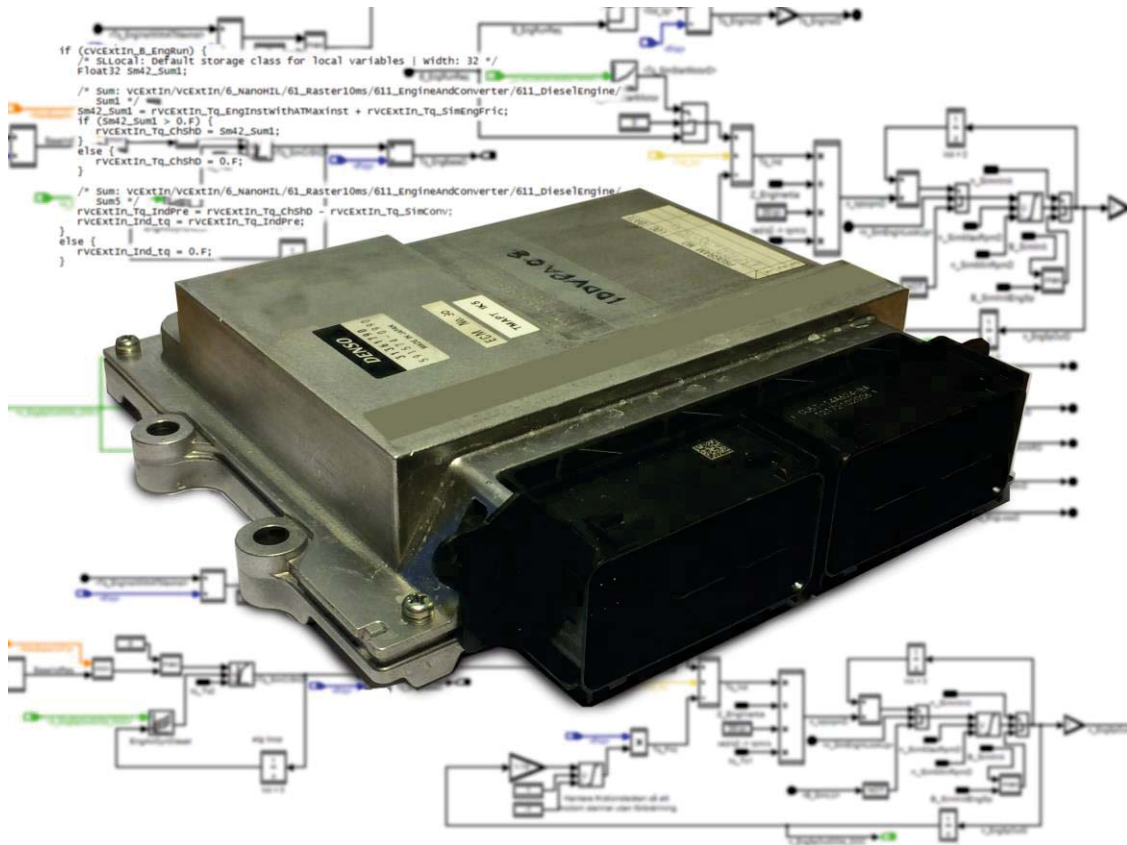




# CHALMERS



## Simulator med fordonsdynamik i ett motorstyrsystem

Vehicle dynamics simulator in an engine  
management system

Examensarbete inom högskoleingenjörprogrammet Mekanik

David Säwemark & Eric Larsson



# FÖRORD

Denna rapport är den avslutade delen på examensarbetet som vi gjorde på institutionen för signaler och system. Examensarbetet är avslutningen på utbildningen mekatronik högskoleingenjör 180hp på Chalmers tekniska högskola. Uppdraget utfördes på utvecklingsavdelningen Torque Management Controls & Calibration hos Volvo Personvagnar AB i Torslanda. Denna avdelning ansvarar för momenthantering och den framåtdrivande funktionen i motorstyrssystemet. Projektet gick ut på att ta fram en simulator för den del av motorstyrssystemet som gruppen ansvarade för. Den ska vara utformad i Simulink/TargetLink för att kunna implementeras i motorstyrenheten.

Vi, David Säwemark och Eric Larsson skulle vilja rikta ett stort tack till vår handledare på Volvo, Per Persson. Han har alltid funnits som stöd med sin expertis genom hela uppdraget. Ett tack till Fredrik Höök som var ansvarig chef för uppdraget och såg till att uppdraget kunde utföras. Sist men inte minst skulle vi vilja tacka Göran Hult som handledare på skolan som har hjälpt oss att utforma rapporten.

David Säwemark & Eric Larsson

## SAMMANFATTNING

Volvo Car Corporation och avdelningen Torque Management Controls & Calibration har för avsikt att ta fram en testtrigg för att tidigt kunna testa nya motorfunktioner. Denna testtrigg skulle möjliggöra att det i ett tidigare skede skulle gå att simulera och testa mjukvaran direkt mot styrenheten. Därför skulle mindre logik- och funktionsfel kunna felsökas och lösas i ett tidigare skede. Det skulle på så vis kunna sparas både tid och resurser genom att slippa boka testbil och ta sig ut från kontoret och koppla upp sig mot en bil. Med gamla Simulink-modeller från Volvo har det konstruerats en simulator som simulerar både motor-, växellåda- och fordonsmodell. Denna är utvecklad i Simulink/TargetLink vilket är verktyget som används för utveckling av motorstyrfunktioner i ECM:en (Engine Control Module, motorstyrenhet). Via ett speciellt funktionsblock manipulerar simulatorn insignalerna till styrenheten till att tro att den sitter inkopplad i en bil. Signaler som påverkas är till exempel gaspådrag, motormoment, motorvartal och hastighet. Denna form av simulatorkörning har tidigare varit obeprövad på avdelningen. Genom detta projekt har det visats att denna form av simulering är fullt möjlig och kan ge en tidig funktionstestning av motorstysystemet. Utöver arbetet med fordon- och motormodeller har det även tagits fram funktioner för bland annat trippmätare, bränsleförbrukning, momentneddrag vid växling samt förarmodell med förprogrammerad gas och bromscykel. Under arbetet har det också till simulatorn kopplats in verkliga motorfunktioner såsom transmission-funktioner och andra allmänna momenthanterande motorfunktioner. För att hålla tidsramen har detta arbete främst riktat sig mot att ta fram en simulator för automatväxlad dieselbil. Med hjälp av denna simulator och tillhörande testtrigg kommer man vid ett tidigare skede kunna testa och funktionssäkra motorprogramvara.

## **ABSTRACT**

Volvo Car Corporation and the department of Torque Management Controls & Calibration intends to develop a test rig for early testing of new engine features. This test rig would enable an early testing stage to simulate and test the software directly in the controller. Therefore, making the smaller logic and functional errors easier to troubleshoot and resolved at an earlier stage. It could in that way be saved both time and resources by not having to book a test car and getting out of the office and connect to a car. With old Simulink-models from Volvo, a simulator which simulates engine-, transmission- and vehicle models has been constructed. This simulator is developed in Simulink/TargetLink which is the tools used for developing engine control functions for the ECM (Engine Control Module). With a special function block manipulates the simulator the input signals to the controller to believe that it is plugged into a car. Signals that are affected, for example, throttle, engine torque, engine speed and speed. This form of simulator driving has previously been untested in the department. Through this project it has been shown that this form of simulation is entirely possible and may provide an early functional testing of engine control system. In addition to work on the vehicle and engine models have also been developed features including trip meter, fuel consumption, torque downsizing when shifting and driver model with pre-programmed acceleration and braking cycle. During the work, it has also been linked to the simulator in real engine functions such as transmission functions and other general torque handling engine functions. To meet the schedule, this work has mainly been directed towards developing a simulator for automatic transmission diesel car. Using this simulator and test rig will be at an earlier stage to test and dependable engine software.

# INNEHÅLLSFÖRTECKNING

<b>BETECKNINGAR .....</b>	<b>1</b>
<b>1. INLEDNING .....</b>	<b>2</b>
1.1 Bakgrund .....	2
1.2 Syfte .....	2
1.3 Avgränsningar .....	2
1.4 Precisering av arbetsuppgiften .....	3
<b>2. TEKNISK BAKGRUND .....</b>	<b>4</b>
2.1 Simulink.....	4
2.2 TargetLink.....	4
2.3 INCA .....	5
2.4 ECM med ETK.....	6
2.5 Utrustning för testtrigg.....	7
2.6 Fordonskomponenter .....	8
Konverter och Lock-Up .....	8
2.5 Namnkonventioner på Volvo Cars .....	8
<b>3. METOD .....</b>	<b>9</b>
3.1 Framtagning av första grundmodellen .....	9
3.2 Omarbetning av grundmodell.....	9
3.3 Datorsimulation med TargetLink .....	9
3.4 Läs- och kalibrerbara block .....	10
3.5 Uppkoppling mot styrenhet.....	10
3.5.1 Kodgenerering och kompilering av kod.....	10
3.5.2 Överföring och datasättning i INCA.....	10
3.6 Testning mot styrenhet .....	10
3.7 Ändring/utbyggnad av simulatormodellen .....	10
3.8 Uppkoppling av simulatormodell mot riktiga motorfunktioner .....	10
<b>4. GENOMFÖRANDE .....</b>	<b>11</b>
4.1 Generellt om simulatormodellen .....	11
4.2 VcExtIn – Placering av Simulatorfunktion .....	12
4.3 Utgångsmaterial .....	13
4.3.1 Engine & Converter .....	13
4.3.2 ATgearbox.....	14
4.3.3 Vehicle .....	15
4.4 Simulators huvudblock.....	16
4.4.1 Pedalmapp.....	16
4.4.2 Engine & Converter .....	17
4.4.3 ATGearbox .....	20
4.4.4 Vehicle .....	21
4.5 Uppkoppling av simulators signaler mot verkliga motorstyrfunktioner .....	22
4.5.1 Outsignals .....	22
4.5.2 Insignals .....	25
4.6 Utbyggnadsfunktioner .....	26
4.6.1 Rasterhantering.....	26
4.6.2 Tidskrets .....	27
4.6.3 Momentneddrag vid växling.....	27
4.6.4 Beräkning av total sträcka .....	28
4.6.5 Beräkning av total och momentan förbrukning .....	28
4.6.6 DSTC-Funktionalitet.....	29

4.6.7 Förarmodell .....	30
4.6.8 Cruise control/farthållare .....	31
<b>5. RESULTAT .....</b>	<b>32</b>
<b>5.1 Körfall.....</b>	<b>32</b>
5.1.1 7-100 km/h, 50 % gaspedal .....	33
5.1.2 7-100 km/h, 100 % gaspedal .....	34
5.1.3 100-0 km/h, 100 % Bromspedal .....	35
<b>5.2 Funktioner .....</b>	<b>36</b>
5.2.1 Cruise control – Cc.....	36
5.2.2 Bränsleförbrukning .....	37
5.2.3 Trippmätare, tid och hastighet .....	38
5.2.4 Momentneddrag vid däcksläpp - "DSTC" .....	39
5.2.5 Momentneddrag vid växling.....	40
5.2.6 Neutralläge .....	41
5.2.7 Förarmodell .....	42
5.2.8 Slag och såg .....	43
<b>6. SLUTSATSER .....</b>	<b>44</b>
6.1 Besvarande av frågeställningar .....	44
6.2 Utvärdering av den framtagna simulatorn .....	45
6.3 Verklighetsförankrade brister och förbättringar .....	45
6.4 Framtida utvecklingsmöjligheter för simulatorn.....	46
<b>7. REFERENSER .....</b>	<b>47</b>

## **BETECKNINGAR**

ECM- Engine Control Module, motorstyrenhet eller styrenhet.

TCM- Transmission Control Module, växellådstyrenhet.

CVC- Complete Vehicle Control.

E&C- Engine & Converter.

INCA- Integrated Calibration and Acquisition systems, dataprogram för kalibrering mot ECM.

HIL- Hardware In Loop.

ETK - Emulator TastKopf, minnes emulator.

Cc – Cruise control, farthållare.

Raster – Uppdateringsintervall/samplingsintervall.



# 1. INLEDNING

En simulator för motorstyrenheten för Volvos bilar har tagits fram för att förenkla utvecklingen av mjukvaran. I denna del förklaras projektets bakgrund, syfte, avgränsningar och precisering av arbetsuppgiften.

## 1.1 Bakgrund

Volvo Personvagnar AB eller Volvo Car Cooperation arbetar med utveckling och produktion av personbilar i premiumsegmentet. Grundades år 1927 som Volvo AB men år 1999 såldes personbilstillverkningen till Ford. Idag ägs Volvo Cars av Zhejiang Geely Holding Group, ägarbytet genomfördes år 2010 [1]. Bakgrunden till detta examensarbete ligger i att Volvo Cars vill effektivisera sin verifiering och provning av egenutvecklad mjukvara till motorstyrssystemet. Genom att simulera mjukvaran kan de lättare upptäcka fel i koden innan provkörning i skarpt läge av koden sker i bilen.

## 1.2 Syfte

Uppdraget går ut på att ta fram en simulator som ska kunna testa mjukvaran till motorstyrningen som utvecklas internt på Volvo Cars. Simulatorens ska tas fram och utvecklas i Simulink/TargetLink som är Volvos utvecklingsverktyg för motorstyrningen. Detta leder till att i ett tidigare skede kan upptäcka fel, vilket medför både kostnads- och tidsbesparing.

## 1.3 Avgränsningar

Modellerna i simulatorens ska på ett enkelt sätt kunna realisera både ett fordons och en motors funktion. Det finns inga krav att det på detaljnivå ska vara en exakt simulering av verkligheten. Det är bedömningen av handledaren på företaget som bestämmer hur noga detaljnivån på simulatorens ska vara. Funktionalitet för dieselmotormodellen kommer ha prioritet över bensen på grund av komplexiteten hos bensenmodellen. Det kommer dock skapas förutsättningar för att kunna konfigurera en bensenmodell i simulatorens för senare påbyggnad.

Volvo har sina programvaror och verktyg specificerade, det finns inga valmöjligheter att använda andra programvaror eller verktyg för att utföra arbetet. Simuleringen och modellerna ska rymmas i en utvecklingsstyrenhet, ingen extra hårdvara skall eller får användas.

## 1.4 Precisering av arbetsuppgiften

Motorstyrningen som utvecklas på avdelningen måste i dagsläget testas direkt i bilen. Tanken är att simulatören ska underlätta för funktionsutvecklaren genom att hitta grundläggande fel i programkoden innan det testas i bilen. Simulationen kommer kunna utföras på avdelningen i form av en liten HIL (Hardware In the Loop).

Frågeställningar:

- *Vad behövs för parametrar/ingångar för att få en realistisk simulator?*
- *Vilken funktionalitet önskas av simulatören?*
- *Kompatibelt för andra fordonsplattformar?*
- *Påbyggnadsmöjligheter efter vårt arbete?*

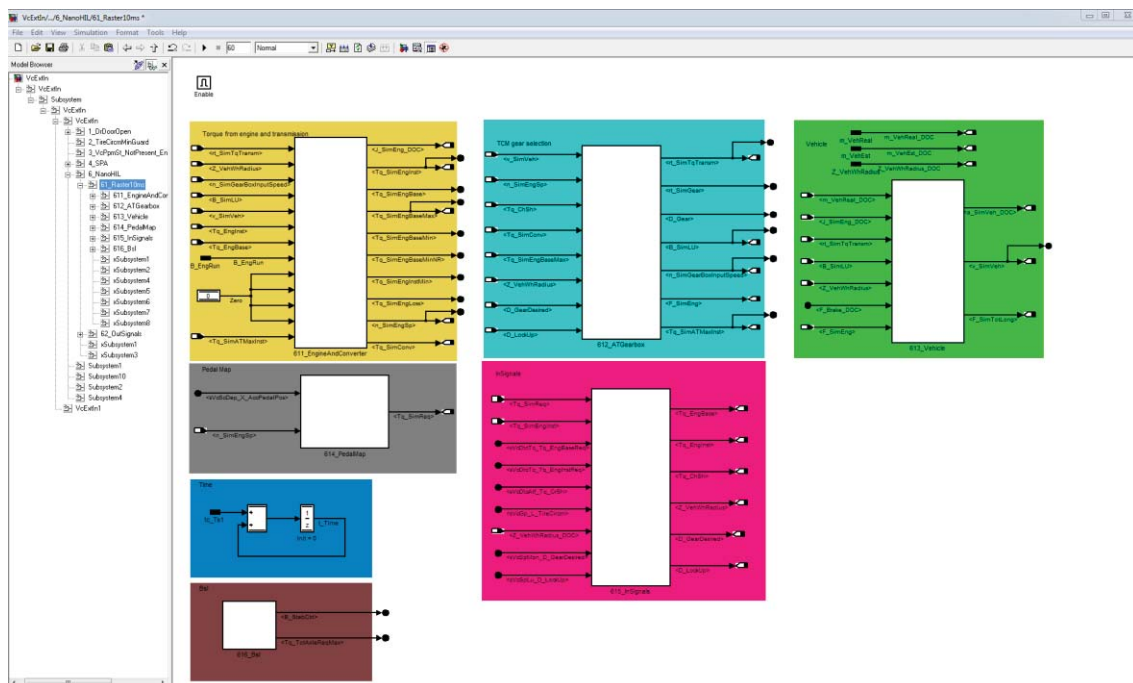
## 2. TEKNISK BAKGRUND

Här beskrivs de tekniska delarna i projektet vilket kan behövas kompletteras för att få en djupare förståelse av rapporten.

Dator, programvaror och licenser som använts under detta arbete har tillhandahållits av Volvo Cars.

### 2.1 Simulink

Simulink är ett tillägg till Matlab som används för simulering, signalbehandling och verifiering av modeller. Basprogrammet Matlab är ett beräkningsprogram från MathWorks som är utvecklat för att kunna utföra både matematiska och tekniska beräkningar. Simulink används i detta arbete som plattform för TargetLink och funktionsutveckling av motorstyrfunktioner. Det används även för simulering av flödesmodeller. Se Figur 2.1. [2]



Figur 2.1. Exempel på Simulinkmodeller, hämtad från simulatorn.

### 2.2 TargetLink

TargetLink är en toolbox till Simulink som används för kodgenerera C-kod direkt från flödesmodeller i Simulink. Möjligheten finns även att simulera TargetLink-block direkt i Simulink. Detta utvecklingsverktyg är speciellt framtagen för utveckling av kod för realtidssystem. För en stabil och säker kodgenerering. [3]

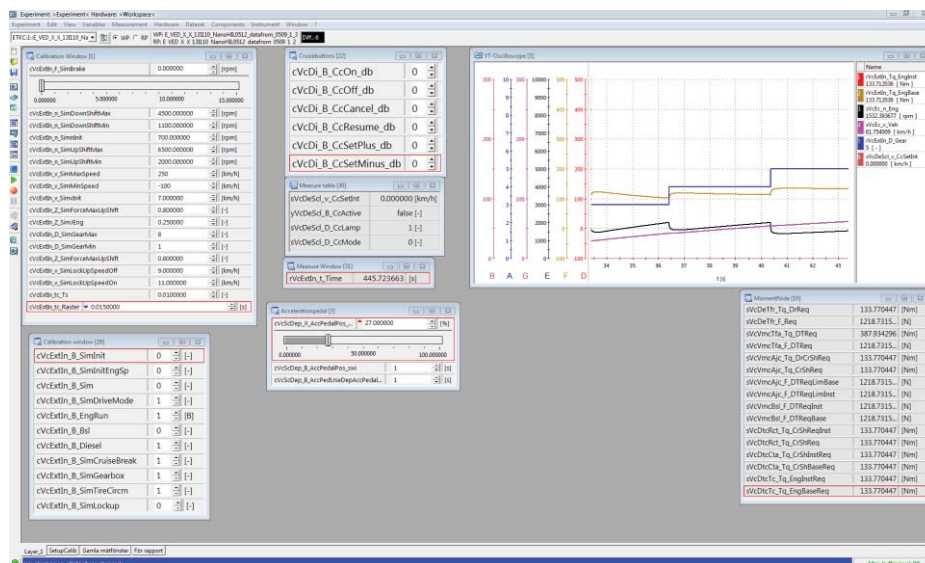
## Symbolbeskrivning Targetlink-block:

- In- och utport från block:
- ”Gåttill” och ”Gåfrån” inom block:
- Multiplikations block:
- Switch:
- Fast konstant:
- Kalibrerbar konstant:
- Merge-block:
- Tabell:
- Begränsare:
- Sampelfördröjning:

## 2.3 INCA

INCA är ett mjukvaruverktyg för kalibrering, mätning och hantering av kalibreringsdata för styrsystem inom fordonsindustrin.

En dator med INCA installerat kopplas upp mot en dubbelportad RAM-enhet som speglar ECM:en via en ETAS-modul. Härifrån kan programmet överföra ny programkod, samla in kördata och kalibrera utvalda variabler i styrenheten. Se simuleringsmiljö i Figur 2.2 [4]



Figur 2.2. Exempel på INCA-miljö för simulering, tagen från ett experiment i projektet.

## 2.4 ECM med ETK

ECM:en är en av många styrenheter i bilen. Se Figur 2.3. Det är den styrenheten som kontrollerar och styr motorn i bilen. Den ansvarar för bland annat bränsleinsprutning, tändning av tändstiften i förbränningskammarna och farthållare. ECM:en läser in gaspedalens position från föraren och omvandlar det till begärt moment till motorn och ser till att det levereras ut på vevaxeln till medbringarplåten eller kopplingen, beroende på om det är en automat- eller manuellväxellåda.

Andra uppgifter som ligger under ECM:en är:

- Tryck och temperatur för motoroljan
- Luftflödesstyrning till motorn
- Dependability (Säkerhet) - övervakning och säkerhet för motorn
- Momentförluster för generator och kompressor för luftkonditionering
- Växlingspunkter
- Körfunktioner – startstop, hybrid med flera

I ECM:en finns det tre olika raster eller uppdateringsintervaller, 10ms, 40ms och 160ms. Beroende på vilket krav som ställs på en funktion så läggs den i rätt raster, vissa funktioner behöver inte lika hög uppdateringshastighet.

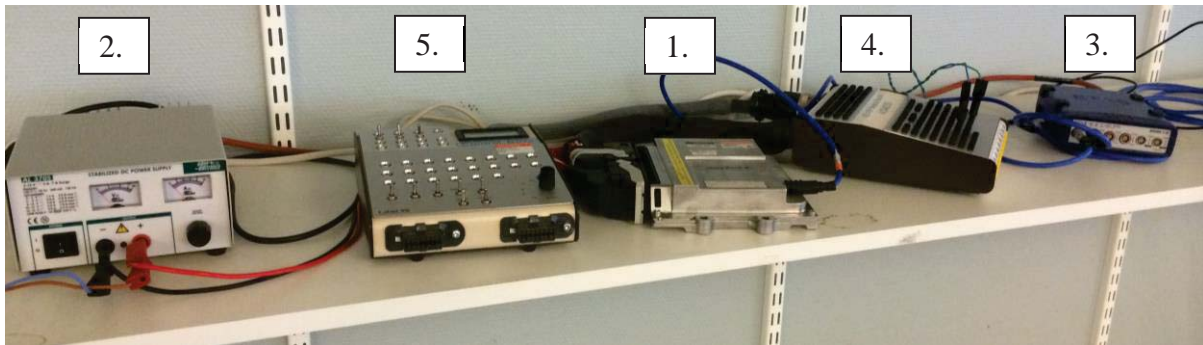


*Figur 2.3. Detta är ECM:en som användes i projektet.*

## 2.5 Utrustning för testrigg

För att kunna genomföra projektet och få igång ECM:en utan att koppla upp den mot en bil behövdes en del testutrustning. All utrustning tillhandahölls av Volvo och var färdig för att användas vid projektets uppstart. En lista med alla enheter i testriggen, numrering refererar till Figur 2.4:

1. En ECM:  
En motorstyrenhet från en bil, den har i uppgift att köra motorns programvara. Den innehåller även projektets simulatorprogramvara i minnet och exekverar den på samma gång som motorprogramvaran.
2. Ett spänningsaggregat:  
För att kunna spänningssätta testriggen.
3. En ETAS - interface module:  
Den har i uppgift att kommunicera med ECM:en och överföra mjukvara. Den utnyttjar ETK modulen i ECM:en för att komma åt det dubbelportade RAM-minnet.
4. En brygga:  
Det är en kopplingsplint som gör det möjligt att komma åt alla signaler som skickas till/från styrenheten via kabel. Denna modul används ej under projektet, utan skickar bara signalerna vidare.
5. En huvudstyrenhet:  
Huvudstyrenhetens funktion i testriggen är att styra spänningen till ECM:en, det finns en switch på den som heter ”ignition” (tändning) och den ersätter startnyckeln i en bil.



Figur 2.4. Testriggen med alla enheter, numreringen hänvisar till nummerlistan ovan.

