

Contribuții la modelarea, simularea și emularea Circuitelor Integrate Dedicate controlului inteligent al pompelor de combustibil de înaltă presiune din domeniul automotive.

Teză de doctorat – Rezumat

pentru obținerea titlului științific de doctor la

Universitatea Politehnica Timișoara

în domeniul de doctorat __Inginerie Electronica si Telecomunicatii__

autor ing. _Laszlo MOLNAR_____

conducător științific Prof.univ.dr.ing. _Aurel GONTEAN_____

luna_07_ anul_2019__

Capitolul 1: Introducere

Dezvoltarea unui ASIC este un proces complex și consumator de timp. În timpul fazei de dezvoltare dar și după fabricare, echipa de ingineri trebuie să verifice implementările digitale și analogice ale tuturor funcțiilor specifice ale ASIC-ului.

Procesul de dezvoltare este prezentat în Fig. 1.1 și începe cu colectarea specificațiilor pentru funcțiile dorite ale ASIC-ului. ASIC-ul va funcționa într-un sistem, astfel încât arhitectul sistemului și echipa de dezvoltare trebuie să colaboreze în această fază pentru a înțelege mai bine funcțiile necesare și interacțiunea cu restul sistemului.

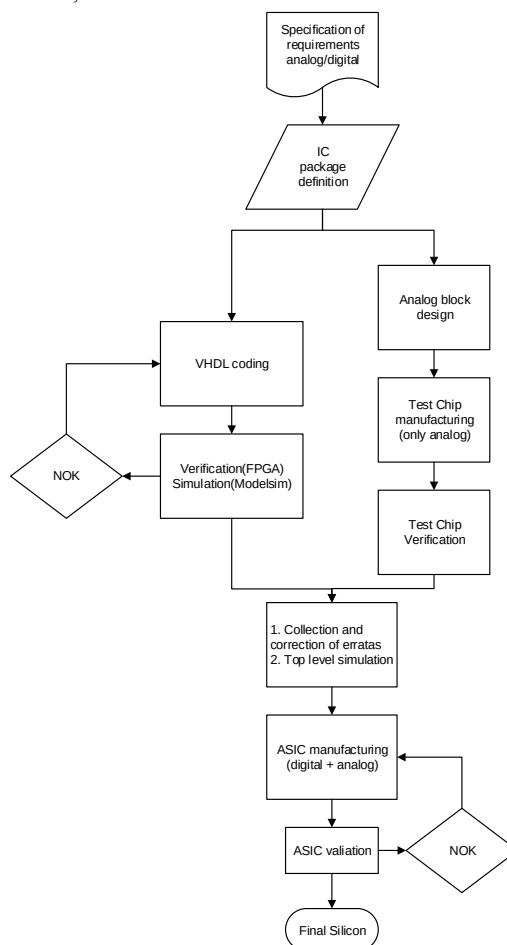


Fig.1.1. Schema bloc a procesului de dezvoltare a unui CI dedicat.

După finalizarea acestui pas, începe implementarea fizică a funcțiilor. Există două procese în paralel, unul, elaborarea codului VHDL pentru a implementa funcțiile digitale definite, iar celălalt, pentru a implementa circuite analogice de mică și mare putere care cooperează cu funcțiile digitale. Circuitele de putere mică sunt comparatoare, surse de referință pentru curent și tensiune, senzori de temperatură, ADC (Analog to Digital Converter), DAC (Digital to Analog Converter) și așa mai departe. Circuitele de mare putere sunt etaje de comandă a porții tranzistoarelor MOSFET, tranzistoare de putere MOSFET N-ch (N channel) și P-ch (P channel) integrate pe pastila de siliciu.

Proiectanții VHDL dar și cei ai blocurilor analogice din ASIC realizează simularea simulării pentru testarea implementării lor.

Suplimentar față de simulare, verificarea codului VHDL se realizează pe plăci FPGA pentru a verifica funcția digitală. În cazul în care sunt găsite erori, codul VHDL este corectat și verificat din nou până când se realizează o implementare fără probleme.

În următorul pas se fabrică un "cip de test" care conține numai blocul analogic, și electronica de putere, care se folosește pentru verificarea implementării analogice. Blocul digital este emulat de echipamente externe pentru a implementa interacțiunea digitală necesară pentru blocul analogic.

Toate constatările sunt colectate ca o listă de erată și corectate în implementarea noii versiuni a CI.

La sfârșit, se realizează o simulare de nivel superior în care blocul digital și cel analog funcționează împreună, dar de data asta și cu adnotări din layout-ul siliciului pentru a evidenția eventualele efecte parazite ale implementării specifice.

Următorul pas este așa numitul „tape-out” atunci când proiectul este înghețat și procesul de fabricație a ASIC-ului (Application Specific Integrated Circuit) începe, dar în această fază doar în număr limitat, până la 100 de bucăți pentru a fi trimise înapoi echipei de proiectare pentru a le verifica.

Capitolul 2: Testarea codului VHDL

În capitolul 1 am prezentat un proces simplificat de proiectare a ASIC-urilor, un pas important fiind testarea codului VHDL. În acest capitol voi extinde acest proces, insistând asupra abordării proprii și contribuțiilor personale

Codul VHDL este testat în faza de proiectare prin simulare și ulterior pe plăci cu FPGA. După finalizarea fazei de proiectare, codul VHDL va fi transferat în pastila de siliciu a ASIC-ului și va deveni o funcție digitală implementată de blocuri logice predefinite cum ar fi: porți logice, MUX (Multiplexor), LUT (Look Up Table), PLA (Programmable Logic Array).

În timpul procesului de fabricare se poate întâmpla ca implementarea VHDL fără bug-uri să devină o funcție defectuoasă datorită unei erori în procesul de fabricație – de exemplu, atunci când se creează un strat de metalizare pentru interconectarea pe pastila de siliciu, un scurtcircuit la o linie de alimentare va afecta funcționarea unei celule (stack-at failure).

Există două tipuri principale de metode de detectare a defecțiunilor: injectare de vectori de test prin echipamente de testare automată, ATE (Automatic Test Equipment) și cel de auto-testare, BIST (Built-In Self Test) implementat în ASIC.

Injectarea vectorilor de test necesită o așa-numită simulare a defecțiunilor pentru a defini un set de vectori de testare care sunt necesari pentru programul de testare care controlează ATE-ul. Simularea defecțiunilor se bazează pe introducerea artificială în codul VHDL (corect) a unor defecțiuni conform modelelor de defecțiuni predefinite. Fiecare descriere voit eronată este apoi simulată și comparată cu rezultatele simulării codului fără defecțiuni. Pe baza acestei comparații se poate analiza ce efect are „injectarea” unei defecțiuni în codul fără erori.

BIST este implementat în structura ASIC-ului și este pornit de fiecare dată când acesta este alimentat. Acest test este consumator de timp și de obicei, testează doar o parte limitată a logicii ASIC-ului (blocuri de siguranță).

BIST este utilizat pentru a detecta defecțiunile apărute în timpul duratei de viață a componentelor care au fost considerate componente "conforme" după ce toate verificările au fost efectuate în timpul procesului de fabricare. În timpul duratei de viață a componentei s-ar putea întâmpla evenimente externe, cum ar fi descărcările ESD (Electro-Static Discharge) sau un scurtcircuit între pinii acestuia astfel încât să afecteze o funcție internă a ASIC-ului; executarea procedurii de BIST la fiecare inițializare a ASIC-ului testează aceste funcții, dezactivând ASIC-ul în cazul în care o funcție testată este afectată.

Următoarele tabele sunt un sumar a ceea ce au făcut diverși autori în domeniul testării codului VHDL.

Capitolul 3: Arhitecturi de validare și simulare a circuitelor integrate dedicate

Capitolul 2 a tratat testarea codului VHDL (la nivel de poartă, dar și comportamental) care stă la baza blocului digital al oricărui ASIC și testarea blocului analogic prin extragerea unor modele comportamentale din schema analogică la nivel de tranzistor a implementării particulare a unui ASIC; toate aceste teste se realizează în faza premergătoare fabricării fizice a ASIC-ului.

