viteză al vehiculului.

Rezultatele simulării sunt indicate în Fig. 7, unde observăm răspunsul în viteza a vehiculului la un semnal treaptă. Sistemul atinge timpul de stabilizare dorit. Cu toate acestea, răspunsul a depășit, în loc să fie puternic amortizat.

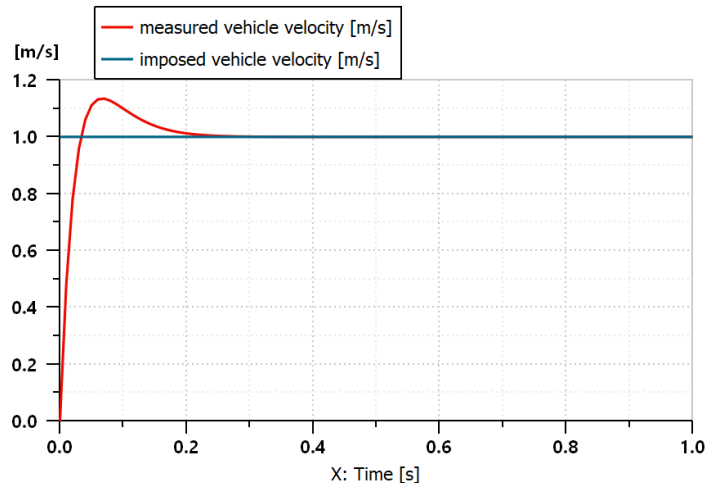


Fig. 7. Răspunsul de viteză pentru sistemul cu buclă închisă PI la semnal tip treaptă.

Observație! Motivul depășirii este pentru că regulatorul PI introduce, de asemenea, în funcția de transfer în buclă închisă un zero la

$$K_{plant}(K_p s + K_i) = 0.$$

Pentru a evita calcule suplimentare, ar putea fi folosit un alt tip de structură de regulator numită IP (Integral-Proportional, care nu este tratat aici).

Controlul curentului din indus

Folosind controlul reglat deja pentru viteza vehiculului, putem genera și semnalul de referință pentru curentul de indus i_{a-ref} , vezi Fig. 3.

Pe baza referinței lui i_{a-ref} , putem formula specificațiile de performanță pentru bucla de curent, similar cu cea a reglării vitezei vehiculului.

Pentru a dezvolta controlul curentului de indus, luați în considerare specificațiile următoare: este de 10 ori mai rapid decât bucla externă pentru un control de viteză (necesar pentru decuplarea buclilor)

- $\sigma < 20 \%$,
- $T_s = 0.01$ [sec].

Un regulator PI pentru urmărirea referinței curentului armăturii poate fi dezvoltat așa cum s-a explicat mai devreme. Testarea unui astfel de regulator pe o simulare locală (Fig. 10), dă răspunsul la un semnal pas de curent indicat în Fig. 11. După cum se poate vedea în Fig. 11, urmărirea referinței se realizează cu specificațiile de performanță dorite.

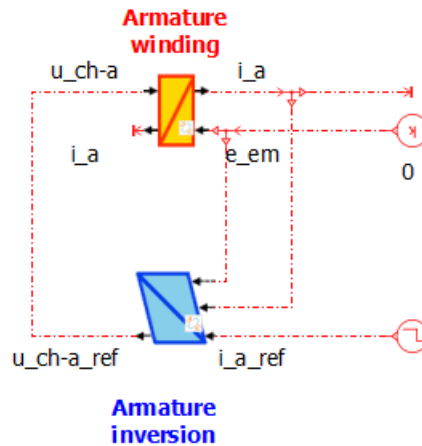


Fig. 10. Diagrama de simulare locală a reglării de curent a indusului.

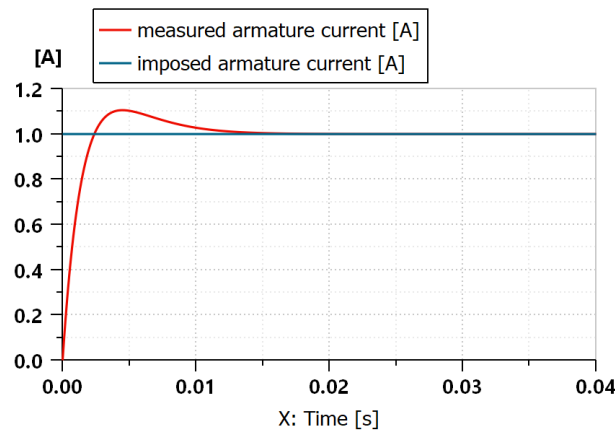


Fig. 11. Răspunsul indusului la un semnal treaptă de curent controlat PI.