## 2.6 Fordonskomponenter

### Konverter och Lock-Up

En konverter är en drivlinekomponent som sköter kraftöverföring mellan motor och automatväxellåda. Den har som uppgift att se till att bilen får en mjuk start från stillastående. Denna realiseras genom att en impeller och en turbin vilka utan mekanisk låsning till varandra omsluts av hydraolja. När ett moment läggs på från motorn börjar impellern snurra och pumpa runt olja i konverterhuset. Turbinen börjar då sakta rotera och föra vidare momentet till växellådan och därefter till hjulen. Denna metod av momentöverföring är ineffektiv men levererar istället en mjuk och oscillationsfri acceleration vid lågfartskörning. Då denna funktion endast är önskvärd vid låg fart finns en mekanisk låsning implementerad. Även en halvsluten låsning som tillåter slirning finns att använda. Denna låsning på konvertern aktiveras då fordonet överstiger en hastighet på ca 10 km/h. Denna mekaniska låsning mellan impeller och turbin kallas Lock-Up. [5]

## 2.5 Namnkonventioner på Volvo Cars

På stora teknikorienterade företag såsom Volvo där många signaler, filer och system cirkulerar är det nödvändigt att använda någon form av systematik vid namngivning och uppbyggnad av mjukvarusystem.

Variabeln kan till exempel heta:  
sVcEc\_v\_Veh

Första bokstaven: 'X'VcEc\_v\_Veh

- r lokal läsbar fysik variabel
- x lokal läsbar logisk variabel
- s global läsbar fysik variabel
- y global läsbar logisk variabel
- c kaliberbar variabel
- t 1-dimensions tabell
- m 2-dimensions tabell

Andra bokstavsparet: s'XX'Ec\_v\_Veh

Vc står för att variabeln tillhör Vehicle Control och indikerar egenutvecklad mjukvara.

Tredje bokstavsparet: sVc'XX'\_v\_Veh

Ec står för vilken funktionsblock variabeln tillhör, nu i detta fall tillhör variabeln Engine Control.

Sjätte bokstaven: sVcEc\_'X'\_Veh

v är enheten för variabeln, i detta fall är det en hastighet.

De sista bokstäverna: sVcEc\_v\_'XXX'

Veh är bara en beteckning för vad den variabeln mer detaljerat tillhör. I detta exempel står det för vehicle (fordon).

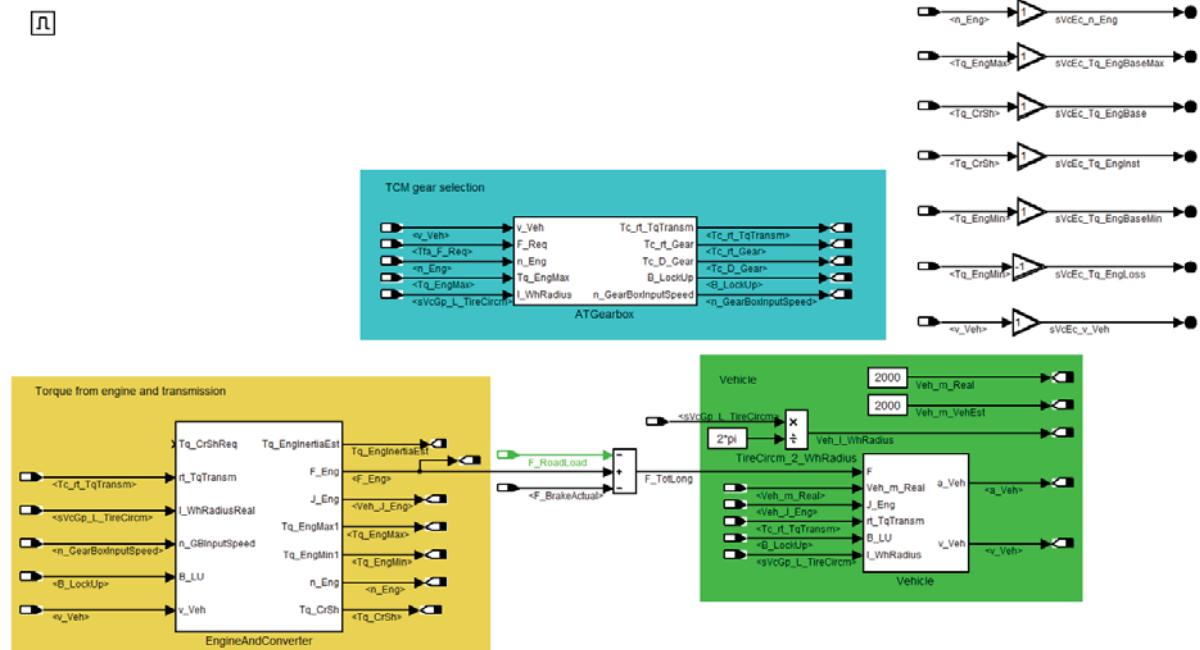
Sammanfattat är variabeln sVcEc\_v\_Veh en global fysik signal som tillhör motorstyrenheten och är en hastighet på fordonet.

### 3. METOD

Arbetet inleddes genom att formulera och förtydliga omfattningen av projektet. Det stod här klart att arbetet skulle bli väldigt skalbart beroende på hur snabbt utvecklandet av modellerna fortskred.

#### 3.1 Framtagning av första grundmodellen

Tillsammans med handledare på Volvo synades de gamla modellerna noggrant och ett urval av funktioner valdes för att fortsätta jobba med. De funktioner som valdes för att bilda grundmodellen sammanslogs i tre övergripande funktioner: *Engine & converter*, *ATgearbox* och *Vehicle*. Se Figur 3.1.



Figur 3.1. Simulink-vy över de tre grundfunktioner vilket ingick i första grundmodellen.

#### 3.2 Omarbetning av grundmodell

Eftersom grundmodellen var skapad för att kunna simuleras i datorn via Simulink behövde den omarbetas från Simulink-block till Targetlink-block. Både automatisk och manuell konvertering var tvungen att användas för att kunna få modellen helt Targetlink-kompatibel.

#### 3.3 Datorsimulation med TargetLink

För att funktionstesta alla ändringar som genomfördes vid övergången från Simulink-block till Targetlink-block behövde modellen provköras i Simulinks simulering. Först behövde dock en signal som representerar gaspedalen kopplas in. Det gick nu via scope att se hur modellen reagerade på olika gaspådrag. Även andra småfunktioner implementerades i modellen.



### **3.4 Läs- och kalibrerbara block**

Innan första kodgenerering och export av modellen skulle genomföras behövde de flesta blocken ställas in för att kunna läsas eller kalibreras. För ett vanligt styrsystem i motorn är det ofta tvunget att prioritera vilka variabler och block som ska vara läsbara på grund av begränsad datamängd i styrenheten. I detta fall kunde det förbises, i princip alla variabler i simulatoren kan läsas för att lättare kunna felsöka och åtgärda fel.

### **3.5 Uppkoppling mot styrenhet**

Dessa steg i underrubrikerna måste genomföras för att överföra ny programvara till ECM:en.

#### **3.5.1 Kodgenerering och kompilering av kod**

TargetLink genererar C-kod från den blocklogik som byggts upp i Simulink. Därefter kompileras C-koden till maskinkod via programmet High-performance Embedded Workshop.

#### **3.5.2 Överföring och datasättning i INCA**

Den kompilerade koden hämtas nu in i INCAs filhanteringssystem. Här datasätts simuleringskoden med rätt kalibreringsvariabler med en toolbox i INCA, CDM (Calibration Data Manager). Dessa kalibreringsvariabler är tagna från Volvos egen kalibreringsdatafil med komplettering från simulatorns variabler. INCA flashar sedan ECM:en och programmerar med den nya koden.

### **3.6 Testning mot styrenhet**

När all datasättning och överföring av programkoden är klar startas ett experiment upp i INCA. I experimentetmiljön kan testningen utföras genom körning av modellen mot ECM:en.

### **3.7 Ändring/utbyggnad av simulatormodellen**

Om fel uppstod vid testning av modellen i styrenheten ändras modellen i Simulink. Skulle funktionaliteten av modellen var som förväntad, påbörjas ytterligare en ny funktionspåbyggnad på modellen i Simulink.

Detta arbetssätt upprepades från punkt 3.5.1 tills en godtycklig grundfunktion uppnåts så att vidare utbyggnad kunde ske.

### **3.8 Uppkoppling av simulatormodell mot riktiga motorfunktioner**

Arbetsgången var efter grundmodellens framställande att i huvudsak steg för steg koppla in och testa funktionaliteten hos varje verkligt motorstyrblock.

## 4. GENOMFÖRANDE

Detta kapitel inleds med en kortare presentation av simulatören och var den kommer bli placerad i ECM:en.

Genomförandet av detta arbete kan sedan delas upp i fyra delar:

**Utgångsmaterial:** beskriver det utgångsmaterial som tillhandahålls av Volvo.

**Simulatorns huvudblock:** behandlar främst uppbyggnad och funktionssäkring av de 4 grundblocken som gjordes under projektet.

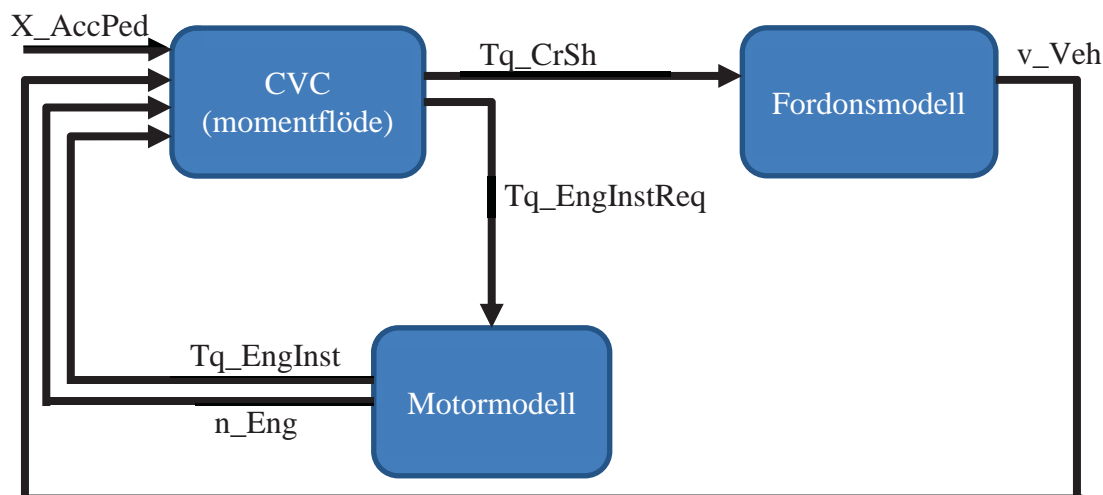
**Uppkoppling med simulatorns signaler mot verkliga motorstyrfunktioner:** behandlar utbyggnad, implementering och återkoppling av in- och utsignalsblock. Detta för att kunna koppla upp mot verkliga/befintliga signaler och motorfunktioner i styrenheten.

**Utbyggnadsfunktioner:** omfattar de funktioner som under projektet skapats och kopplats mot styrenheten.

Då flera av de modeller som presenteras i de kommande kapitlen är relativt stora kan det vara svårt att utläsa innehållet. Därför kommer större och tydligare bilder presenteras som bilagor.

### 4.1 Generellt om simulatören

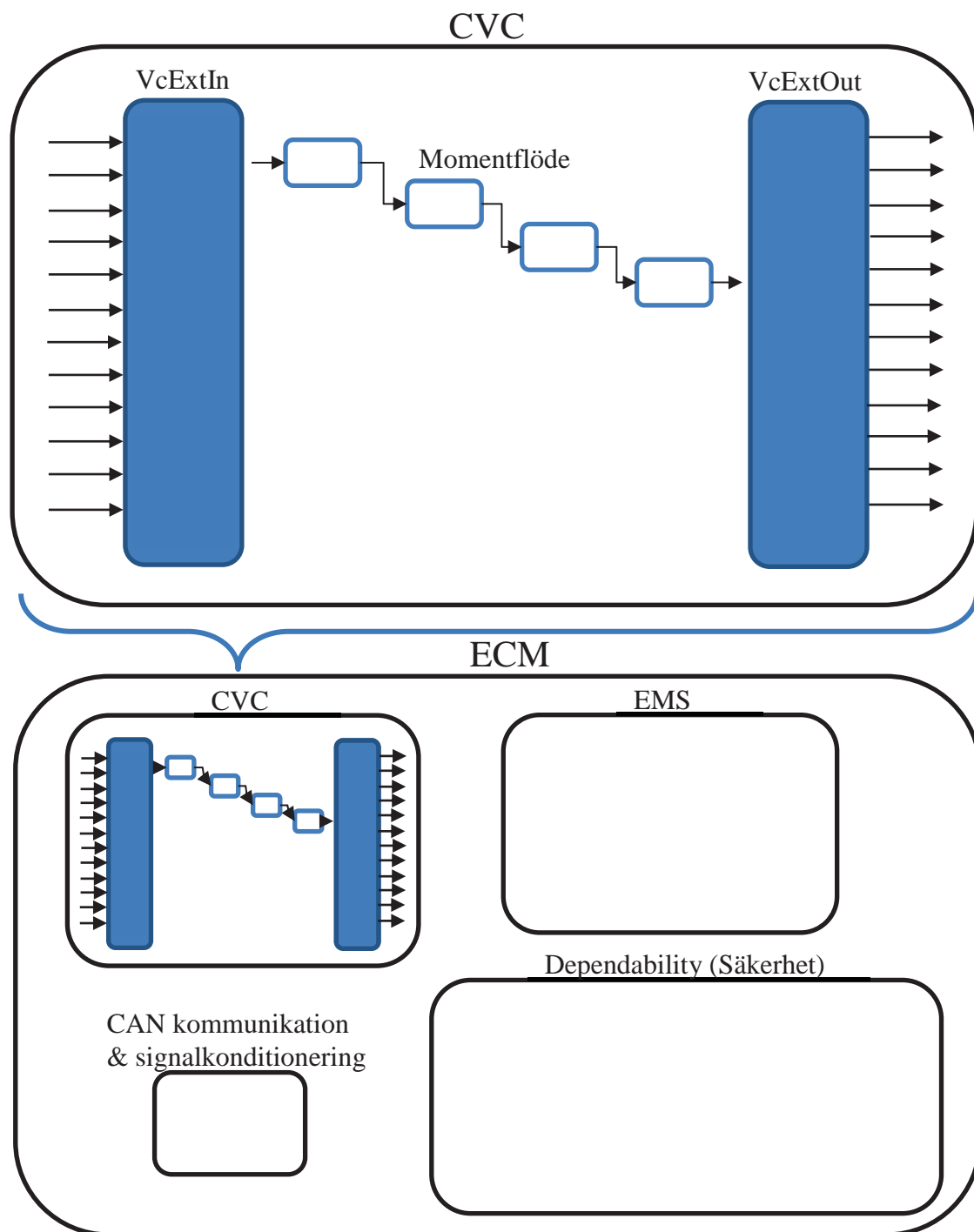
Simulatören har som uppgift att simulera de fordons- och motorsignaler som i vanliga fall styrenheten skulle få från bilen. I Figur 4.1 finns en översiktsbild av signalutbytet som sker mellan simulatören och motorstyrsystemet. Accelerationssignalen  $X_{AccPed}$  tas in i CVC och behandlas till ett begärt moment  $Tq_{EngInstReq}$ . Denna signal skickas sedan vidare till motormodellen och omvandlas till producerat moment  $Tq_{EngInst}$  och motorvarvtal  $n_{Eng}$ . Dessa signaler leds sedan in i CVC och behandlas ytterligare en gång och omvandlas till vevaxelmoment  $Tq_{CrSh}$ . Därefter tar fordonsmodellen emot signalen och omvandlar den till en hastighet  $v_{Veh}$  vilken skickas tillbaka till CVC. Detta är en enkel modell över hur signalutbytet sker.



Figur 4.1. Denna schematiska bild förklarar enkelt signalutbytet mellan modellerna.

## 4.2 VcExtIn – Placering av Simulatorfunktion

VcExtIn är en funktion som finns med i ECM:en för att kunna göra speciella ingrepp på utvalda signaler. Det är via den som man kan "lura" ECM:en att den sitter inkopplad mot en bil och får in signaler som till exempel gaspådrag, hastighet och motorvarv. Därför är det i VcExtIn som alla simulatorfunktioner måste placeras för att de senare ska kunna kopplas ut mot de verkliga styrfunktionerna. I Figur 4.2 illustreras VcExtIn's placering i CVC-blocket och sedan CVC-blockets placering i ECM:en med några exempel på andra funktioner som ECM:en har hand om.



Figur 4.2. Förenklad översiktsbild på VcExtIn-blocket i ECM:en.

De variabler vilka simulator-funktionerna simulerar och beräknar skickas ut och skriver över de verkliga signalerna till momentflödet (motorstyrfunktionerna).

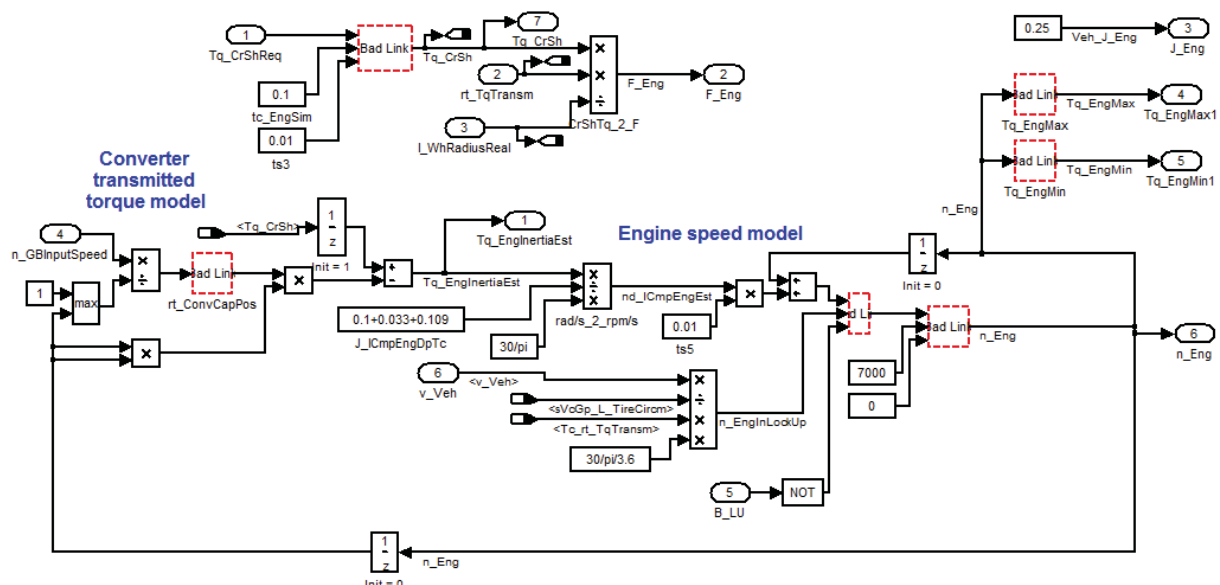
I ECM:en exekveras VcExtIn i alla raster, detta skulle komma att orsaka ett problem då tidshanterande funktioner fick oregelbunden exekvering. Detta problem är något som behandlas senare i rapporten.

## 4.3 Utgångsmaterial

Under denna rubrik beskrivs originalmodellerna som tillhandahölls av Volvo vid projektets start. Dessa granskades och inventerades tidigt i projektet. De tre originalmodellerna kallades: Engine & converter, ATgearbox och Vehicle. Under arbetets gång skulle dessa funktioner komma att både byggas ut och ökas i komplexitet. Tyvärr har vissa block en "Bad Link" symbol, detta beror på att utgångsmaterialet mellan gamla matlab-versioner tappat tabellbilden.

### 4.3.1 Engine & Converter

Detta subsystem innehåller en generell motormodell som tar emot begärt moment från föraren och ger ut ett motorvarv, se Figur 4.3. Den innehåller också en konvertermodell som är aktiv då fordonet håller en hastighet under ca 10 km/h. Begärt moment,  $Tq\_CrShReq$  skickas in i motormodellen. Om konvertern är aktiverad subtraheras  $Tq\_CrSh$  med det moment som konvertern förbrukar. Därefter beräknas signalen om från radianer per sekund till rpm och samplas. För att sedan adderas i en integrerande-loop vilken innehåller motorns varvantal  $n\_Eng$ . Motorvarvtalet begränsas genom ett begränsnings-block mellan 0 och 7000 rpm. Då Lock-Up:en är aktiverad ( $B\_LU=1$ ) räknas istället motorvarvet ut via fordonshastigheten.

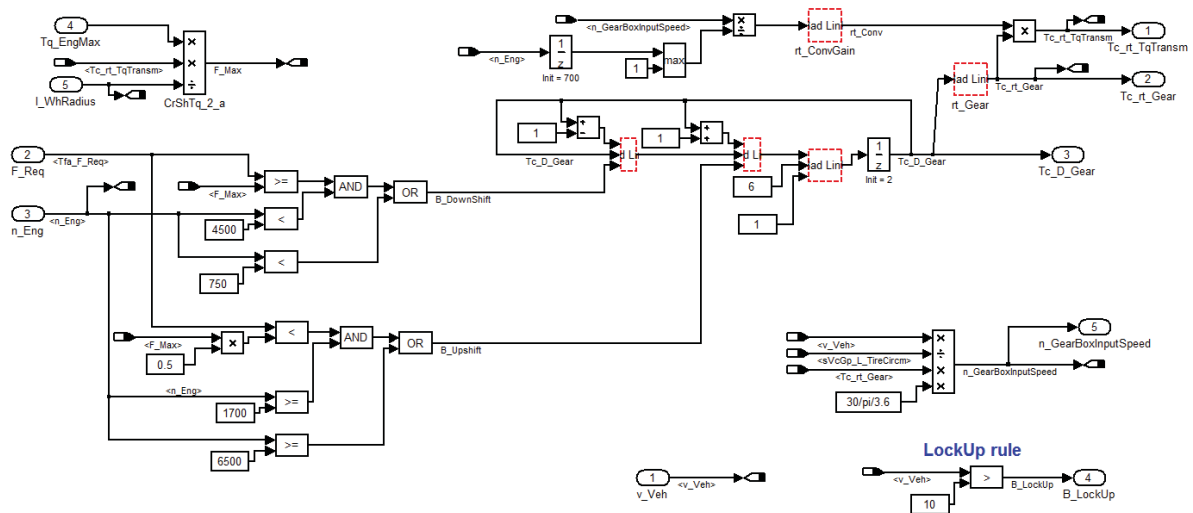


Figur 4.3. Gammla Engine & Converter modell.

### 4.3.2 ATgearbox

Grundfunktionen för automatväxellådan är att behandla motorvarvet  $n_{Eng}$  från E&C och den tillgängliga kraften från motor  $F_{Req}$ , se Figur 4.4. Dessa två variabler jämförs i villkor för att bestämma om det ska ske en uppväxling eller nedväxling. När en växel  $D_{Gear}$  är vald bestäms en utväxling  $Tc_{rt\_Gear}$  för växeln. Även en utväxling  $Tc_{rt\_TqTransm}$  som är kompenserad med konverterförluster. Den variabeln används bland annat för att räkna ut den maximala dragkraften för fordonet  $F_{Max}$ , där den multipliceras med motorns maximala vridmoment  $Tq_{EngMax}$  och delas med hjulradie.

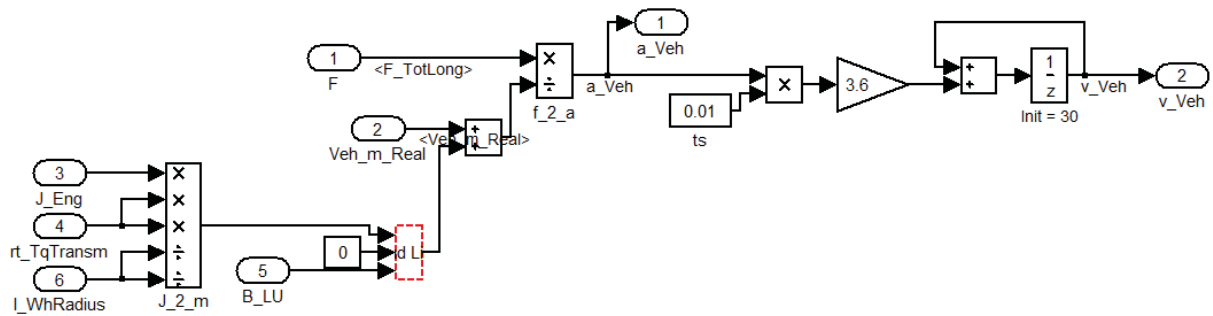
I detta block beräknas även växellådans ingående varvtal  $n_{GearBoxInputSpeed}$  med en matematisk uträkning genom de ingående parametrarna: hastighet, hjulradie, växelförhållande och  $30/\pi/3.6$ . Funktionen LockUp bestäms i detta subsystem den logiken är uppbyggd genom jämförelse om hastigheten är större än 10 km/h. Om villkoret är uppfyllt är LockUp:en aktiverad.



Figur 4.4. Gammal automatväxellådmodell.

### 4.3.3 Vehicle

Fordonsmodellen tar in den kvarvarande simulerade kraften vilket var fordonets kraft, se Figur 4.5. Denna kraft är subtraherad med väglast och bromskraft. Den återstående kraften  $F_{TotLong}$  är den som driver bilen framåt. Den räknas om till bilens acceleration genom att dela med bilens massa och efter det räknas om till hastighet. Vilken används i en integrerande-loop för att få ut den slutliga hastigheten för fordonet. Hastigheten läggs ut på en utgångsport från blocket.



Figur 4.5. Gammal fordonmodell.