În acest capitol voi prezenta metodele folosite în prezent la testarea ASIC-ului, atunci când el este deja fabricat, dar voi propune și o metodă alternativă de testare atunci când doar codul VHDL este disponibil și se cunoaște arhitectura analogică, dar implementarea în siliciu încă nu este disponibilă, tot ASIC-ul fiind emulat de diverse componente (de exemplu blocul digital de un FPGA, iar blocul analogic de componente discrete interconectate conform arhitecturii ASIC-ului).

Suplimentar față de testarea fizică prin emulare, prezint metodele actuale și propun o metodă alternativă de simulare a blocului digital și cel analogic pentru testarea arhitecturii, atunci când încă nici ASIC-ul nu este fabricat și nici setup-ul care emulează funcția ASIC-ului nu sunt disponibile.

Cel mai important proces în faza de dezvoltare a unui ASIC după faza de proiectare este verificarea. Pentru un ASIC complex, procesul de verificare poate dura mai mult decât faza de proiectare.

Propunerea dezvoltată în cadrul tezei este potrivită pentru validarea întregului sistem într-o fază timpurie, când numai TC-ul (Test Chip) este fabricat fizic. Pentru această fază, propun arhitectura de validare MTP (Modular Test Platform) cu platforme individuale pentru fiecare componentă principală a ECU-ului, după cum se arată în Fig. 3.1.

Platforma de validare este construită pe o placă principală, având rolul de interconectare și suport mecanic pentru diferitele PCB-uri (Printed Circuit Board) implementând diversele blocuri necesare. Există o placă cu sursa de alimentare principală, care generează tensiuni de alimentare de 1V2, 3V3 și 5V dintr-o baterie sau o sursă externă de alimentare.

Comunicarea dintre utilizator și ASIC este gestionată de MDB (Microcontroller Development Board), care conține un transceiver CAN (Control Area Network). Interfața cu utilizatorul este un GUI (Graphical User Interface) bazat pe Microsoft Excel utilizând macro-comenzi Visual Basic.

Un adaptor USB-CAN generează interfața electrică diferențială CAN corespunzătoare PC-MTP.

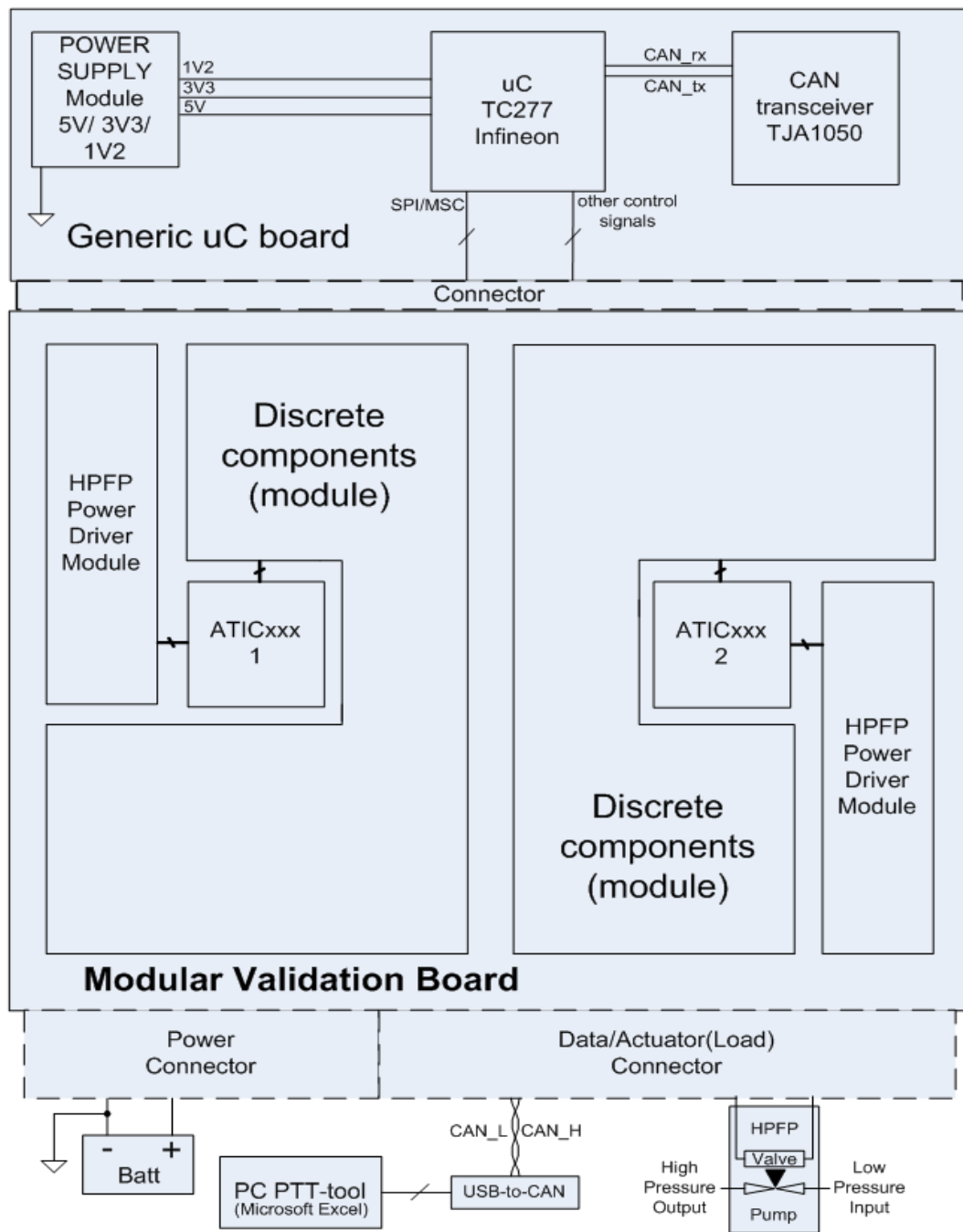


Fig. 3.1 Platforma modulară de test propusă.

Emularea ASIC se realizează prin două componente. Blocul digital este emulat de o placă de dezvoltare FPGA care implementează codul VHDL. Pentru emularea blocului analog, sunt disponibile două opțiuni diferite.

Prima opțiune este de a construi un PCB cu componente discrete, care îndeplinesc funcțiile blocului analogic ASIC; avantajul acestei opțiuni este testarea blocului digital înainte de fabricarea TC-ului (design analogic). Propun o a doua opțiune care ar putea folosi însăși TC-ul care conține doar blocurile analogice ale ASIC-ului, dar aceasta înseamnă că designul digital și analog trebuie să fie terminat aproape în același timp. Principalul avantaj comparativ cu soluția anterioară este că blocul analog este foarte asemănător cu ASIC-ul, deci comportamentul circuitului este același.

În ambele cazuri, blocul analogic este conectat la placa de dezvoltare cu FPGA, deci la codul VHDL care definește funcțiile blocului digital.

Avantajul utilizării TC-ului ca bloc analogic este faptul că acesta conține circuitul analogic exact care va fi în interiorul ASIC-ului și astfel toate efectele parazitare vor fi reflectate în funcționalitatea analogică.

ASIC-ul ar putea avea diferite funcții, de exemplu așa cum se arată în Fig. 3.1 un circuit de comandă în buclă închisă cu control de curent pentru HPFP (High Pressure Fuel Pump). Circuitul de putere pentru comanda HPFP este proiectat pe un PCB care utilizează componente discrete. Acest modul este o topologie Half-Bridge care conține un MOSFET cu canal-P (HSD - High Side Driver) și un MOSFET cu canal-N (LSD - Low Side Driver) și circuite suplimentare, snubber, recircularea energiei (free-wheel pe HS și active clamping pe LS), rezistențe de măsură a curentului.

A doua propunere de simulare la nivel de sistem are arhitectura „All-In-One” folosind doar SaberRD®, după cum se arată în Fig. 3.2.

Următoarele capitole vor prezent implementarea unei platforme virtuale a arhitecturii HW din Fig. 3.1, bazată pe metodologia de simulare a subsistemului „All-In-One”.

