

Vedlegg

A Forprosjektsrapport

FORPROSJEKT - RAPPORT

FOR BACHELOROPPGAVE

TITTEL:

Modell asynkronmaskin

KANDIDATNUMMER(E):
SINGHARAT BOONKRONG
TOMAS SKAAR VADSET
OLA EDVARD NORDVIK
HÅVARD LERVIK

DATO:

1.02.2019

EMNEKODE:

IE303612

EMNE:

Bacheloroppgave

DOKUMENT TILGANG:

- Åpen

STUDIUM:

**BACHELOR I INGENIØRFAG,
ELKRAFTSYSTEM**

ANT SIDER/VEDLEGG:

9/2

BIBL. NR:

- Ikke i bruk -

OPPDRAAGSGIVER(E)/VEILEDER(E):

STADT AS

OPPGAVE/SAMMENDRAG:

Firmaet STADT AS ønsker å ha en modell for en asynkronmaskin for å utrede resultatet av å koble om mellom forskjellige driftsmodus ved ulike tidspunkt og driftssituasjoner. Det skal gjøres ved at den leser av strøm, strøm og fasevinkel. De ønsker også å få mest mulig informasjon fra motoren. Denne bacheloroppgaven ble gitt til studentene ved avdeling for ingeniørfag og realfag, elkraftsystem.

Oppgaven går ut på å lage en slik modell med verktøy som gjør det mulig for oppdragsgiver å ta i bruk eller videreutvikle denne modellen til det arbeidet som de ønsker. Det skal også bli undersøkt om hvilke alternativer som allerede finnes.

Denne oppgaven er en eksamensbesvarelse utført av student(er) ved NTNU i Ålesund.

Postadresse
Høgskolen i Ålesund
N-6025 Ålesund
Norway

Besøksadresse
Larsgårdsvegen 2
Internett
www.hials.no

Telefon
70 16 12 00
Epostadresse
postmottak@hials.no

Telefax
70 16 13 00

Bankkonto
7694 05 00636
Foretaksregisteret
NO 971 572 140

INNHold

1 INNLEDNING	3
2 BEGREPER	3
3 PROSJEKTORGANISASJON.....	3
3.1 PROSJEKTGRUPPE	3
3.2 STYRINGSGRUPPE (VEILEDER OG KONTAKTPERSON OPPDRAGSGIVER)	4
4 AVTALER.....	4
4.1 AVTALE MED OPPDRAGSGIVER	4
4.2 ARBEIDSSTED OG RESSURSER	4
4.3 GRUPPENORMER – SAMARBEIDSREGLER – HOLDNINGER	4
5 PROSJEKTBESKRIVELSE	4
5.1 PROBLEMSTILLING - MÅLSETTING - HENSIKT	4
5.2 KRAV TIL LØSNING ELLER PROSJEKTRESULTAT – SPESIFIKASJON	4
5.3 PLANLAGT FRAMGANGSMÅTE(R) FOR UTVIKLINGSARBEIDET – METODE(R)	4
5.4 INFORMASJONSINNSAMLING – UTFØRT OG PLANLAGT	5
5.5 VURDERING – ANALYSE AV RISIKO	5
5.6 HOVEDAKTIVITETER I VIDERE ARBEID	5
5.7 FRAMDRIFTSPLAN – STYRING AV PROSJEKTET	5
5.8 BESLUTNINGER – BESLUTNINGSPROSESS	6
6 DOKUMENTASJON	6
6.1 RAPPORTER OG TEKNISKE DOKUMENTER	6
7 PLANLAGTE MØTER OG RAPPORTER.....	6
7.1 MØTER.....	6
7.2 PERIODISKE RAPPORTER	6
8 PLANLAGT AVVIKSBEHANDLING	6
9 UTSTYRSBEHOV/FORUTSETNINGER FOR GJENNOMFØRING.....	7
10 REFERANSER	7
VEDLEGG	7

1 INNLEDNING

Gruppen har valgt oppgaven Modell asynkronmaskin. Formålet med denne oppgaven er å utvikle en enkel modell av en asynkronmaskin. Denne modellen kan brukes til simulering av fremdriftssystem om bord et fartøy i prosjekteringsfasen, der en ikke har nødvendig maskinvare til å teste styringssystemet. Oppgaven er gitt av STADT AS.

2 BEGREPER

- **Asynkronmotor** er en kortslutningsmotor som drives rundt av et roterende magnetisk felt, men hvor rotoren ved belastning får en langsommere hastighet enn det roterende feltet (sakking). Det er derfor motoren kalles *asynkron*. [1]

3 PROSJEKTORGANISASJON

3.1 Prosjektgruppe

Studentnummer(e)
476117 – Tomas Skaar Vadset (Sekretær)
476119 – Håvard Lervik
997529 – Singharat Boonkrong (Prosjekt leder)
247309 – Ola Edvard Nordvik

Tabell: Studentnummer(e) for alle i gruppen som leverer oppgaven for bedømmelse i faget IE 303612

3.1.1 Oppgaver for prosjektgruppen - organisering

Alle i gruppen har samme ansvar for at de ulike oppgavene i prosjektet blir gjort. Alle medlemmene skal også oppdatere prosjektlederen med fremdriften, og eventuelt avvik som oppstår.

3.1.2 Oppgaver for prosjektleder

Prosjektleder har ansvar for å kalle inn til møte med styringsgruppen og holde styringsgruppen oppdatert med fremdrift.

3.1.3 Oppgaver for sekretær

Sekretær har ansvar for å skrive referat fra møtet med styringsgruppen og skrive fremdriftsrapport.

3.1.4 Oppgaver for øvrige medlem(mer)

Bistå prosjektlederen og sekretæren ved behov.

3.2 Styringsgruppe (veileder og kontaktperson oppdragsgiver)

Styringsgruppen består av Eike M. Garbe fra NTNU i Ålesund, Hallvard Lidset Slettevoll og Rudi M. Slettevoll fra STADT AS.

4 AVTALER

4.1 Avtale med oppdragsgiver

Oppgaven er å lage en modell for en asynkronmaskin. Modellen skal brukes av STADT som et utgangspunkt som videre kan utvikles til å implementeres i deres systemer. Ved hjelp av denne modellen skal STADT kunne hente ut informasjon om hvordan ulike driftsmodi av asynkronmaskin har innvirkning på strøm og spenning.

4.2 Arbeidssted og ressurser

Prosjektet vil bli gjennomført på NTNU i Ålesund. Her er det et rom reservert for bachelorgrupper hver uke, fra tirsdag til fredag. Det er også tilgang til en lab på Fagskolen i Ålesund der gruppen kan teste ut modellene som blir laget i mindre skala. Veilederen er tilgjengelig på campus i Ålesund og på mail innenfor arbeidstid. Oppdragsgiver fra STADT er tilgjengelig på mail og til møter når det er nødvendig. Før hvert møte med styringsgruppen vil det bli sendt en fremdriftsrapport minst 24 timer i forveien som inneholder agenda for møtet, fremdrift og problemer prosjektgruppen har oppdaget.

4.3 Gruppenormer – samarbeidsregler – holdninger

Gruppen skal møtes hver ukedag kl. 08.15 til 16.00 de dagene det er mulig. Noen uker vil dette bli vanskelig på grunn av obligatorisk oppmøte i andre fag. I dette tidsrommet skal alle medlemmene være til stede ved NTNU Ålesund. Dette vil gi god kommunikasjon og samarbeid i gruppen. Det vil også bli nødvendig å jobbe utenom disse tidene, men ved disse tidene skal alle møtes.

Alle i gruppen har rett til å si sin mening. Medlemmene i gruppen skal gjøre sitt beste for at prosjektet kommer i mål.

5 PROSJEKTBESKRIVELSE

5.1 *Problemstilling - målsetting - hensikt*

Problemstillingen kan deles i to deler. Den første delen består av å utlede den matematiske modellen for en asynkronmaskin. Den andre delen består av implementering av matematiske modellen i et program som visualiserer resultatene fra simuleringen.

Problemstilling:

- Den matematiske modellen av en asynkronmaskin kan være kompleks. Kan vi forenkle denne modellen?
- Kan vi implementere dette i et åpen-kilde program som kunden kan bruke videre?

Effektmål:

Produsere en modell som kan senere brukes til videreutvikling, eller eventuelt tilbys som tjeneste av oppdragsgiver til deres kunder.

Resultatmål:

Formålet er å komme fram til en enkel modell av en asynkronmaskin som er fleksibel og kan implementeres i forskjellige driftssystemer. Modellen skal være implementert i et program med relativt billig lisens.

Prosessmål:

Prosessmålet er å bruke kunnskapen som gruppen har blitt tilegnet i løpet av studietiden og kunnskapen som har blitt hentet fra litteratur til å løse problemstillingen.

5.2 *Krav til løsning eller prosjektresultat – spesifikasjon*

Prosjektet anses som fullført når gruppen har en grov modell av asynkronmaskin. Modellen skal produsere verdier som samsvarer med virkeligheten, og skal være laget i et program som STADT har tilgang til.

5.3 *Planlagt framgangsmåte(r) for utviklingsarbeidet – metode(r)*

Det første gruppen ser på er hva er det bedriften ønsker fra prosjektet. Hvilke krav er det som skal følges? Deretter må det sjekkes om gruppen har den nødvendige informasjonen til å løse prosjektet. Hvilke aktiviteter som blir prioritert presiseres i Gantt-skjemaet. Det vil bli satt opp minst to medlemmer per aktivitet, der en har hovedansvar. Dersom aktiviteten ikke går som planlagt, skal hovedansvarlig informere prosjektleder. Deretter blir det kallet inn til et møte av prosjektleder der gruppen prøver å løse problemet. Om det oppstår problem som gruppen selv ikke klarer å løse, vil prosjektleder ta kontakt med styringsgruppen.

Noen aktiviteter kan ta lengre tid enn først antatt, derfor vil det bli gjort endringer i Gantt-skjemaet i løpet av prosjektet. Det kan også oppstå aktiviteter som ikke er i Gantt-skjemaet.

5.4 Informasjonsinnsamling – utført og planlagt

Alle gruppe medlemmene skal anskaffe relevant litteratur under forprosjekt perioden. Dette innebærer blant annet generelt informasjon om asynkronmaskin og eksisterende matematisk modell av maskinen.

Det er også nødvendig å undersøke de forskjellige metodene som brukes til å starte en asynkronmaskin og tilhørende påvirkning på kraftnettet. Disse metodene kan være for eks. Direktestart, Stjerne-Trekant kobling og Softstarters.

I tillegg til punktene over, skal gruppen også finne en løsning for å implementere matematisk modellen i et simuleringsprogram som MATLAB og Simulink.

5.5 Vurdering – analyse av risiko

Det er på nåværende tidspunkt ikke avklart detaljer i hvor presis denne modellen skal være, annet enn at det skal være en grov tilnærming. Det er her trolig en del valg som må tas underveis i prosessen. Spesielt gjelder dette for modellens oppbygging og hvilke parametere i asynkronmaskinen som kan estimeres, eller sees bort ifra for å oppnå et ønsket resultat.

Muligheten til å realisere denne modellen av asynkronmaskin i et annet program enn MATLAB er heller ikke utredet på nåværende tidspunkt. Det kan derfor oppstå en del uventede problemer her og det er vanskelig å vite hvor lang tid som kreves for å sette seg inn i, og å få implementert modellen i dette programmet.

Suksessfaktorer:

- God kommunikasjon innad i gruppen og med veiledere.
- Strukturert arbeid med klare mål og god arbeidsfordeling.
- Beregne tiden godt

Trusler:

- Gjøre den teoretiske modellen for vanskelig.
- Dårlig kommunikasjon mellom styringsgruppe og prosjektgruppe.
- Dårlig kommunikasjon innad i prosjektgruppen.
- Feilberegning av tid og arbeidsmengde

5.6 Hovedaktiviteter i videre arbeid

A. Litteratursøk

- Teori bak asynkronmaskin
- Hva finnes av slike verktøy?
- Eksisterende matematisk modell?

B. Lage Modell

- Utrede kontrollsystem for modellen
- Implementering av matematisk modell i systemet
- Teste/sammenligne med resultat fra Designbase

C. Lab oppsett

- Koble opp et system med asynkronmaskin
- Kjøre systemet og lagre måleresultatene
- Teste/sammenligne med resultatene fra modellen

D. Jobbe videre med annen software

- Teste annen software

E. Rapportskriving

- Finpuss av rapport
- Forprosjektsrapport

F. Lage presentasjon

5.7 Framdriftsplan – styring av prosjektet

5.7.1 Hovedplan

Hovedplanen er lagt ved i et Gantt-skjema som er laget i Microsoft Excel. Dette diagrammet er delt opp i hovedaktiviteter og delaktiviteter der det er satt opp estimert startdato og sluttdato. Det er også presisert faktiske start- og sluttdatoer, slik at gruppen kan se hvor krevende de ulike delene av prosjektet egentlig er. Det blir sett på som en milepæl når en hovedaktivitet er fullført. Skjemaet inneholder også hvilke gruppedlemmer som skal jobbe med de ulike delene og hvem som har hovedansvar.

5.7.2 Styringshjelpemidler

Bruker Gantt-skjema laget i Microsoft Excel

5.7.3 Utviklingshjelpemidler

Til dette prosjektet vil gruppen blant annet bruke MATLAB og Simulink til å lage en simulering av en asynkronmotor.

Gruppen vil også benytte asynkronmaskin og annet utstyr som er tilgjengelig på labben ved fagskolen i Ålesund.

5.7.4 Intern kontroll – evaluering

Gruppen vil møtes flere ganger i uken på skolen, der det vil bli gjort oppfølging av fremdriften til de ulike oppgavene.