## 4.4 Simulators huvudblock

Under denna rubrik beskrivs de funktioner som är utbyggda eller kompletterade under projektets gång. Det förtydligas även vad som behövs för att få grundfunktionen hos simulatoren att fungera i slutna simulering. Med det menas att simulatormodellen arbetar helt fristående från ECM:ens motorstyrning. Detta för att säkra upp simulatorns funktion innan motorstyrningen implementeras.

Då den tidigare datormodellen över motor-, växellåds- och fordonmodellen var gjord i Simulink behövde alla block konverteras till TargetLink-block. Detta var tvunget att göras i två delar.

- Automatisk konvertering av standardblock från Simulink till TargetLink.
- Manuell konvertering av tabeller och matriser.

När alla block var konverterade till TargetLink säkerställdes de nya blockens funktionalitet genom en kodgenerering som bland annat hittar logikfel och algebraiska loopar.

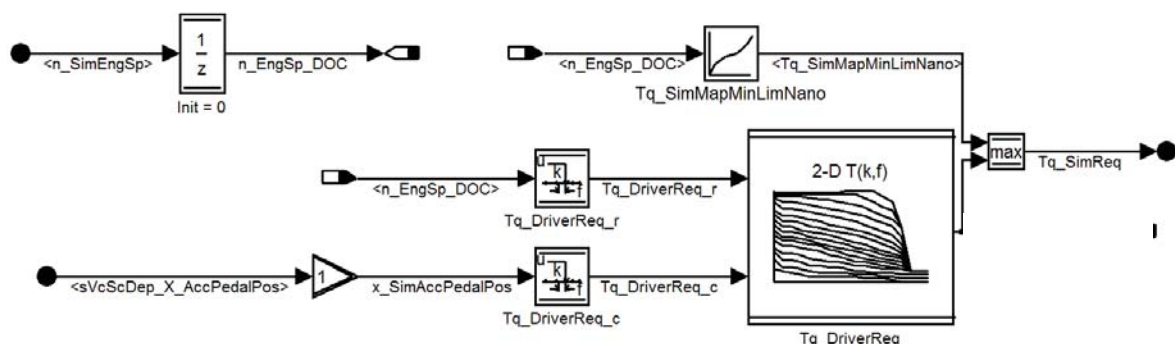
För att få en bättre översikt på blockstrukturen se Bilaga B sida B1-B8.

### 4.4.1 Pedalmapp

Denna funktion har till uppgift att ta in gaspedalens position och varvtalet från motorn med ett sampels fördröjning, se Figur 4.6. Dessa signaler slås upp i en tabell och ett begärt förarmoment skickas ut. Parallellt i flödet tas bara motorvarvtalet in och slås upp i en tabell för att få ut begärt moment vid tomgångskörning. Sedan tas det största värdet av de begärda momenten och skickas vidare till motorn.

Vid simulation i Simulink används ett signalbyggnadsblock som ersätter gaspedalens signal. Den ska ligga mellan 0-100%, där 0 inte motsvarar något gaspådrag och 100 är maximalt.

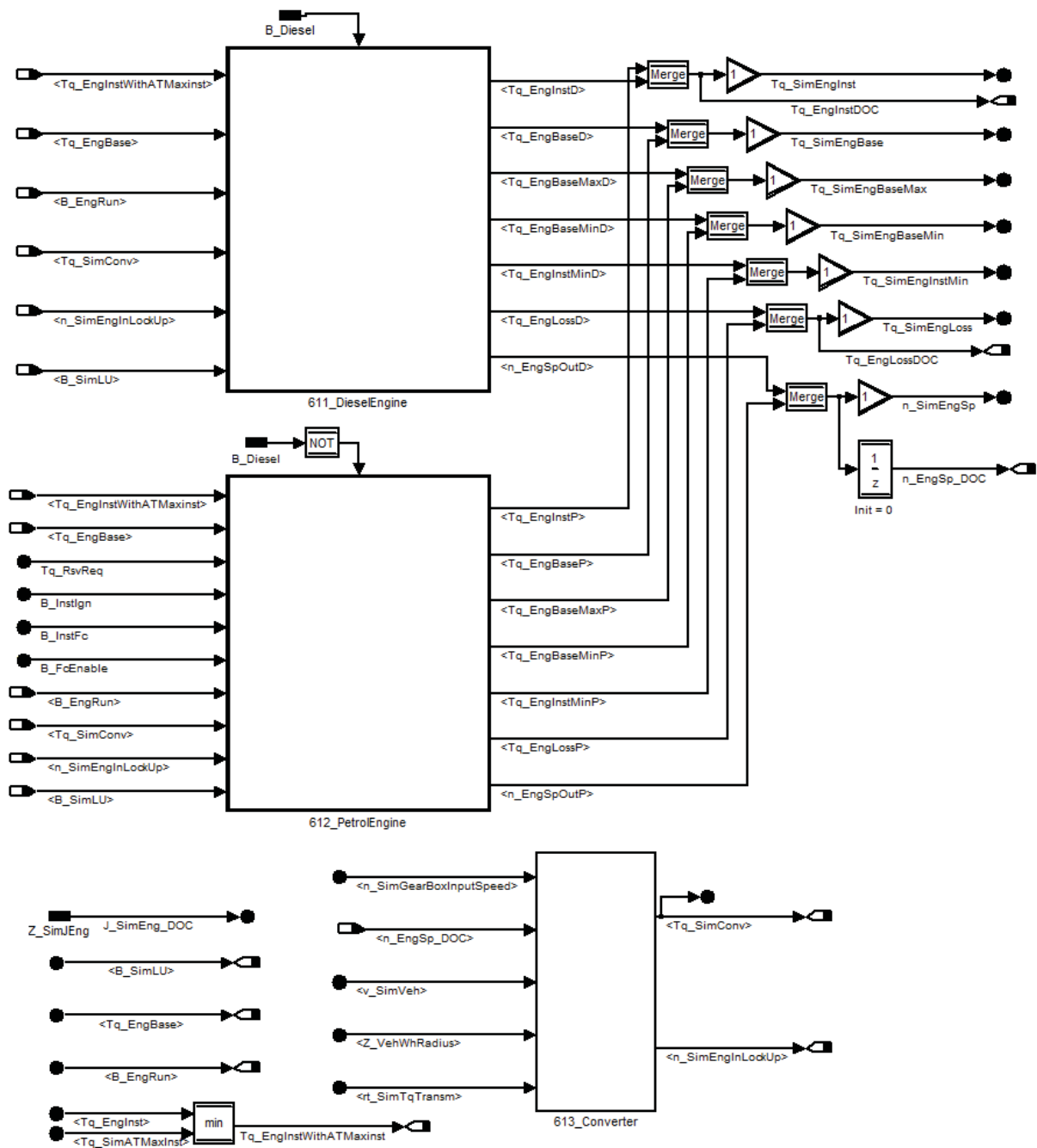
Denna funktion används bara vid slutna simulering. När simulatoren är uppkopplad mot motorstyrningen används pedalmappen som finns i motorstyrningen och detta block inaktiveras.



Figur 4.6. Accelerationspedalmapp i simulator.

## 4.4.2 Engine & Converter

Då den första motormodellen endast var generell behövdes det implementeras två nya modeller, en diesel- och en bensinmotor. Via handledare erhöles två block vilka relativt likt kunde återspegla de två motorerna. Dessa modellers in- och utgångar behövde kopplas ihop med de signaler som var anslutna till den tidigare motormodellen. Då det ska gå att välja vilken modell som ska användas, behövdes utportarna från båda blocken kopplas ihop på samma utgångar. För att detta skulle kunna realiseras användes ett antal Merge-block. Detta block tar alltid det senaste värdet från den ena eller den andra ingången. Funktionaliteten är konstruerad så att bara en motor kan vara aktiv åt gången. *B\_Diesel* används vid val av motormodell. Vid 1 aktiveras diesel och 0 bensin.



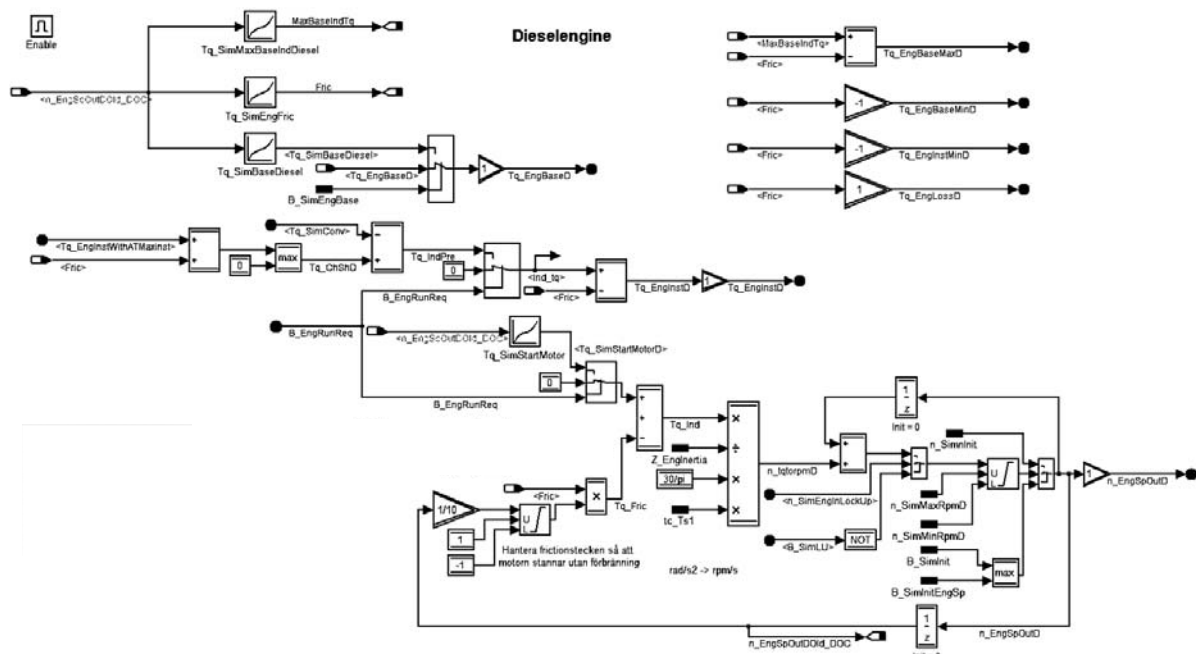
Figur 4.7. Konverter, diesel och bensinblock. Bilaga B sida B4.



## Dieselmotell

Det begärda momentet skickas in via  $Tq\_EngInstWithATMaxInst$  och adderas med motorns friktion  $Fric$ . Signalen subtraheras med konvertermomentet  $Tq\_SimConv$ . För att denna signal ska kunna ledas vidare måste signalen för motor aktiv  $B\_EngRunReq$  vara satt. Om denna signal är aktiv skickas  $Ind\_tq$  till adderings/subtraherings-blocket där friktionen dras bort och eventuellt startmotormoment  $Tq\_SimStartMotorD$  adderas. Summan av denna momentsignal räknas om till varvtalet  $n\_tqtorpmD$  vilket sedan adderas i integrerande-loopen för motorvarvtalet  $n\_EngSpOutD$ . Motorvarvtalet begränsas genom ett begränsnings-block mellan  $n\_SimMinRpmD$  och  $n\_SimMaxRpmD$ , vilka båda är kalibrerbara variabler. Då Lock-Up:en är aktiverad  $B\_SimLU$  räknas istället motorvarvet ut via fordonshastigheten vilken kommer från konvertermodellen  $n\_SimEngInLockUp$ . Det finns även en nollställande funktion som används vid simulering. Detta görs via  $B\_SimInit$  eller  $B\_SimInitEngSp$ . [6]

Tabellerna  $Tq\_SimMaxBaseIndDiesel$ ,  $Tq\_SimEngFric$  och  $Tq\_SimBaseDiesel$  baseras på en sampel gammal signal från motorvarvtalet  $n\_EngSpOutOld\_DOC$ .



Figur 4.8. Logik för dieselmotor. Bilaga B sida B5.

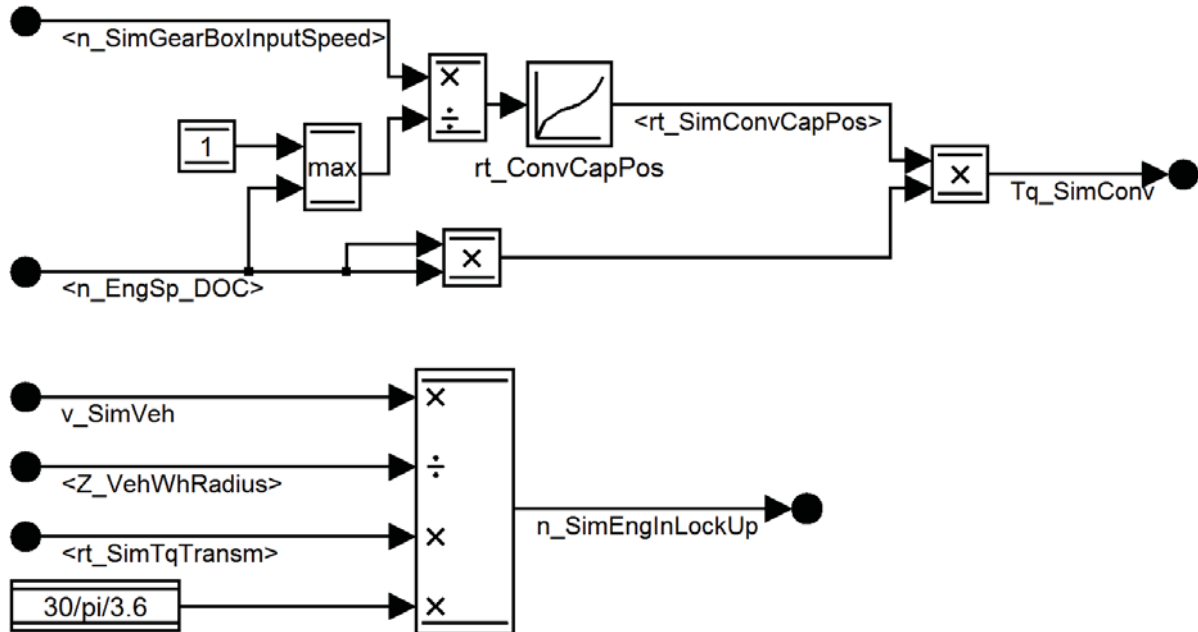
## Bensinmodell

Då denna modell inte behandlats i detta projekt kommer enbart modellen bifogas i Bilaga B sida B6. Anledningen till att in- och utsignalerna är inkopplade är att möjliggöra senare implementering av bensinmotorn.

## Konvertermodell

Konverterlogiken låg tidigare med i den generalla motormodellen. Det blev ett problem när det ska gå att byta mellan diesel- och bensinmotor. Därför bröts den funktionen ut från motormodellen och placerades i ett separat block.

Det konvertern beräknar ut är ett förhållande mellan växellådans varvtal  $n_{SimGearBoxInputSpeed}$  och motornsvarvtal  $n_{EngSp\_DOC}$ , se Figur 4.9. Det förhållandet används för att ta fram ett värde som sedan multipliceras med varvtalet upphöjt med två. Då fås konverterns momentpåverkan. Varvtalet för motorn beräknas parallellt ut när Lock-Up:en är aktiv. Det görs genom att räkna baklänges utifrån fordonets hastighet. [7]

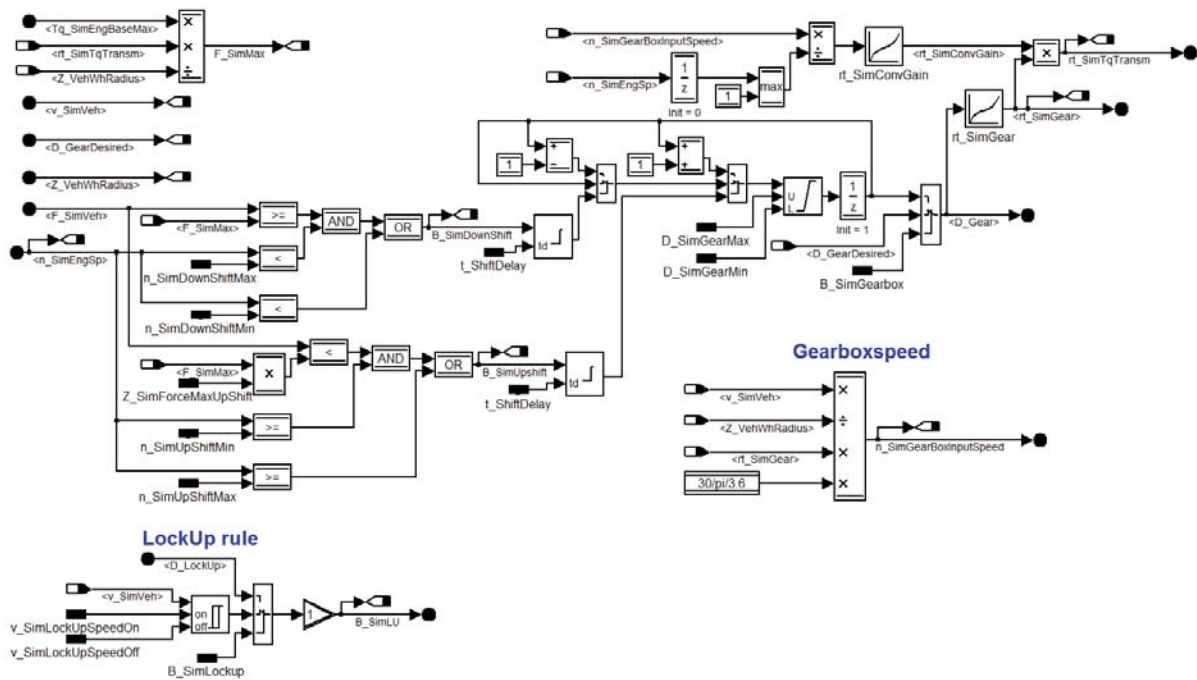


Figur 4.9. Logik för konverter.

### 4.4.3 ATGearbox

Automatlådmodellens grundfunktion behövde inte ändras från grundmodellen, se Figur 4.10. Endast mindre funktionstillägg gjordes. Beräkningen av fordonstragkraften flyttades till Vehicle-blocket. Lock-Up funktionen byggdes om för att kunna aktiveras och inaktiveras vid olika hastigheter och kunna styras från en yttre signal från TCM:en. För att kunna styra fördröjningen av en växling lades en delayfunktion till vid  $B\_SimUpShift$  och  $B\_SimDownShift$ .

När projektet nått det stadiet då vi kunde ta in transmissionsfunktionens växlingssignal som finns i CVC, gjordes det möjligt att välja vilken växlingssignal som ska användas. Antingen väljs signalen från simuleringsmodellen eller transmissionsfunktionen från CVC. [8]



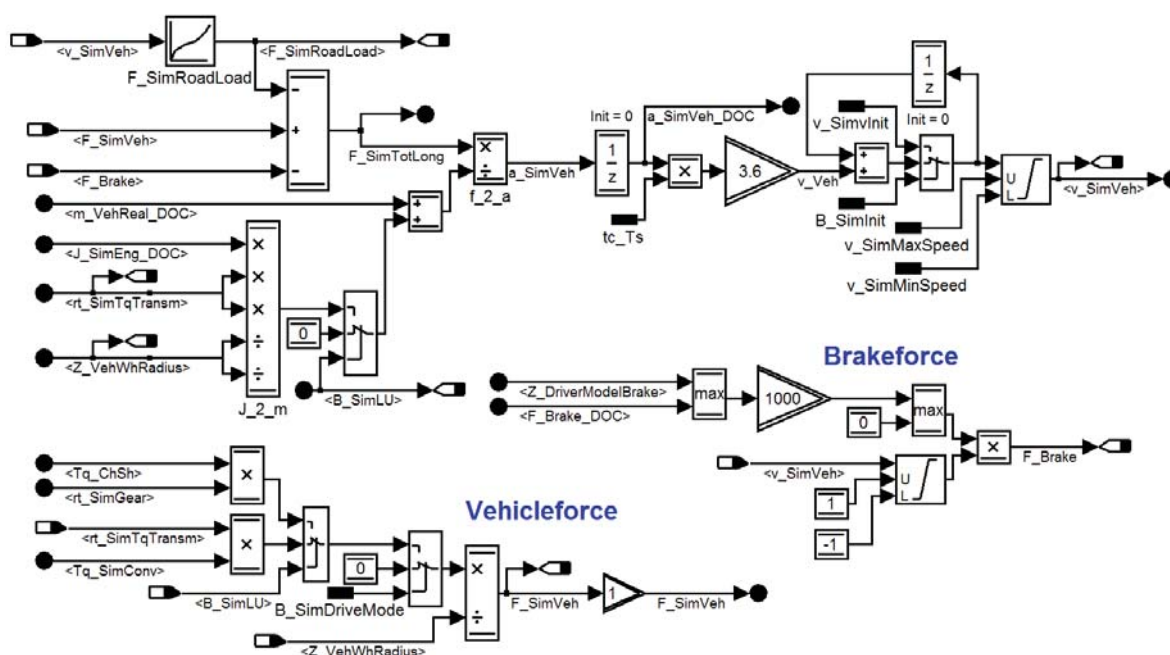
Figur 4.10. Logik för automatväxellåda. Bilaga B sida B7.

#### 4.4.4 Vehicle

Fordonsmodellen behövde inte många korrigeringar utifrån grundmodellen. Grundfunktionen behölls och det adderades några funktioner. Blocket som beräknar fordonskraften  $F_{SimVeh}$  och det följande subtraherande blocket med väglasten  $F_{SimRoadLoad}$  och bromskraften  $F_{Brake}$  flyttades in i fordonsmodellen. Väglasten simulerades med hjälp utav en tabell som berodde på hastigheten. Bromskraften kommer från en insignal  $F_{Break\_DOC}$  som generas från en kalibrerbar variabel. Bromssignalen kan ligga mellan 0-20, sedan multipliceras med 1000 för att få en större bromsverkan. För att få bilen att stanna med 0 i hastighet multipliceras bromskraften med ett värde mellan 1 och -1 vilket är beroende av hastigheten. Detta ger resultat när hastigheten närmar sig noll, då multipliceras bromskraften ett mycket litet tal och bilen stannar på hastigheten noll med ett litet kvarstående fel.

Funktionen för att kunna nollställa hastigheten behövdes vid simulering. Det löstes med en switch som skickar in värdet noll i integrerande-loopen vid en signal från en kalibrerbar variabel.

Fordonkraften som tidigare låg i E&C och hette  $F_{Eng}$ , fick byggas om för att den signalen som tidigare användes inte var uppbyggd på samma sätt. Beräkningen använder nu vid aktiv Lock-Up momentsingalen  $Tq\_ChSh$  multiplicerat med förhållandet  $rt_{SimGear}$ . Motsvarande inaktiv Lock-Up momentsignalen  $Tq_{SimConv}$  och förhållandet  $rt_{SimTqTransm}$ . För att kunna få simulatoren att simulera Neutralt läge, då motorn är igång men bilen står still utan att bromsa. Detta löstes genom att sätta en switch vilket bryter kraftberäkningen och skickar ut värdet noll.



Figur 4.11. Logik för fordon. Bilaga B sida B8.

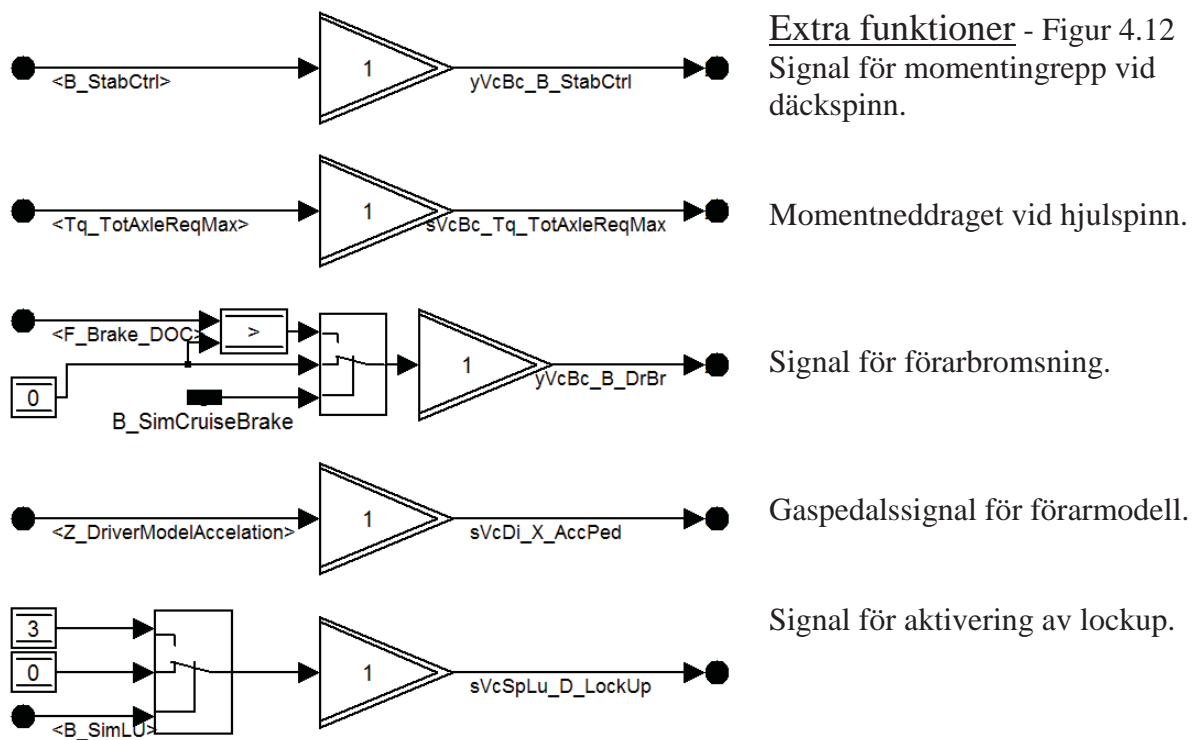
## 4.5 Uppkoppling av simulatorns signaler mot verkliga motorstyrfunktioner

I detta kapitel beskrivs de funktioner som skapats och tagits fram under projektets gång. Första steget för att kunna koppla ihop simuleringsmodellen med det verkliga motorstyrsystemet var att skriva över de globala variabler som används av de funktionsblocken i styrenheten. De lokaliserades genom att steg för steg grundligt tyda insignaler på de styrmodeller som skulle användas.