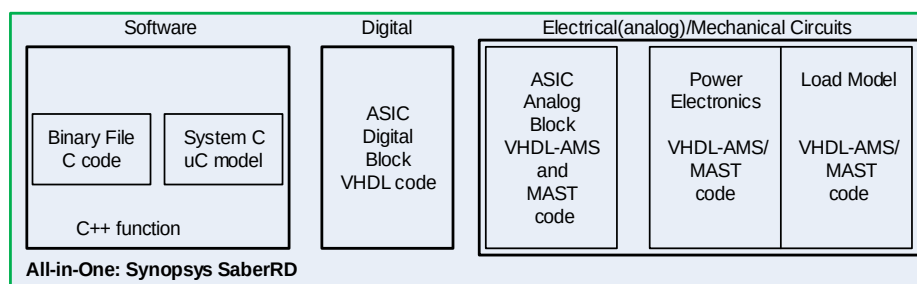


Fig. 3.2 Arhitectura de simulare propusă (original).

Această metodologie prezintă o arhitectură de simulare promițătoare, bazată pe experiența mea anterioară de utilizare a SaberRD® ca simulator pentru codul VHDL. În plus, acesta este un simulator electric specializat pentru circuite electronice, având incluse biblioteci conținând o gamă largă de modele de componente electronice. Modelele electronice (analogice) sunt construite folosind MAST (limbajul de descriere a hardware-ului proprietar Synopsys) sau VHDL-AMS. Totodată permite și traducerea automată a modelelor Spice în MAST pentru a fi utilizate cu succes modele de componente electronice puse la dispoziție de către producătorul lor.

Rularea unui model SystemC al uC-ului împreună cu fișierul binar (software compilat) în SaberRD® este o metodă complet nouă. Simulatorul este capabil să apeleze o funcție C++; pentru aceasta este necesar să se creeze în prealabil funcția C++ și să fie stocată în folder-ul de simulare sub forma unui fișier ".dll". Scopul meu este de a implementa modelul uC împreună cu software-ul corespunzător care rulează pe acesta într-o funcție C++ care are o listă de parametri transferați din simulare, după care funcția este executată și la final rezultatele sunt returnate către simulare. Rezultatele simulării subsistemelor sunt comparate cu circuitul real utilizând MTP propus în Fig. 3.1.

Scopul final al comparației este verificarea acurateței și eficienței simulării sistemului în comparație cu mediul software și hardware care conține și sarcina (HPFP) care va fi utilizată.

O altă provocare este aceea de a modela sarcina electro-mecanică în ceea ce privește parametrii esențiali și relevanți din sistemul hidraulic, deoarece ei pot dezvălui schimbări cantitative a inductanței supapei de scurgere a pompei; de exemplu modificările presiunii din rampa comună pot duce la variația inductanței valvei sau a timpului de închidere.

Capitolul 4: Controlul pompelor de injecție

În domeniul automobilelor există două tipuri de motoare cu combustie internă: benzină și motorină. Pentru vehiculele de pasageri cel mai frecvent utilizat este motorul pe benzină din cauza emisiilor mai mici de gaze de eșapament și control mai bun al emisiilor pentru a atinge normele de poluare.

Motoarele pe benzină moderne sunt clasificate în două categorii principale: motoare mai ieftine și cu performanțe reduse cu sistem SPtI (Single Point Injection) sau MPI (Multi Point Injection) și cel GDI (Gasoline Direct Injection), sistem de injecție directă mai scump, cu un control mai bun al consumului de combustibil, oferind motoare mai puternice la aceeași capacitate cilindrică.

Sistemele SPtI și MPI injectează combustibilul în galeria de admisie, pentru SPtI în zona comună a galeriei de admisie, iar pentru MPI în fiecare ramificație individuală a galeriei de admisie, chiar în amonte de supapa de admisie. În aceste cazuri, combustibilul se amestecă cu aerul în afara cilindrului. Presiunea combustibilului injectată este de maxim 4 bari, iar injectoarele folosite sunt actuatori cu solenoid cu timp de răspuns lent.

Sistemul GDI este descris de Sarwar, Sankavaram și Lu [35], cu un accent deosebit pe pompa de combustibil de înaltă presiune (HPFP).

Sistemul GDI utilizează injectoare de înaltă performanță pentru a injecta combustibil direct în camera de ardere, cu o presiune a combustibilului de până la 500Bar. Datorită injecției directe, capul pistonului este răcit de combustibil, permițând rapoarte mai mari de compresie și o aprindere prin scânteie mai devreme producând putere mai mare.

Injectoarele GDI pot fi acționate prin solenoid sau piezo. Injecțiile multiple sunt posibile cu ambele, în timpul unui ciclu motor, în fiecare cilindru individual. Pentru a permite mai multe injecții pe ciclu motor, injectoarele trebuie să aibă un răspuns rapid, în special la închidere, deoarece dozarea combustibilului depinde de o procedură de comandă bazată exclusiv pe timp, astfel încât controlul electronic al injectoarelor este important și trebuie să fie eficient și precis.

Injectoarele piezoelectrice și cele cu solenoid sunt alimentate de tensiuni mai mari decât tensiunea de 12V a sistemelor auto. Surse de tensiune în comutație de tip ridicător (boost) sunt utilizate pentru a crește nivelul de tensiune la 65V pentru injectoarele solenoidale și la 400V pentru injectoarele piezo, permițând astfel o deschidere rapidă și o închidere rapidă prin limitarea valorii maxime (clamping) a tensiunii electromotoare induse a injectorului solenoid la 65V.

În Fig.3.2 se prezintă componentele de bază ale unui sistem GDI. Benzina din rezervorul de combustibil este pompată de o pompă de combustibil de joasă presiune, LPFP (Low Pressure Fuel Pump), prin conducte cu o presiune maximă de 4 bari către pompa de combustibil de înaltă presiune, HPFP, care este activată de arborele cu came al motorului generând o presiune de până la 500Bar. Combustibilul cu înaltă presiune este acumulat în rampa comună (tampon de combustibil), iar injectoarele sunt alimentate din rampa comună.

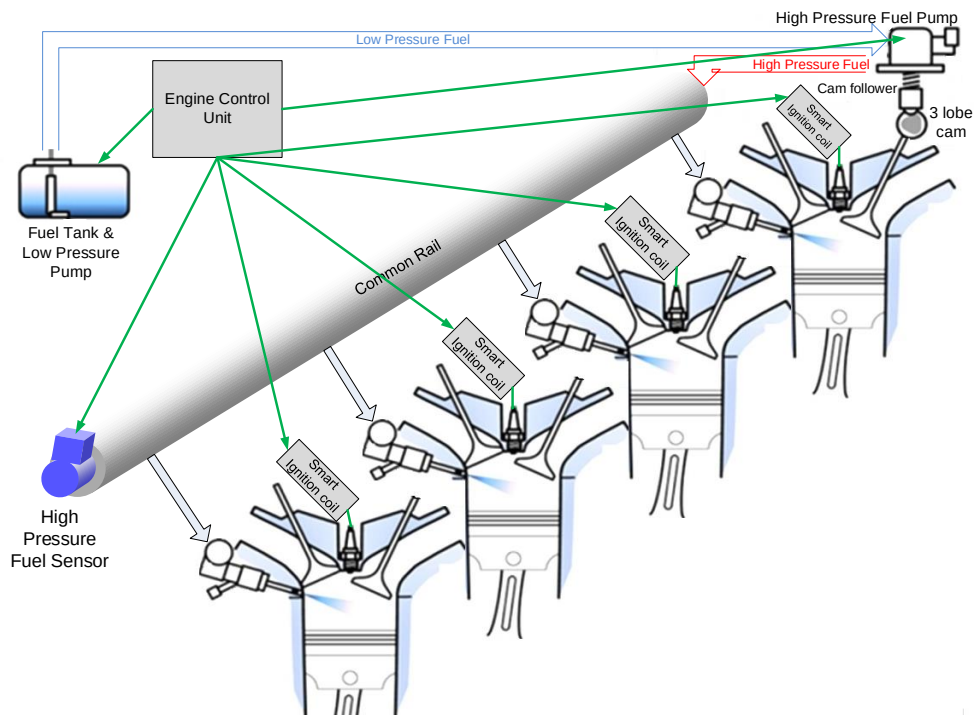


Fig.3.2 Arhitectura sistemului GDI.

Pentru a avea o injecție precisă a combustibilului, presiunea în sistemul cu rampă comună trebuie să fie în mod ideal constantă pentru a avea întotdeauna aceeași cantitate de combustibil pulverizat în cilindru de fiecare injector având timpul t_{ON} de activare (injecție).