5.8 Beslutninger – beslutningsprosess

Avgjørelser relatert til definering og avgrensing av oppgaven har blitt tatt i plenum. Gruppedeltakerne har hatt lik påvirkningskraft på avgjørelser uavhengig av ansvarsfordeling.

Ved problem som påvirker utfallet av hovedprosjektet, vil styringsgruppen involveres før prosjektgruppen tar en avgjørende beslutning.

6 DOKUMENTASJON

6.1 Rapporter og tekniske dokumenter

Planlagte rapporter:

- Møtereferat
- Fremdriftsrapport
- Forprosjektrapport
- Testprotokoll/rapport fra lab

Alle rapporter og tekniske dokumenter vil lagres i skyløsning som alle gruppedeltakere har tilgang til.

7 PLANLAGTE MØTER OG RAPPORTER

7.1 Møter

Interne prosjektmøter anses som nødvendig, siden gruppen skal samarbeide med hverandre gjennom hele bachelorperioden.

Møter med styringsgruppen vil bli tatt når det er nødvendig. Siden STADT AS ligger i Gjerdsвика vil det bli mye reising dersom man skal ta møter veldig ofte. Møter vil derfor bli avtalt etter hvert som det er nødvendig. Det vil også være mulig å bruke program som Skype til møter.

7.2 Periodiske rapporter

7.2.1 Framdriftsrapporter (inkl. milepæl)

Det blir laget en fremdriftsrapport hver andre uke. I tillegg til denne rapporten vil det bli lagt ved en møteagenda i forkant av møtet med styringsgruppe, som inneholder plan for møtet. Rapport og møteagenda skal leveres til styringsgruppen minst 24 timer før et møte.

8 PLANLAGT AVVIKSBEHANDLING

Ved større avvik i planen skal prosjektlederen varsles. Medlemmene blir deretter innkallet til et internt gruppemøte for å løse problemet som har oppstått. Dersom problemet ikke er løst etter møtet med prosjektgruppen, skal styringsgruppen informeres, slik at de kan komme med innspill og eventuelle avgjørelser.

9 UTSTYRSBEHOV/FORUTSETNINGER FOR GJENNOMFØRINGS

Prosjektet består i hovedsak av matematisk utledning og implementering av matematiske modellen i et program. Dette gjør at utstyrsbehovet for dette prosjektet er minimalt. Gruppen trenger likevel noen programvarer/lisens og tilgang til eksisterende elkraft-lab ved fagskolen i Ålesund.

- MATLAB & Simulink
- Designbase 6.0
- Testlab
- Cassy Lab 2

Ved ytterligere behov, vil dette bli tatt opp på et møte med styringsgruppen.

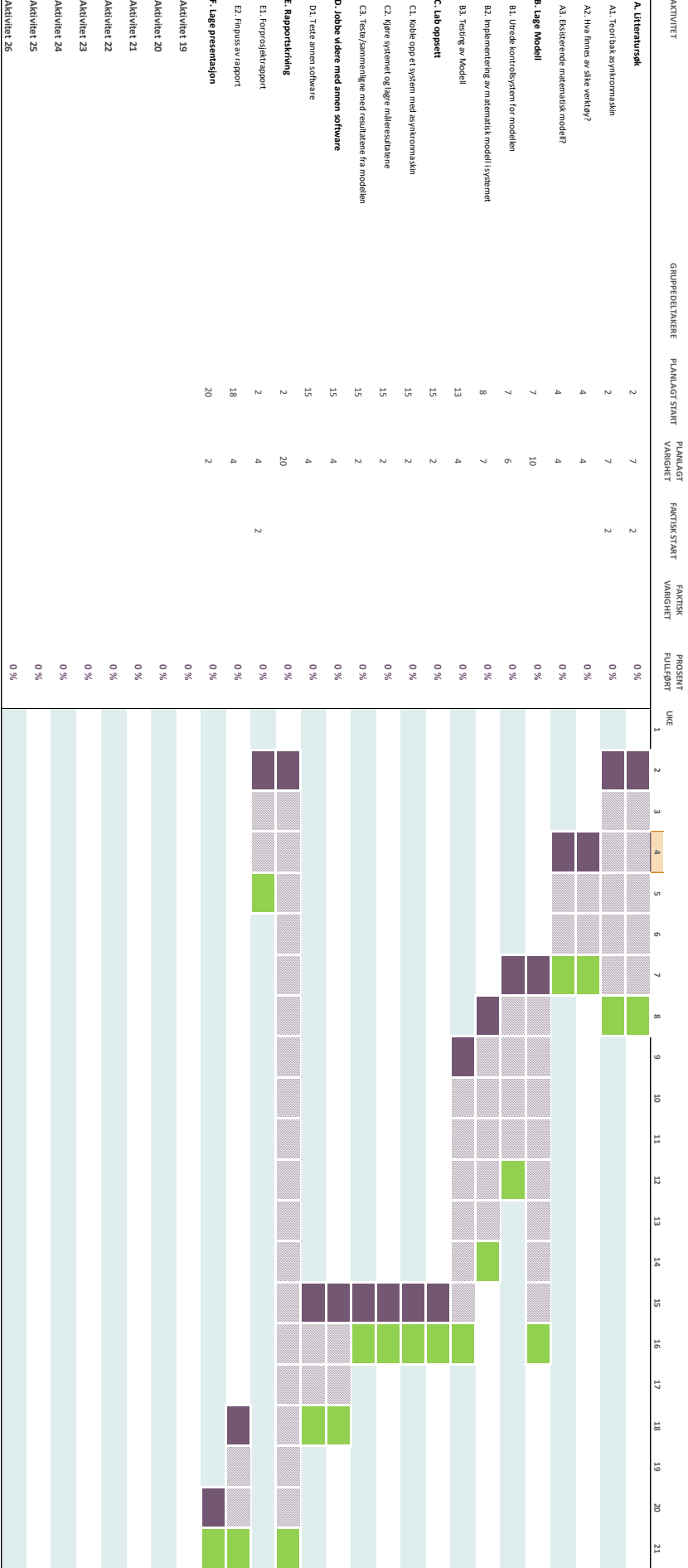
10 REFERANSER

- [1] Rosvold, Knut A. (2018, 29. oktober). Asynkronmotor. I Store norske leksikon. Hentet 17. januar 2019 fra <https://snl.no/asynkronmotor>.

B Gantt diagram

Prosjektplanlegging

Veg i en periode som skal utveies, til høyre. En forklaring beskriver følgende diagram.



C Fremdriftsrapporter

IB303312	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Bacheloroppgave			STADT AS	1 av 2
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	Prosjektgruppe (navn)	Dato
Framdriftsrapport				

Hovedhensikt / fokus for arbeidet i denne perioden

Første perioden handlet om å komme i gang med prosjektet. Perioden startet 08.01.2019. Gruppen skulle finne den nødvendige informasjonen til å komme i gang med prosjektet. Forprosjektrapporten skulle også skrives i denne perioden

Planlagte aktiviteter i denne perioden

1. Gjøre ferdig forprosjektrapport
2. Finne informasjon om asynkronmaskin
3. Finne informasjon om lignende verktøy som allerede finnes
4. Starte på modell i MATLAB

Virkelig gjennomførte aktiviteter i denne perioden

1. Fullført forprosjektrapport
2. Funnet litt informasjon om asynkronmaskin
3. Funnet litt om lignende verktøy som allerede finnes
4. Startet på modell i MATLAB

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

Beskrivelse av/begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

Gruppen har fått mer innsikt i hva dette prosjektet vil kreve. Det har også blitt mer klart hva oppdragsgiver ønsker

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil være å bygge videre på modellen i MATLAB SIMULINK som allerede er laget og se på andre muligheter.

Planlagte aktiviteter neste periode

1. Bygge videre på MATLAB modellen

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato

2. Skrive på rapporten

Annet

Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Bacheloroppgave			STADT AS	1 av 2
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	Prosjektgruppe (navn)	Dato
Framdriftsrapport				18.02.19

Hovedhensikt / fokus for arbeidet i denne perioden

Andre perioden handlet om å bygge videre på det som ble gjort i første periode. Skal også starte på rapporten med den informasjonen som ble funnet i den første perioden. Gruppen skal også fortsette å lete etter informasjon til de ulike komponentene og ulike verktøy som finnes.

Planlagte aktiviteter i denne perioden

1. Fortsette med Simulink modell i MATLAB
2. Fortsette med informasjonssanking om asynkronmaskin
3. Starte med rapport

Virkelig gjennomførte aktiviteter i denne perioden

1. Fortsette med Simulink modell i MATLAB
2. Funnet mer informasjon om asynkronmaskin
3. Startet på rapport
4. Startet å finne informasjon om DC motor
5. Statet med modell i skript i MATLAB

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

Etter møte med midlertidig veileder ble det anbefalt at gruppens startet med å lese seg opp på DC motor for å få lettere forståelse for asynkronmaskinen. Ble også anbefalt å lage modellen i skript siden dette er lettere å implementere i andre program og også billigere.

Beskrivelse av /begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

Gruppen har lært litt om DC motorer, lært mer om asynkronmaskin

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil være å få lage til et script for den matematiske modellen, og kanskje få testet denne.

Planlagte aktiviteter neste periode

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				18.02.19

1. Lage til script i MATLAB
2. Skrive på rapporten
3. Finne informasjon

Annet

Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fralogg)	STADT AS	1 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				04.03.19

Hovedhensikt / fokus for arbeidet i denne perioden

Tredje perioden handlet om gjøre ferdig modellen for en DC motor som ble startet på i andre periode og gå videre til AC. Ble også skrevet en del på rapporten med den informasjonen som har blitt funnet i løpet av perioden.

Planlagte aktiviteter i denne perioden

1. Gjøre ferdig DC modell i MATLAB
2. Fortsette med informasjonssanking om asynkronmaskin
3. Starte med AC modell i MATLAB
4. Skrive på rapport
5. Lære om DQ0 transformasjon

Virkelig gjennomførte aktiviteter i denne perioden

1. Gjorde ferdig DC modell i MATLAB
2. Funnet mer informasjon om asynkronmaskin
3. Fortsatt på rapport
4. Startet med AC modell i MATLAB
5. Lest om DQ0 transformasjon

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

Beskrivelse av /begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

I løpet av perioden har gruppen lært om oppsettet for en DC motor i MATLAB og fått et innblikk i hvordan det vil bli å sette dette opp for en AC motor.

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil være å få laget til et script for den matematiske modellen til en AC motor

Planlagte aktiviteter neste periode

1. Lage til script for AC motor i MATLAB

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				04.03.19

2. Skrive på rapporten
3. Finne informasjon

Annet

Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Bacheloroppgave			STADT AS	1 av 2
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	Prosjektgruppe (navn)	Dato
Framdriftsrapport				18.03.19

Hovedhensikt / fokus for arbeidet i denne perioden

Den fjerde perioden handlet om å fortsette med AC modellen som ble startet med i forrige periode. I løpet av denne perioden skulle det også bli gjort litt testing på lab for å se om det oppførte seg slik som forventet i virkeligheten.

Planlagte aktiviteter i denne perioden

1. Fortsette med AC modell i MATLAB
2. Fortsette med informasjonssanking om asynkronmaskin
3. Teste på lab
4. Skrive på rapport

Virkelig gjennomførte aktiviteter i denne perioden

1. Fortsatt med AC modell i MATLAB
2. Funnet mer informasjon om asynkronmaskin
3. Fortsatt på rapport
4. Undersøkt utstyr på lab
5. Startet på Simulink modell med hjelp av kun en pakke

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

Den opprinnelige planen var å kun lage modellen i script i MATLAB, men etter et møte med opprinnelige veilederen ble det enighet om å lage en modell i Simulink der man kan bruke Simulinkpakken. Dette er for å holde kostnadene lavest mulig. På grunn av manglende informasjon på asynkronmaskinene på labben ble det kun gjort undersøkelser på hvilke komponenter som blir nødvendig og hvordan oppkoblingen vil bli

Beskrivelse av /begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

I løpet av perioden har gruppen lært litt om hvordan man må gå frem for å sette opp en AC modell i MATLAB, men det er fortsatt usikkerhet rundt hvordan man bruker de verdiene man måler til å finne ut alle verdiene som er ønsket

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil fortsatt være å få laget til et script for den matematiske modellen til en AC motor

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				18.03.19

Planlagte aktiviteter neste periode
<ol style="list-style-type: none">1. Lage til script for AC motor i MATLAB2. Lage modell i Simulink med kun en pakke3. Skrive på rapporten4. Finne informasjon
Annet
Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Bacheloroppgave			STADT AS	1 av 2
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fralogg)	Prosjektgruppe (navn)	Dato
Framdriftsrapport				01.04.19

Hovedhensikt / fokus for arbeidet i denne perioden

Den femte perioden handlet om å bygge videre på de to AC modellene som ble startet med i forrige periode. I løpet av denne perioden skulle det også bli gjort litt testing på lab for å se om det oppførte seg slik som forventet i virkeligheten.

Planlagte aktiviteter i denne perioden

1. Fortsette med AC modell i MATLAB Simulink
2. Fortsette med AC modell i MATLAB skript.
3. Fortsette med informasjonssanking om asynkronmaskin
4. Teste på lab
5. Skrive på rapport

Virkelig gjennomførte aktiviteter i denne perioden

1. Fortsatt med AC modell i MATLAB simulink
2. Fortsatt med AC modell i MATLAB skript
3. Funnet mer informasjon om asynkronmaskin
4. Fortsatt på rapport
5. Undersøkt utstyr på lab
6. Testet på labb

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

Testing på lab har ikke gått som planlagt på grunn av manglende verdi på motorer, ikke tilgang til personer med den nødvendige informasjonen. Det ble forklart en del av opprinnelig veileder over e-post, men har blitt mer usikkerhet etter at opprinnelig veileder ikke lenger er tilgjengelig.