Simularea vehiculului complet

Regulatele locale dezvoltate pot fi testate pe simularea completă a vehiculului, odată ce toate au fost proiectate.

Întregul control bazat pe inversiune poate fi testat pe profilul vitezei de referință din Fig. 4.

Fig. 12 și 13 arată că viteza controlată a vehiculului este capabilă să urmărească profilul de referință impus. Fig. 12 prezintă răspunsul de viteză pe întregul profil, în timp ce Fig. 13 prezintă o versiune concentrată pe un anumit sector a aceluiași grafic.

Fig. 14 arată că specificațiile de performanță ale accelerației vehiculului sunt îndeplinite, astfel încât accelerația vehiculului intră în starea de echilibru înainte de a avea loc modificările treptelor următoare.

Fig. 15 prezintă referința de curent generată și că sistemul este capabil să o urmărească, cu specificațiile dorite îndeplinite.

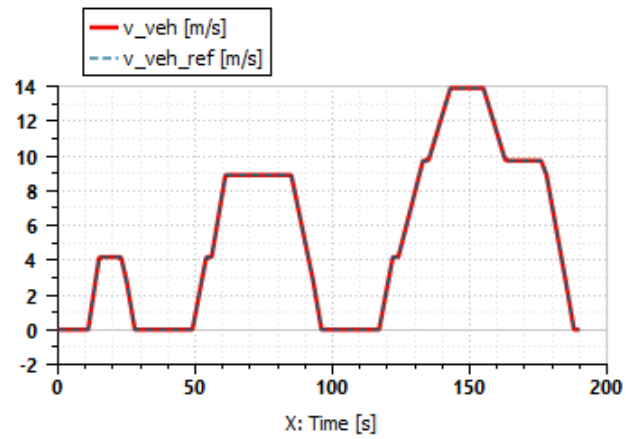


Fig. 12. Răspunsul complet la viteza vehiculului.

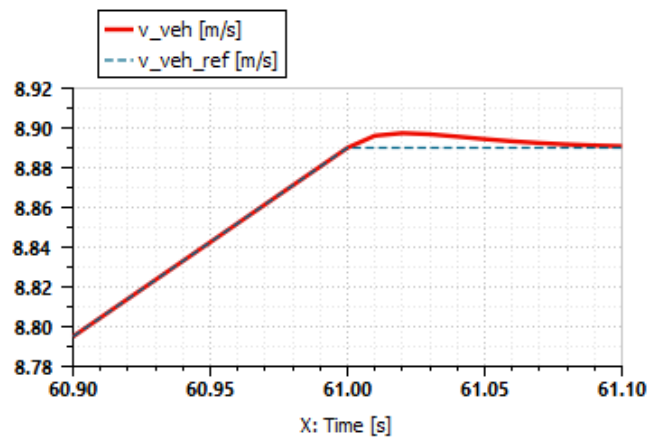


Fig. 13. Răspuns complet la viteza vehiculului, secțiune focalizată.

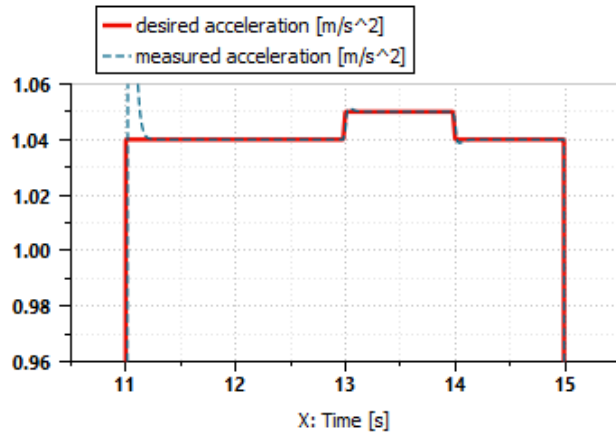


Fig. 14. Urmărirea profilului de accelerație.