În Fig.3.3 este prezentată schema simplificată a unei pompe HPFP constând în pompa în sine și valva DIV (Digital Inlet Valve) de control a cantității de combustibil .

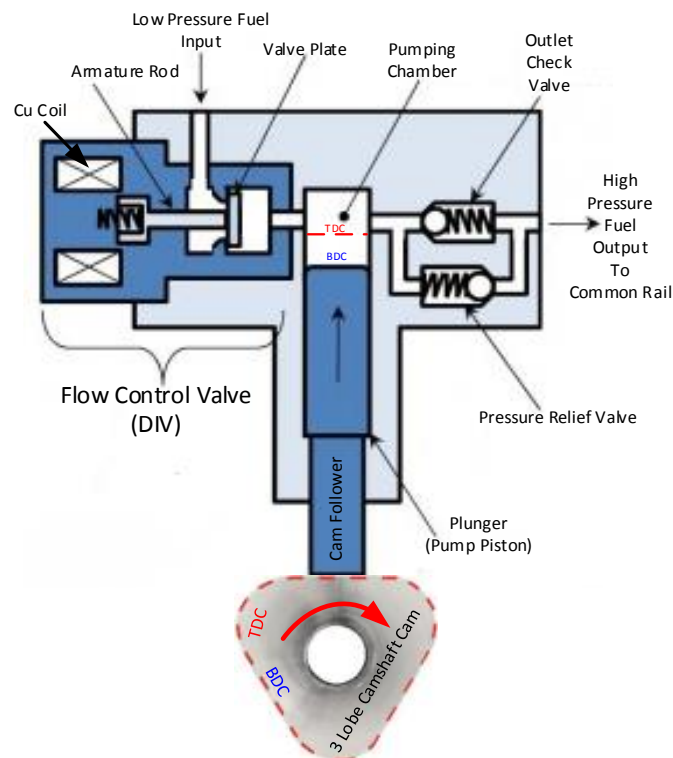


Fig.3.3 Detaliu HPFP .

Pentru a menține o presiune constantă în sistemul cu rampă comună, reglarea presiunii pornește de la controlul supapei electro-mecanice atașate pompei HPFP. Pentru a controla cantitatea de combustibil de joasă presiune care intră în camera de presiune a HPFP, valva electro-mecanică (DIV), normal deschisă, este activată după o durată Δt_1 față de punctul mort superior (TDC – Top Dead Center) al poziției pistonului pompei. LPFP pompează combustibilul în cilindrul de compresie a HPFP, iar apoi algoritmul de control al presiunii implementat în microcontrolerul unității de control al motorului (ECU) calculează cantitatea de combustibil necesară în funcție de sarcina și viteza motorului.

ECU-ul comandă închiderea supapei electro-mecanice DIV prin intermediul unui profil de curent „peak and hold”, permițând astfel ca rotirea lobilor arborelui cu came care acționează pistonul pompei HPFP să comprime combustibilul din camera de presiune. Dacă valva de scurgere DIV nu este închisă (profil de curent activ), combustibilul va fi pompat înapoi în sistemul de joasă presiune.

La ieșirea HPFP-ului există o supapă mecanică de reținere preîncărcată de un arc.

Atunci când presiunea din camera pompei HPFP este mai mare decât presiunea din rampa comună, combustibilul este transferat în rampa comună.

Algoritmul de control al presiunii combustibilului utilizează informațiile de la un senzor de presiune montat pe rampa comună.

Pe baza vitezei și încărcării motorului, ECU comandă injectoarele individual pentru fiecare cilindru; când se detectează poziția corectă a pistonului motorului termic, bobina de inducție generează o scânteie prin bujie în camera de combustie, rezultând o explozie a amestecul de combustibil.

În Fig.3.4 este prezentată pompa de presiune înaltă Continental, capabilă să producă o presiune de până la 350Bar în rampa comună (benzină).

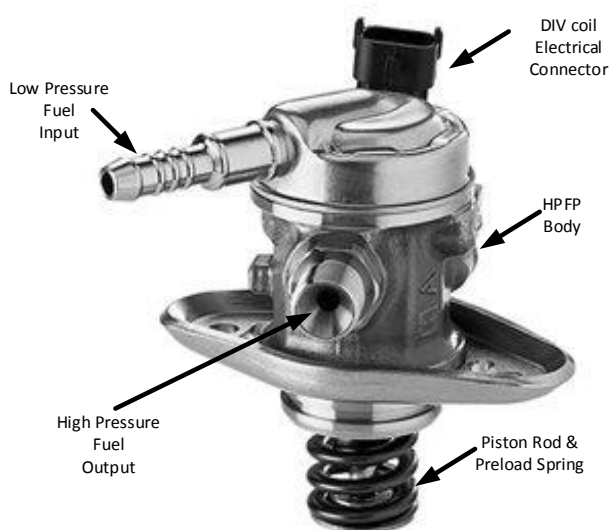


Fig.3.4 Pompa de presiune înaltă Continental-350Bar .

Capitolul 5: Rezultate experimentale

În acest capitol analizez implementarea celor două platforme propuse în Capitolul 3, platforma de simulare cu un singur simulator multi-domeniu și platforma MTP de emulare a ASIC-ului.

Ambele platforme facilitează testarea în fază timpurie a procesului de dezvoltare a ASIC-ului evitând propagarea eventualelor erori/defecte de concept/arhitectură în producția ASIC-ului.

În cele ce urmează voi prezenta schema globală de simulare organizată ierarhic, voi intra în detalii specifice referitoare la fiecare bloc component al ierarhiei. Un subiect important va fi dezvoltarea modelului electric de simulare pentru valva electro-mecanică DIV. Următorul pas va fi explicarea implementării fizice a platformei de emulare a ASIC-ului organizată tot pe blocuri ierarhice. În final voi compara rezultatele simulării cu cele ale implementării fizice

Conform practicii uzuale am început prin a simula în simulatorul mixt Saber® (Synopsys), așa cum este prezentat în Fig. . Scopul meu a fost să împart simularea pe blocuri funcționale, astfel încât să fie posibilă migrarea blocurilor în orice moment într-un proces de dezvoltare a circuitelor integrate mixte. Fiecare bloc funcțional poate fi ulterior extins la o implementare specifică într-un proces tehnologic de fabricare a circuitelor integrate.

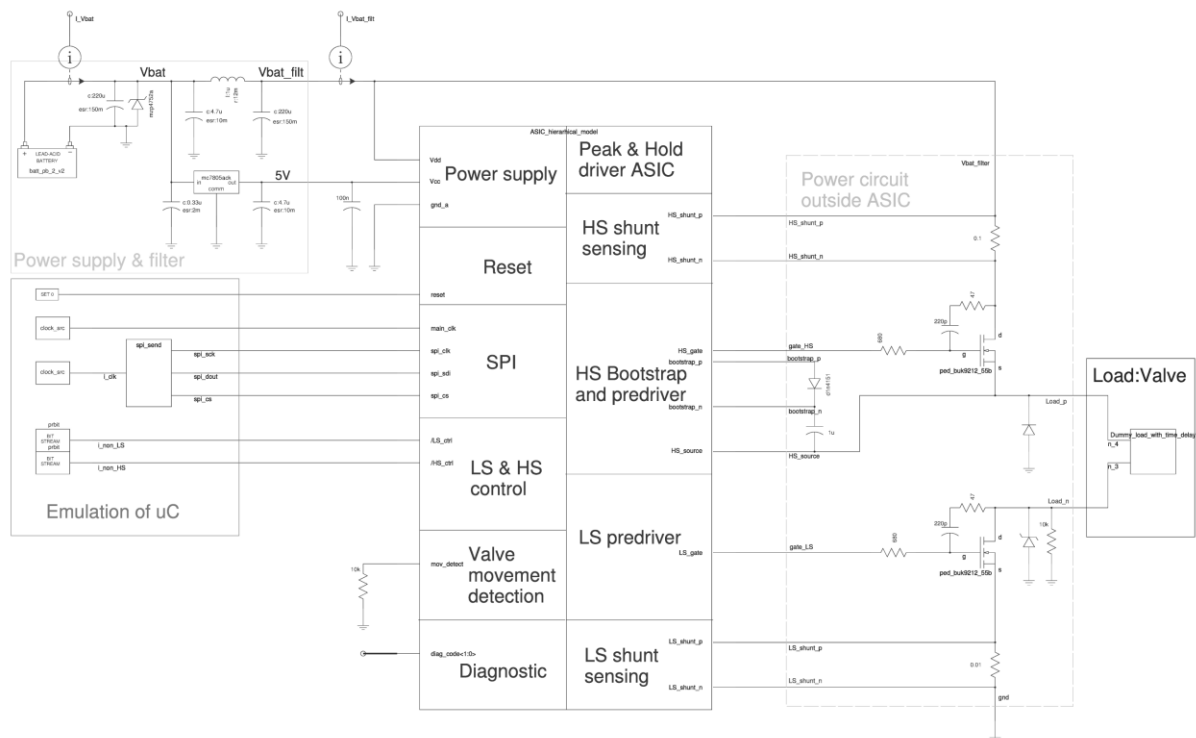


Fig. 5.1 schema ierarhică de simulare.