Beskrivelse av/begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

Gruppen har lært litt om hvordan man går frem på testing av lab. Har også funnet mer ut om hvordan man kan bruke de målte verdiene til å regne ut verdier som ikke er mulig å måle.

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil være å finne ut om modellene som er laget er realistiske i forhold til virkeligheten. Det blir også fokus på rapportskrivning.

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				01.04.19

Planlagte aktiviteter neste periode
<ol style="list-style-type: none">1. Fortsette med MATLAB modeller2. Finne andre modeller som kan brukes til sammenligning3. Skrive på rapporten4. Finne informasjon
Annet
Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	1 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				15.04.19

Hovedhensikt / fokus for arbeidet i denne perioden

Denne perioden handlet bygge videre på modellene til asynkronmotoren som har blitt laget. Det har også vært fokus på å informere ny veileder om prosjektet og hva som har blitt gjort. Som i alle andre perioder er det også fokus på rapportskriving og informasjonssanking. Det ble også gjort tester på lab

Planlagte aktiviteter i denne perioden

1. Fortsette med modeller i MATLAB og Simulink.
2. Fortsette med informasjonssanking om asynkronmaskin
3. Teste på lab
4. Skrive på rapport

Virkelig gjennomførte aktiviteter i denne perioden

1. Fortsatt med modeller i MATLAB og Simulink
2. Funnet mer informasjon om asynkronmaskin
3. Fortsatt på rapport
4. Testet på lab

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

Beskrivelse av /begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

Gruppen har nå mer kunnskap om modellene og hvordan de ulike kurvene ser ut. Det er også mer kunnskap om hvordan sluttproduktet skal se ut.

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil være å gjøre modellen i Simulink mer fleksibel slik at det lett kan simuleres ulike scenarioer.

Planlagte aktiviteter neste periode

1. Gjøre modell i Simulink mer fleksibel
2. Teste modellen i ulike scenarioer
3. Skrive på rapporten
4. Finne informasjon

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				15.04.19

Annet
Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	1 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				29.04.19

Hovedhensikt / fokus for arbeidet i denne perioden

Denne perioden handlet om å gjøre modellen mer brukervennlig. Det skal bli lettere å legge til å endre på komponenter. Det vil som i de andre periodene også bli fokus på rapportskrivning og informasjonssanking. Muligheten for å lage til generator og tap i kabelen.

Planlagte aktiviteter i denne perioden

1. Gjøre modellen i Simulink mer brukervennlig
2. Fortsette med informasjonssanking
3. Se på generator og tap i kabel
4. Skrive på rapport

Virkelig gjennomførte aktiviteter i denne perioden

1. Gjort modellen mer brukervennlig
2. Funnet mer informasjon
3. Fortsatt på rapport
4. Sett på generator og tap i kabel

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

I møtet med veileder og oppdragsgiver ble det snakket om muligheten for å lage til en modell for generatoren siden gruppen så langt har brukt sterkt nett. Gruppen fant ut at dette ikke var mulig med den tiden og den planen som har blitt lagt på hvordan modellen for asynkronmaskinen skal løses.

Beskrivelse av/begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

I løpet av perioden har gruppen lært litt om hvordan man gjør modeller i MATLAB mer brukervennlig, ved hjelp av masker. Det har også blitt erfart at det blir for mye arbeid i forhold til tiden som er igjen å prøve å lage til en modell for generator og tap i kablene

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil bli å ferdigstille den modellen gruppen har bestemt seg for å lage. Det vil også bli fokus på skriving av rapport.

Planlagte aktiviteter neste periode

1. Ferdigstille modellen i Simulink

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				29.04.19

2. Skrive rapport

Annet

Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	1 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				13.05.19

Hovedhensikt / fokus for arbeidet i denne perioden

Denne perioden handlet om å teste de ulike casene og ferdigstille modellen. Perioden handlet også om å skrive videre på rapporten

Planlagte aktiviteter i denne perioden

1. Ferdigstille modell
2. Teste caser i Simulink
3. Skrive rapport

Virkelig gjennomførte aktiviteter i denne perioden

1. Ferdigstilt modeller
2. Testet caser i Simulink
3. Teste caser i Simscape
4. Skrevet på rapport

Beskrivelse av/begrunnelse for eventuelle avvik mellom planlagte og virkelige aktiviteter

Det ble bestemt at det skal brukes en modell i Simscape til å sammenligne og teste noen av casene for å sjekke hvor realistisk modellen er

Beskrivelse av /begrunnelse for endringer som nå ønskes i selve prosjektets innhold eller i den videre framgangsmåten - eller framdriftsplanen

Erfaring fra denne perioden

Gruppen har erfart at modellen i Simulink ikke nødvendigvis klarer å simulere alt som gruppen håpte på. Det blir derfor også brukt modell i Simscape for å gjøre ulike tester og bruke til sammenligning

Hovedhensikt/fokus neste periode

Hovedfokus for neste periode vil være å gjøre ferdig rapport

Planlagte aktiviteter neste periode

1. Ferdigstille rapport

Annet

Ønske om /behov for veiledning, tema i undervisningen – drøfting ellers

IB303312 Bacheloroppgave	Prosjekt	Antall møter denne periode 1).	Firma - Oppdragsgiver	Side
Rapport fra prosess	Periode/uke(r)	Antall timer denne per. (fra logg)	STADT AS	2 av 2
Framdriftsrapport			Prosjektgruppe (navn)	Dato
				13.05.19

D Møtereferater

Møtereferat

Dato: 16.01.2019

Sted: Gjerdsvika

Deltakere: Eike Matthias Garbe, Rudi M. Slettevoll, Hallvard Lidset Slettevoll, Tomas Skaar Vadset, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Oppstartsmøte

Onsdag 16.01.2019 besøkte gruppen og veileder STADT AS i Gjerdsvika. Vi fikk en rask omvisning før det ble et møte med Hallvard Lidset Slettevoll og Rudi M. Slettevoll. Under møtet ble det gjort mer klart hva STADT ønsket fra dette prosjektet. De ønsker en grovmodell av en asynkronmotor som skal se når det passer seg best å koble om mellom det forskjellige driftmodi. Gruppen skal se på hvilke slike verktøy som finnes og se om det er noe som kan brukes eller lage et eget verktøy ved hjelp av ulike simuleringsverktøy. Dette møtet var veldig viktig med tanke på hvor gruppen skal starte med prosjektet. Før dette møtet var ikke gruppen helt sikker på hva som var forventet fra prosjektet og det ble brukt mye tid på bare generell informasjonsheving.

Møtereferat

Dato: 13.02.2019

Sted: NTNU Ålesund

Deltakere: Ibrahim A. Hameed, Tomas Skaar Vadset, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Informere midlertidig veileder

På grunn av uforutsette hendelser har gruppen fått Ibrahim A. Hameed som midlertidig veileder på til Eike M. Garbe er tilbake. Formålet med dette møte var da å informere midlertidig veileder om oppgaven og hva gruppen har fokusert på så langt. Det ble diskutert at gruppen syntes at det var vanskelig å finne et sted å starte med den kompliserte matematiske modellen. Midlertidig veileder foreslå da at gruppen skulle starte med å forstå en matematisk modell for DC-motor. Denne modellen er en del lettere men en del av prinsippene vil være det samme som for AC. Det ble også enighet om at gruppen burde fokusere mer på å lage et MATLAB script, enn en simulinkmodell.

Møtereferat

Dato: 22.02.2019

Sted: NTNU Ålesund

Deltakere: Ibrahim A. Hameed, Rachid Oucheikh, Tomas Skaar Vadset, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Starte med AC-motor

Gruppen har brukt tiden siden forrige møte for å se på hvordan man legger inn en modell for en DC motor i MATLAB. DC motor er mindre komplisert, men det er samme prinsippene når man legger inn i MATLAB som for en AC motor. Hovedfokuset på møtet var da hvordan gruppen kunne bygge videre på det som har blitt gjort i for DC og gjøre noe lignende for AC. Midlertidig Veileder tok kontakt med Rachid Oucheikh som har god kontroll på elektriske maskiner. Han skal være med å veilede fremover.

Møtereferat

Dato: 13.03.2019

Sted: NTNU Ålesund (Skype-møte)

Deltakere: Eike Matthias Garbe, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Avklaring om bruk av lab og tilpassing av asynkronmodell.

Gruppen har siden sist møte hatt fokus på å prøve å forstå asynkronmodeller som vi har funnet på nett. Det har blitt unngått å bruke Simulink-modeller siden dette er en relativt dyr MATLAB lisens. Etter uker med undersøkning uten særlig fremgang i script format, har gruppen nå ønske om å lage en modell i Simulink. Modellen skal inneholde selvbygde komponenter i stedet for tilleggspakker for Simulink, dette er for å holde kostnader nede.

Det ble også tatt opp tilgjengeligheten for bruk av elkraft lab på fagskolebygget. Dette handlet mest om hvem gruppen skulle kontakte for tilgang, og om anskaffelse av mer detaljerte spesifikasjoner på asynkronmotorer på labben.

Møtereferat

Dato: 05.04.2019

Sted: Skype

Deltakere: Robert Nilssen, Tomas Skaar Vadset, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Informere Robert Nilssen om oppgaven og hva gruppen har gjort så langt.

Gruppen får hjelp fra Robert Nilssen som er professor i elektriske maskiner ved NTNU i Trondheim. Møtet gikk da ut på å informere om hva oppgaven gikk ut på og hva gruppen har fått gjort så langt. Det ble der lagt en plan for hva gruppen skulle se på videre. Frem til neste møte skal gruppen se på PSS Sincal og lignende program som kan brukes til å simulere nettet på båten. Gruppen skal også eksperimentere litt mer med modellene som ble laget i MATLAB og Simulink og se om det er mulig å simulere ulike situasjoner. Det ble avtalt et nytt møte fredag 12.04 der forhåpentligvis oppdragsgiver også vil delta. Planen for møtet er at Robert Nilssen får høre Stadt AS sin synsvinkel på oppgaven. Det skal også diskuteres litt hvilke komponenter som blir brukt og hva planen videre skal være.

Møtereferat

Dato: 12.04.2019

Sted: Skype

Deltakere: Robert Nilssen, Rudi M. Slettevoll, Tomas Skaar Vadset, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Legge en plan for hva oppdragsgiver vil at gruppen skal se på videre

I møtet har gruppen sammen med veileder og oppdragsgiver lagt en plan på hva gruppen skal se på videre og hva oppdragsgiver ønsker fra sluttproduktet. Det ble bestemt at fokuset fremover skal være å bygge videre på modellen i Simulink og gjøre det lett å legge til å fjerne komponenter. Stadt AS ønsker at det skal være enkelt å bruke denne modellen på flere ulike typer motorer og scenarioer. Modellen har så langt brukt det som heter stivt nett, som betyr at det bare blir gitt en fast spenning hele tiden. På et skip er det svakt nett, som betyr at spenningen ikke vil være konstant. Gruppen skal legge inn en generator som skal representere et svakt nett. Videre skal det også testes med flere generatorer og motorer med ulike parameterer. Fremover skal gruppen teste ulike scenarioer for å se hvordan de ulike verdiene reagerer.

Møtereferat

Dato: 29.04.2019

Sted: Skype

Deltakere: Robert Nilssen, Rudi M. Slettvoll, Hallvard Slettvoll, Tomas Skaar Vadset, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Diskutere hvilke eksempler som skal bli sett på i modellen

I møtet ble det snakket om hva gruppen bør gjøre videre med modellen. Gruppen har gjort modellen mer brukervennlig siden forrige møte, og viste dette frem til oppdragsgiver. Det nærmer seg slutt på prosjektet. Derfor er det viktig at det blir enighet med oppdragsgiver hvilke scenarioer som er ønsket at gruppen skal se på og skrive om. Det var usikkerhet i gruppen om hva som var relevant å ha med i rapporten. Det skal derfor lages en disposisjon der gruppen skriver ned det de tenker er relevant for oppgaven. Denne disposisjonen skal levers til veileder til neste møte som er fredag 03.05.2019.

Møtereferat

Dato: 03.05.2019

Sted: Skype

Deltakere: Robert Nilssen, Rudi M. Slettvoll, Tomas Skaar Vadset, Ola Edvard Nordvik, Håvard Lervik og Singharat Boonkrong.

Mål med møte: Diskutere disposisjon til rapport og hva rapporten burde inneholde

I forkant av møtet hadde gruppen laget en disposisjon med det som de tenkte burde være med i den ferdige rapporten. Det var i løpet av møtet tatt opp hvilke «caser» som gruppen kan og burde se på. Casene det var snakk om var å koble inn ulike motortyper med og uten last, se på innkoblingstid og også se om det er mulig å se på kortslutning.