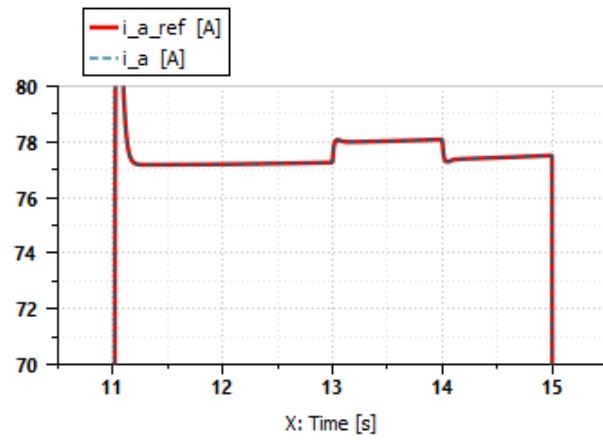


Fig. 15. Urmărirea referinței curentului de indus

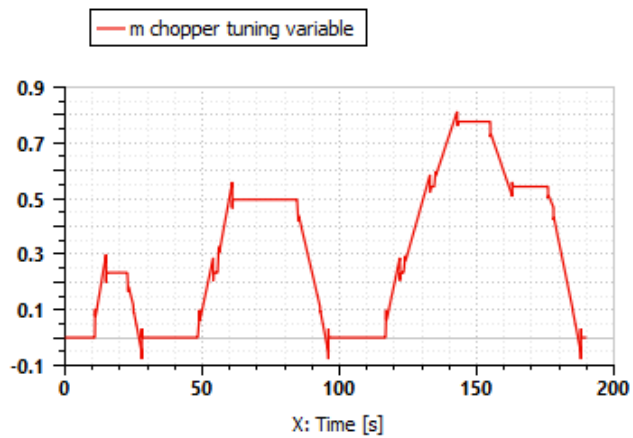


Fig. 16. Variabila de reglare (pentru convertor)

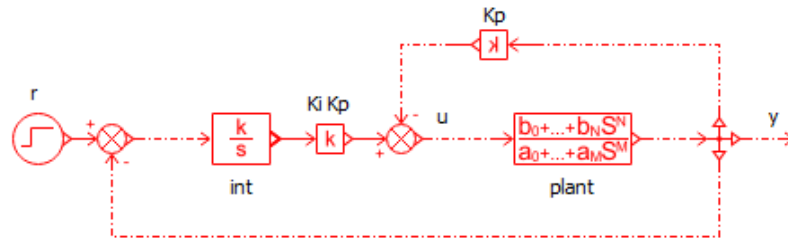


Fig. 17. Schema de control Integral-Proportional

Vârful din graficul pentru accelerația măsurată (Fig. 14) și curentul măsurat (Fig. 15) în jurul a 11 secunde, se datorează zero-ului nedorit care este introdus implicit în bucla închisă de către regulatorul PI al vitezei vehiculului. Acesta a fost motivul depășirii răspunsului la viteza vehiculului (Fig. 13). După cum s-a menționat mai devreme, pentru a evita zero-ul nedorit introdus în bucla închisă de către controlerul PI, poate fi utilizat un alt controler numit IP (vezi Fig. 17).

Concluzii

În acest exercițiu, s-a ilustrat procesul de dezvoltare a reguletoarelor PI folosind plasarea polilor. Am văzut că astfel de reguletoare utilizate în schema de control bazată pe inversare sunt ușor de gestionat și ar putea asigura o funcționare performantă a sistemului pe baza condițiilor de operare dorite.

Bibliografie

- A. Bouscayrol, J. P. Hautier, B. Lemaire-Semail, "Graphic Formalisms for the Control of Multi-Physical Energetic Systems", Systemic Design Methodologies for Electrical Energy, tome 1, Analysis, Synthesis and Management, Chapter 3, ISTE Willey editions, October 2012, ISBN: 9781848213883.
- B. A. Bouscayrol, X. Guillaud, P. Delarue, B. Lemaire-Semail, "Energetic Macroscopic Representation and inversion-based control illustrated on a wind energy conversion systems using Hardware-in-the-loop simulation", IEEE TIE, vol. 56, no. 12, Dec. 2009, pp. 4826-4835.
- C. [9] M. Ruba, R. O. Nemes, S. M. Ciornei, C. Martis, A. Bouscayrol and H. Hedesiu, "Digital Twin Real-Time FPGA Implementation for Light Electric Vehicle Propulsion System Using EMR Organization," 2019 IEEE Vehicle Power and Propulsion Conference (VPPC), 2019, pp. 1-6, doi: 10.1109/VPPC46532.2019.8952428
- D. <http://www.emrwebsite.org/>
- E. Amesim Help pentru funcții EMR