Simularea este organizată într-o arhitectură ierarhică care constă dintr-o schemă de nivel superior și un simbol principal (simbolul ASIC-ului care efectuează funcția de comandă a supapei DIV).

Schema de nivel superior este împărțită în 5 blocuri:

- Alimentare și filtru
- Emularea uC
- Bloc comportamental ASIC (Peak & Hold driver)
- Componente electronice discrete de putere (care nu fac parte din ASIC)
- Modelul electric a supapei DIV

Implementarea fizică a circuitului este prezentată în figura 5.2.

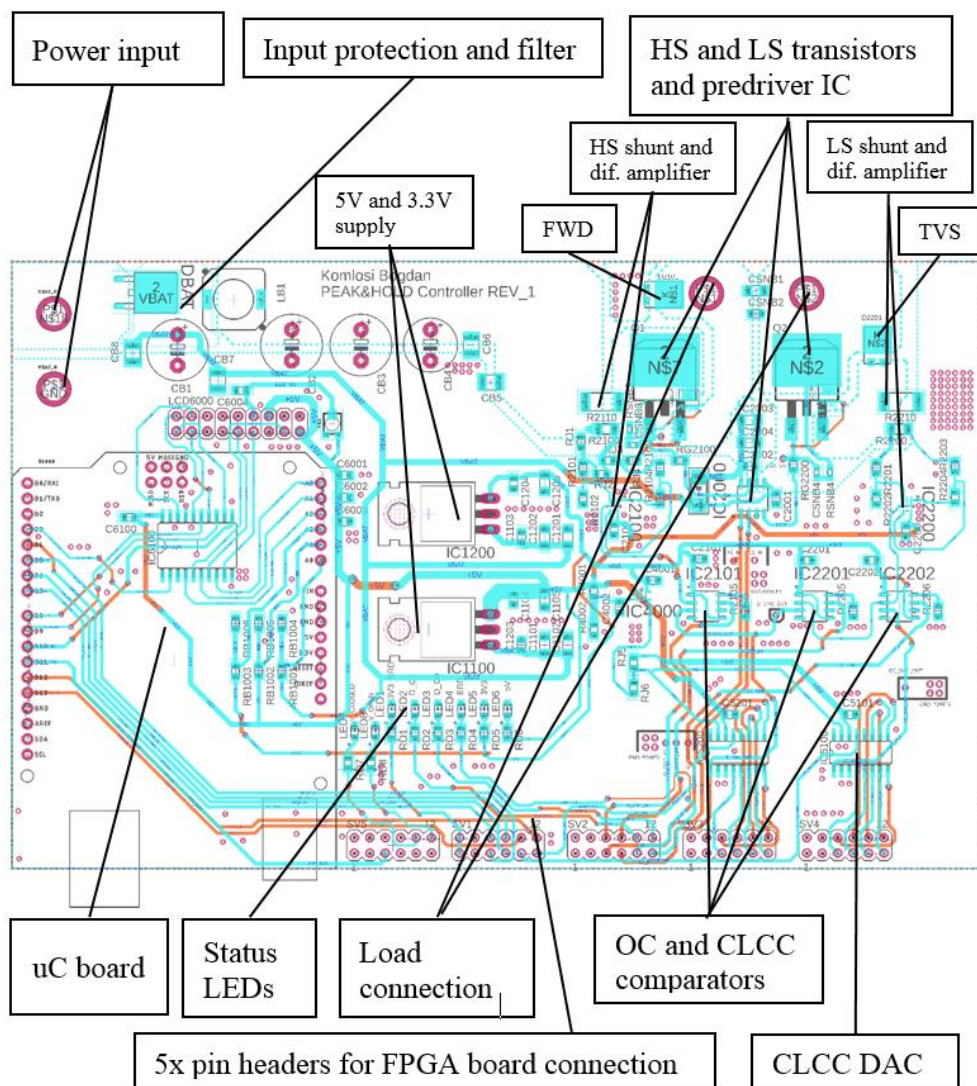


Fig.5.2 Implementarea circuitului imprimat.

În Fig.5.3 sunt comparate rezultatele reale cu rezultatele simulării aceluiași semnale folosind date de configurare identice cu cele pentru FPGA .

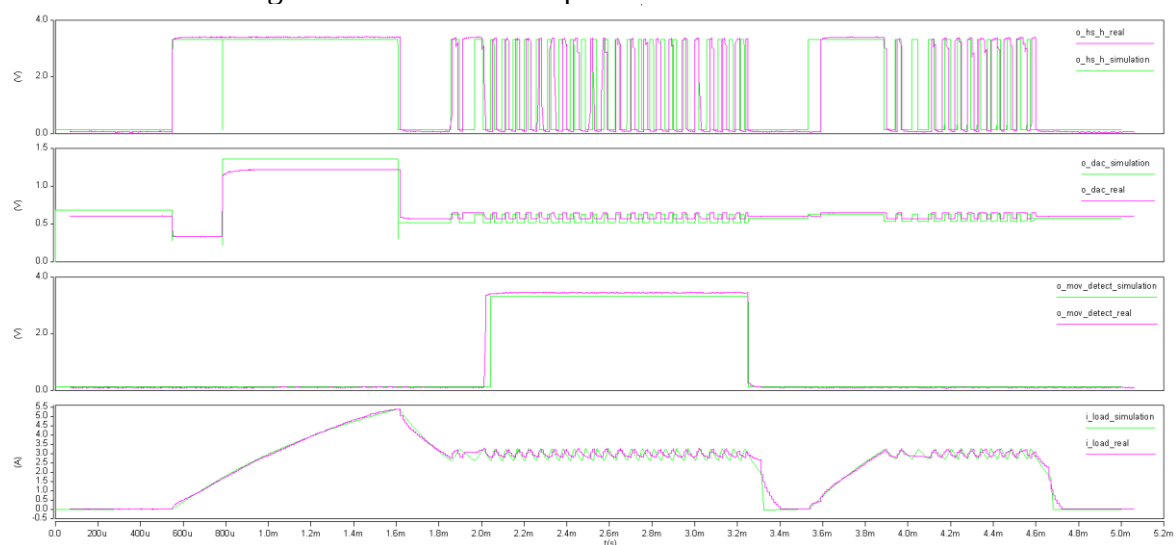


Fig. 5.3 Rezultate de simulare în comparație cu cele reale .

Începând din partea de jos, semnalele "i load" (reprezentând curentul de sarcină) se potrivesc foarte bine între ele, existând o diferență în faza de „hold” atunci când supapa se închide, curentul real are o schimbare specifică a inductanței rezultând o fază de ON mai lungă pentru controlul HS. Modelul de sarcină nu este capabil să reproducă exact același efect, dar își schimbă valoarea de inductanță, rezultând după aceea o comutare mai rapidă exact ca în cazul validării reale.

Cealaltă diferență este la sfârșitul fiecărui puls de curent, panta curentului în faza de clamping este diferit reprodusă în comparație cu cea reală (în simulare disiparea puterii datorită tensiunii electromotoare induse este mai rapidă chiar dacă fenomenul a pornit de la același curent, indicând o inductanță puțin mai mică la sfârșitul profilului de curent pentru modelul de simulare comparație cu bobina reală DIV).