Alla utsignaler samlades ihop i ett eget block i Simulink för att enkelt kunna byta ut de globala adresserna så det går att implementera simulatoren på en annan motorplattform.

### 4.5.1 Outsignals

Utsignalsblocket "Outsignals" används till att skriva över de signaler som behövs för att få motorstyrningen att funka. Antingen skrivs en signal över med simulerad signal eller med kalibrerbar konstant. De signaler som behöver skrivas över är de som avläses från CAN-bussen och läggs på variabler i ECM:en. Tanken med att samla alla utsignaler i samma block är att det går lätt att byta ut de signaler som ska skrivas över till andra motsvarande signaler för andra plattformar. I det följande tre figurerna beskrivs utsignalerna.



Figur 4.12. Innehåll i Outsignals.

## Signaler till Engine Control

- Figur 4.13

Tillgängligt bränslemoment från motorn.

Tillgängligt luftmoment från motorn.

Tillgängligt maximalt luftmoment från motorn.

Tillgängligt minimalt luftmoment från motorn.

Tillgängligt minimalt bränslemoment från motorn.

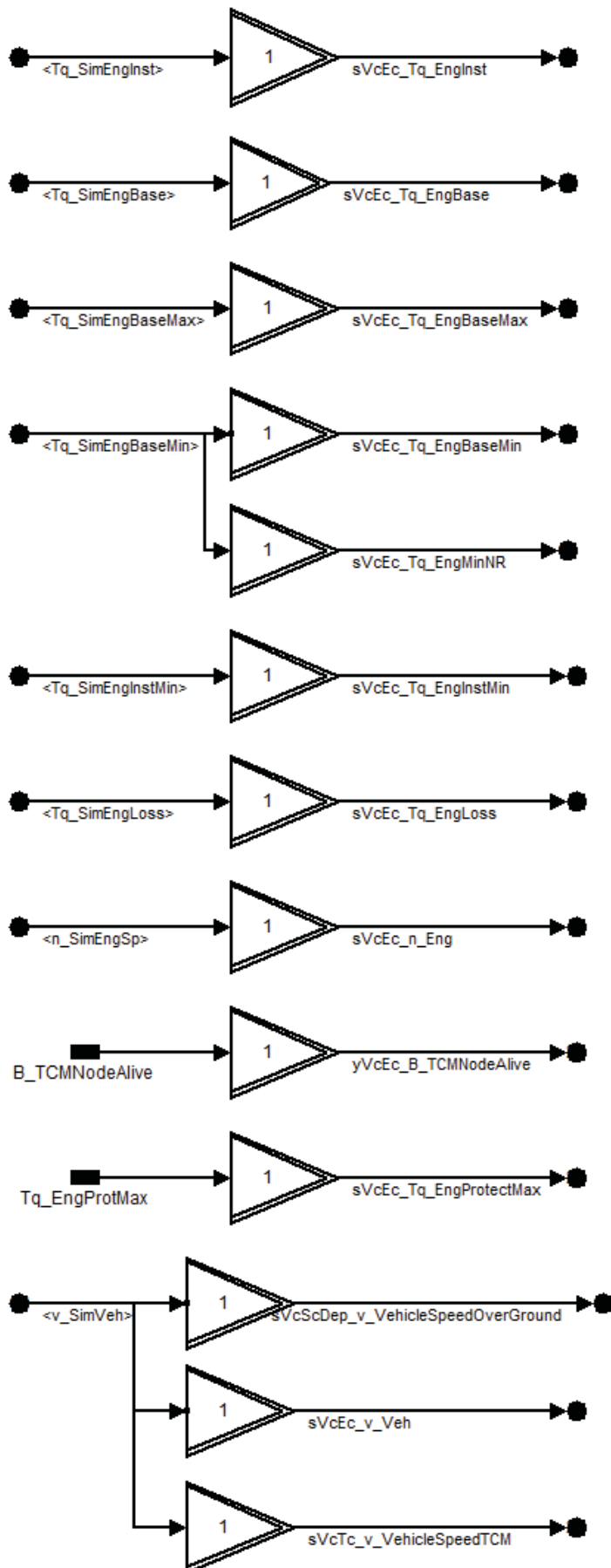
Momentförluster i motorn.

Varvtal från motorn.

Boolesk signal som indikerar att TCM skickar signaler.

Motorskydd för maxmoment.

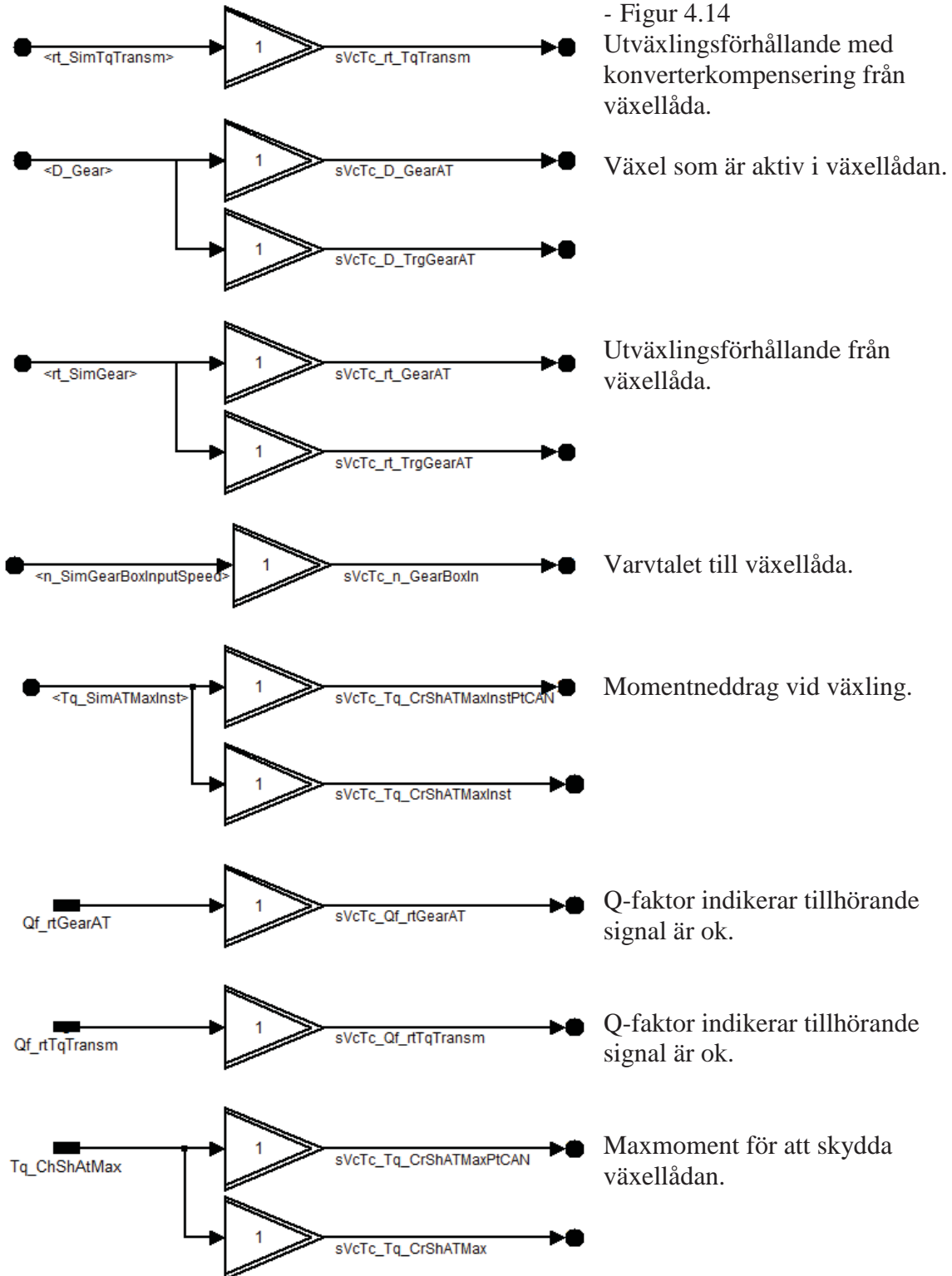
Fordonshastigheten skickas ut till flera signaler som behövs i olika funktioner.



Figur 4.13. Innehåll i Outsignals.

Simuleradesignaler från  
Transmission Control

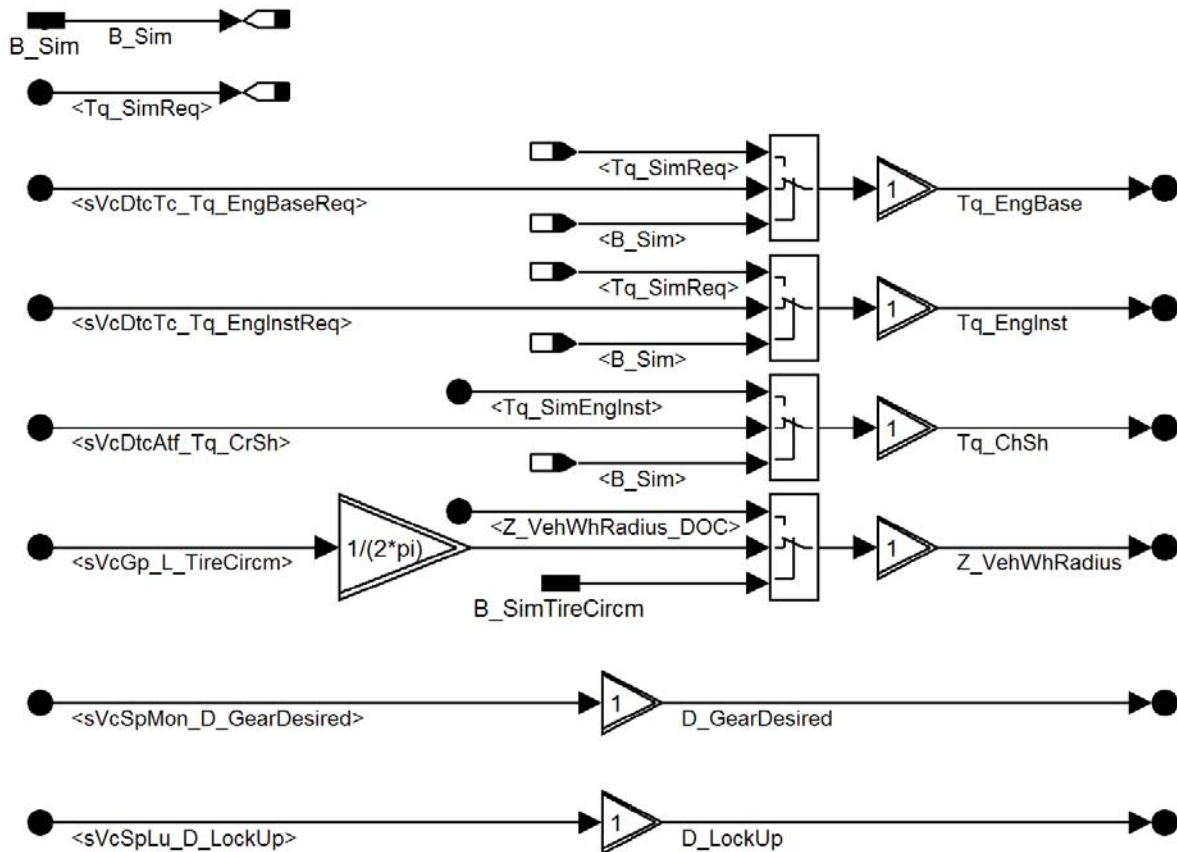
- Figur 4.14



Figur 4.14. Innehåll i Outsignals.

## 4.5.2 Insignals

I insignalsblocket "Insignals" väljs vilka signaler som ska styra simulatorm, se Figur 4.15. Genom att påverka den kalibrerbara konstanten  $B\_Sim$  växlar simulatorm mellan att simulera motorstyrningens signaler eller simulera interna signalerna. Detta var fördelaktigt i projektets början när fokus låg på att få en korrekt funktion hos simulatorm. Därmed att inte blanda in motorstyrningens signaler, vilket kan medföra en krångligare felsökning.



Figur 4.15. Innehåll i Insignals.



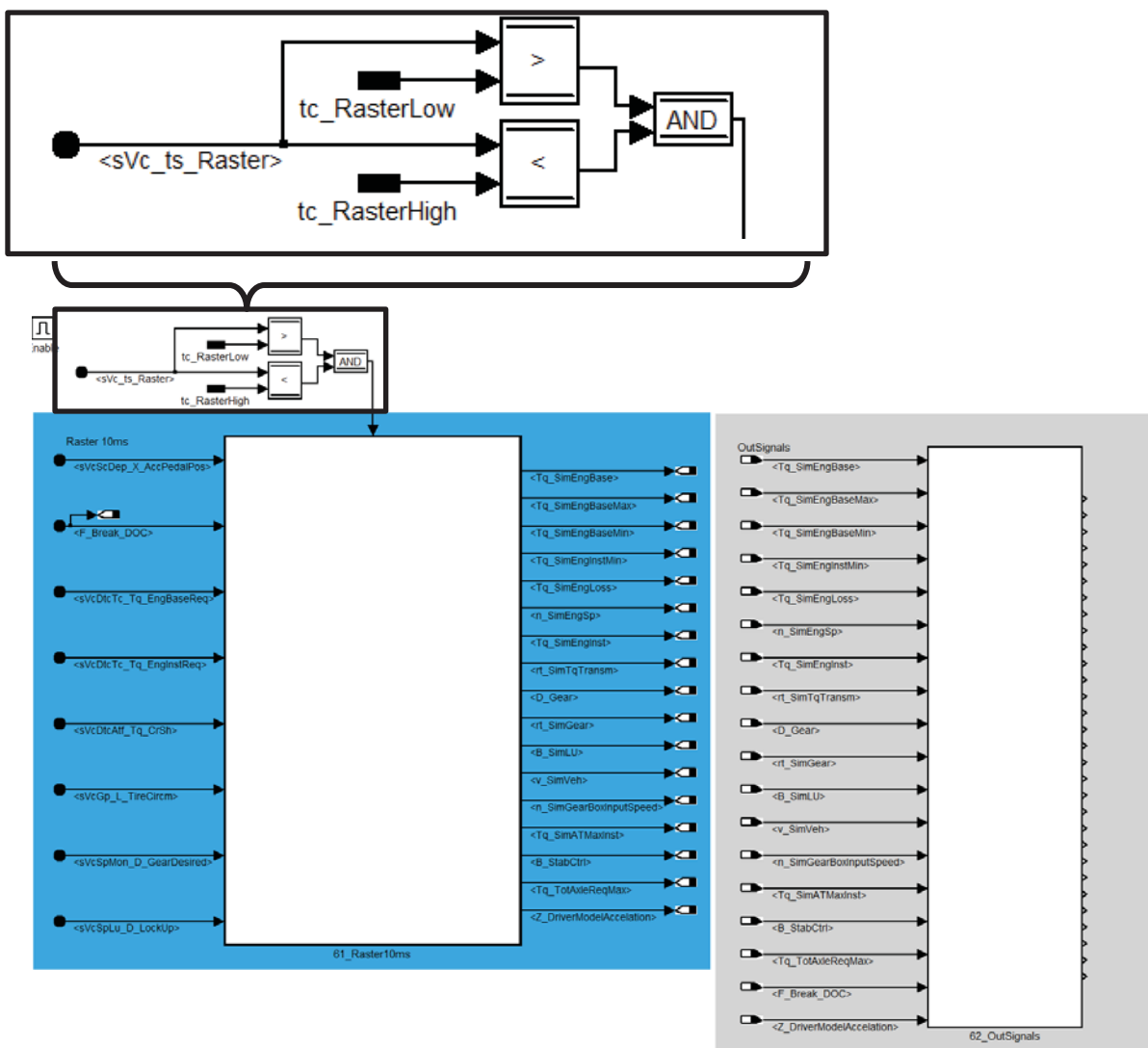
## 4.6 Utbyggnadsfunktioner

Här beskrivs de utbyggnadsfunktioner, som utökar simulatorns funktionalitet utöver grundfunktionen. Det står också förtydligt under varje funktion om den är realiserad genom uppkoppling mot motorstyrfunktioner. De flesta funktioner har ofta en kombination av både egenutvecklade block och uppkoppling mot motorfunktioner.

### 4.6.1 Rasterhantering

Det är mycket viktigt att simulationen endast utförs i 10 ms rastret. Om denna funktion inte skulle finnas hade den uppdateras i 10 ms-, 40 ms- och 160 ms rastret. Detta eftersom VcExtIn utförs i alla raster. Detta skulle medföra att exekveringen utförs oregelbundet. Outsignals-blocket ska fortfarande exekveras vid varje raster för att bibehålla rätt värden på utportarna till momentflödet.

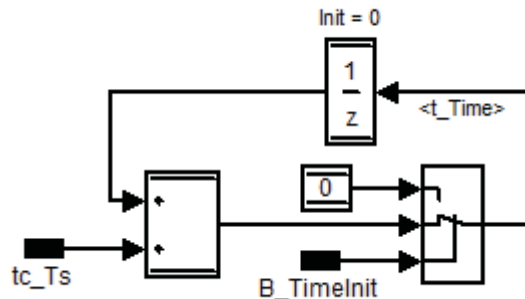
På *sVc\_ts\_Raster* kommer det ett värde på vilket raster som är aktiv. Villkoret för att *61\_Raster10ms* ska exekveras är om det värdet ligger mellan 0 och 15ms vilket ställs in i variablerna *tc\_RasterLow* och *tc\_RasterHigh*. Se Figur 4.16.



Figur 4.16. Logik för att styra simulatorn till att exekveras i rätt raster. Bilaga B sida B2.

## 4.6.2 Tidskrets

En tidtagningsfunktion, se Figur 4.17. Den är enkelt uppbyggd genom att samplings tiden 10 ms  $tc\_Ts$  adderas varje exekvering med ett sampel gammal tid. Exekveringen sker varje 10 ms. Då fås en tidtagning i variabeln  $t\_Time$ . Det går även att nollställa  $t\_Time$  genom att påverka den booleska variabeln  $B\_TimeInit$ .



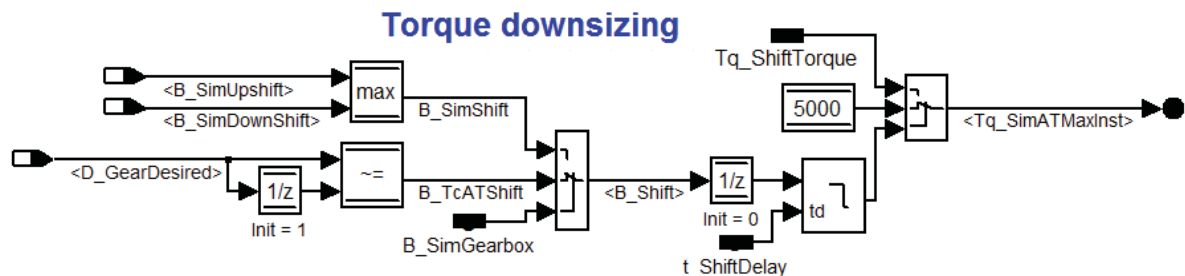
Figur 4.17. Logik för tidtagning.

## 4.6.3 Momentneddrag vid växling

Vid växling med automatlåda måste momentet från motorn under en kort period reduceras för att inte skada någon del av drivlinan. En funktion var därför tvungen att skapas för att uppnå detta krav, se Figur 4.18. Signalerna som aktiverar momentneddraget kan komma antingen från den interna ATgearboxmodellen eller från motorstyrningens växelssignal. Detta är beroende på vilket värde  $B\_SimGearbox$  är kalibrerad till.

Från den interna modellen kommer växelssignalen antingen då en upp eller nedväxling blir triggad. Signalen som kommer från motorstyrningarna bär ett värde som representerar vilken växel som är aktiverad. Denna signal jämförs med ett gammalt sampel och om ändring skett kommer växling inträffa.

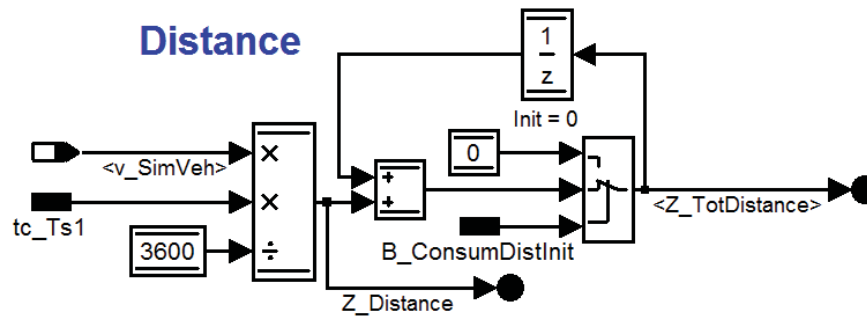
En fördröjningsfunktion ser till att momentet reduceras under tillräckligt lång period så växlingen hinner genomföras. Hur mycket funktionen ska sänka momentet med kan kalibreras till önskat värde.



Figur 4.18. Logik för momentneddrag vid växling.

#### 4.6.4 Beräkning av total sträcka

Att kunna beräkna den totala sträckan som det simulerade fordonet kört är en viktig och intressant grundfunktion med stor utbyggningspotential. Insignaler till denna funktion är bilens hastighet, samplingstiden och en division som omvandlar km/h till m/s. Därefter används en enkel integrerande-loop som även kan nollställas via en kalibrerbar variabel. Se Figur 4.19.



Figur 4.19. Logik för distansberäkning.

#### 4.6.5 Beräkning av total och momentan förbrukning

Teorin bakom denna funktion är att genom ett antal matematiska beräkningar räkna fram antalet liter som totalt förbrukats. Insignaler till bränslematrisen *EngConsum* är motorvarvtal tillsammans med summan av tillgängligt moment och friktionsförluster i motorn, se Figur 4.21. Bränslematrisen skickar sedan ut ett värde med enheten mg/stroke. Det värdet används i beräkningen och omvandlingen till liter, se Ekv.4.1. Efter det sista beräkningsblocket kommer en integrerande-loop vars funktion är att efter varje loop addera antalet liter.

$$\frac{q * n * a * b}{60 * \rho * 10^6} * t = l \quad (4.1)$$

q = insprutat bränsle [mg/stroke]  
n = varvtal [rpm]  
a = antal cylindrar (4)  
b = fyrtaktsprincipen (0.5)  
ρ = densitet diesel (0.81) [kg/dm<sup>3</sup>]  
t = samplingstid (0.01) [s]

Bränslematrisen som erhöles från Volvos bränsleavdelning var uppbyggd av axlarna Y=mg/stroke X=Varvtal, vilket gav ett moment. Storleken på matrisen var 30X30. För att realisera denna förbrukningsfunktion behövdes dock en matris med axlarna Y=Moment, X=Varvtal, eftersom mg/stroke är okänt. Se Bilaga C sida C1. Ett kortare Matlabscript togs då fram för att kunna konvertera matrisen. Se Figur 4.20

```
%a varvtal (X)
%b flöde (Y)
%c tabell (Moment)

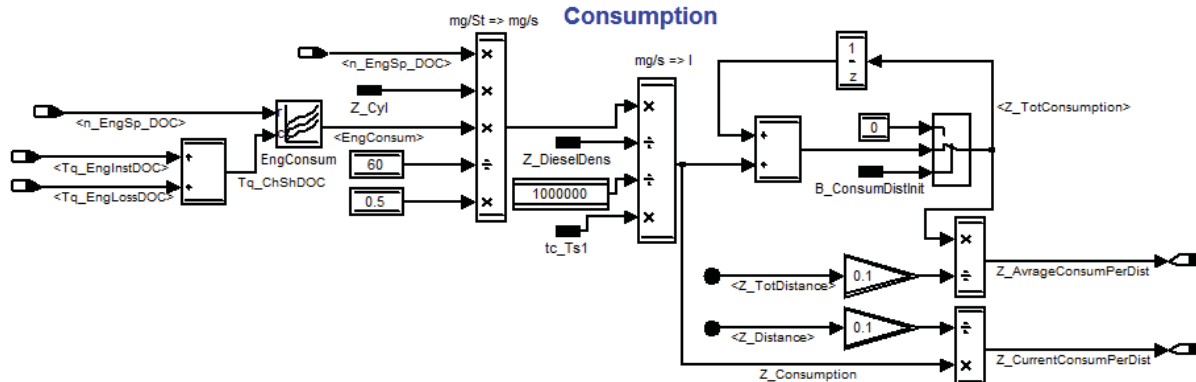
tq = linspace(min(c(1, :)),
max(c(end, :)), 30)';

invc = [];
for ix = 1:size(c,2)
    tmp = c(:, ix);
    tmpinv = interp1(tmp, b, tq);

    invc = [invc, tmpinv];
end;
```

Figur 4.20. Matlabscript för matrisomvandling.