E Kildekode

MATLAB

```
%% Induction motor odesolver

close all; clear all;
tspan=[0 30]; %simulation time
x0=[0; 0; 0; 0; 0]; % Initial conditions for the statevariables

[t, x]=ode45(@IModefunction,tspan,x0);

%Torque equation
Lm=0.9750/(2*pi*60);
P=4;
T=P*Lm/3*(x(:,3).*x(:,2)-x(:,4).*x(:,1));

%dq-stator current absolute value
Idqs_abs= 2/3*(sqrt(x(:,1).^2+x(:,2).^2));

% plots
figure
plot(t,Idqs_abs);
title('dq-stator abs current')
xlabel('time')
ylabel('Ampere')

figure
subplot(5,1,1)
plot(t,x(:,1));
title('Sd-Current')
subplot(5,1,2)
plot(t,x(:,2));
title('Sq-Current')
subplot(5,1,3)
plot(t,x(:,3));
title('Rd-Current')
subplot(5,1,4)
plot(t,x(:,4));
title('Rq-Current')
%subplot(5,1,5)
figure
plot(t,x(:,5));
title('Angular velocity')

figure
plot(t,T);
title('Torque vs time')
xlabel('time');
ylabel('torque');

figure
plot(x(:,5),T);
```

```
title('Torque speed curve')  
xlabel('angular velocity')  
ylabel('Torque');
```

InductionMotorODEfunction

```
function dxdt=IModefunction(t,x)
%motor parameters
Xls=0.0320; %leakage reactance stator
Xlr=0.0100; %leakage reactance rotor
Xm=0.9750; %mutual reactance

Xr=Xlr+Xm; %Self reactance rotor
Xs=Xls+Xm; %Self reactance stator

Lr=Xr/(2*pi*60); %Self inductance rotor
Ls=Xs/(2*pi*60); %Self inductance stator
Lm=Xm/(2*pi*60); %Mutual inductance
Rs=0.0041; %Resistance stator
Rr=0.0009; %Resistance rotor
P=4; %number of poles
Jm=152; %motor moment of inertia
Jl=0; %load moment of inertia
Tl=0; %load torque
cf=0.85; % friction coefficient
V=sqrt(2)*690/sqrt(3); %Peak value phase voltage
J= Jl+Jm; %

%matrices for state equation

A_inv=[Ls 0 Lm 0 0;
        0 Ls 0 Lm 0;
        Lm 0 Lr 0 0;
        0 Lm 0 Lr 0;
        0 0 0 0 J];

A=inv(A_inv);

B=[Rs 0 0 0 0;
    0 Rs 0 0 0;
    0 (P*Lm*x(5))/2 Rr (P*x(5)*Lr)/2 0;
    -(P*x(5)*Lm)/2 0 -(P*x(5)*Lr)/2 Rr 0;
    0 0 -(P*Lm*x(2))/3 (P*Lm*x(1))/3 cf ];

%input setup
Vas =V* sin(2*pi*60*t);
Vbs =V* sin(2*pi*60*t - 2/3*pi);
Vcs =V* sin(2*pi*60*t + 2/3*pi);
%transform dq stationary(Clark transform)
Vds= [1 -1/2 -1/2]*[Vas;Vbs;Vcs];
Vqs= [0 sqrt(3)/2 -sqrt(3)/2]*[Vas;Vbs;Vcs];

%input Vector
u=[Vds;Vqs;0;0; -Tl];

%state vector (or use x(1:5) in the state equation)
x=[x(1);x(2);x(3);x(4);x(5)];
```

```
%state equation
dxdt=A*(u-B*x);

end
```

Simulink

Induction_motor

Design Description

Induction_motor: Design Description

by

Published 18-May-2019 18:00:06

Copyright © 2019

Table of Contents

Chapter 1. Model Version.....	1
Chapter 2. Root System.....	2
Blocks.....	2
Parameters.....	2
Block Execution Order.....	7
Chapter 3. Subsystems.....	9
3-phase sine wave.....	9
Blocks.....	9
Clarke Transformation.....	12
Blocks.....	12
Electrical sub-model.....	15
Blocks.....	15
Matrix 2.....	23
Blocks.....	23
Mechanical sub-model.....	26
Blocks.....	26
Motor, High speed.....	31
Blocks.....	32
Torque sub-model.....	39
Blocks.....	39
Chapter 4. System Design Variables.....	44
Design Variable Summary.....	44
Chapter 5. Requirements.....	45
Chapter 6. System Model Configuration.....	46
Chapter 7. Glossary.....	77
Chapter 8. About this Report.....	78
Report Overview.....	78
Root System Description.....	78
Subsystem Descriptions.....	79
State Chart Descriptions.....	79

Chapter 1. Model Version

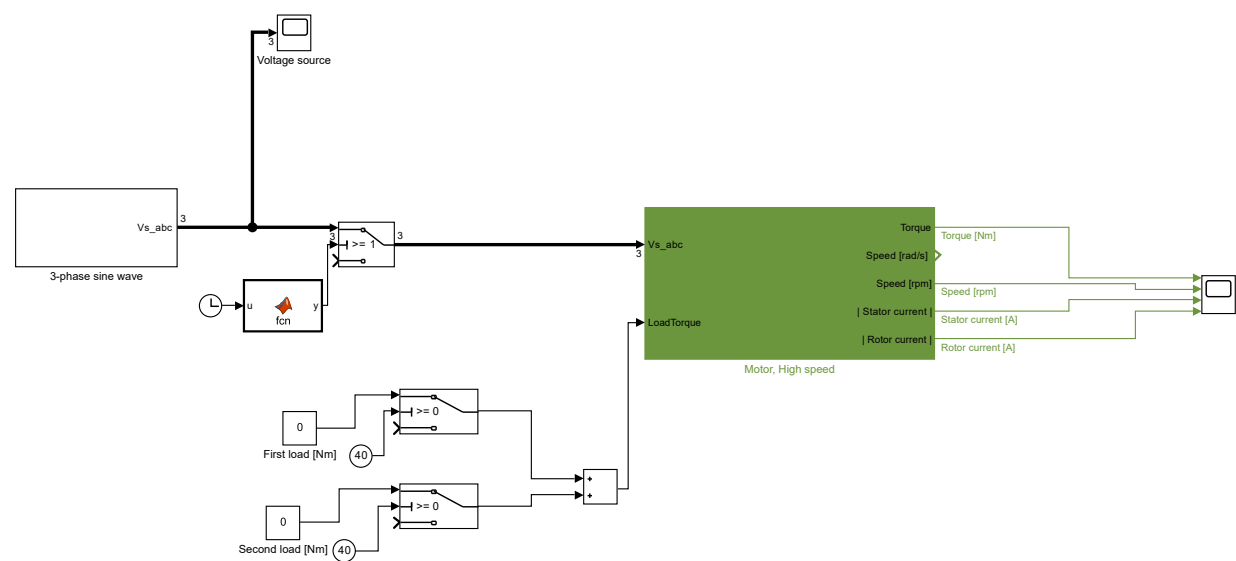
Version: 1.455

Last modified: Sat May 18 17:59:34 2019

Checksum: 2960570121 3956389500 3487773027 110457026

Chapter 2. Root System

Figure 2.1. Induction_motor



Blocks

Parameters

"3-phase sine wave " (SubSystem)

Table 2.1. "3-phase sine wave " Parameters

Parameter	Value
Voltage, line-line, rms	690
Frequency	60

"Add" (Sum)

Table 2.2. "Add" Parameters

Parameter	Value
Icon shape	rectangular
List of signs	++

Parameter	Value
Sum over	All dimensions
Dimension	1
Require all inputs to have the same data type	off
Accumulator data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock data type settings against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Clock" (Clock)

Table 2.3. "Clock" Parameters

Parameter	Value
Display time	on
Decimation	60

"Clock1" (Clock)

Table 2.4. "Clock1" Parameters

Parameter	Value
Display time	off
Decimation	10

"Clock2" (Clock)

Table 2.5. "Clock2" Parameters

Parameter	Value
Display time	on
Decimation	60

"First load [Nm]" (Constant)

Table 2.6. "First load [Nm]" Parameters

Parameter	Value
Constant value	0
Interpret vector parameters as 1-D	on
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit from 'Constant value'
Lock output data type setting against changes by the fixed-point tools	off
Sample time	inf
Frame period	inf

"MATLAB Function" (MATLAB Function)

Table 2.7. MATLAB Function Function Properties

Property	Value
Update Method	INHERITED
Sample Time	
Support variable-size arrays	1
Saturate on integer overflow	1
Treat these inherited Simulink signal types as fixed-point objects	Fixed-point
MATLAB Function block fimath	Same as MATLAB Default
Input fixed-point math	fimath(...)
Description	

Table 2.8. MATLAB Function Argument Summary

Name	Scope	Port	Data Type	Size
u	Input	1	double	1

Name	Scope	Port	Data Type	Size
y	Output	1	double	1

MATLAB Function Function Script

```
function y = fcn(u)
if (u > 14) && (u < 16)
    y = 0;
elseif (u > 19) && (u < 30)
    y = 0;
else
    y = 1;
end
```

"Motor, High speed " (SubSystem)

Table 2.9. "Motor, High speed " Parameters

Parameter	Value
Stator Reactance, Xs	0.0320
Rotor Reactance, Xr	0.0100
Mutual Reactance, Xm	0.9750
Rotor Inertia, Jm	152
Load Inertia, Jl	0
Stator Resistance, Rs	0.0041
Rotor Resistance, Rr	0.0009
Poles, P	4
Friction Coefficient, Cf	5
Frequency, f	60

"Second load [Nm]" (Constant)

Table 2.10. "Second load [Nm]" Parameters

Parameter	Value
Constant value	0
Interpret vector parameters as 1-D	on
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit from 'Constant value'

Parameter	Value
Lock output data type setting against changes by the fixed-point tools	off
Sample time	inf
Frame period	inf

"Switch" (Switch)

Table 2.11. "Switch" Parameters

Parameter	Value
Criteria for passing first input	$u_2 \geq \text{Threshold}$
Threshold	0
Require all data port inputs to have the same data type	off
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Enable zero-crossing detection	on
Sample time (-1 for inherited)	-1
Allow different data input sizes (Results in variable-size output signal)	off

"Switch1" (Switch)

Table 2.12. "Switch1" Parameters

Parameter	Value
Criteria for passing first input	$u_2 \geq \text{Threshold}$
Threshold	1
Require all data port inputs to have the same data type	off
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule

Parameter	Value
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Enable zero-crossing detection	on
Sample time (-1 for inherited)	-1
Allow different data input sizes (Results in variable-size output signal)	off

"Switch2" (Switch)

Table 2.13. "Switch2" Parameters

Parameter	Value
Criteria for passing first input	u2 >= Threshold
Threshold	0
Require all data port inputs to have the same data type	off
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Enable zero-crossing detection	on
Sample time (-1 for inherited)	-1
Allow different data input sizes (Results in variable-size output signal)	off

Block Execution Order

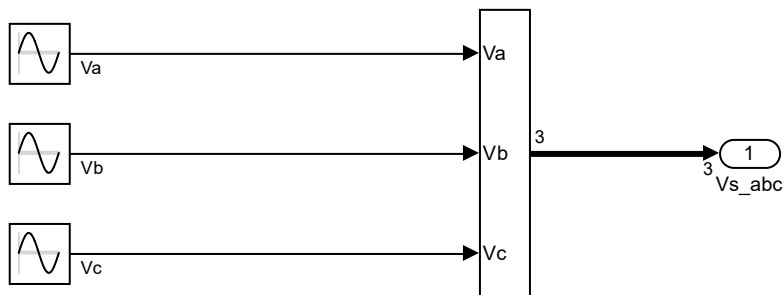
1. [Integrator](#) (Integrator)
2. [Product](#) (Product)
3. [Product1](#) (Product)
4. [Subtract](#) (Sum)
5. [Gain](#) (Gain)
6. [Mutual inductance Lm](#) (Gain)
7. [Integrator](#) (Integrator)
8. [Integrator 1](#) (Integrator)

9. [Add](#) (Sum)
10. [conversion to rpm](#) (Gain)
11. [Stator current equation](#) (Fcn)
12. [Gain1](#) (Gain)
13. [Rotor current equation](#) (Fcn)
14. [Gain3](#) (Gain)
15. [Scope](#) (Scope)
16. [Sine Wave Function](#) (Sin)
17. [Sine Wave Function1](#) (Sin)
18. [Sine Wave Function2](#) (Sin)
19. [Voltage source](#) (Scope)
20. [First load \[Nm\]](#) (Constant)
21. [Clock](#) (Clock)
22. [Switch](#) (Switch)
23. [Second load \[Nm\]](#) (Constant)
24. [Clock2](#) (Clock)
25. [Switch2](#) (Switch)
26. [Add](#) (Sum)
27. [Clock1](#) (Clock)
28. [MATLAB Function](#)
 1. [SFunction](#) (S-Function)
29. [Switch1](#) (Switch)
30. [Gain1](#) (Gain)
31. [Gain](#) (Gain)
32. [Scope](#) (Scope)
33. [Gain2](#) (Gain)
34. [ids, iqs, idr, iqr](#) (Scope)
35. [Constant1](#) (Constant)
36. [Constant2](#) (Constant)
37. [Matrix 1](#) (Constant)
38. [Inverse fcn](#)
 1. [SFunction](#) (S-Function)
39. [Gain](#) (Gain)
40. [Fcn](#) (Fcn)
41. [Fcn1](#) (Fcn)
42. [Fcn2](#) (Fcn)
43. [Fcn3](#) (Fcn)
44. [Subtract](#) (Sum)
45. [Product](#) (Product)
46. [Friction Coefficient](#) (Gain)
47. [Subtract](#) (Sum)
48. [Total inertia⁻¹](#) (Gain)
49. [Total inertia⁻²](#) (Gain)

Chapter 3. Subsystems

3-phase sine wave

Figure 3.1. Induction_motor/3-phase sine wave



Blocks

Parameters

"Mux1" (Mux)

Table 3.1. "Mux1" Parameters

Parameter	Value
Number of inputs	3
Display option	signals

"Sine Wave Function" (Sin)