Capitolul 6: Detecția mișcării, reducerea vibrației și diagnoza electrică a valvei DIV

Capitolul 6 tratează îmbunătățirea metodelor de control a solenoidului DIV, reducerea zgomotului produs de vibrațiile supapei valvei DIV la închidere și deschidere și găsirea unei metode de diagnoză electrică mai ieftină pentru circuitul de comandă a solenoidului DIV. Cu ajutorul platformei de simulare și a modelelor dezvoltate în Capitolul 5 am propus și simulat un algoritm de detecție a mișcării supapei valvei DIV; totodată am propus și simulat o metodă de diagnostic și protecție a circuitului de comandă. În final utilizând platforma MTP realizată am testat implementarea celor 3 contribuții originale: algoritmul de reducere a zgomotului, algoritmul de detecție a mișcării și diagnoza/protecția circuitului de comandă a solenoidului DIV.

Dat fiind faptul că sarcina controlată (valva DIV) este un element electro-mecanic cu elemente cinetice, care se mișcă în momentul în care bobina este energizată, apar fenomene de vibrație când părțile care se mișcă își ating poziția/limitele de capăt. Aceste vibrații generează efecte fonice nedorite, de aceea s-a propus o modalitate de frânare a mișcării fără a afecta funcționalitatea normală a valvei și totodată a reduce zgomotul produs de vibrații atunci când valva se închide sau deschide.

Pentru a ne asigura că procedura de reducere a zgomotului nu afectează funcționarea normală a valvei este nevoie să ne asigurăm că valva s-a deschis/închis pentru a asigura cantitatea de combustibil necesar în rampa comună, de aceea am implementat un algoritm care detectează acest lucru. Următoarele două subcapitole vor trata subiectul reducerii zgomotului produs de supapa valvei DIV și a algoritmului de detecție a mișcării supapei.

În Fig. 6.1 este prezentată modalitatea de reducere a zgomotului prin aplicarea a unui profil cu două pulsuri de curent controlate în buclă închisă. Este vizibil că scade amplitudinea vibrației la închiderea valvei și aproape că dispare zgomotul la deschiderea valvei în cazul în care se aplică algoritmul de reducere a zgomotului.

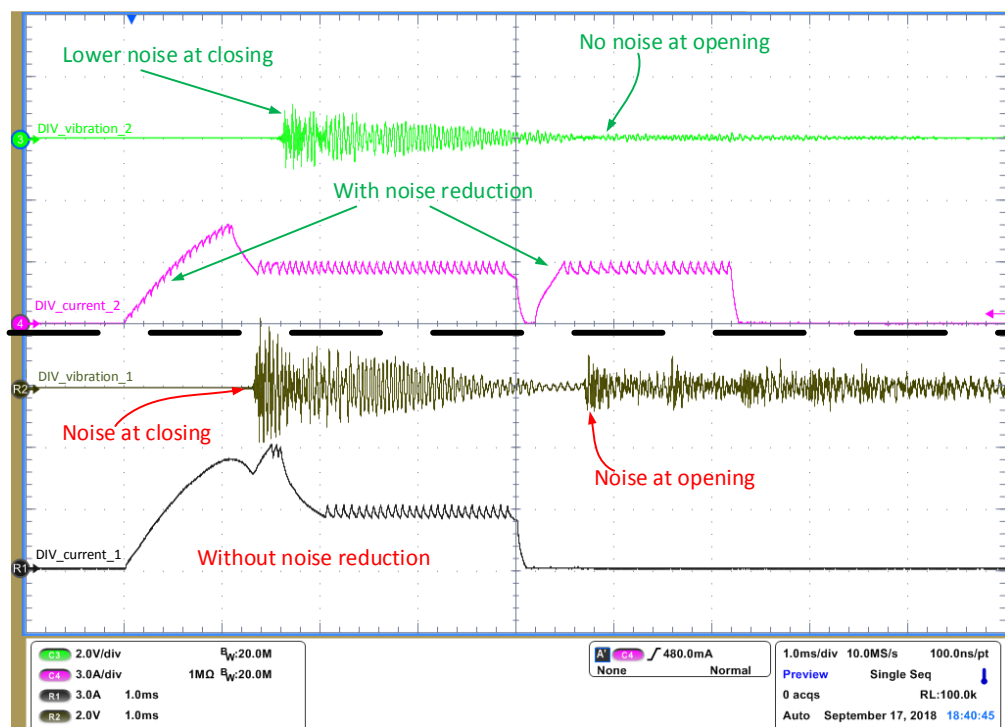


Fig. 6.1 Zgomotul generat de închiderea/deschiderea valvei DIV cu și fără reducerea zgomotului

Adițional propun o metodă de diagnoză; detectarea supra-curentului (OC) și a sarcinii-neconectate(OL), adecvată pentru ambele tipuri de sarcini comandate, cele cu caracter predominant inductiv, dar și pentru cele cu caracter rezistiv. În cazul sarcinilor rezistive se reutilizează DAC-ul pe 8 biți din blocul de CLCC (oricum neutilizat pentru că în cazul în care este controlată o sarcină rezistivă funcția de CLCC nu se poate folosi).

În Fig. 6.2 se prezintă metoda propusă de diagnostic folosind un numărător implementat în blocul digital în locul unui convertor digital-analog implementat în tehnologie analogică.

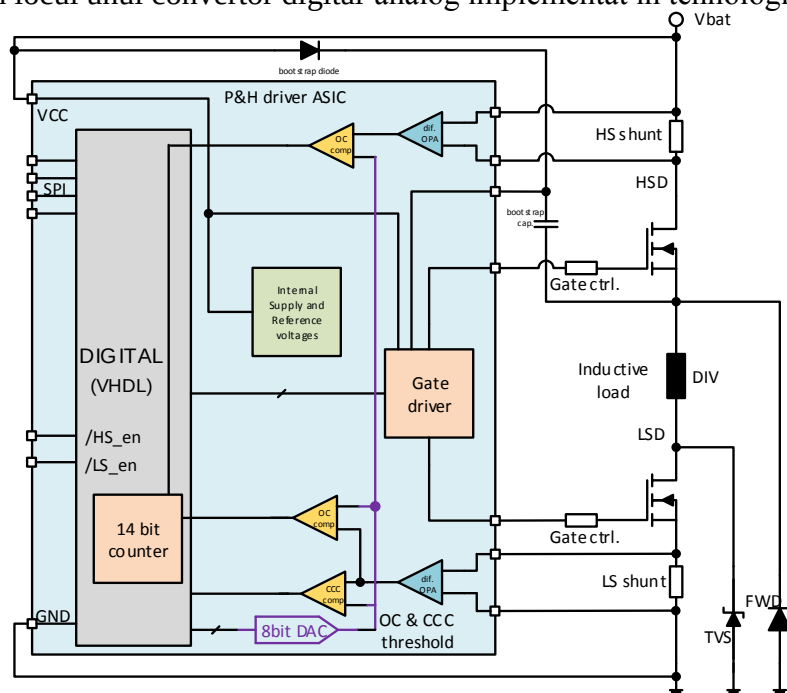


Fig. 6.2 Arhitecturade diagnoză și protecție propusă

Simularea a avut un rol foarte important în dezvoltarea algoritmului de detectare a mișcării. În FIG 6.3 este prezentată diagrama de stare a algoritmului de detectare a mișcării supapei din valva DIV.

Algoritmul este implementat numai pentru pulsul de închidere a valvei DIV și pornește de îndată ce mașina de stare (implementată în VHDL) care controlează blocul CLCC sare de la faza "idle" (inactiv) la "peak" (vârf). Toate contoarele și variabilele utilizate pentru implementarea algoritmului sunt inițializate în acest moment. Algoritmul așteaptă până când mașina de stare va sări de la faza "peak" la "hold"(menținere). De îndată ce faza "hold" este activă, există două contoare concurente, unul măsurând durata de ON a tranzistorul HS iar celălalt durata de OFF.

Scopul algoritmului este de a detecta în faza "hold" schimbarea de la frecvență de comutarea lentă la mai rapidă a tranzistorului HS (pentru a menține aceleași limite de curent programate), cauzată de scăderea inductanței după închiderea valvei.