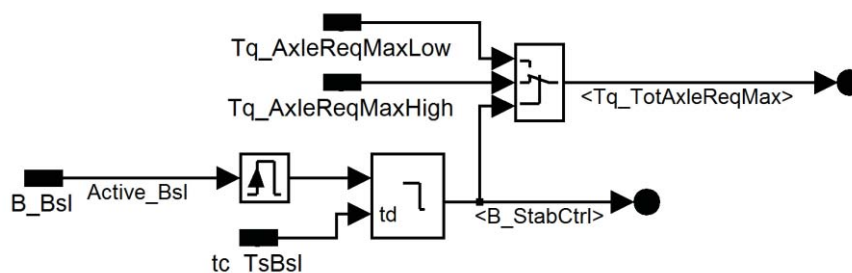
Då det även kan vara intressant att veta den momentana och genomsnittliga förbrukningen behöver distansfunktionen utnyttjas. Med en enkel division erhålls både  $Z\_AvrageConsumPerDist$  och  $Z\_CurrentConsumPerDist$ . Där  $Z\_AvrageConsumPerDist$  är produkten av divisionen mellan den totala förbrukningen i liter genom totala körda sträckan.  $Z\_CurrentConsumPerDist$  är produkten av divisionen mellan liter/sampel och meter/sampel.



Figur 4.21. Logik för beräkning av bränsleförbrukning.

#### 4.6.6 DSTC-Funktionalitet

Det kan finnas intresse av att kunna erhålla ett momentneddrag vid ett simulerat hjulspinn. Genom att manuellt påverka  $B\_Bsl$  kan det begärda motormomentet dras ner. Detta sköts i motorstyrningen och inte i simulatoren. Det funktionen gör är att skicka en boolesk-signal på  $B\_StabCtrl$  och en momentsignal  $Tq\_AxleReqMax$  som är ett neddrag. Dessa två signaler går till en funktion i motorstyrningen och där i påverkar det begärda motormomentet. Tiden på neddraget kommer från  $tc\_TsBsl$ , den ligger på 2 s. Storleken på neddraget kommer från  $Tq\_AxleReqMaxLow$  och  $Tq\_AxleReqMaxHigh$ . Se Figur 4.22.

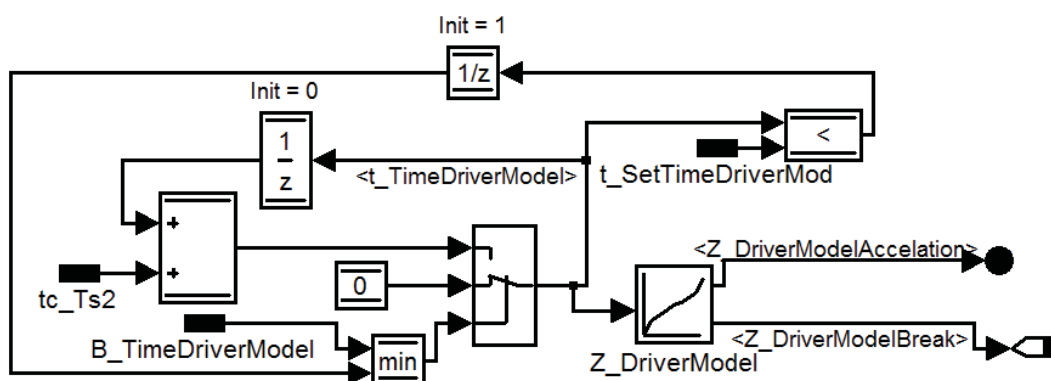


Figur 4.22. Logik för ett momentneddrag för hjulspinn.

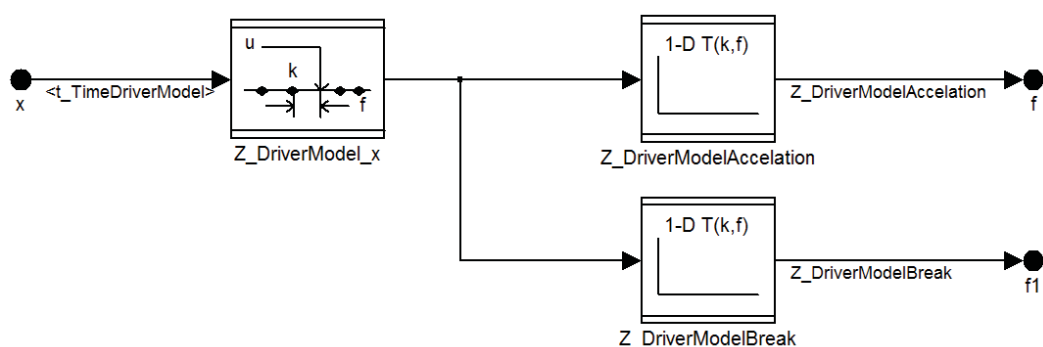
## 4.6.7 Förarmodell

En körprofil eller förarmodell repeterar exakta kopior av ett körfall genom att ändra accelerationspedalens och bromspedalens signaler. Det är en väsentlig funktion för att kunna jämföra olika motorstyrssystem och parametrar såsom bränsleförbrukning, växlingar, hasighet och motorvarvtal. Den består av två tabeller vilka är beroende av samma tidsfunktion.

Huvudfunktionen hos förarmodellen är en tidkrets vilken har presenterats tidigare i rapporten, det som skiljer denna tidkrets från tidigare är att den har en start och nollställningsfunktion, se Figur 4.23. Startvillkoret är att variabeln  $B\_TimeDriverModel$  påverkas, då startar tidräkningen. Den räknar ända tills nollställningsvillkoret uppfylls genom variabeln  $t\_SetTimeDriverMod$  som innehåller maximal tid innan den startar om, i detta fall 60 s. Tidtagningen används i tabellerna för förarmodellen, se Figur 4.24. När tidtagningen startar börjar tabellerna skicka ut värden på variablerna för accelerationspedalen och brompedalen. Tabellerna vilka togs fram för detta projekt visas i (Bilaga A Sid A1). Accelerationssignalen varierar mellan 0-100 % och bromsignalen mellan 0-20 och körfallet pågår i 60 sekunder. Den repeterar körfallet tills  $B\_TimeDriverModel$  är inaktiv.



Figur 4.23. Logik för förarmodell.

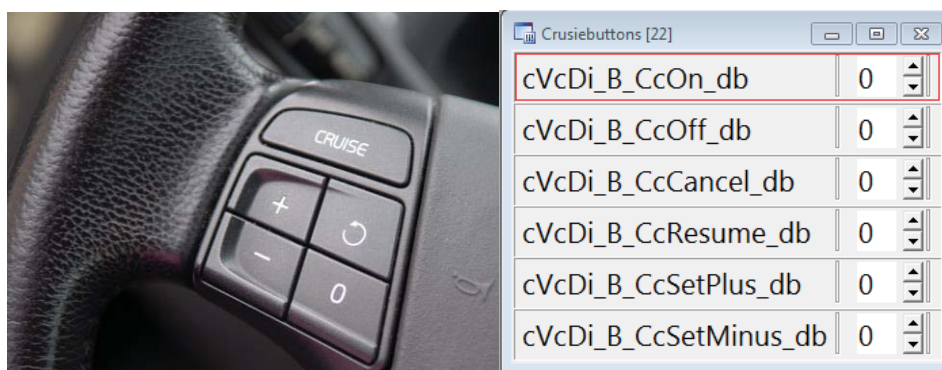


Figur 4.24. Underliggande tabeller i  $Z\_DriverModel$ .

#### 4.6.8 Cruise control/farthållare

Cruise control är en funktion vilken kan hålla hastigheten konstant och därmed behöver inte accelerationspedalen påverkas för att hålla hastigheten. Denna funktion ligger i motorstyrningen, men kan styras från simulatormiljön i INCA finns motsvarande knappsignaler som finns på ratten i bilen, se Figur 4.25.

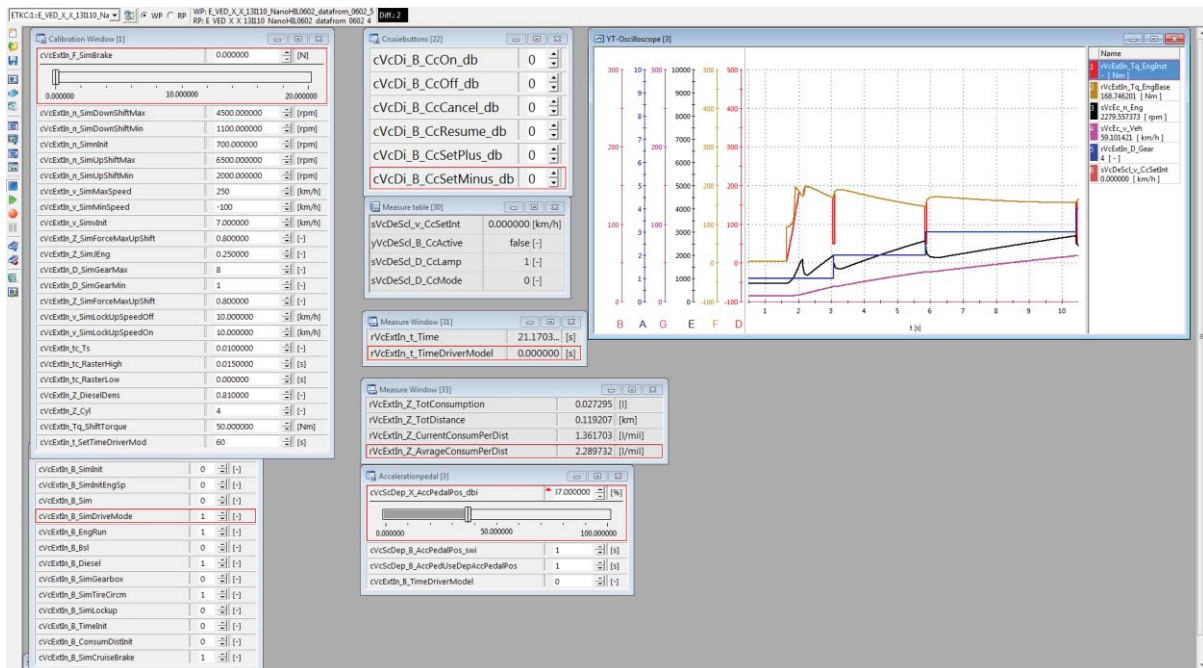
$B\_CcOn$  och  $B\_CcOff$  => Cruise-knapp  
 $B\_CcCancel$  => 0-knapp  
 $B\_CcResume$  => ”cirkulerande pil”-knapp  
 $B\_CcSetPlus$  => Plus-knapp  
 $B\_CcSetMinus$  => Minus-knapp  
Alla signaler ska tiggas med en puls.



Figur 4.25. Knappar till cruise control i bil till vänster och till höger motsvarande signaler i simuleringsmiljön i INCA.

## 5. RESULTAT

Resultatet i denna del presenteras i form av grafer som samlats in genom testkörning i INCA mot styrenheten. Se Figur 5.1. Resultatet har delats upp i två delar: körfall och funktioner. Under körfall demonstreras hur fordonet beter sig under acceleration och inbromsning. Under funktioner visas specifika funktioner vilket simulatoren är utrustad med.



Figur 5.1. Experiment-vy i INCA.

### 5.1 Körfall

I de tre körfallen som framställs i denna del har alla körts med rullande start. Anledningen är att simulatoren ska återspegla en automatväxlad bil, vilken i tomgångskörning håller ca 7 km/h om varken bromsen eller gaspedalen är påverkad.

I de följande tre fallen kommer signalerna vara färgkodade, se Figur 5.2:

1. Svart = Accelerationspedal.
2. Röd = Begärt moment till motorn.
3. Blå = Växel.
4. Lila = Fordons hastighet.
5. Grön = Motorvarvtal.

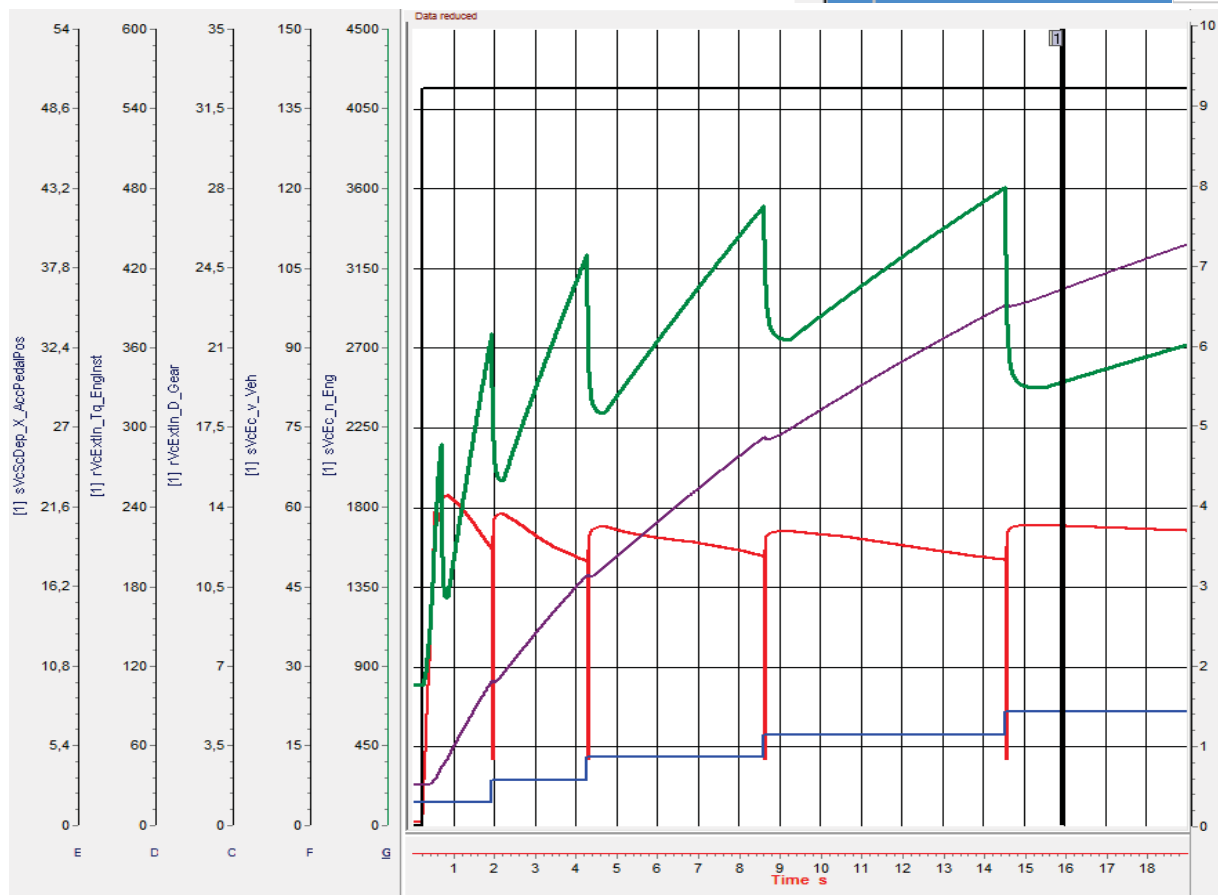
No.	Color	Name	Units
1	Black	[1] sVcScDep_X_AccPedalPos	%
2	Red	[1] rVcExtIn_tq_EngInst	Nm
3	Blue	[1] rVcExtIn_D_Gear	-
4	Purple	[1] sVcEc_v_Veh	km/h
5	Green	[1] sVcEc_n_Eng	rpm

Figur 5.2. Lista med grafsignaler.

### 5.1.1 7-100 km/h, 50 % gaspedal

Den svarta lodräta linjen visar där hastigheten uppnår 100 km/h. Detta tar ca 16 s i detta körfall. Se Figur 5.3. Vid växling kan man se att motormomentet dras ner och motorvarvet sjunker för att sedan öka igen. Fordonet hinner komma upp till femte växeln. Hastigheten har en brant stigning i början av accelerationen för att sedan avta lite. Detta beror på att det är bara 50 % gaspedal och börjar närma sig sin hastighet. Det kan jämföras med nästa körfall under kapitel 5.1.2 med 100 % gaspedal.

No	Color	Name	Units
1	Black	[1] sVcScDep_X_AccPedalPos	%
2	Red	[1] rVcExtIn_Tq_EngInst	Nm
3	Blue	[1] rVcExtIn_D_Gear	.
4	Purple	[1] sVcEc_v_Veh	km/h
5	Green	[1] sVcEc_n_Eng	rpm



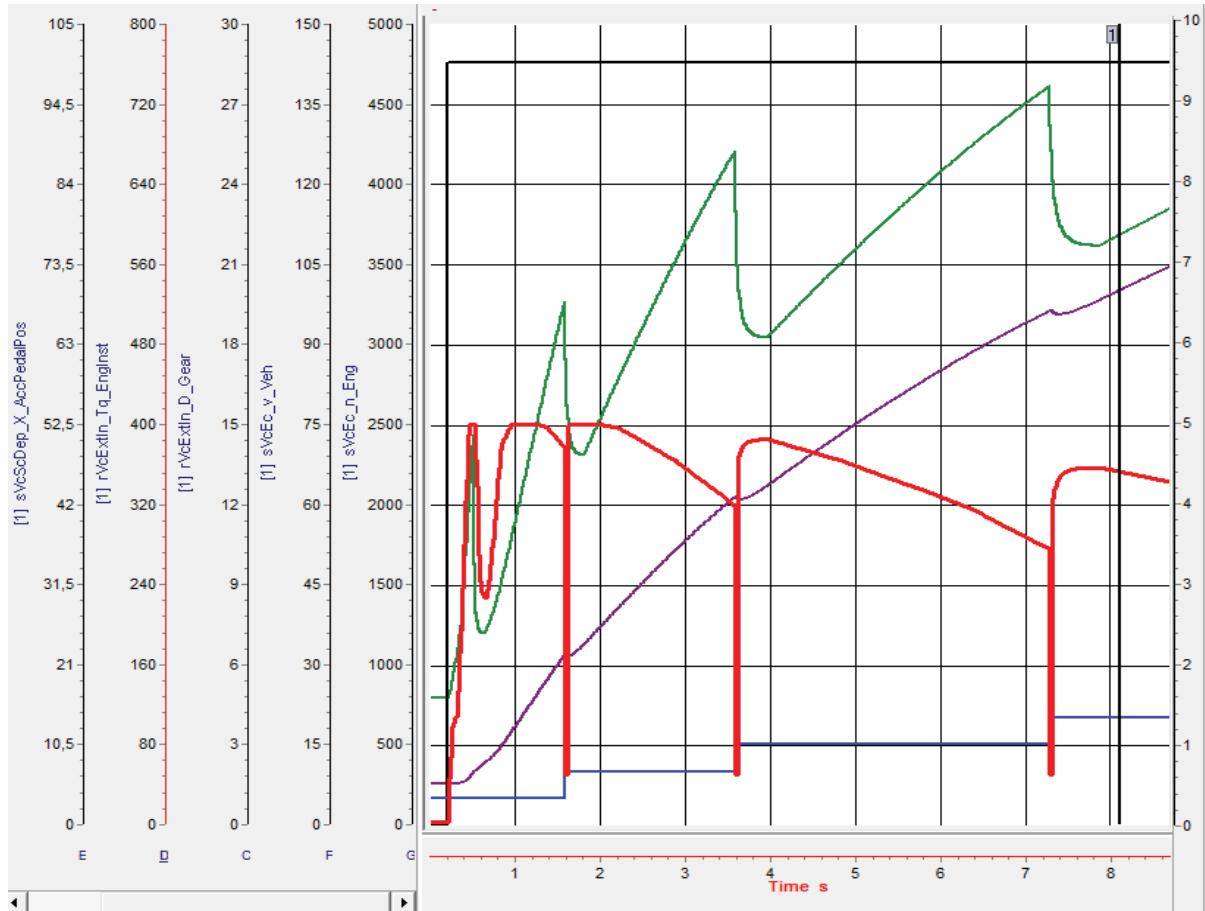
Figur 5.3. Körfall 7-100 km/h 50 % accelerationspådrag.



### 5.1.2 7-100 km/h, 100 % gaspedal

Den svarta lodräta linjen visar där hastigheten uppnår 100 km/h. Detta tar ca 8 s i detta körfall. Se Figur 5.4. Vid växling kan man se att motormomentet dras ner och motorvarvet sjunker för att sedan öka igen. Fordonet kommer endast upp till fjärde växeln, detta beror på att fordonet varvar upp till ett högre vartal innan det sker en uppväxling. Hastigheten har en jämnare stigning upp till 100 km/h med 100 % gaspedal än tidigare körfall med 50 % gaspedal.

No.	Color	Name	Units
1	Black	[1] sVcScDep_X_AccPedalPos	%
2	Red	[1] rVcExtIn_Tq_EngInst	Nm
3	Blue	[1] rVcExtIn_D_Gear	-
4	Purple	[1] sVcEc_v_Veh	km/h
5	Green	[1] sVcEc_n_Eng	rpm

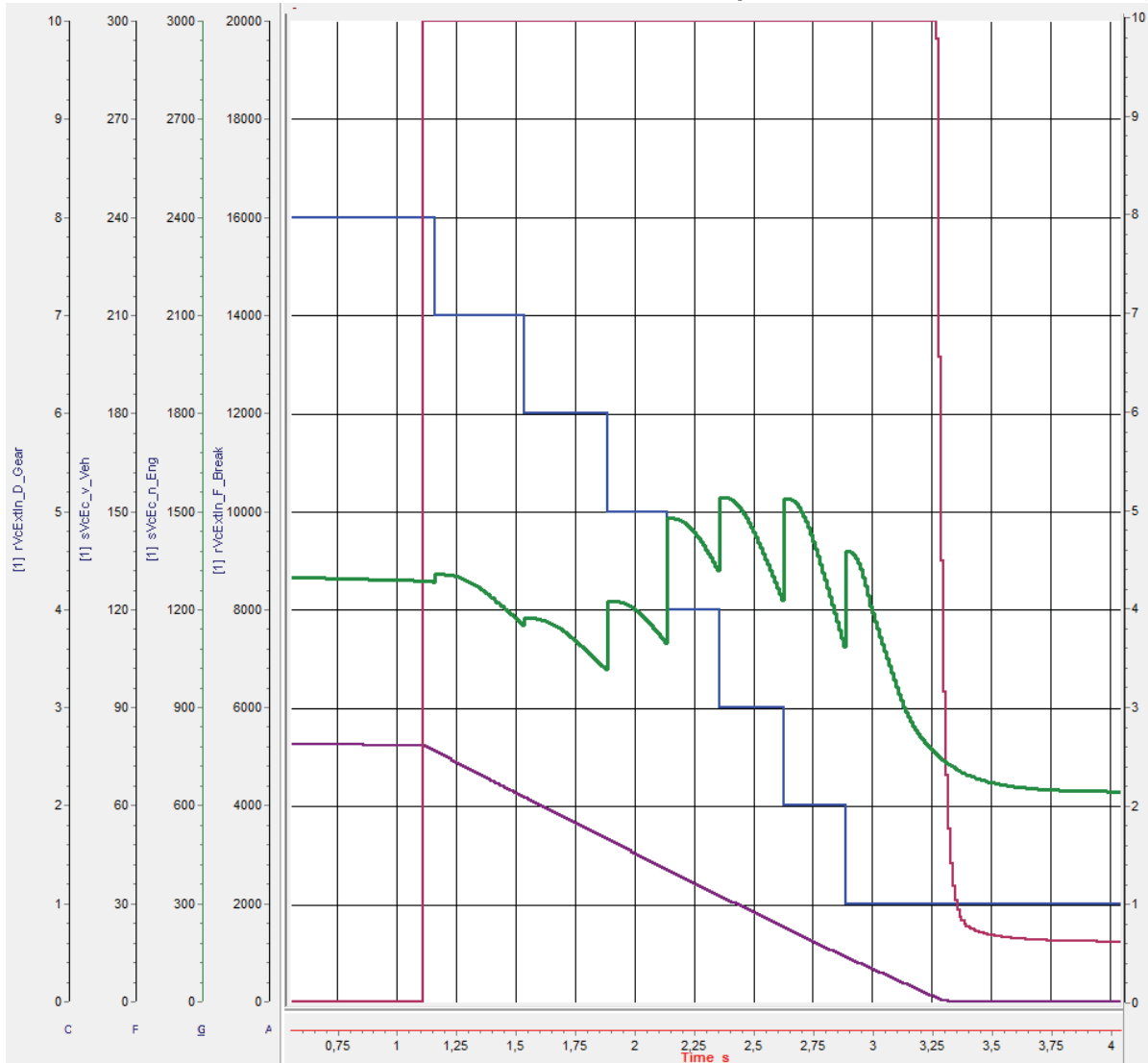


Figur 5.4. Körfall 7-100 km/h 100 % accelerationspådrag.

### 5.1.3 100-0 km/h, 100 % Bromspedal

Vid tiden 1 s läggs maximal broms på och hastighetsminskningen inleds. Nedväxlingarna sker succesivt då hastigheten minskar. Utläst från grafen tar det ca 3 s för att få stopp på bilen. Se Figur 5.5.