Table 3.2. "Sine Wave Function" Parameters

Parameter	Value
Sine type	Time based
Time (t)	Use simulation time
Amplitude	$(V_{rms} \cdot \sqrt{2}) / \sqrt{3}$
Bias	0
Frequency (rad/sec)	$2 \cdot \pi \cdot f$
Phase (rad)	0

Parameter	Value
Samples per period	10
Number of offset samples	0
Sample time	0
Interpret vector parameters as 1-D	on

"Sine Wave Function1" (Sin)

Table 3.3. "Sine Wave Function1" Parameters

Parameter	Value
Sine type	Time based
Time (t)	Use simulation time
Amplitude	$(V_{rms} \cdot \sqrt{2}) / \sqrt{3}$
Bias	0
Frequency (rad/sec)	$2 \cdot \pi \cdot f$
Phase (rad)	$-2 \cdot \pi / 3$
Samples per period	10
Number of offset samples	0
Sample time	0
Interpret vector parameters as 1-D	on

"Sine Wave Function2" (Sin)

Table 3.4. "Sine Wave Function2" Parameters

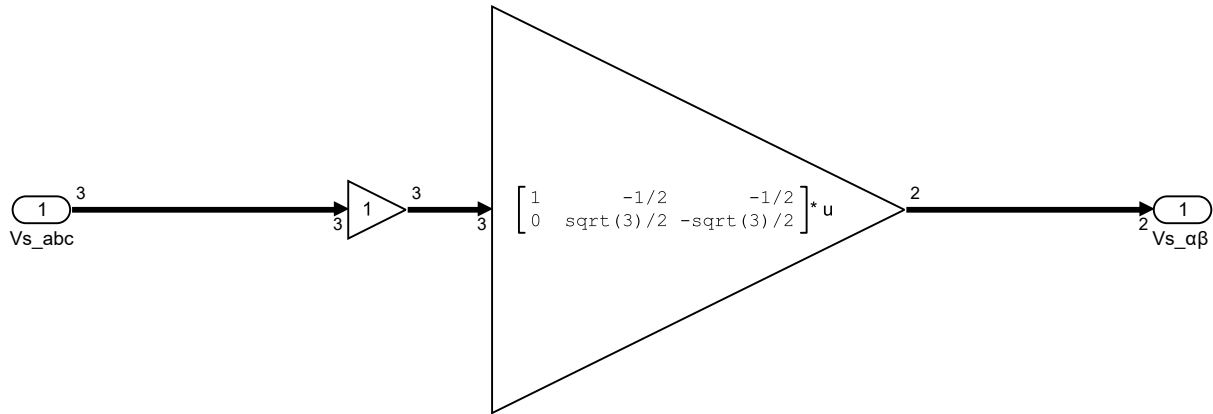
Parameter	Value
Sine type	Time based
Time (t)	Use simulation time
Amplitude	$(V_{rms} \cdot \sqrt{2}) / \sqrt{3}$
Bias	0
Frequency (rad/sec)	$2 \cdot \pi \cdot f$
Phase (rad)	$2 \cdot \pi / 3$
Samples per period	10
Number of offset samples	0
Sample time	0
Interpret vector parameters as 1-D	on

"Vs_abc " (Output)**Table 3.5. "Vs_abc " Parameters**

Parameter	Value
Port number	1
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

Clarke Transformation

Figure 3.2. Induction_motor/Motor, High speed /Clarke Transformation



Blocks

Parameters

"Gain" (Gain)

Table 3.6. "Gain" Parameters

Parameter	Value
Gain	[1 -1/2 -1/2; 0 sqrt(3)/2 -sqrt(3)/2]
Multiplication	Matrix(K*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off

Parameter	Value
Sample time (-1 for inherited)	-1

"Gain1" (Gain)**Table 3.7. "Gain1" Parameters**

Parameter	Value
Gain	1
Multiplication	Element-wise($K \cdot u$)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Vs_abc" (Inport)**Table 3.8. "Vs_abc" Parameters**

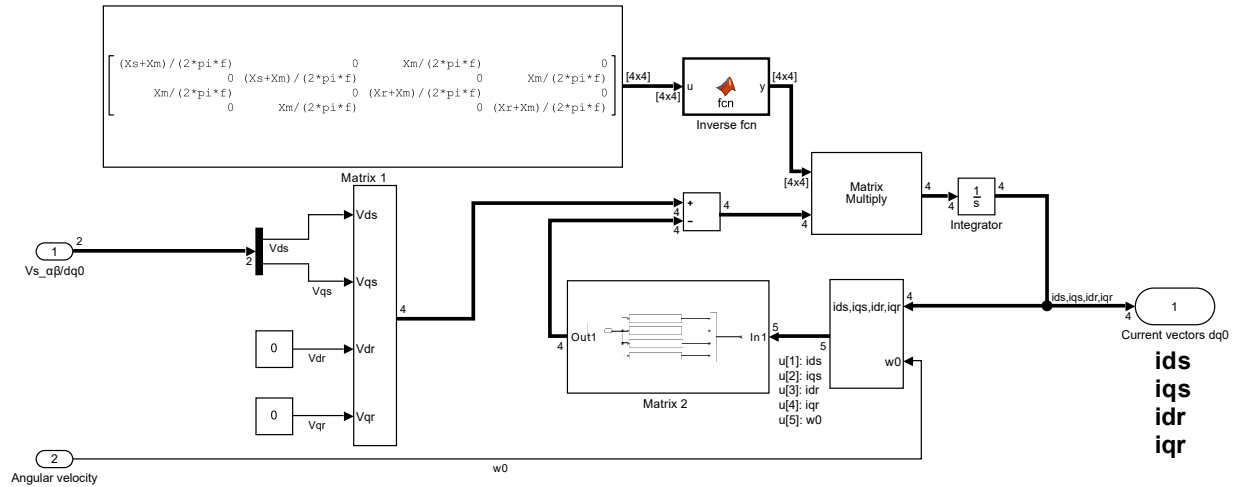
Parameter	Value
Port number	1
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

"Vs_αβ " (Outport)**Table 3.9. "Vs_αβ " Parameters**

Parameter	Value
Port number	1
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

Electrical sub-model

Figure 3.3. Induction_motor/Motor, High speed /Electrical sub-model



Blocks

Parameters

"Angular velocity " (Inport)

Table 3.10. "Angular velocity " Parameters

Parameter	Value
Port number	2
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

"Constant1" (Constant)

Table 3.11. "Constant1" Parameters

Parameter	Value
Constant value	0

Parameter	Value
Interpret vector parameters as 1-D	on
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit from 'Constant value'
Lock output data type setting against changes by the fixed-point tools	off
Sample time	inf
Frame period	inf

"Constant2" (Constant)

Table 3.12. "Constant2" Parameters

Parameter	Value
Constant value	0
Interpret vector parameters as 1-D	on
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit from 'Constant value'
Lock output data type setting against changes by the fixed-point tools	off
Sample time	inf
Frame period	inf

"Current vectors dq0 " (Outport)

Table 3.13. "Current vectors dq0 " Parameters

Parameter	Value
Port number	1
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit

Parameter	Value
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

"Demux" (Demux)

Table 3.14. "Demux" Parameters

Parameter	Value
Number of outputs	2
Display option	bar
Bus selection mode	off

"Integrator " (Integrator)

Table 3.15. "Integrator " Parameters

Parameter	Value
External reset	none
Initial condition source	internal
Initial condition	0
Limit output	off
Upper saturation limit	inf
Lower saturation limit	-inf
Wrap state	off
Wrapped state upper value	pi
Wrapped state lower value	-pi
Show saturation port	off
Show state port	off
Ignore limit and reset when linearizing	off

Parameter	Value
Enable zero-crossing detection	on
State Name (e.g., 'position')	"

"Inverse fcn" (MATLAB Function)

Table 3.16. Inverse fcn Function Properties

Property	Value
Update Method	INHERITED
Sample Time	
Support variable-size arrays	1
Saturate on integer overflow	1
Treat these inherited Simulink signal types as fixed-point objects	Fixed-point
MATLAB Function block fimath	Same as MATLAB Default
Input fixed-point math	fimath(...)
Description	

Table 3.17. Inverse fcn Argument Summary

Name	Scope	Port	Data Type	Size
u	Input	1	double	[4, 4]
y	Output	1	double	[4, 4]

Inverse fcn Function Script

```
function y = fcn(u)
```

```
y = inv(u);
```

Table 3.18. Inverse fcn Supporting Functions

Function	Defined By	Path
abs	MATLAB	
char	MATLAB	
coder.internal.applyScalarFunction	MATLAB	

Chapter 3. Subsystems

Function	Defined By	Path
<code>coder.internal.applyScalarFunctionInPlace</code>	MATLAB	
<code>coder.internal.avoidArrayFlattening</code>	MATLAB	
<code>coder.internal.blas.inline</code>	MATLAB	
<code>coder.internal.blas.int</code>	MATLAB	
<code>coder.internal.blas.isNullEmpty</code>	MATLAB	
<code>coder.internal.blas.ixamax</code>	MATLAB	
<code>coder.internal.blas.threshold</code>	MATLAB	
<code>coder.internal.blas.use_refblas</code>	MATLAB	
<code>coder.internal.blas.xger</code>	MATLAB	
<code>coder.internal.blas.xgeru</code>	MATLAB	
<code>coder.internal.blas.xswap</code>	MATLAB	
<code>coder.internal.blas.xtrsm</code>	MATLAB	
<code>coder.internal.charCastCheck</code>	MATLAB	
<code>coder.internal.div</code>	MATLAB	
<code>coder.internal.error</code>	MATLAB	
<code>coder.internal.floatModel</code>	MATLAB	
<code>coder.internal.flt2str</code>	MATLAB	
<code>coder.internal.indexMinus</code>	MATLAB	
<code>coder.internal.indexPlus</code>	MATLAB	
<code>coder.internal.indexTimes</code>	MATLAB	
<code>coder.internal.isBuiltInNumeric</code>	MATLAB	
<code>coder.internal.isFloatClass</code>	MATLAB	
<code>coder.internal.isaUint</code>	MATLAB	
<code>coder.internal.lapack.info_t</code>	MATLAB	
<code>coder.internal.lapack.threshold</code>	MATLAB	
<code>coder.internal.lapack.use_lapack</code>	MATLAB	
<code>coder.internal.lapack.xgetrf</code>	MATLAB	
<code>coder.internal.length</code>	MATLAB	
<code>coder.internal.nan</code>	MATLAB	

Chapter 3. Subsystems

Function	Defined By	Path
<code>coder.internal.narginchk</code>	MATLAB	
<code>coder.internal.refblas.ixamax</code>	MATLAB	
<code>coder.internal.refblas.xcabs1</code>	MATLAB	
<code>coder.internal.refblas.xger</code>	MATLAB	
<code>coder.internal.refblas.xgerx</code>	MATLAB	
<code>coder.internal.refblas.xswap</code>	MATLAB	
<code>coder.internal.refblas.xtrsm</code>	MATLAB	
<code>coder.internal.reflapack.xzgetrf</code>	MATLAB	
<code>coder.internal.scalar.abs</code>	MATLAB	
<code>coder.internal.scalar.floor</code>	MATLAB	
<code>coder.internal.scalarEg</code>	MATLAB	
<code>coder.internal.unsignedClass</code>	MATLAB	
<code>coder.internal.warning</code>	MATLAB	
<code>colon</code>	MATLAB	
<code>eps</code>	MATLAB	
<code>floor</code>	MATLAB	
<code>full</code>	MATLAB	
<code>intmax</code>	MATLAB	
<code>intmin</code>	MATLAB	
<code>inv</code>	MATLAB	
<code>isequal</code>	MATLAB	
<code>isfi</code>	MATLAB	
<code>isfimath</code>	MATLAB	
<code>ismatrix</code>	MATLAB	
<code>isnan</code>	MATLAB	
<code>isnumericitype</code>	MATLAB	
<code>issparse</code>	MATLAB	
<code>norm</code>	MATLAB	
<code>realmin</code>	MATLAB	

"Matrix 1 " (Constant)**Table 3.19. "Matrix 1 " Parameters**

Parameter	Value
Constant value	$[(X_s+X_m)/(2\pi f) \ 0 \ X_m/(2\pi f) \ 0; \ 0 \ (X_s+X_m)/(2\pi f) \ 0 \ X_m/(2\pi f); \ X_m/(2\pi f) \ 0 \ (X_r+X_m)/(2\pi f) \ 0; \ 0 \ X_m/(2\pi f) \ 0 \ (X_r+X_m)/(2\pi f)]$
Interpret vector parameters as 1-D	on
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit from 'Constant value'
Lock output data type setting against changes by the fixed-point tools	off
Sample time	inf
Frame period	inf

"Mux1" (Mux)**Table 3.20. "Mux1" Parameters**

Parameter	Value
Number of inputs	2
Display option	signals

"Mux3" (Mux)**Table 3.21. "Mux3" Parameters**

Parameter	Value
Number of inputs	4
Display option	signals

"Product" (Product)**Table 3.22. "Product" Parameters**

Parameter	Value
Number of inputs	2
Multiplication	Matrix(*)

Parameter	Value
Multiply over	All dimensions
Dimension	1
Require all inputs to have the same data type	off
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Subtract" (Sum)

Table 3.23. "Subtract" Parameters

Parameter	Value
Icon shape	rectangular
List of signs	+ -
Sum over	All dimensions
Dimension	1
Require all inputs to have the same data type	off
Accumulator data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock data type settings against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Vs_αβ/dq0 " (Inport)