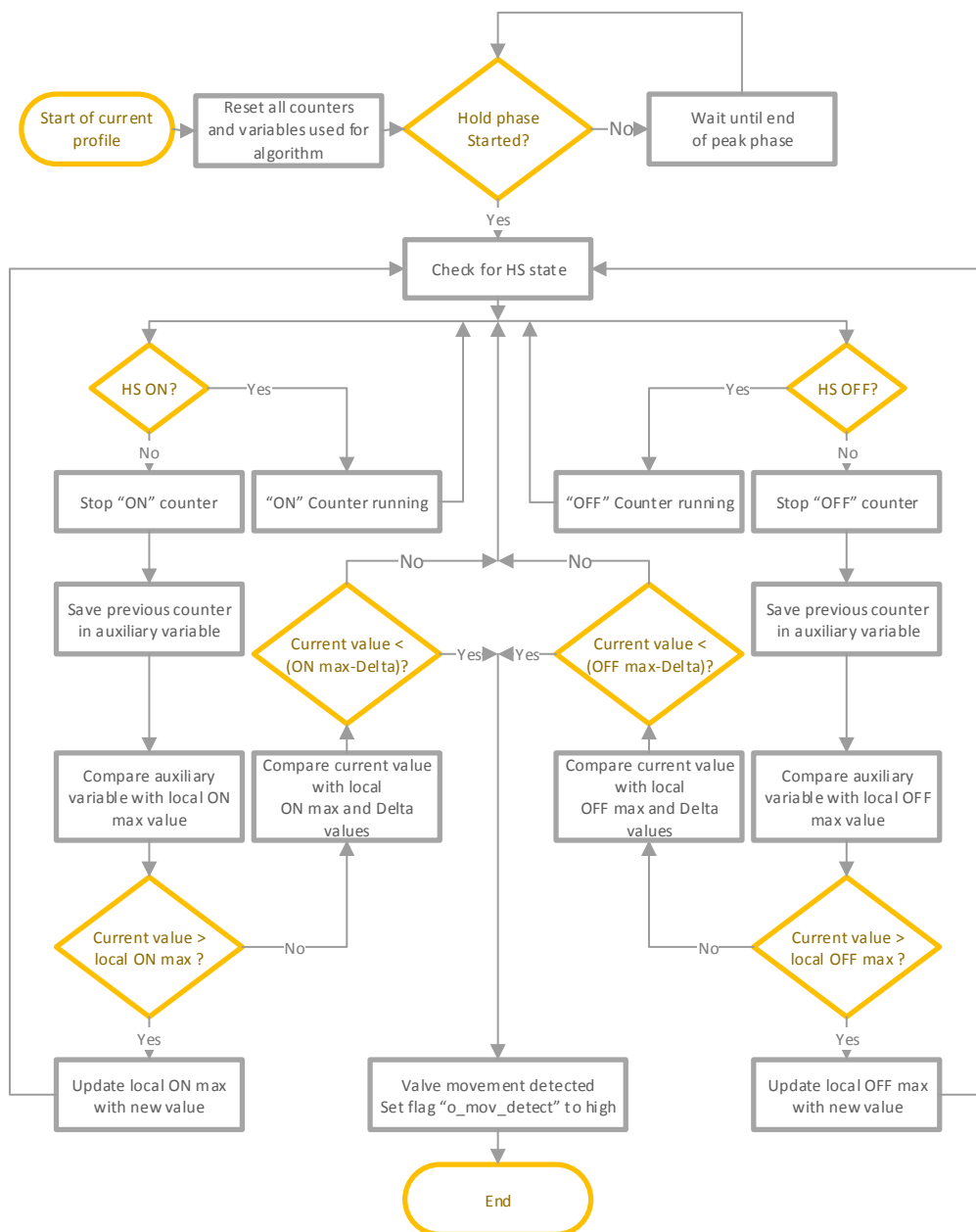


Fig. Error! No text of specified style in document.3 Algoritmul de detecție a mișcării supapei: diagrama de stări .

Datorită profilului de control specific, când curentul scade de la valoarea de „peak” până la curentul de „hold” prima durată de OFF a tranzistorului HS este ignorată, deoarece ar rezulta o valoare maximă prea mare comparativ cu restul duratelor de OFF din faza de „hold”. Pornind de la cel de-al doilea eveniment de OFF și primul eveniment de ON al tranzistorului HS, algoritmul măsoară constant durata de OFF și ON a tranzistorului HS.

Pentru a evita detectarea falsă a mișcării supapei datorită unei diferențe prea mici între două măsurări consecutive, este necesară o "valoare delta", care este predefinită într-un registru al ASIC-ului.

Contributii Personale

Capitolul 3

1. Propun pe baza lucrării proprii „3” platforma de validare MTP care poate îmbunătăți performanța de validare datorită utilizării directe a TC-ului pentru a emula blocul analogic al ASIC-ului. MTP poate fi reutilizat cu ușurință pentru noile proiecte datorită structurii sale modulare.
2. Propun tot pe baza lucrării proprii „3” două arhitecturi alternative de simulare a sistemelor fără Co-simulare cu număr redus de simulatoare utilizând un simulator multi-domeniu pentru a realiza simularea de arhitectură a ASIC-ului atunci când încă nu este disponibilă platforma MTP de emulare a ASIC-ului pentru a realiza teste și a valida funcționalitatea conceptuală corectă în faze timpurii ale dezvoltării ASIC-ului.

Capitolul 4

3. Prezint și explic pe baza lucrării proprii „6” funcționarea sistemelor GDI cu detalii specifice referitoare la acționarea mecanică și comanda electronică a pompei de benzină de înaltă presiune, respectiv a valvei electro-mecanice constituate a acesteia.

Capitolul 5

4. Am efectuat un număr semnificativ de scenarii de funcționare cu diferite medii de testare și am analizat datele așa cum se arată în lucrarea mea „4”.
5. Am propus un model nou de simulare comparativ cu modelul simplu R-L așa cum se arată în „4”. Noul model este capabil să urmărească mai bine forma curentului prin valvă fizică utilizată datorită structurii tabelului de căutare.
6. Am modelat de asemenea comportamentul de saturație al inductanței, sporind performanța generală a simulării.
7. Modelul de simulare electrică a bobinei DIV propus în „4” a fost îmbunătățit prin introducerea efectului temporal pentru a simula închiderea supapei. Închiderea supapei s-a introdus prin modificarea inductivității bobinei în funcție de un semnal sincronizat cu începutul profilului de curent și se manifestă prin modificarea frecvenței de comutație în faza de „hold” atunci când se folosește strategia de CLCC pentru reglarea de curent prin bobină așa cum se arată în subcapitolul 5.1.5.7.
8. Am realizat o platformă de simulare conform cu cea propusă în lucrarea mea „3” și explicată în subcapitolul 3.2.3
9. Am realizat o platformă de validare conform cu cea propusă în „3” și explicată în subcapitolul 3.1.1 cu excepția implementării etapei în care am propus să se înlocuiască blocul analogic emulat cu componente discrete cu o placă care să conțină Test Chip-ul (Test Chip-ul este un CI care conține doar blocuri analogice a fi implementate în ASIC).
10. Am comparat rezultatele platformei de simulare cu rezultatele platformei de validare. Comparăția arată o potrivire bună a rezultatelor, demonstrând eficiența platformei de validare.

Capitolul 6

11. Am analizat stadiul actual al metodelor existente de reducere a zgomotului, așa cum se arată în lucrarea mea „6”.
12. Am a propus o metodă alternativă de reducere a zgomotului, așa cum se arată în lucrarea mea „6”.
13. Metoda alternativă propusă de mine și publicată în „6” se bazează pe [19], însă nu necesită o intrare de ADC pentru monitorizarea constantă a curentului prin bobina DIV.
14. În metoda propusă de mine în „6”, ADC-ul necesar în [19] este înlocuit cu un numărător implementat în VHDL, mult mai ieftin și simplu față de un ADC. Acest lucru este posibil datorită controlului de curent în buclă închisă care aproximativ își dublează frecvența de comutare în momentul în care se închide valva DIV. Această frecvență de comutație este monitorizată de numărătorul implementat în VHDL care detectează pe baza algoritmului prezentat în Fig. Error! No text of specified style in document.. mișcarea valvei DIV.
15. Metoda propusă de mine aplică ambele pulsuri, cel de închidere și cel de deschidere cu control de curent în buclă închisă.
16. Am testat pe o valvă DIV folosită în sistemele de 250bari eficiența metodei propuse de reducere a zgomotului, așa cum se arată în lucrarea mea „6”.
17. Am pus accent pe rezolvarea problemelor (mișcarea precoce a valvei și variația timpului natural de închidere/deschidere a valvei) găsite la implementarea modalității propuse de reducere a zgomotului, așa cum se arată în „6”.