No.	Color	Name	Units
1	Blue	[1] rVcExtIn_D_Gear	-
2	Purple	[1] sVcEc_v_Veh	km/h
3	Green	[1] sVcEc_n_Eng	rpm
4	Red	[1] rVcExtIn_F_Brake	N



Figur 5.5. Körfall 100-0 km/h 100 % broms.

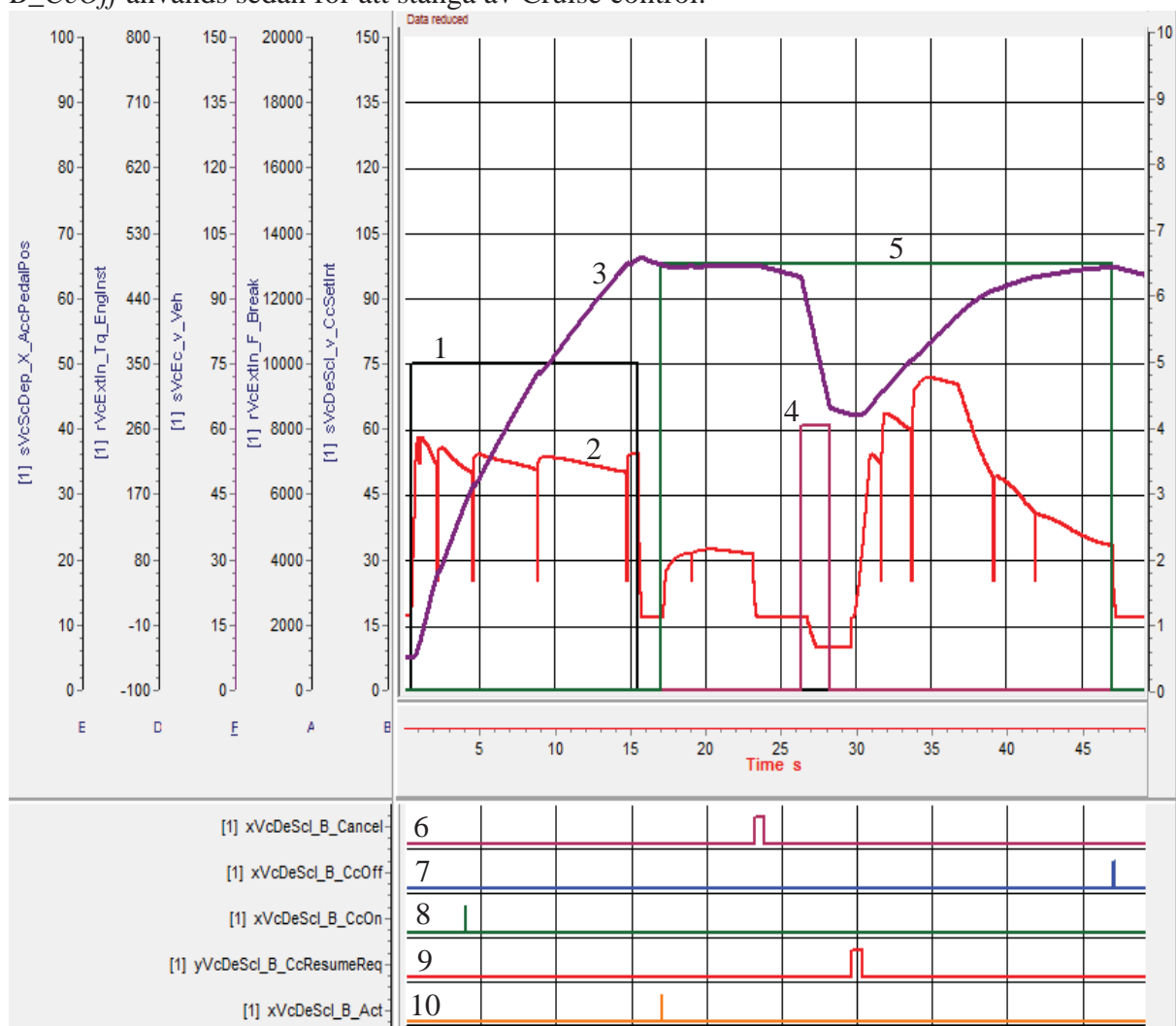
## 5.2 Funktioner

### 5.2.1 Cruise control – Cc

För att demonstrera cruise controlfunktionen behövde ett antal ”knappar” aktiveras. Förloppet inleddes med en manuell acceleration till ca 100 km/h. Detta kan ses på den lila linjen som representerar hastigheten. Se Figur 5.6. Under hastighetsökningen aktiverades *B\_CcOn*, vilket möjliggör cruisekörning. När önskad hastighet uppnåtts triggas *B\_CcSetPlus* efter 17 s. I grafen kan man se att den gröna Cc-linjen sätts på ca 100 km/h. Accelerationspedalen kan nu släppas helt och hastigheten behålls.

No.	Color	Name	Units
1	—	[1] sVcScDep_X_AccPedalPos	%
2	—	[1] rVcExtIn_Tq_EngInst	Nm
3	—	[1] sVcEc_v_Veh	km/h
4	—	[1] rVcExtIn_F_Brake	N
5	—	[1] sVcDeScI_v_CcSetInt	km/h
6	—	[1] xVcDeScI_B_Cancel	-
7	—	[1] xVcDeScI_B_CcOff	-
8	—	[1] xVcDeScI_B_CcOn	-
9	—	[1] yVcDeScI_B_CcResumeReq	-
10	—	[1] xVcDeScI_B_Act	-

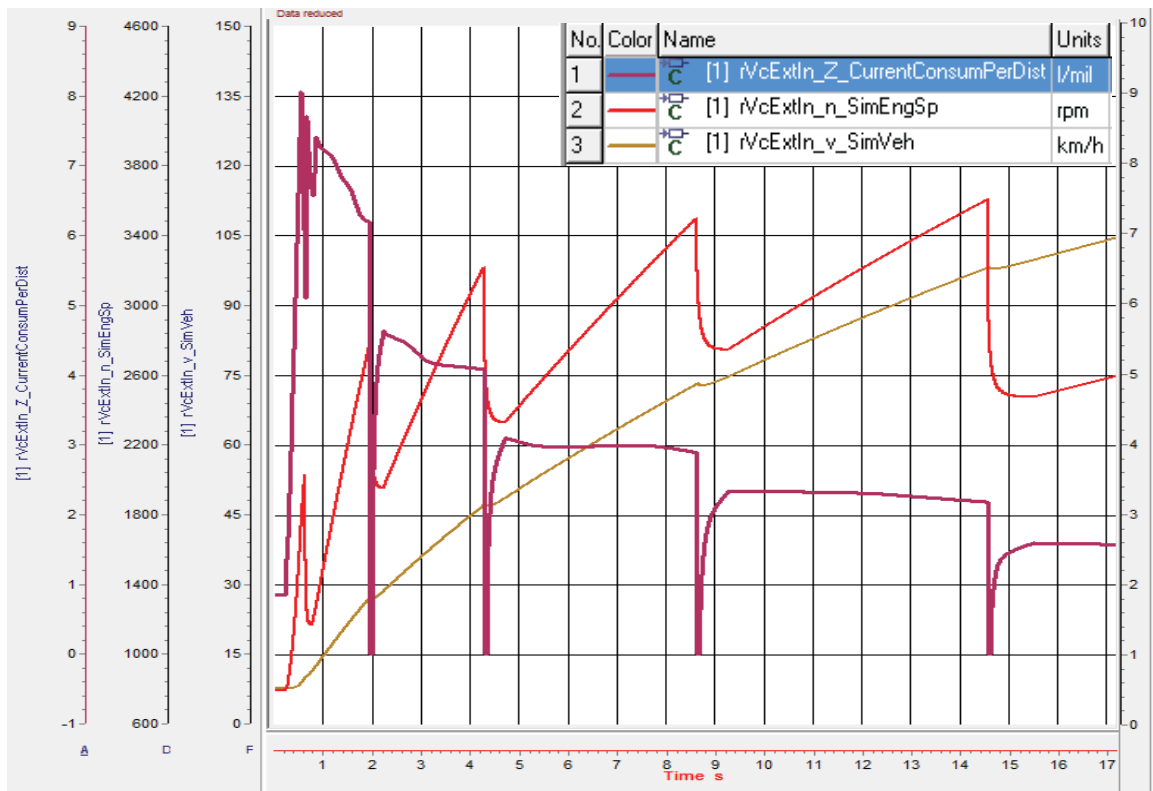
Vid ca 23 s aktiveras *B\_CcCancel*, vilket stänger av Cc. Därefter läggs en bromskraft på manuellt. Vid ca 30 s triggas *B\_CcResume* och Cc börjar jobba för att nå upp till den tidigare inställda hastigheten. *B\_CcOff* används sedan för att stänga av Cruise control.



Figur 5.6. Demonstration av cruise control.

## 5.2.2 Bränsleförbrukning

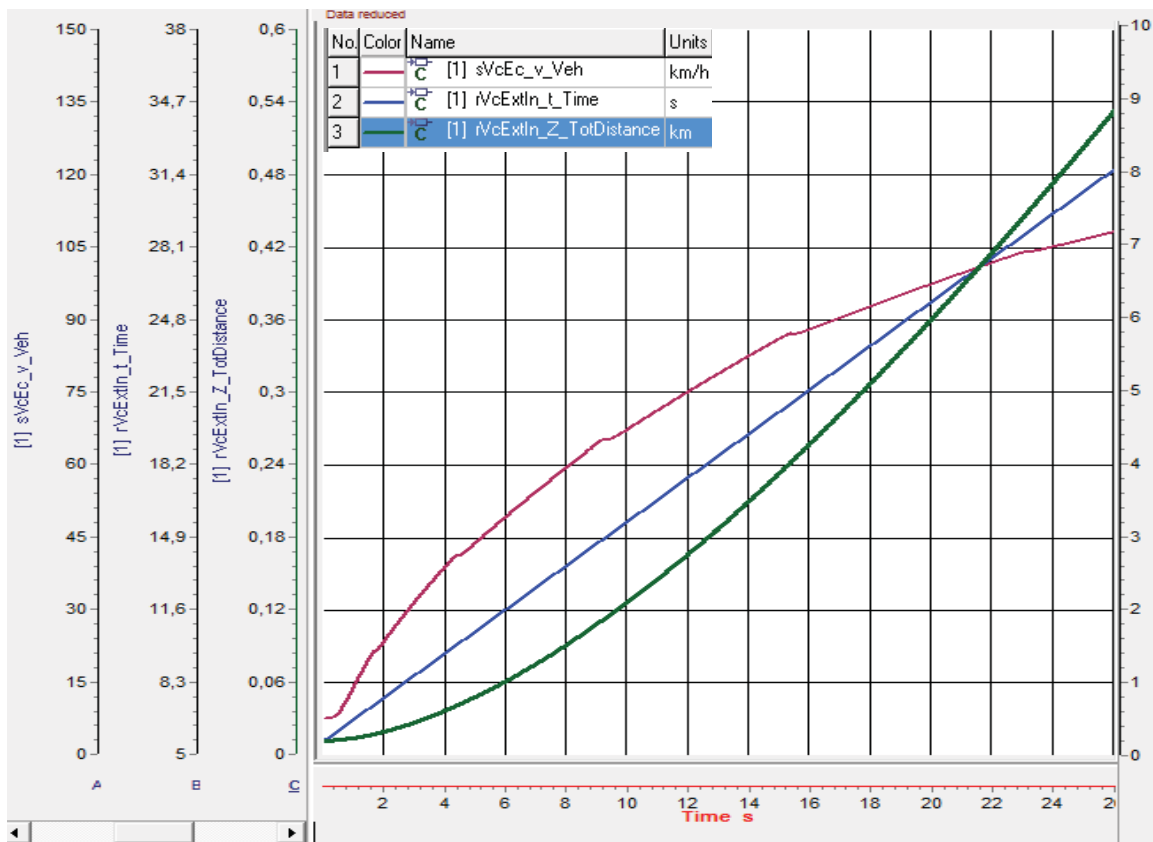
I denna graf presenteras förhållandet mellan momentanförbrukning och varvtal då 40 % gaspådrag är pålagt. Det kan då lätt utläsas att konverterern under accelerationen tar anspråk på en stor del av momentet på grund av sin ineffektivitet. Vid varje växling ser man också ett tydligt neddrag på den momentana förbrukningen. Förbrukningen sjunker även när fordonet närmar sig sin målhastighet. Se Figur 5.7.



Figur 5.7. Exempel på bränsleförbruknings vid 40 % accelerationspådrag.

### 5.2.3 Trippmätare, tid och hastighet

Här visas sambanden mellan sträcka, hastighet och tid. Se Figur 5.8. Detta experiment utfördes med ett konstant gaspådrag 40 %.

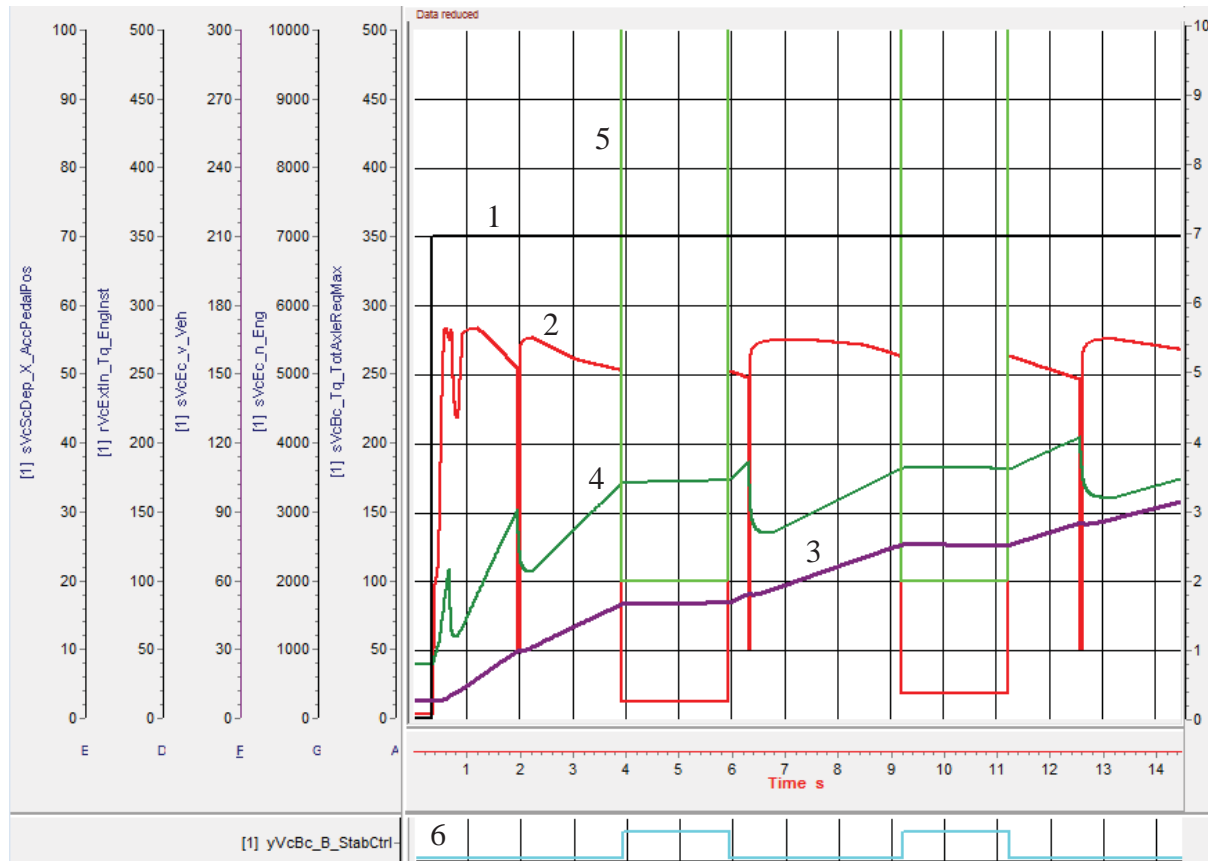


Figur 5.8. Graf för sträcka, hastighet och tid, vid 40 % accelerationspådrag.

## 5.2.4 Momentneddrag vid däcksläpp - "DSTC"

Här visas exempel på ett momentneddrag då fordonet utsätts för däcksläpp. Det syns ett tydligt utslag på både begärda momentet och hastigheten när det simulerade däcksläppet inträffar. Se Figur 5.9.

No.	Color	Name	Units
1	Black	[1] sVcScDep_X_AccPedalPos	%
2	Red	[1] rVcExtIn_Tq_EngInst	Nm
3	Purple	[1] sVcEc_v_Veh	km/h
4	Green	[1] sVcEc_n_Eng	rpm
5	Light Green	[1] sVcBc_Tq_TotAxleReqMax	Nm
6	Light Blue	[1] yVcBc_B_StabCtrl	.

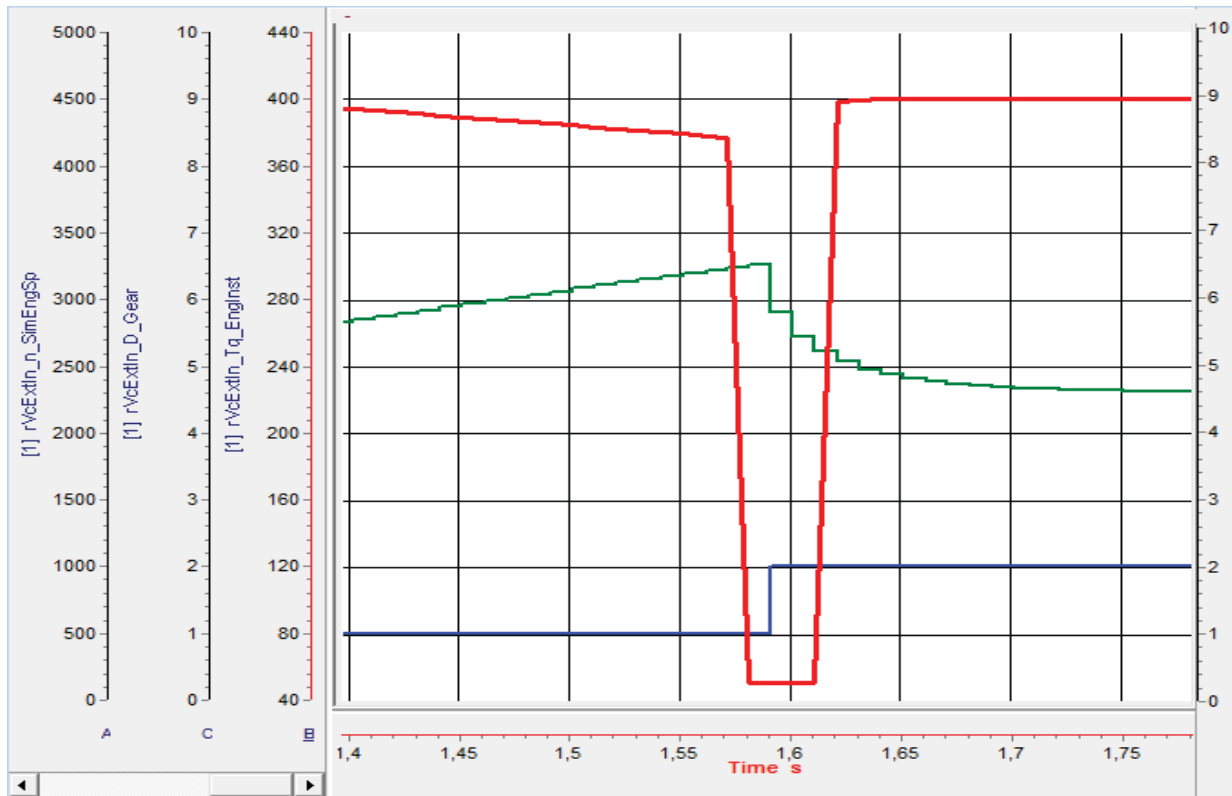


Figur 5.9. Exempel på momentneddrag vid däcksläpp.

## 5.2.5 Momentneddrag vid växling

Strax innan växling ska det ske ett neddrag på det begärda momentet, detta visas i *Figur 5.10*. I detta exempel sker växlingen mellan växel 1 och upp till växel 2 *D\_Gear*. Precis innan växlingen dras momentet *Tq\_EngInst* till 50 Nm för att skydda växellådan. När växlingen är klar återgår momentet.

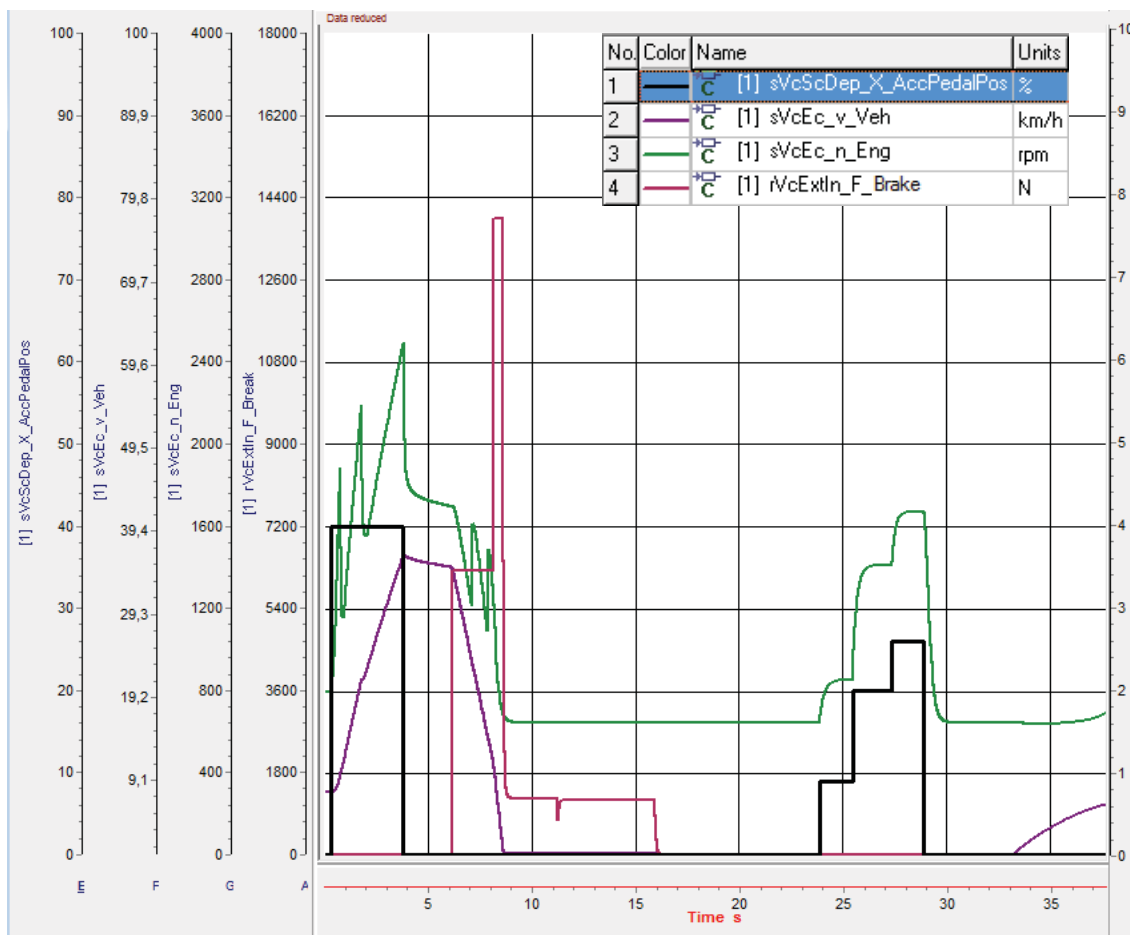
No.	Color	Name	Units
1	Green	[1] rVcExtIn_n_SimEngSp	rpm
2	Blue	[1] rVcExtIn_D_Gear	-
3	Red	[1] rVcExtIn_Tq_EngInst	Nm



Figur 5.10. Exempel på momentneddrag vid växling.

## 5.2.6 Neutralläge

I vissa lägen vill man kunna stanna bilen och släppa bromsen med motorn fortfarande igång. Därför behövdes en funktion för neutralläge tas fram. Grafen nedan presenterar ett körfall där fordonet första accelererar till lämplig hastighet och sedan bromsar bilen till stillastående och där efter läggs neutralväxel i. Sen är neutralväxel i under hela förloppet, gas påläggs 24 s, motorns varvtal ökar och bilen står still. Sista delen vid 33 s läggs växel i igen och fordonet rullar iväg. Se Figur 5.11.