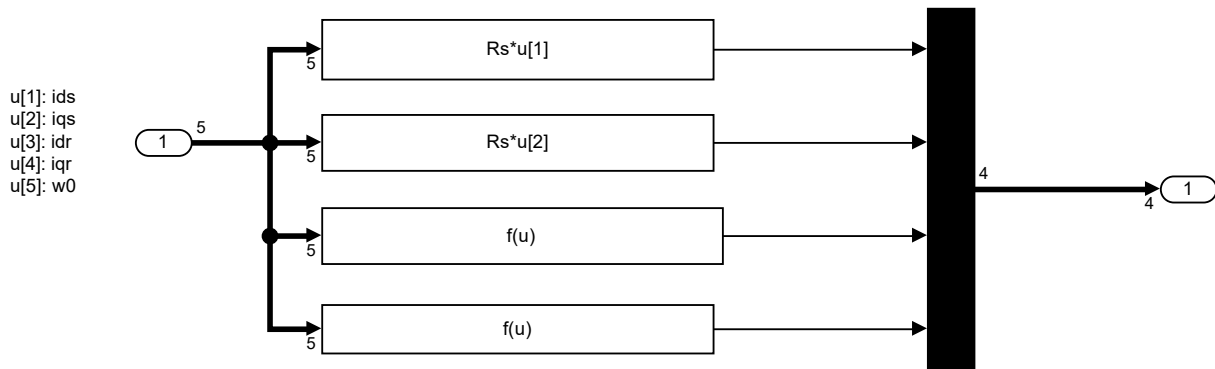
Table 3.24. "Vs_αβ/dq0 " Parameters

Parameter	Value
Port number	1
Port dimensions (-1 for inherited)	-1

Parameter	Value
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

Matrix 2

Figure 3.4. Induction_motor/Motor, High speed /Electrical sub-model /Matrix 2



Blocks

Parameters

"Fcn" (Fcn)

Table 3.25. "Fcn" Parameters

Parameter	Value
Expression	$R_s \cdot u[1]$
Sample time (-1 for inherited)	-1

"Fcn1" (Fcn)

Table 3.26. "Fcn1" Parameters

Parameter	Value
Expression	$R_s \cdot u[2]$

Parameter	Value
Sample time (-1 for inherited)	-1

"Fcn2" (Fcn)

Table 3.27. "Fcn2" Parameters

Parameter	Value
Expression	$P/2 \cdot (X_m / (2 \cdot \pi \cdot f)) \cdot u[2] \cdot u[5] + R \cdot r \cdot u[3] + P/2 \cdot ((X_r + X_m) / (2 \cdot \pi \cdot f)) \cdot u[4] \cdot u[5]$
Sample time (-1 for inherited)	-1

"Fcn3" (Fcn)

Table 3.28. "Fcn3" Parameters

Parameter	Value
Expression	$-P/2 \cdot (X_m / (2 \cdot \pi \cdot f)) \cdot u[1] \cdot u[5] - P/2 \cdot ((X_r + X_m) / (2 \cdot \pi \cdot f)) \cdot u[3] \cdot u[5] + R \cdot r \cdot u[4]$
Sample time (-1 for inherited)	-1

"In1" (Inport)

Table 3.29. "In1" Parameters

Parameter	Value
Port number	1
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

"Mux" (Mux)

Table 3.30. "Mux" Parameters

Parameter	Value
Number of inputs	4

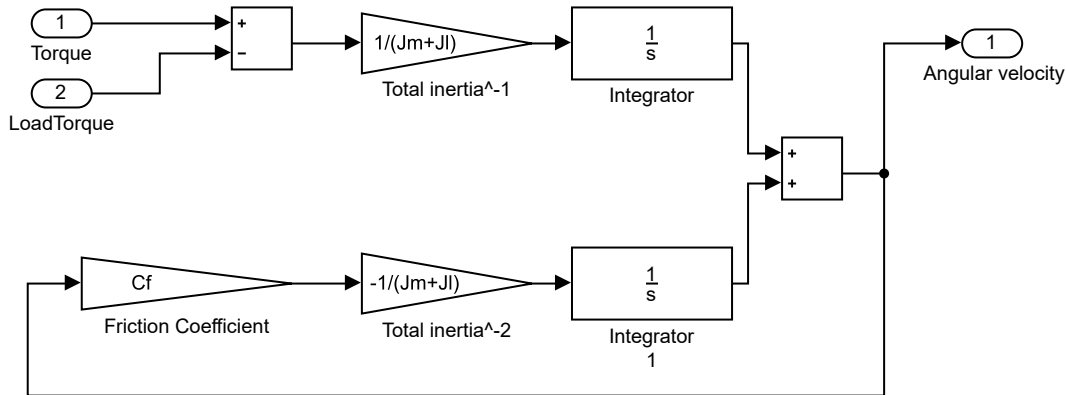
Parameter	Value
Display option	bar

"Out1" (Outport)**Table 3.31. "Out1" Parameters**

Parameter	Value
Port number	1
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

Mechanical sub-model

Figure 3.5. Induction_motor/Motor, High speed /Mechanical sub-model



Blocks

Parameters

" Angular velocity " (Output)

Table 3.32. " Angular velocity " Parameters

Parameter	Value
Port number	1
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s^2, N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outputport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]

Parameter	Value
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

"Add" (Sum)**Table 3.33. "Add" Parameters**

Parameter	Value
Icon shape	rectangular
List of signs	++
Sum over	All dimensions
Dimension	1
Require all inputs to have the same data type	off
Accumulator data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock data type settings against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Friction Coefficient" (Gain)**Table 3.34. "Friction Coefficient" Parameters**

Parameter	Value
Gain	Cf
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule

Parameter	Value
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Integrator " (Integrator)

Table 3.35. "Integrator " Parameters

Parameter	Value
External reset	none
Initial condition source	internal
Initial condition	1
Limit output	off
Upper saturation limit	inf
Lower saturation limit	-inf
Wrap state	off
Wrapped state upper value	pi
Wrapped state lower value	-pi
Show saturation port	off
Show state port	off
Ignore limit and reset when linearizing	off
Enable zero-crossing detection	on
State Name (e.g., 'position')	"

"Integrator 1" (Integrator)

Table 3.36. "Integrator 1" Parameters

Parameter	Value
External reset	none
Initial condition source	internal
Initial condition	1
Limit output	off
Upper saturation limit	inf
Lower saturation limit	-inf
Wrap state	off

Parameter	Value
Wrapped state upper value	pi
Wrapped state lower value	-pi
Show saturation port	off
Show state port	off
Ignore limit and reset when linearizing	off
Enable zero-crossing detection	on
State Name (e.g., 'position')	"

"LoadTorque " (Inport)

Table 3.37. "LoadTorque " Parameters

Parameter	Value
Port number	2
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

"Subtract" (Sum)

Table 3.38. "Subtract" Parameters

Parameter	Value
Icon shape	rectangular
List of signs	+ -
Sum over	All dimensions
Dimension	1
Require all inputs to have the same data type	off
Accumulator data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock data type settings against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off

Parameter	Value
Sample time (-1 for inherited)	-1

"Torque " (Inport)**Table 3.39. "Torque " Parameters**

Parameter	Value
Port number	1
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

"Total inertia⁻¹" (Gain)**Table 3.40. "Total inertia⁻¹" Parameters**

Parameter	Value
Gain	$1/(J_m + J_l)$
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

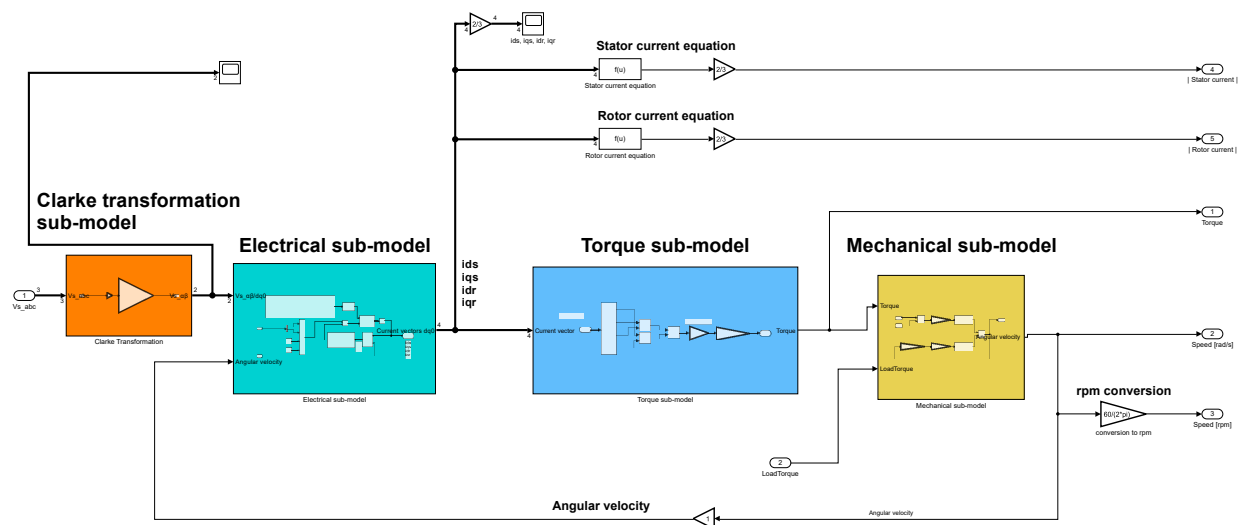
"Total inertia⁻²" (Gain)

Table 3.41. "Total inertia⁻²" Parameters

Parameter	Value
Gain	-1/(Jm+Jl)
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

Motor, High speed

Figure 3.6. Induction_motor/Motor, High speed



Blocks

Parameters

"conversion to rpm " (Gain)

Table 3.42. "conversion to rpm " Parameters

Parameter	Value
Gain	$60/(2*\pi)$
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Gain" (Gain)

Table 3.43. "Gain" Parameters

Parameter	Value
Gain	1
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule

Parameter	Value
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Gain1" (Gain)**Table 3.44. "Gain1" Parameters**

Parameter	Value
Gain	2/3
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Gain2" (Gain)**Table 3.45. "Gain2" Parameters**

Parameter	Value
Gain	2/3
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]

Parameter	Value
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Gain3" (Gain)**Table 3.46. "Gain3" Parameters**

Parameter	Value
Gain	2/3
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"LoadTorque " (Inport)**Table 3.47. "LoadTorque " Parameters**

Parameter	Value
Port number	2
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]

Parameter	Value
Data type	Inherit: auto

"Rotor current equation " (Fcn)

Table 3.48. "Rotor current equation " Parameters

Parameter	Value
Expression	$\sqrt{(u[3])^2 + (u[4])^2}$
Sample time (-1 for inherited)	-1

"Speed [rad/s] " (Outport)

Table 3.49. "Speed [rad/s] " Parameters

Parameter	Value
Port number	2
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

"Speed [rpm]" (Output)**Table 3.50. "Speed [rpm]" Parameters**

Parameter	Value
Port number	3
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

"Stator current equation " (Fcn)**Table 3.51. "Stator current equation " Parameters**

Parameter	Value
Expression	$\sqrt{(u[1])^2 + (u[2])^2}$
Sample time (-1 for inherited)	-1

"Torque" (Output)**Table 3.52. "Torque" Parameters**

Parameter	Value
Port number	1

Parameter	Value
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s^2, N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

"Vs_abc " (Inport)

Table 3.53. "Vs_abc " Parameters

Parameter	Value
Port number	1
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

"| Rotor current |" (Outport)

Table 3.54. "| Rotor current |" Parameters

Parameter	Value
Port number	5
Icon display	Port number

Parameter	Value
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outputport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

" | Stator current | " (Output)

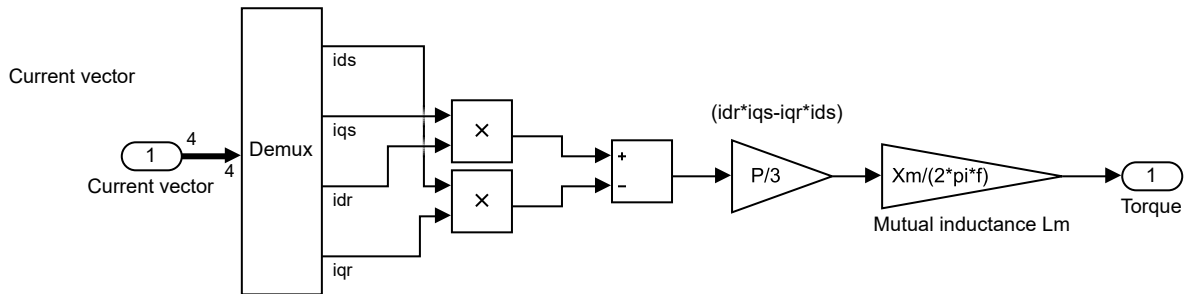
Table 3.55. " | Stator current | " Parameters

Parameter	Value
Port number	4
Icon display	Port number
Minimum	[]
Maximum	[]
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outputport is virtual	off
Source of initial output value	Dialog
Output when disabled	held

Parameter	Value
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

Torque sub-model

Figure 3.7. Induction_motor/Motor, High speed /Torque sub-model



Blocks

Parameters

"Current vector" (Inport)

Table 3.56. "Current vector" Parameters

Parameter	Value
Port number	1
Port dimensions (-1 for inherited)	-1
Sample time (-1 for inherited)	-1
Minimum	[]
Maximum	[]
Data type	Inherit: auto

"Demux" (Demux)**Table 3.57. "Demux" Parameters**

Parameter	Value
Number of outputs	4
Display option	none
Bus selection mode	off

"Gain" (Gain)**Table 3.58. "Gain" Parameters**

Parameter	Value
Gain	P/3
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Mutual inductance Lm " (Gain)**Table 3.59. "Mutual inductance Lm " Parameters**