Lucrări științifice publicate în volumele unor manifestări științifice

1. L. Molnar and A. Gontean, "Functional Simulation methodes," 2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC), Timisoara, 2016, pp. 198-202. (WOS:000390717800044)
2. L. Molnar and A. Gontean, "Fault simulation methodes," 2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC), Timisoara, 2016, pp. 194-197. (WOS:000390717800043)
3. L. Molnar and A. Gontean, "ASIC validation architectures: Hardware versus simulation," 2017 21st International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, 2017, pp. 671-676. (WOS:000427419900111)
4. L. Molnar, A. Gontean, A. Kotlar and P. Svasta, "Simulation model for automotive high pressure fuel pump," 2017 IEEE 23rd International Symposium for Design and Technology in Electronic Packaging (SIITME), Constanta, 2017, pp. 231-234. (WOS:000428032300048)
5. L. Molnar, O. Luca and A. Gontean, "Simulation Method to Evaluate Thermal Runaway of MOSFETs," 2018 41st International Spring Seminar on Electronics Technology (ISSE), Zlatibor, 2018, pp. 1-6. (WOS:000449866600011)
6. L. Molnar and A. Gontean, "Hardware Difficulties and Improvements for High Pressure Fuel Pump Solenoid Valve Noise Cancellation," 2018 IEEE 24th International Symposium for Design and Technology in Electronic Packaging (SIITME), Iasi, 2018, pp. 84-89. (WOS:000466960400015)
7. L. Molnar, A. Kotlar and S. Lica, "High Power Switch using IGBT and GaN MOSFET A solution proposal," 2018 IEEE 24th International Symposium for Design and Technology in Electronic Packaging (SIITME), Iasi, 2018, pp. 43-48. (WOS:000466960400008)

Lista citărilor

Citări pentru lucrarea:

„2” L. Molnar and A. Gontean, "Fault simulation methodes," *2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC)*, Timisoara, 2016, pp.194-197.
doi: 10.1109/ISETC.2016.7781089

1. Rahul Bhattacharya, S. H. M. Ragamai, Subindu Kumar, *VLSI Design and Test*, vol. 711, pp. 179, 2017.
2. V.I. HAHANOV, I.V. IEMELIANOV, M.M. LIUBARSKYI, S.V. CHUMACHENKO, E.I. LITVINOVA, TAMER BANI AMER, "Qubit Method for Deductive Fault Analysis of Logic Circuits.", *Elektronnoe modelirovanie*, vol. 39, pp. 59, 2017

Listă patente

1. L.Molnar, B.Komlosi, „High Pressure Fuel Pump Solenoid Valve Movement Detection (2019)”, 2019E03508 RO
2. L.Molnar, B.Komlosi, „Time Constant Based Over current and Open Load diagnostic for inductive loads”, 2019E03495 RO
3. L.Molnar, B.Komlosi, ” High Pressure Fuel Pump Solenoid Valve Noise cancelling with two current controlled_current_pulses and soft landing featuring Movement Detection”, 2019E04397 RO

Bibliografie selectivă:

- [1] R. J. Hayne , “Presynthesis test generation using VHDL behavioral fault models”, Southeastcon, 2011 Proceedings of IEEE, pp. 264-267, 2011
- [2] C. Marino, M. Forliti, A. Rocchi, A. Giambastiani, F. Iozzi, M. De Marinis, L. Fanucci, “Mixed signal behavioral verification using VHDL-AMS”, Research in Microelectronics and Electronics, 2005 PhD (Volume:2), pp. 115-118, 2005
- [3] V. Jusas; T. Neverdauskas, “Stimuli generator for testing processes in VHDL”, NORCHIP, 2014, pp.1-4, 2014
- [4] Verification Methodology Manual 1.2 Standard Library User Guide, Version E-2011.03, March 2011
- [5] Verification Methodology Manual for SystemVerilog, Hunter, Alan, Nightingale, Andy, Cerny, Eduard, Bergeron, Janick, Springer (2005), ISBN 10: 0028600789
- [6] Universal Verification Methodology 1.1 User’s Guide, May 18, 2011 http://accellera.org/images/downloads/standards/uvm/uvm_users_guide_1.1
- [7] Open Verification Methodology 2.0 <https://www.doulos.com/knowhow/sysverilog/ovm/>
- [8] Jaehyun Park, Kyeongchan Ra, Younggwon Lee and Sungjoon Park, “Development of mixed signal ESC system on chip”, 2015 IEEE 11th International Conference on Power Electronics and Drive Systems, Pages: 143 - 147, DOI: 10.1109/PEDS.2015.7203571, 2015
- [9] Eric Chesters, “Role of the verification team throughout the ASIC development life cycle”, 2009 46th ACM/IEEE Design Automation Conference, Pages: 216 - 219, DOI: 10.1145/1629911.1629971, 2009
- [10] Fei Gong, Meenal Vaidya, Rishvanth Kora, Daniel Harshbarger, Brad Ulery and William Meyer, “ FPGA based prototype verification in automotive mixed signal integrated circuit development”, 2013 IEEE 56th International Midwest Symposium on Circuits and Systems (MWSCAS), 2013
- [11] Andreas Mauderer, Marvin Freier, Jan-Hendrik Oetjens and Wolfgang Rosenstiel, “Efficient digital design for automotive mixed-signal ASICs using simulink”, 2012 IEEE 15th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), Pages: 372 - 377, DOI: 10.1109/DDECS.2012.6219090, 2012
- [12] S. Alpe, G. Botto and M. De Giuseppe, “FPGA based ASIC fast prototyping system for new generation piezo injectors control units”, 6th Conference on Ph.D. Research in Microelectronics & Electronics, Pages: 1 – 4, 2010
- [13] Alejandro Cook, Dominik Ull, Melanie Elm, Hans-Joachim Wunderlich, Helmut Randoll and Stefan Döhren, “Reuse of Structural Volume Test Methods for In-System Testing of Automotive ASICs”, 2012 IEEE 21st Asian Test Symposium, Pages: 1200 - 1203, DOI: 10.1109/MWSCAS.2013.6674869, 2013
- [14] Yoichi Iihoshi, Shin Yamauchi, Ryusei Miura, Yoshikuni Kurashima, Toshio Hori,” Model-Based Control for High-Pressure Fuel Pumps”, ICROS-SICE, Fukuoka, Japan, 2009, pp. 4120-4123.
- [15] Alessandro di Gaeta, Giovanni Fiengo, Angelo Palladino, Veniero Giglio, “A control oriented model of a Common-Rail System for Gasoline Direct Injection Engine”, CDC, Shanghai, China, 2009, pp. 6614 – 6619
- [16] Tarek A. Tutunji a, Ashraf Saleem “A methodology for identification and control of electro-mechanical actuators”, MethodsX, Volume 2, 2015, pp. 219-231
- [17] A. Sarwar, C. Sankavaram and X. Lu, "Control adaptation approach for fault detection and isolation in SIDI high pressure fuel pump," 2017 IEEE International Conference on Prognostics and Health Management (ICPHM), Dallas, TX, 2017, pp. 117-123.

- [18] Jaehyun Park, Kyeongchan Ra, Younggwon Lee and Sungjoon Park, “Development of mixed signal ESC system on chip”, 2015 IEEE 11th International Conference on Power Electronics and Drive Systems, Pages: 143 - 147, DOI: 10.1109/PEDS.2015.7203571, 2015
- [19] Kraft T., Chia T. K. B., Sassler W. (2013), European Patent No. DE102013207162 (A1), Retrieved from Google Patents.
- [20] Takashi Okamoto, Hiroyuki Yamada, Kousaku Shimada, Koji Matsufuji (2002), US Patent No. US7299790B2, Retrieved from Google Patents.