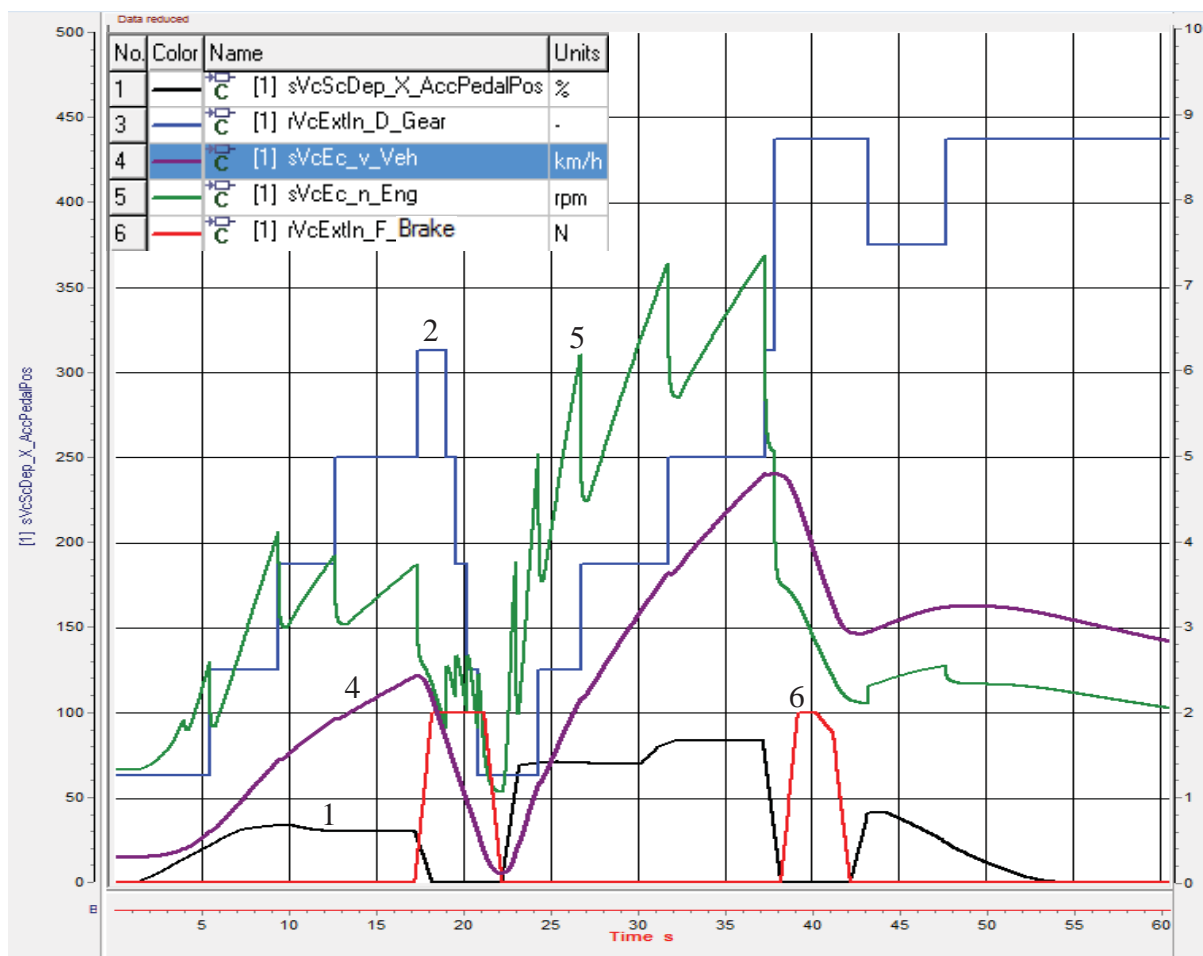
Figur 5.11. En demonstration om neutralväxlens funktion.



## 5.2.7 Förarmodell

Här presenteras graferna till den förarmodell som tagits fram. Se Figur 5.12. Den förprogrammerade körprofil sträcker sig över 60 s och testar både acceleration, broms och växlingar. Utifrån denna förarmodell kan intressant data tas ut. Eftersom modellen kan köras med exakt samma körprofil varje gång kan man lättare analysera resultatet av att ändra en eller flera parametrar/ kalibrerbar variabler. Den förutbestämda körprofilen för gaspedal och bromspedal finns att hitta i Bilaga A sida A1.

Alla axlar på vänstersidan förutom  $X\_AccPedalPos$  har skalats bort från denna graf för öka bildytan och underlätta förståelsen för grafen. En bild med alla axlar återfinns i Bilaga A sida A2.

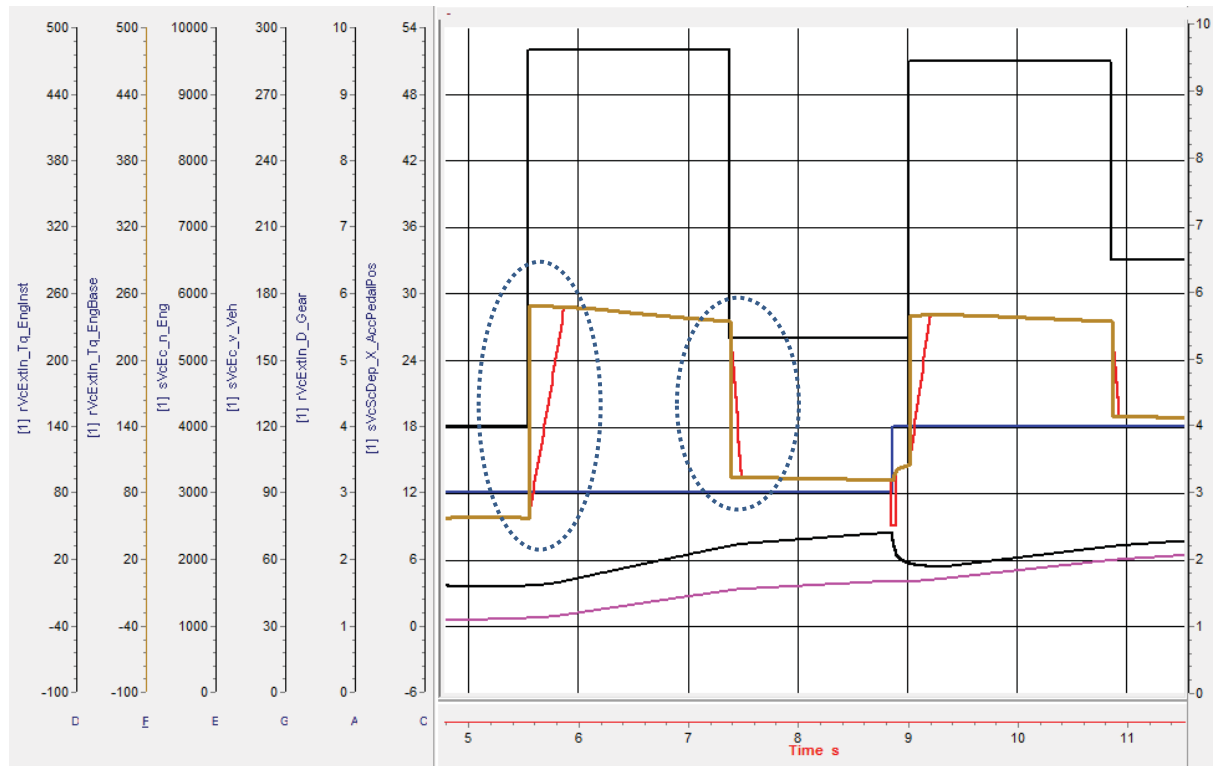


Figur 5.12. Graf för förprogramerad förarmodell.

## 5.2.8 Slag och såg

Då det alltid eftersträvas så mjuk körning som möjligt måste det finnas vissa begränsningar då ett mer eller mindre kraftigt gaspådrag sker. Därför finns en viss funktionalitet som hanterar dessa momentförändringar. I Figur 5.13 nedan ser man hur  $Tq_{EngInst}$  är begränsad genom rampning vid både gaspådrag och snabb minskning av gas.

No.	Color	Name	Units
1	Red	[1] rVcExtIn_Tq_EngInst	Nm
2	Blue	[1] rVcExtIn_Tq_EngBase	Nm
3	Black	[1] sVcEc_n_Eng	rpm
4	Magenta	[1] sVcEc_v_Veh	km/h
5	Dark Blue	[1] rVcExtIn_D_Gear	-
6	Black	[1] sVcScDep_X_AccPedalPos	%



Figur 5.13. Demonstration hur slag och såg-funktioner fungerar.

## 6. SLUTSATS

Under denna rubrik besvaras frågeställningarna som ställdes under rubriken ”precisering av arbetsuppgiften” och en diskussion om den framtagna simulatorm.

### 6.1 Besvarande av frågeställningar

#### *Vad behövs för parametrar/ingångar för att få en realistisk simulator?*

Det måste först definieras vad som är en realistisk nivå. Den realistiska nivån har i detta projekt även en korrelation till hur omfattande arbetet skulle komma att bli. Detta eftersom allt fler parametrar/funktioner skulle behövas implementeras för att uppnå en högre realistiska nivå på simulatorm.

Nivån för vår simulator fastställdes gemensamt med handledaren på Volvo. För att erhålla en kvalitativt godtagbar modell fastlades det även att det är bra att använda: massa, max/min-moment från motorn, enkel momentomvandlarmodell (konverter) och utväxlingsförhållanden.

#### *Vilken funktionalitet önskas av simulatorm?*

För att simulatorm ska kunna vara användbar för Volvo krävs det att tillräckligt många funktioner är implementerade för att kunna utföra enklare tester mot styrenheten. Modellen som tagits fram ska även lätt kunna underhållas om funktioner i motorstyrenheten byts ut. Fler funktioner ska kunna byggas ut om mer funktioner efterfrågas.

Slutprodukten av detta arbete är en simulator vilken kommer kunna hjälpa funktionsutvecklare på avdelningen att verifiera sin mjukvara tidigare. Genom detta kan man även innan funktionsduglig testbil finns att tillgå kunna testa sina funktioner mot styrenheten. Motorsystemet är en väldigt grundläggande del i bilarna eftersom den hanterar många säkerhetsfunktioner. Det finns därför mycket som kan gå fel och eventuellt skada utrustning på bilen om mjukvaran inte är funktionssäkrad. Med vår simulator kan man nu testa enklare funktioner som tidigare inte hade varit aktuella för verifiering i bil.

#### *Kompatibelt för andra fordonsplattformar?*

Den nuvarande EUCD-plattformen har snart uppnått sin pensionsålder och kommer modell för modell systematiskt bytas ut mot den nya SPA-plattformen (Scalable Platform Architecture). Det kommer därför fortsätta rulla ut bilar med den gamla plattformen tills alla modeller är utbytta.

Den plattform vi har utvecklat simulatorm till är den äldre EUCD-plattformen. Anledningen till detta är att EUCD är en stabil och beprövad fordonsplattform som även är enklare att felsöka. Vi har däremot under arbetets gång förebyggt möjligheten att även kunna köra simulatorm mot SPA-plattform. Detta har vi gjort genom att samla alla in- och utgångar till alla verkliga motorfunktioner i två separata block så man relativt enkelt ska kunna byta ut de adresser som signalerna ska skrivas över på. Förändringarna kommer att vara dels nya signaler och till viss del ny signalering.

### *Påbyggnadsmöjligheter efter vårt arbete?*

Då denna simulator endast var avsedd att beröra de allra mest grundläggande funktionaliteterna i motorstyrsystemet finns det därför många påbyggnadsmöjligheter.

Nedan listas de större utbyggningsmöjligheter som finns för simulatorm:

- Bensinmotor
- Manuell växellåda
- Hybrid funktionalitet
- Start-Stopp-funktionalitet
- Däck och vägunderlagsmodell
- SPA-Plattform
- Utförligare emissionsberäkningar
- Oscillation och fördröjningar på grund av flexibilitet i drivlina

## **6.2 Utvärdering av den framtagna simulatorm**

Den slutgiltiga simulator som tagits fram i detta arbete har uppvisat en tillräckligt verklighetstrogen karaktär för att kunna klassas som realistisk. Denna uppfattning grundar sig i både handledarens expertutlåtande och en överskådlig jämförelse mellan mätdata från verklig bil och simulator. Handledaren har även under arbetets gång gjort sitt utlåtande vid större milstolpar i projektet för att verifiera att modellen uppvisar en verklighetstrogen karaktär.

## **6.3 Verklighetsförankrade brister och förbättringar**

Då det alltid är svårt att återskapa exakta verklighetstroga modeller finns ständigt saker som kan förbättras. De saker vilka avsiktligt utelämnats i detta arbete är funktioner som av handledare ansetts vara tillräckligt utförliga. Det insågs tidigt i arbetet att vissa förenklingar och begränsningar var tvungna att göras för att inte projektet skulle bli för omfattande. Några av dessa förenklingar som kan förbättras är:

- Växlingstid
- Väglast
- Bränsleförbrukning
- Temperaturförhållanden (kallstart)
- Höghöjdsmapping (Luftryck)
- Friktion i växellåda

## **6.4 Framtida utvecklingsmöjligheter för simulatorn**

Utöver frågeställningens utbyggnadsmöjligheter finns vissa ytterligare förbättringar som inte berör funktionsutveckling.

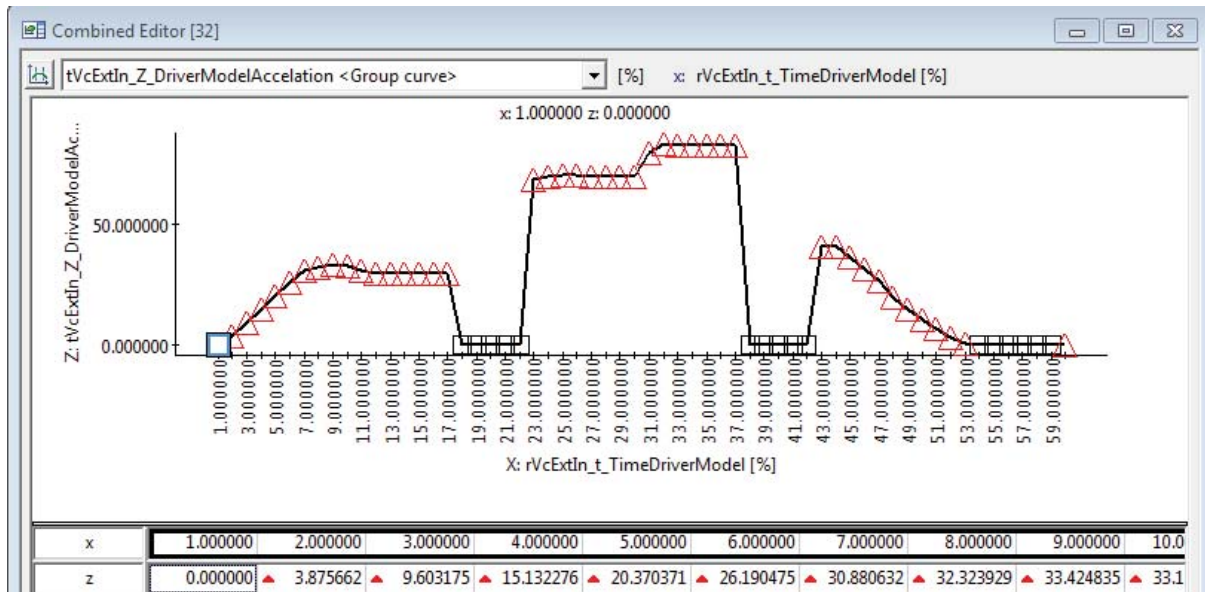
Mellan handledare och gruppchef har det diskuterats om att inrätta en teststation på avdelningen som alla funktionsutvecklare kan koppla upp sig mot för att testa sina funktioner. Genom att ha en central station eller ett nätverk som alla i gruppen har tillgång till kan man lättare börja använda detta som ett arbetssätt för testning och verifiering av motorfunktioner.

Det bör även tas fram tydliga instruktioner för hur simulatorn används och fungerar.

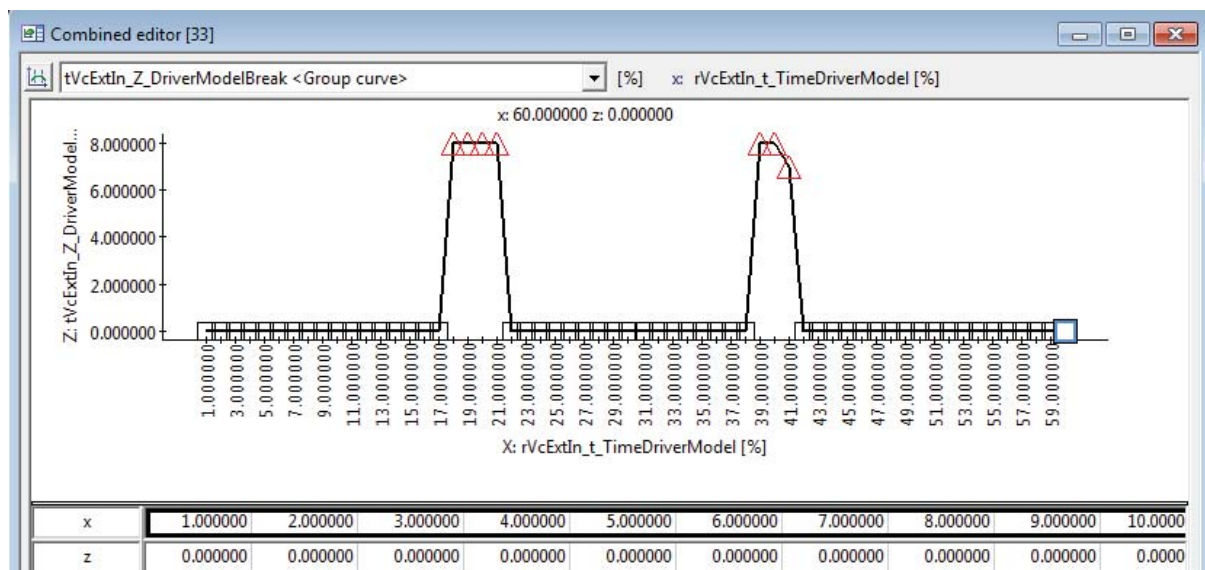
## 7. REFERENSER

- [1] "Wikipedia," Volvo Personvagnar, [Online]. Available: [http://sv.wikipedia.org/wiki/Volvo\\_Personvagnar](http://sv.wikipedia.org/wiki/Volvo_Personvagnar). [Använd 05 06 2014].
- [2] "Wikipedia," Simulink, [Online]. Available: <http://en.wikipedia.org/wiki/Simulink>. [Använd 05 06 2014].
- [3] "Wikipedia," TargetLink, [Online]. Available: <http://en.wikipedia.org/wiki/TargetLink>. [Använd 05 06 2014].
- [4] "Wikipedia," INCA, [Online]. Available: [http://en.wikipedia.org/wiki/INCA\\_\(software\)](http://en.wikipedia.org/wiki/INCA_(software)). [Använd 05 06 2014].
- [5] K. Nice, "Howstuffworks," Konverter och Lock-Up, [Online]. Available: <http://auto.howstuffworks.com/auto-parts/towing/towing-capacity/information/torque-converter.htm>. [Använd 05 06 2014].
- [6] D. J. Dohner, "A Mathematical Engine Model for Development of Dynamic Engine Control," SAE Technical Paper, Detroit, 1980.
- [7] D. Cho och J. K. Hedrick, "Automotive Powertrain Modeling for Control," 1989.
- [8] R. Rajamani, Vehicle Dynamics and Control, Springer, 2006.

# BILAGA A. Förarmodell Sida A1 (2)

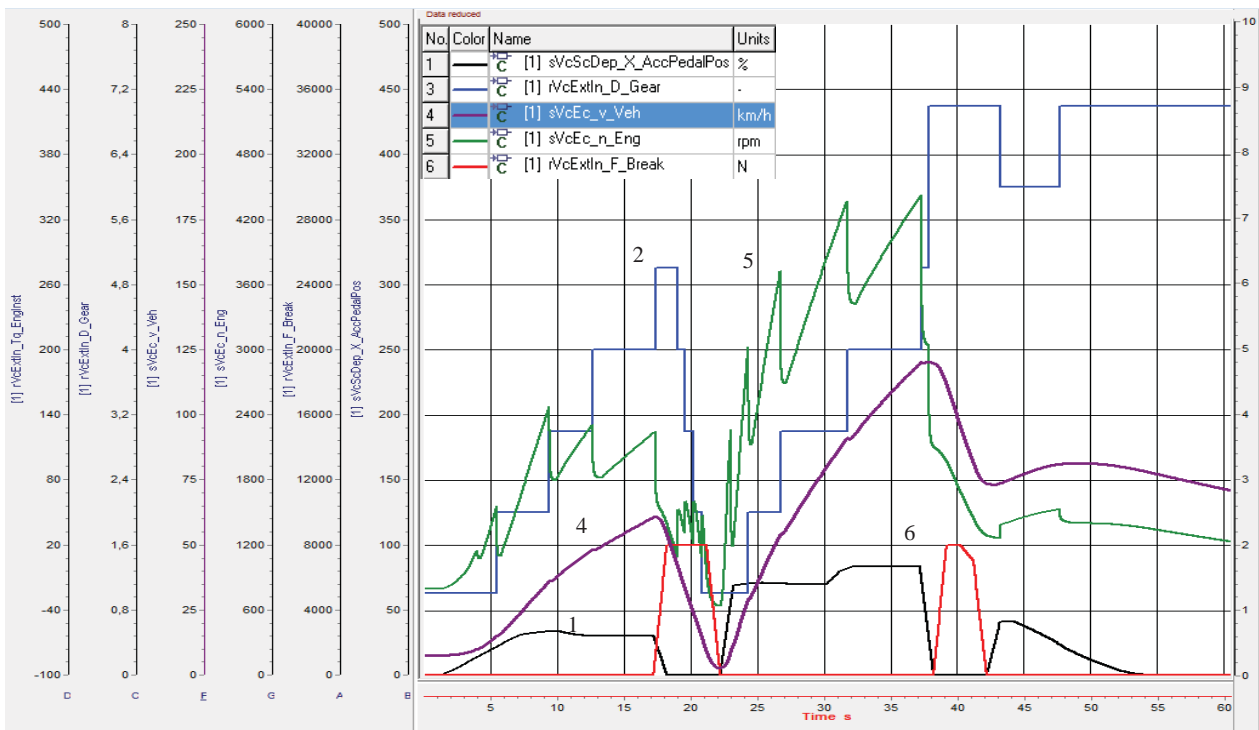


*Gaspedaltabell.*



*Bromspedaltabell*

## BILAGA A. Förarmodell Sida A2 (2)



Detta är en förstoring av grafen för förarmodellen med alla axlar.