Parameter	Value
Gain	$X_m/(2 \cdot \pi \cdot f)$
Multiplication	Element-wise(K.*u)
Parameter minimum	[]
Parameter maximum	[]
Parameter data type	Inherit: Inherit via internal rule

Parameter	Value
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Product" (Product)**Table 3.60. "Product" Parameters**

Parameter	Value
Number of inputs	2
Multiplication	Element-wise(.*)
Multiply over	All dimensions
Dimension	1
Require all inputs to have the same data type	off
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Product1" (Product)**Table 3.61. "Product1" Parameters**

Parameter	Value
Number of inputs	2
Multiplication	Element-wise(.*)
Multiply over	All dimensions
Dimension	1

Parameter	Value
Require all inputs to have the same data type	off
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock output data type setting against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Subtract" (Sum)

Table 3.62. "Subtract" Parameters

Parameter	Value
Icon shape	rectangular
List of signs	+ -
Sum over	All dimensions
Dimension	1
Require all inputs to have the same data type	off
Accumulator data type	Inherit: Inherit via internal rule
Output minimum	[]
Output maximum	[]
Output data type	Inherit: Inherit via internal rule
Lock data type settings against changes by the fixed-point tools	off
Integer rounding mode	Floor
Saturate on integer overflow	off
Sample time (-1 for inherited)	-1

"Torque" (Outport)

Table 3.63. "Torque" Parameters

Parameter	Value
Port number	1
Icon display	Port number
Minimum	[]
Maximum	[]

Chapter 3. Subsystems

Parameter	Value
Data type	Inherit: auto
Lock output data type setting against changes by the fixed-point tools	off
Output as nonvirtual bus in parent model	off
Unit (e.g., m, m/s ² , N*m)	inherit
Port dimensions (-1 for inherited)	-1
Variable-size signal	Inherit
Sample time (-1 for inherited)	-1
Ensure outport is virtual	off
Source of initial output value	Dialog
Output when disabled	held
Initial output	[]
MustResolveToSignalObject	off
Specify output when source is unconnected	off
Constant value	0
Interpret vector parameters as 1-D	off

Chapter 4. System Design Variables

Design Variable Summary

Table 4.1. Functions used in Design Variable Expressions

Function Name	Parent Blocks	Calling character vector
P	Motor, High speed	P

Chapter 5. Requirements

Induction_motor does not contain requirements traceability links.

Chapter 6. System Model Configuration

Source: Model
Source Name: Induction_motor

Table 6.1. Induction_motor Configuration Set

Property	Value
Description	
Components	[Induction_motor Configuration Set.Components(1) , Induction_motor Configuration Set.Components(2) , Induction_motor Configuration Set.Components(3) , Induction_motor Configuration Set.Components(4) , Induction_motor Configuration Set.Components(5) , Induction_motor Configuration Set.Components(6) , Induction_motor Configuration Set.Components(7) , Induction_motor Configuration Set.Components(8) , Induction_motor Configuration Set.Components(9) , Induction_motor Configuration Set.Components(10) , Induction_motor Configuration Set.Components(11)]
Name	Configuration
SimulationMode	normal
ConfigType	Model

Table 6.2. Induction_motor Configuration Set.Components(1)

Property	Value
Name	Solver
Description	
Components	
StartTime	0.0
StopTime	40
AbsTol	auto
AutoScaleAbsTol	on
FixedStep	auto
InitialStep	auto
MaxNumMinSteps	-1
MaxOrder	5
ZcThreshold	auto
ConsecutiveZCsStepRelTol	10*128*eps

MaxConsecutiveZCs	1000
ExtrapolationOrder	4
NumberNewtonIterations	1
MaxStep	auto
MinStep	auto
MaxConsecutiveMinStep	1
RelTol	1e-3
SolverMode	SingleTasking
EnableMultiTasking	off
EnableExplicitPartitioning	off
EnableConcurrentExecution	on
ConcurrentTasks	off
Solver	VariableStepAuto
SolverName	VariableStepAuto
SolverType	Variable-step
SolverJacobianMethodControl	auto
ShapePreserveControl	DisableAll
ZeroCrossControl	UseLocalSettings
ZeroCrossAlgorithm	Nonadaptive
SolverResetMethod	Fast
PositivePriorityOrder	off
AutoInsertRateTranBlk	off
SampleTimeConstraint	Unconstrained
InsertRTBMode	Whenever possible
SampleTimeProperty	
DecoupledContinuousIntegration	off
MinimalZcImpactIntegration	off

Table 6.3. Induction_motor Configuration Set.Components(2)

Property	Value
Name	Data Import/Export
Description	
Components	
Decimation	1
ExternalInput	[t, u]
FinalStateName	xFinal
InitialState	xInitial

LimitDataPoints	off
MaxDataPoints	1000
LoadExternalInput	off
LoadInitialState	off
SaveFinalState	off
SaveCompleteFinalSimState	off
SaveFormat	Dataset
SaveOutput	on
SaveState	off
SignalLogging	on
DSMLogging	on
InspectSignalLogs	off
SaveTime	on
ReturnWorkspaceOutputs	off
StateSaveName	xout
TimeSaveName	tout
OutputSaveName	yout
SignalLoggingName	logout
DSMLoggingName	dsmout
OutputOption	RefineOutputTimes
OutputTimes	[]
ReturnWorkspaceOutputsName	out
Refine	1
LoggingToFile	off
DatasetSignalFormat	timeseries
LoggingFileName	out.mat
LoggingIntervals	[-inf, inf]

Table 6.4. Induction_motor Configuration Set.Components(3)

Property	Value
Name	Optimization
Description	
Components	
BlockReduction	on
BooleanDataType	on
ConditionallyExecuteInputs	on
DefaultParameterBehavior	Tunable

Chapter 6. System Model Configuration

InlineParams	off
UseDivisionForNetSlopeComputation	off
UseFloatMulNetSlope	off
DefaultUnderspecifiedDataType	double
UseSpecifiedMinMax	off
InlineInvariantSignals	off
OptimizeBlockIOStorage	on
BufferReuse	on
GlobalBufferReuse	on
GlobalVariableUsage	None
StrengthReduction	off
AdvancedOptControl	
EnforceIntegerDowncast	on
ExpressionFolding	on
BooleansAsBitfields	off
BitfieldContainerType	uint_T
EnableMemcpy	on
MemcpyThreshold	64
PassReuseOutputArgsAs	Structure reference
PassReuseOutputArgsThreshold	12
FoldNonRolledExpr	on
LocalBlockOutputs	on
RollThreshold	5
StateBitsets	off
DataBitsets	off
ActiveStateOutputEnumStorageType	Native Integer
UseTempVars	off
ZeroExternalMemoryAtStartup	on
ZeroInternalMemoryAtStartup	on
InitFltsAndDblsToZero	off
NoFixptDivByZeroProtection	off
EfficientFloat2IntCast	off
EfficientMapNaN2IntZero	on
LifeSpan	auto
EvaldLifeSpan	Inf
MaxStackSize	Inherit from target
BufferReusableBoundary	on
SimCompilerOptimization	off

AccelVerboseBuild	off
OptimizeBlockOrder	off
OptimizeDataStoreBuffers	on
BusAssignmentInplaceUpdate	on
DifferentSizesBufferReuse	off
OptimizationLevel	level2
OptimizationPriority	Balanced
OptimizationCustomize	on
UseRowMajorAlgorithm	off
LabelGuidedReuse	off

Table 6.5. Induction_motor Configuration Set.Components(4)

Property	Value
Name	Diagnostics
Description	
Components	
RTPrefix	error
ConsistencyChecking	none
ArrayBoundsChecking	none
SignalInfNanChecking	none
StringTruncationChecking	error
SignalRangeChecking	none
ReadBeforeWriteMsg	UseLocalSettings
WriteAfterWriteMsg	UseLocalSettings
WriteAfterReadMsg	UseLocalSettings
AlgebraicLoopMsg	warning
ArtificialAlgebraicLoopMsg	warning
SaveWithDisabledLinksMsg	warning
SaveWithParameterizedLinksMsg	warning
CheckSSInitialOutputMsg	on
UnderspecifiedInitializationDetection	Simplified
MergeDetectMultiDrivingBlocksExec	error
CheckExecutionContextRuntimeOutputMsg	off
SignalResolutionControl	UseLocalSettings
BlockPriorityViolationMsg	warning
MinStepSizeMsg	warning
TimeAdjustmentMsg	none

Chapter 6. System Model Configuration

MaxConsecutiveZCsMsg	error
MaskedZcDiagnostic	warning
IgnoredZcDiagnostic	warning
SolverPrmCheckMsg	none
InheritedTsInSrcMsg	warning
MultiTaskDSMMsg	error
MultiTaskCondExecSysMsg	error
MultiTaskRateTransMsg	error
SingleTaskRateTransMsg	none
TasksWithSamePriorityMsg	warning
SigSpecEnsureSampleTimeMsg	warning
CheckMatrixSingularityMsg	none
IntegerOverflowMsg	warning
Int32ToFloatConvMsg	warning
ParameterDowncastMsg	error
ParameterOverflowMsg	error
ParameterUnderflowMsg	none
ParameterPrecisionLossMsg	warning
ParameterTunabilityLossMsg	warning
FixptConstUnderflowMsg	none
FixptConstOverflowMsg	none
FixptConstPrecisionLossMsg	none
UnderSpecifiedDataTypeMsg	none
UnnecessaryDatatypeConvMsg	none
VectorMatrixConversionMsg	none
InvalidFcnCallConnMsg	error
FcnCallInpInsideContextMsg	error
SignalLabelMismatchMsg	none
UnconnectedInputMsg	warning
UnconnectedOutputMsg	warning
UnconnectedLineMsg	warning
UseOnlyExistingSharedCode	error
SFcnCompatibilityMsg	none
FrameProcessingCompatibilityMsg	error
UniqueDataStoreMsg	none
BusObjectLabelMismatch	warning
RootOutportRequireBusObject	warning
AssertControl	UseLocalSettings

Chapter 6. System Model Configuration

Echo	
EnableOverflowDetection	off
AllowSymbolicDim	on
ModelReferenceIOMsg	none
ModelReferenceVersionMismatchMessage	none
ModelReferenceIOMismatchMessage	none
ModelReferenceCSMismatchMessage	none
ModelReferenceSimTargetVerbose	off
UnknownTsInhSupMsg	warning
ModelReferenceDataLoggingMessage	warning
ModelReferenceSymbolNameMessage	warning
ModelReferenceExtraNoncontSigs	error
StateNameClashWarn	none
SimStateInterfaceChecksumMismatchMsg	warning
SimStateOlderReleaseMsg	error
InitInArrayFormatMsg	warning
StrictBusMsg	ErrorLevel1
BusNameAdapt	WarnAndRepair
NonBusSignalsTreatedAsBus	none
SFUnusedDataAndEventsDiag	warning
SFUnexpectedBacktrackingDiag	error
SFInvalidInputDataAccessInChartInitDiag	warning
SFNoUnconditionalDefaultTransitionDiag	error
SFTransitionOutsideNaturalParentDiag	warning
SFUnconditionalTransitionShadowingDiag	warning
SFUnreachableExecutionPathDiag	warning
SFUndirectedBroadcastEventsDiag	warning
SFTransitionActionBeforeConditionDiag	warning
SFOutputUsedAsStateInMooreChartDiag	error
SFTemporalDelaySmallerThanSampleTimeDiag	warning
SFUnconditionalPathOutOfParentDiag	warning
SFSelfTransitionDiag	warning
SFExecutionAtInitializationDiag	warning
SFMachineParentedDataDiag	warning
SFUnreachableStateOrJunctionDiag	warning
SFDanglingTransitionDiag	warning
IntegerSaturationMsg	warning
AllowedUnitSystems	all

UnitsInconsistencyMsg	warning
AllowAutomaticUnitConversions	on
RCSCRenamedMsg	warning
RCSCObservableMsg	warning
ForceCombineOutputUpdateInSim	off
UnderSpecifiedDimensionMsg	none

Table 6.6. Induction motor Configuration Set.Components(5)

Property	Value
Name	Hardware Implementation
Description	
Components	
ProdBitPerChar	8
ProdBitPerShort	16
ProdBitPerInt	32
ProdBitPerLong	32
ProdBitPerLongLong	64
ProdBitPerFloat	32
ProdBitPerDouble	64
ProdBitPerPointer	64
ProdBitPerSizeT	64
ProdBitPerPtrDiffT	64
ProdLargestAtomicInteger	Char
ProdLargestAtomicFloat	Float
ProdIntDivRoundTo	Zero
ProdEndianess	LittleEndian
ProdWordSize	64
ProdShiftRightIntArith	on
ProdLongLongMode	off
ProdHWDeviceType	Intel->x86-64 (Windows64)
TargetBitPerChar	8
TargetBitPerShort	16
TargetBitPerInt	32
TargetBitPerLong	32
TargetBitPerLongLong	64
TargetBitPerFloat	32
TargetBitPerDouble	64

TargetBitPerPointer	32
TargetBitPerSizeT	32
TargetBitPerPtrDiffT	32
TargetLargestAtomicInteger	Char
TargetLargestAtomicFloat	None
TargetShiftRightIntArith	on
TargetLongLongMode	off
TargetIntDivRoundTo	Undefined
TargetEndianness	Unspecified
TargetWordSize	32
TargetPreprocMaxBitsSint	32
TargetPreprocMaxBitsUint	32
TargetHWDeviceType	Specified
TargetUnknown	off
DenormalBehavior	Default
ProdEqTarget	on
UseEmbeddedCoderFeatures	on
UseSimulinkCoderFeatures	on