## BILAGA B. Förstoringar av modeller. Sida B1 (8)

### Blockträd

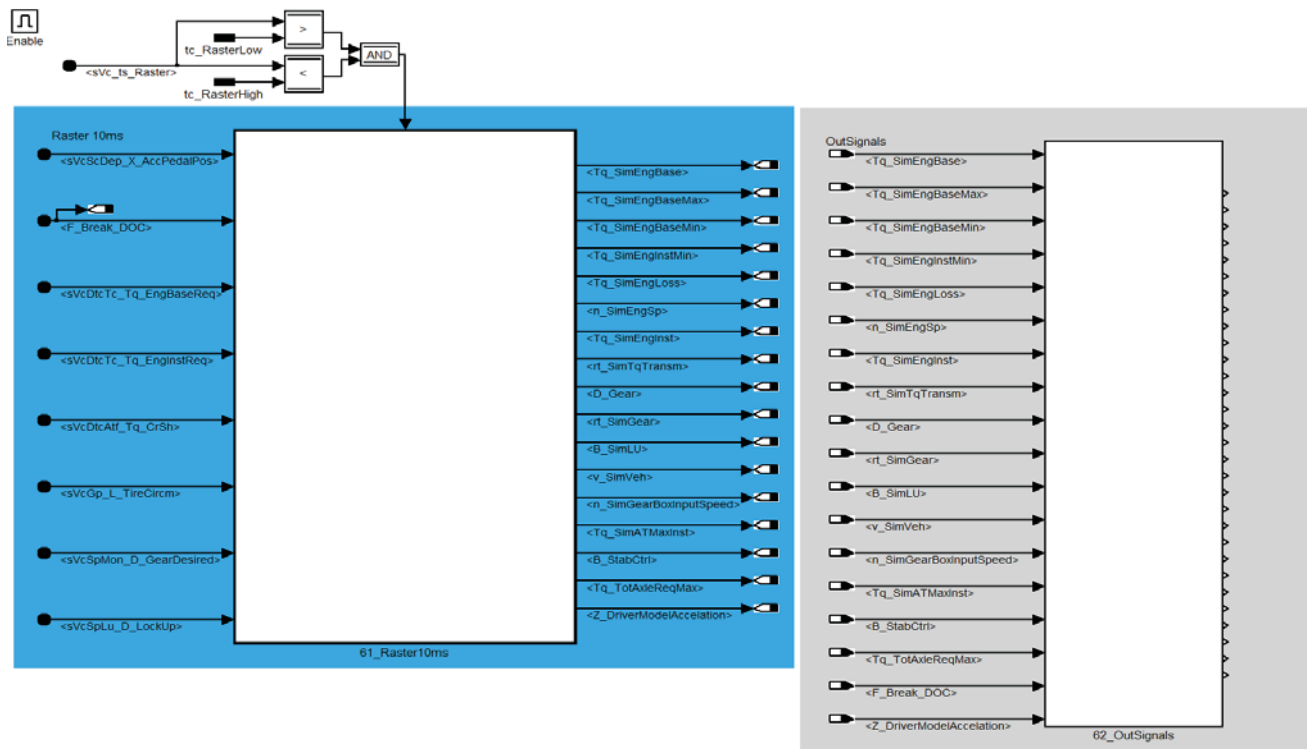
#### 6\_Simulator

- 61\_Raster10ms
  - 611\_EngineAndConverter
    - 6111\_DieselEngine
    - 6112\_PetrolEngine
    - 6113\_Converter
  - 612\_ATGearbox
  - 613\_Vehicle
  - 614\_PedalMap
  - 615\_InSignals
  - 616\_Bsl
- 62\_OutSignals

*Här beskrivs blockträds uppbyggnaden för simulatoren i Simulink/TargetLink.*

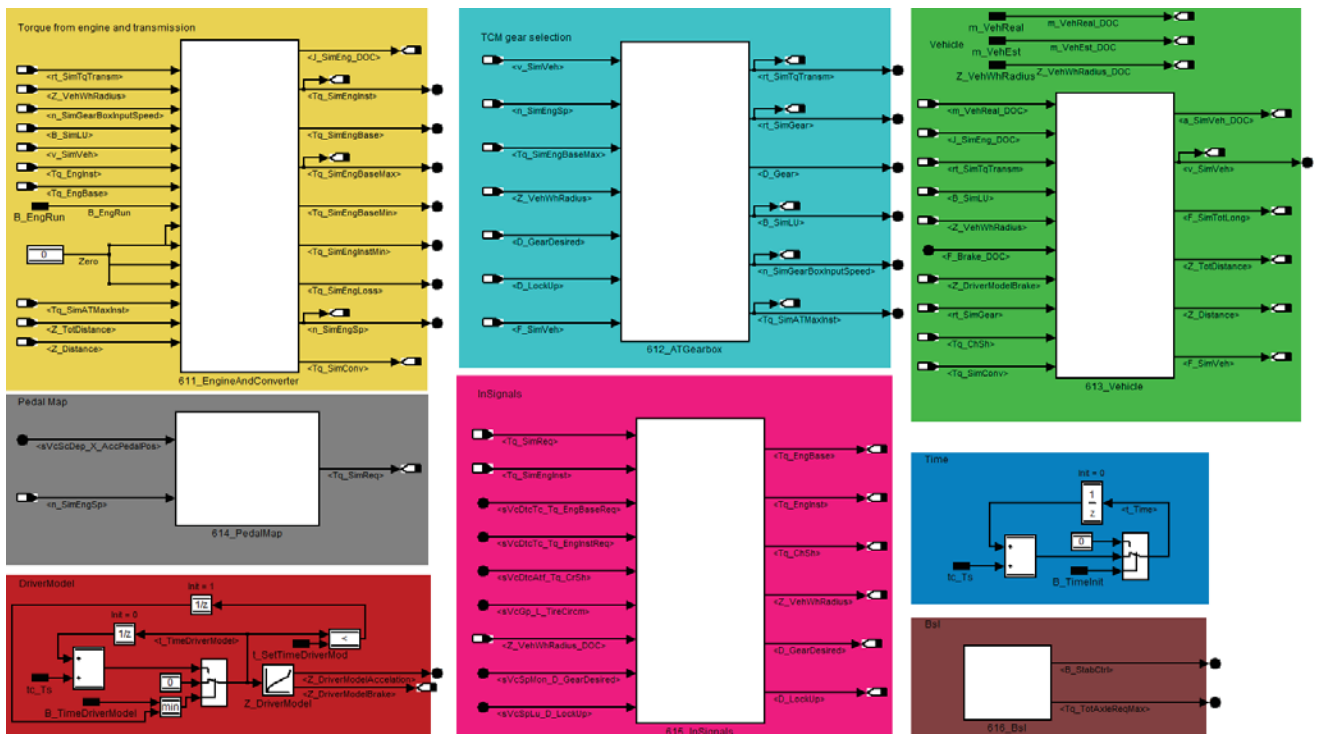
.

## BILAGA B. Förstorningar av modeller. Sida B2 (8)



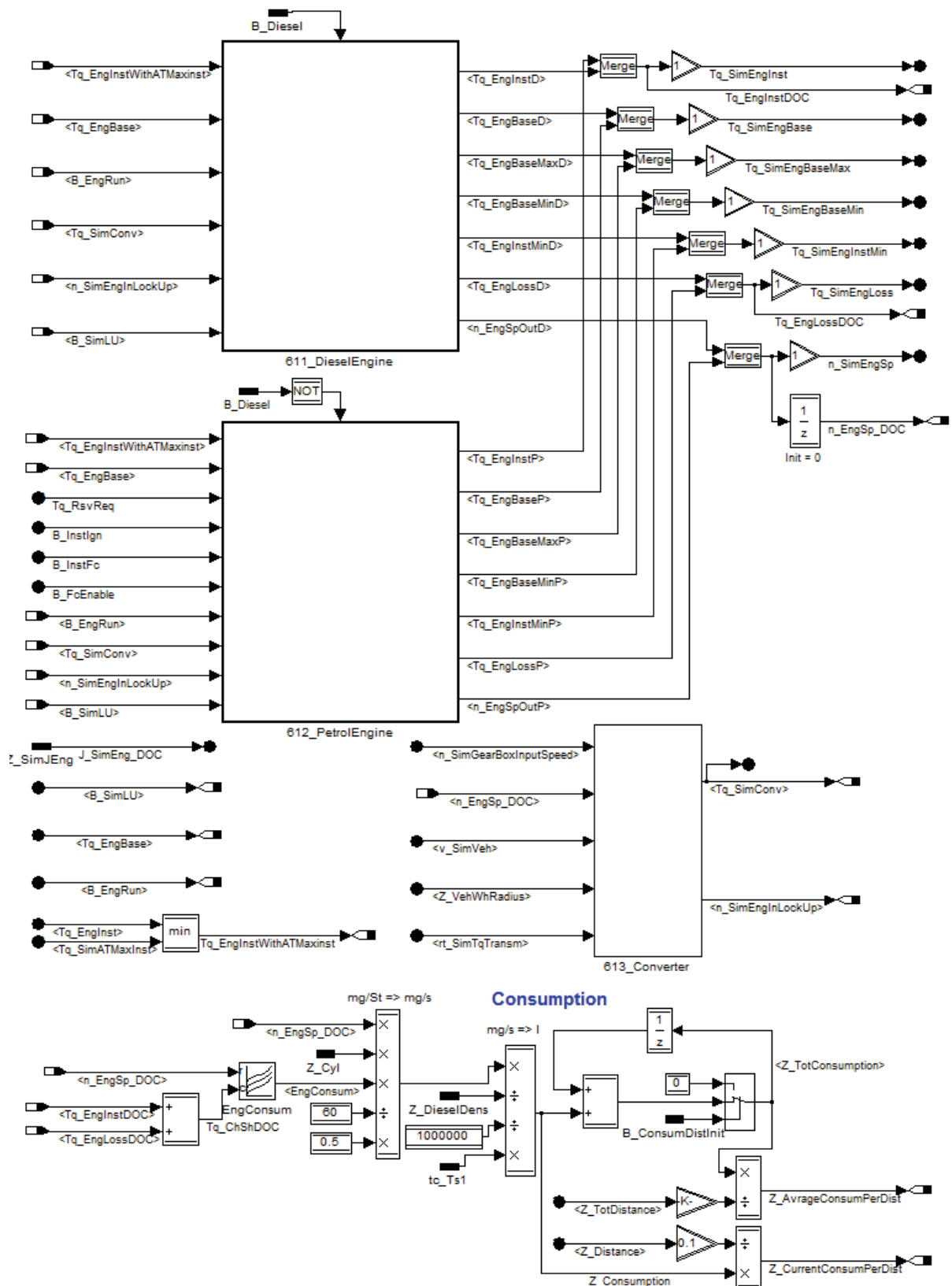
Detta är huvudnivån i simulatorns blockträd, här styrs vilka delar av simulatorn som ska utföras i vilket raster.

## BILAGA B. Förstorningar av modeller. Sida B3 (8)



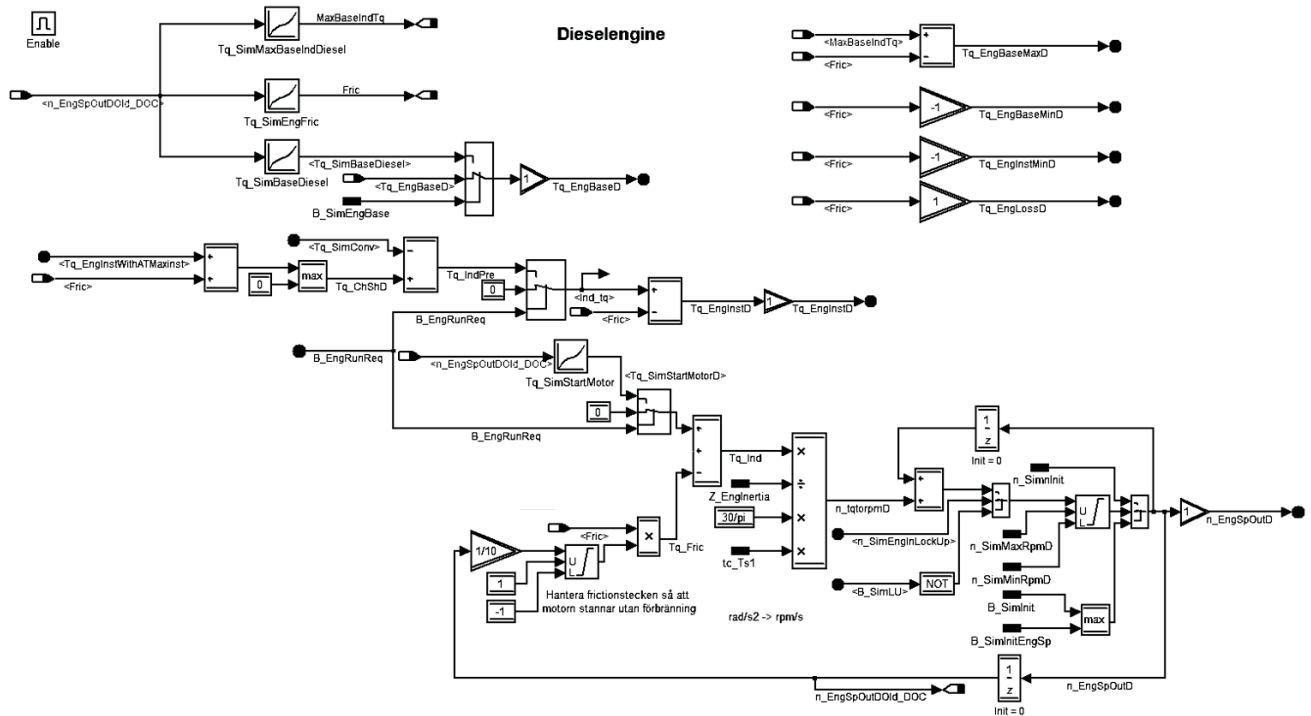
Dessa block ligger under 61\_Raster10ms och här ligger själva modellerna över motorn, växellådan och fordonet plus några andra funktioner.

## BILAGA B. Förstöringar av modeller. Sida B4 (8)



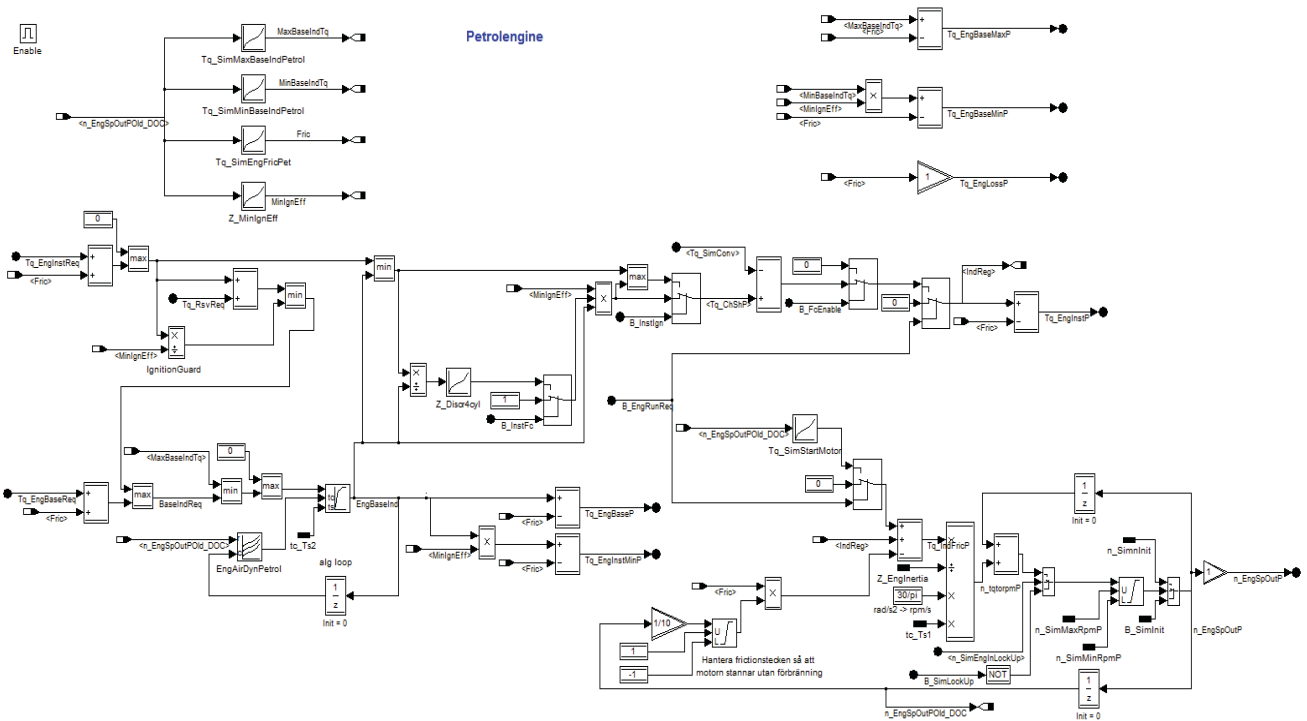
Detta är innehållet under 611\_EngineAndConverter, här ligger även beräkningen av bränsleförbrukningen.

## BILAGA B. Förstorningar av modeller. Sida B5 (8)



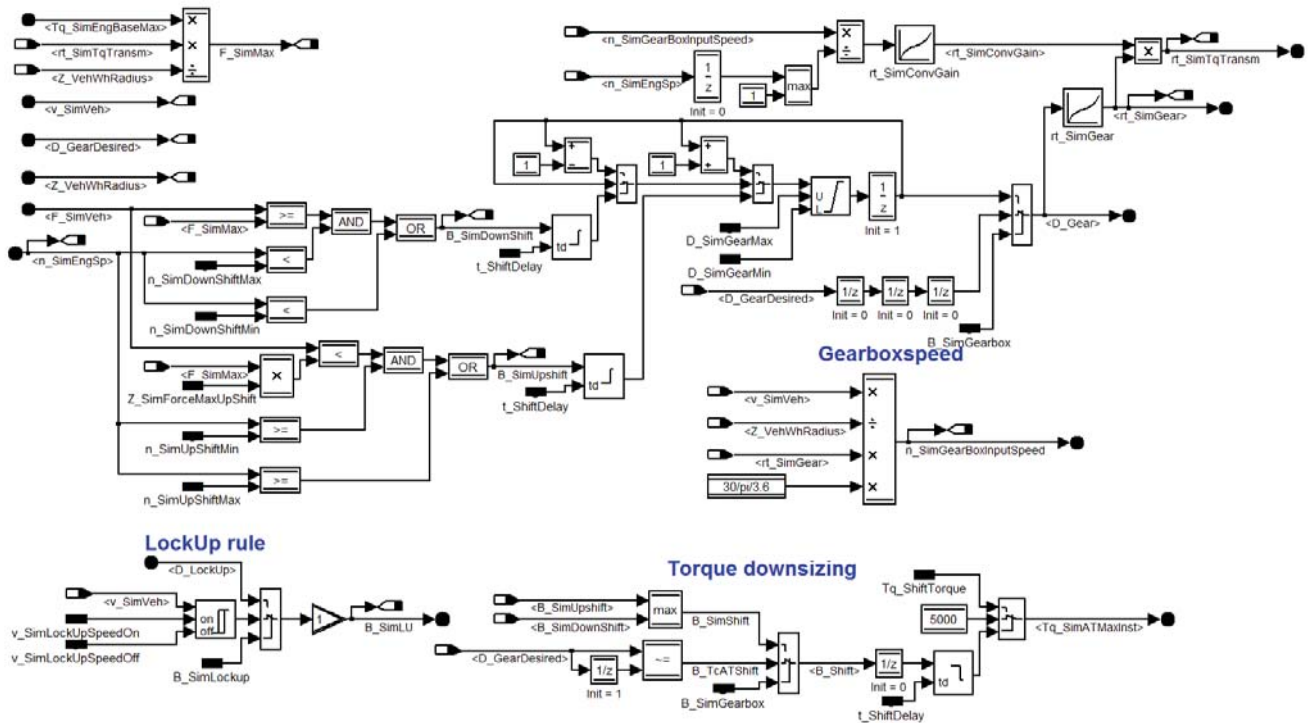
Detta är logiken för Dieselmotorn (6111\_DieselEngine).

## BILAGA B. Förstorningar av modeller. Sida B6 (8)



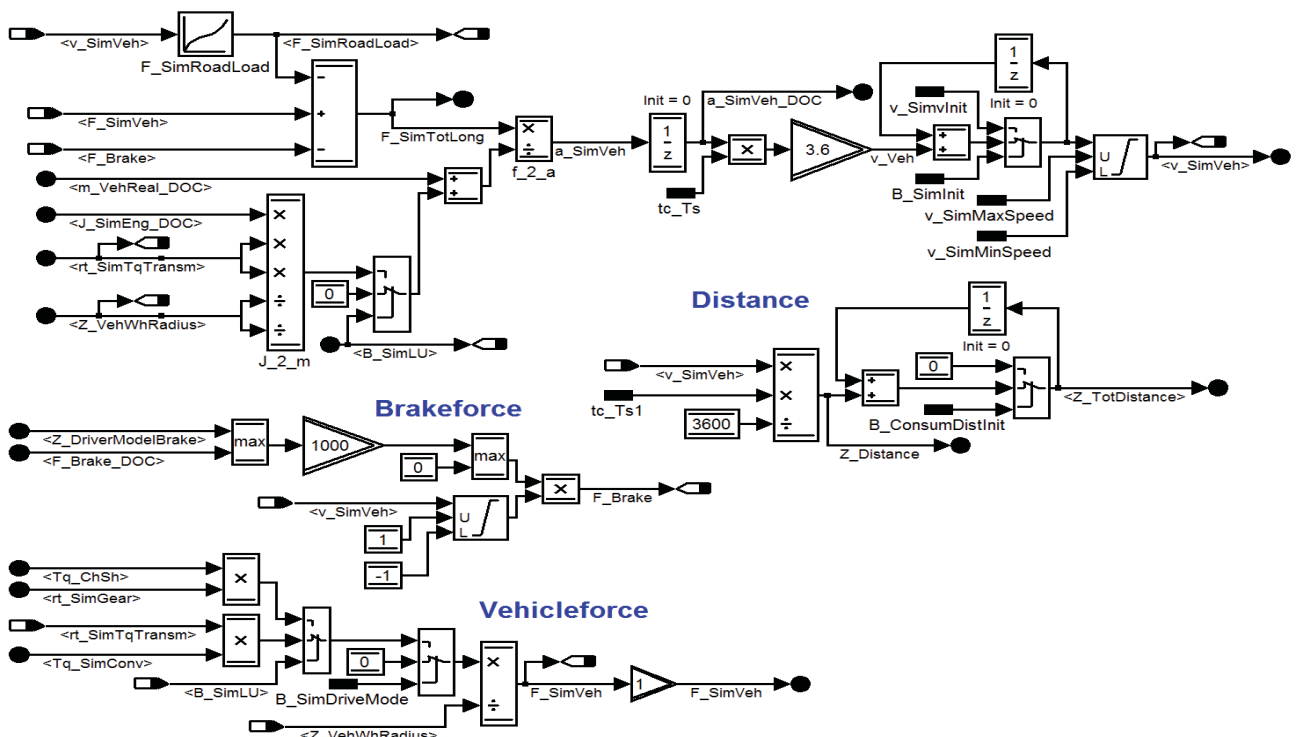
Detta är logiken för bensinmotorn, den har inte behandlats under projektet, utan så såg den ut när den erhöles från Volvo (6112\_PetrolEngine).

**BILAGA B. Förstöringar av modeller. Sida B7 (8)**



Detta är logiken för växellådan, här ligger även momentneddraget för växlingar (612\_ATGearbox).

**BILAGA B. Förstorningar av modeller. Sida B8 (8)**



Detta är logiken för fordonet, här ligger även beräkningen för trippmätaren (613\_Vehicle).



### BILAGA C. Bränsle tabell. Sida C1 (1)

Rpm/Nm	500	600	700	800	900	1000	1125	1250	1375	1500	1625	1750	1875	2000	2250	2500	2625	2750
-35,0313	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-15,9127	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3,2058	0.548	0.548	0.548	0.503	0.474	0.4802	0.4932	0.4848	0.4697	0.4731	0.4811	0.4951	0.4941	0.4758	0.4571	0.5527	0.6106	0.6950
22,3244	3.666	3.666	3.666	3.380	3.211	3.2478	3.3222	3.2749	3.1822	3.1723	3.2243	3.3182	3.3691	3.3447	3.2533	3.6360	3.7688	3.9536
41,4429	6.960	6.960	6.960	6.345	5.919	5.9573	6.0344	5.9238	5.7890	5.6817	5.7332	5.8559	6.0131	6.1539	6.1355	6.2367	6.3005	6.4457
60,5614	10.27	10.27	10.27	9.981	9.538	9.4330	9.2401	8.9164	8.8929	8.8370	8.5952	8.4547	8.5658	8.7689	8.9230	8.7649	8.8109	9.0007
79,68	13.01	13.01	13.01	13.08	13.32	13.409	13.018	12.397	12.274	12.405	12.111	11.584	11.380	11.451	11.886	11.591	11.529	11.681
98,7985	16.17	16.17	16.17	16.43	16.79	17.017	16.980	16.448	15.942	15.907	15.915	15.642	15.010	14.603	14.894	14.912	14.834	14.907
117,917	20.18	20.18	20.18	20.42	20.72	20.775	20.597	20.184	19.667	19.377	19.260	19.147	18.821	18.274	17.629	18.322	18.558	18.741
137,036	25.90	25.90	25.72	25.11	24.63	24.189	23.750	23.321	22.776	22.317	22.036	21.856	21.813	21.631	20.628	21.105	21.431	21.688
156,154	33.61	33.61	31.90	29.36	28.80	27.871	27.243	26.799	26.152	25.616	25.266	24.935	24.673	24.490	24.324	24.206	24.430	24.727
175,273	41.75	41.75	40.99	36.07	33.93	32.212	31.292	30.713	30.109	29.767	29.468	28.982	28.198	27.623	28.099	27.948	27.886	28.044
194,391	46.36	46.36	45.99	42.47	37.59	36.513	35.529	34.728	33.696	33.355	33.047	32.558	31.992	31.465	31.294	31.199	31.061	31.132
213,51	49.64	49.64	49.31	46.41	42.88	39.957	38.777	38.232	37.244	36.486	36.160	35.895	35.582	35.245	34.464	34.209	34.127	34.260
232,628	52.95	52.95	52.60	49.51	46.62	44.388	41.633	40.950	40.327	39.474	39.201	39.038	38.667	38.288	37.615	36.914	36.908	37.127
251,747	56.26	56.26	55.91	52.94	49.65	47.319	44.157	43.290	43.275	43.123	42.834	42.409	41.725	41.188	40.703	39.536	39.569	39.865
270,865	59.56	59.56	59.22	56.38	53.38	50.835	47.317	45.689	45.846	45.932	45.789	45.436	44.722	44.102	43.898	42.641	42.559	42.821
289,984	63.01	63.01	62.65	59.77	56.99	55.031	51.798	49.266	48.446	48.301	48.147	47.873	47.449	47.017	46.830	46.216	46.038	46.061
309,102	66.56	66.56	66.18	63.38	60.49	58.547	55.877	53.376	51.615	51.006	50.683	50.431	50.256	50.018	49.598	49.882	49.662	49.398
328,221	70.18	70.18	69.81	67.09	64.30	62.359	59.497	57.379	55.077	54.296	53.916	53.716	53.643	53.443	52.764	53.307	53.227	53.081
347,339	73.74	73.74	73.37	70.83	68.23	66.408	63.658	61.431	59.014	57.698	57.225	57.037	56.955	56.755	56.053	56.593	56.616	56.633
366,458	77.38	77.38	76.99	74.45	72.02	70.459	67.887	65.600	63.313	61.220	60.606	60.393	60.217	59.971	59.467	59.750	59.840	59.987
385,577	81.14	81.14	80.72	78.18	75.71	74.127	71.811	69.745	67.389	64.924	64.150	63.854	63.496	62.958	62.752	62.786	62.889	63.084
404,695	85.16	85.16	84.73	82.13	79.53	77.977	75.490	73.362	71.178	68.836	68.217	68.273	67.498	66.364	65.994	65.814	65.901	66.098
423,814	89.58	89.58	89.12	86.39	83.75	82.134	79.531	77.296	74.815	72.671	72.097	72.217	71.603	70.583	69.194	68.849	68.848	68.978

Detta är en del av bränsletabellen som användes i projektet, tyvärr kunde inte hela bifogas på grund utav storleken 30X30.

Men den ger en inblick om värdena, det utlästa värdet har enheten mg/Stroke.