Table 6.7. Induction motor Configuration Set.Components(6)

Property	Value
Name	Model Referencing
Description	
Components	
UpdateModelReferenceTargets	IfOutOfDateOrStructuralChange
SkipRefExpFcnMdlSchedulingOrderCheck	off
EnableRefExpFcnMdlSchedulingChecks	on
CheckModelReferenceTargetMessage	error
EnableParallelModelReferenceBuilds	off
ParallelModelReferenceErrorOnInvalidPool	on
ParallelModelReferenceMATLABWorkerInit	None
ModelReferenceNumInstancesAllowed	Multi
PropagateVarSize	Infer from blocks in model
ModelDependencies	
ModelReferencePassRootInputsByReference	on
ModelReferenceMinAlgLoopOccurrences	off
PropagateSignalLabelsOutOfModel	on

SupportModelReferenceSimTargetCustomCode	off
------------------------------------------	-----

Table 6.8. Induction_motor Configuration Set.Components(7)

Property	Value
Name	Simulation Target
Description	
Components	
SimCustomSourceCode	
SimCustomHeaderCode	
SimCustomInitializer	
SimCustomTerminator	
SimReservedNameArray	
SimUserSources	
SimUserIncludeDirs	
SimUserLibraries	
SimUserDefines	
SFSimEnableDebug	off
SFSimOverflowDetection	on
SFSimEcho	on
SimBlas	on
SimCtrlC	on
SimExtrinsic	on
SimIntegrity	on
SimUseLocalCustomCode	on
SimParseCustomCode	on
SimAnalyzeCustomCode	off
SimBuildMode	sf_incremental_build
SimDataInitializer	
SimGenImportedTypeDefs	off
CompileTimeRecursionLimit	50
EnableRuntimeRecursion	on
MATLABDynamicMemAlloc	on
MATLABDynamicMemAllocThreshold	65536
CustomSymbolStrEMXArray	nothing
CustomSymbolStrEMXArrayFcn	nothing
CustomCodeFunctionArrayLayout	

DefaultCustomCodeFunctionArrayLayout	NotSpecified
--------------------------------------	--------------

Table 6.9. Induction_motor Configuration Set.Components(8)

Property	Value
Name	Code Generation
SystemTargetFile	grt.tlc
HardwareBoard	None
TLCOptions	
CodeGenDirectory	
GenCodeOnly	off
MakeCommand	make_rtw
GenerateMakefile	on
PackageGeneratedCodeAndArtifacts	off
PackageName	
TemplateMakefile	grt_default_tmf
PostCodeGenCommand	
Description	
GenerateReport	off
SaveLog	off
RTWVerbose	on
RetainRTWFile	off
ProfileTLC	off
TLCDebug	off
TLCCoverage	off
TLCAssert	off
ProcessScriptMode	Default
ConfigurationMode	Optimized
ProcessScript	
ConfigurationScript	
ConfigAtBuild	off
RTWUseLocalCustomCode	on
RTWUseSimCustomCode	off
CustomSourceCode	
CustomHeaderCode	
CustomInclude	
CustomSource	
CustomLibrary	

Chapter 6. System Model Configuration

CustomDefine	
CustomBLASCallback	
CustomLAPACKCallback	
CustomFFTCallback	
CustomInitializer	
CustomTerminator	
Toolchain	Automatically locate an installed toolchain
BuildConfiguration	Faster Builds
CustomToolchainOptions	
IncludeHyperlinkInReport	off
LaunchReport	off
RecursionLimit	50
PortableWordSizes	off
GenerateErtSFunction	off
CreateSILPILBlock	None
CodeExecutionProfiling	off
CodeExecutionProfileVariable	executionProfile
CodeProfilingSaveOptions	SummaryOnly
CodeProfilingInstrumentation	off
CodeCoverageSettings	Induction_motor Configuration Set.Components(8).CodeCoverageSettings
SILDebugging	off
TargetLang	C
IncludeERTFirstTime	off
GenerateTraceInfo	off
GenerateTraceReport	off
GenerateTraceReportSl	off
GenerateTraceReportSf	off
GenerateTraceReportEml	off
GenerateCodeInfo	off
GenerateWebview	off
GenerateCodeMetricsReport	off
GenerateCodeReplacementReport	off
RTWCompilerOptimization	off
ObjectivePriorities	
RTWCustomCompilerOptimizations	
CheckMdlBeforeBuild	Off
CustomRebuildMode	OnUpdate

DataInitializer	
Components	[Induction_motor Configuration Set.Components(8).Components(1), Induction_motor Configuration Set.Components(8).Components(2)]

Table 6.10. Induction_motor Configuration Set.Components(9)

Property	Value
Description	Simulink Coverage Configuration Component
Components	
Name	Simulink Coverage
CovEnable	off
CovScope	EntireSystem
CovIncludeTopModel	on
RecordCoverage	off
CovPath	/
CovSaveName	covdata
CovCompData	
CovMetricSettings	dwe
CovFilter	
CovHTMLOptions	
CovNameIncrementing	off
CovHtmlReporting	off
CovForceBlockReductionOff	on
CovEnableCumulative	on
CovSaveCumulativeToWorkspaceVar	off
CovSaveSingleToWorkspaceVar	off
CovCumulativeVarName	covCumulativeData
CovCumulativeReport	off
CovSaveOutputData	on
CovOutputDir	slcov_output/\$ModelName\$
CovDataFileName	\$ModelName\$_cvdata
CovShowResultsExplorer	on
CovReportOnPause	on
CovModelRefEnable	off
CovModelRefExcluded	
CovExternalEMLEnable	on
CovSFcnEnable	on

CovBoundaryAbsTol	1.0000e-05
CovBoundaryRelTol	0.0100
CovUseTimeInterval	off
CovStartTime	0
CovStopTime	0
CovMetricStructuralLevel	Decision
CovMetricLookupTable	off
CovMetricSignalRange	off
CovMetricSignalSize	off
CovMetricObjectiveConstraint	off
CovMetricSaturateOnIntegerOverflow	off
CovMetricRelationalBoundary	off
CovLogicBlockShortCircuit	off
CovUnsupportedBlockWarning	on
CovHighlightResults	off
CovMcdcMode	Masking

Table 6.11. Induction motor Configuration Set.Components(10)

Property	Value
Description	HDL Coder custom configuration component
Components	
Name	HDL Coder

Table 6.12. Induction motor Configuration Set.Components(11)

Property	Value
Description	
Components	[Induction_motor Configuration Set.Components(11).Components(1) , Induction_motor Configuration Set.Components(11).Components(2)]
Name	Simscape
EditingMode	Full
ExplicitSolverDiagnosticOptions	warning
GlobalZcOffDiagnosticOptions	warning
SimscapeNormalizeSystem	on
SimscapeNominalValues	{{"value":"1","unit":"A"}, {"value":"1","unit":"bar"}, {"value":"1","unit":"cm^2"}, {"value":"1","unit":"cm^3/s"}, {"value":"1","unit":"k/kg"}, {"value":"1","unit":"kW"}, {"value":"1","unit":"l"}, {"value"

Chapter 6. System Model Configuration

	:"1","unit":"N"},{"value":"1","unit":"N*m"},{"value":"1","unit":"V"}]}
SimscapeLogType	none
SimscapeLogSimulationStatistics	off
SimscapeLogToSDI	off
SimscapeLogOpenViewer	off
SimscapeLogName	simlog
SimscapeLogDecimation	1
SimscapeLogLimitData	on
SimscapeLogDataHistory	5000
SimscapeUseOperatingPoints	off
SimscapeOperatingPoint	
SelectedTab	
Version	1.0
ComponentsAttached	true
Listener	[Induction_motor Configuration Set.Components(11).Listener(1), Induction_motor Configuration Set.Components(11).Listener(2), Induction_motor Configuration Set.Components(11).Listener(3), Induction_motor Configuration Set.Components(11).Listener(4), Induction_motor Configuration Set.Components(11).Listener(5), Induction_motor Configuration Set.Components(11).Listener(6), Induction_motor Configuration Set.Components(11).Listener(7), Induction_motor Configuration Set.Components(11).Listener(8), Induction_motor Configuration Set.Components(11).Listener(9), Induction_motor Configuration Set.Components(11).Listener(10), Induction_motor Configuration Set.Components(11).Listener(11), Induction_motor Configuration Set.Components(11).Listener(12), Induction_motor Configuration Set.Components(11).Listener(13), Induction_motor Configuration Set.Components(11).Listener(14), Induction_motor Configuration Set.Components(11).Listener(15), Induction_motor Configuration Set.Components(11).Listener(16)]
someListenersNotInstalled	false
instanceId	

Table 6.13. Induction motor Configuration Set.Components(8).CodeCoverageSettings

Property	Value
TopModelCoverage	off
ReferencedModelCoverage	off
CoverageTool	None

Table 6.14. Induction motor Configuration
Set.Components(8).Components(1)

Property	Value
Name	Code Appearance
Description	
Components	
ForceParamTrailComments	off
GenerateComments	on
CommentStyle	Auto
IgnoreCustomStorageClasses	on
IgnoreTestpoints	off
IncHierarchyInIds	off
MaxIdLength	31
ShowEliminatedStatement	off
OperatorAnnotations	off
IncAutoGenComments	off
SimulinkDataObjDesc	off
SFDataObjDesc	off
MATLABFcnDesc	off
IncDataTypeInIds	off
PrefixModelToSubsysFcnNames	on
MangleLength	1
SharedChecksumLength	8
CustomSymbolStr	\$R\$N\$M
CustomSymbolStrGlobalVar	\$R\$N\$M
CustomSymbolStrType	\$N\$R\$M_T
CustomSymbolStrField	\$N\$M
CustomSymbolStrFcn	\$R\$N\$M\$F
CustomSymbolStrSimulinkFcn	\$R\$N
CustomSymbolStrFcnArg	rt\$I\$N\$M
CustomSymbolStrBlkIO	rtb_\$N\$M
CustomSymbolStrTmpVar	\$N\$M
CustomSymbolStrMacro	\$R\$N\$M
CustomSymbolStrUtil	\$N\$C
CustomSymbolStrEmxType	emxArray_\$M\$N
CustomSymbolStrEmxFcn	emx\$M\$N
CustomUserTokenString	

CustomCommentsFcn	
DefineNamingRule	None
DefineNamingFcn	
ParamNamingRule	None
ParamNamingFcn	
SignalNamingRule	None
SignalNamingFcn	
InsertBlockDesc	off
InsertPolySpaceComments	off
SimulinkBlockComments	on
BlockCommentType	BlockPathComment
StateflowObjectComments	off
MATLABSourceComments	off
EnableCustomComments	off
InternalIdentifier	Shortened
InlinedPrmAccess	Literals
ReqsInCode	off
UseSimReservedNames	off
ReservedNameArray	

Table 6.15. Induction motor Configuration
Set.Components(8).Components(2)

Property	Value
Name	Target
Description	
Components	
IsERTTarget	off
TargetLibSuffix	
TargetPreCompLibLocation	
GenFloatMathFcnCalls	NOT IN USE
TargetLangStandard	C99 (ISO)
TargetFunctionLibrary	NOT IN USE
CodeReplacementLibrary	None
UtilityFuncGeneration	Auto
ERTMultiwordTypeDef	System defined
MultiwordTypeDef	System defined
ERTMultiwordLength	2048

Chapter 6. System Model Configuration

MultiwordLength	2048
DynamicStringBufferSize	256
GenerateFullHeader	on
InferredTypesCompatibility	off
ExistingSharedCode	
SharedCodeLocation	
GenerateSampleERTMain	off
GenerateTestInterfaces	off
ModelReferenceCompliant	on
ParMdlRefBuildCompliant	on
CompOptLevelCompliant	on
ConcurrentExecutionCompliant	on
IncludeMdlTerminateFcn	on
CombineOutputUpdateFcns	on
CombineSignalStateStructs	off
GroupInternalDataByFunction	off
SuppressErrorStatus	off
ERTFirstTimeCompliant	off
IncludeFileDelimiter	Auto
ERTCustomFileBanners	off
SupportAbsoluteTime	on
LogVarNameModifier	rt_
MatFileLogging	on
MultiInstanceERTCode	off
CodeInterfacePackaging	Nonreusable function
PurelyIntegerCode	off
SupportNonFinite	on
SupportComplex	on
SupportContinuousTime	on
SupportNonInlinedSFcns	on
RemoveDisableFunc	off
RemoveResetFunc	off
SupportVariableSizeSignals	off
ParenthesesLevel	Nominal
CastingMode	Nominal
PreserveStateflowLocalDataDimensions	off
GenerateClassInterface	off
ModelStepFunctionPrototypeControlCompliant	off

CPPClassGenCompliant	on
GRTInterface	off
GenerateAllocFcn	off
UseToolchainInfoCompliant	on
GenerateSharedConstants	on
LUTObjectStructOrderExplicitValues	Size,Breakpoints,Table
LUTObjectStructOrderEvenSpacing	Size,Breakpoints,Table
ArrayLayout	Column-major
UnsupportedSFcnMsg	error
ERTHeaderFileRootName	\$R\$E
ERTSourceFileRootName	\$R\$E
ERTDataFileRootName	\$R_data
ExtMode	off
ExtModeStaticAlloc	off
ExtModeTesting	off
ExtModeStaticAllocSize	1000000
ExtModeTransport	0
ExtModeMexFile	ext_comm
ExtModeMexArgs	
ExtModeIntrfLevel	Level1
RTWCAPISignals	off
RTWCAPIParams	off
RTWCAPIStates	off
RTWCAPIRootIO	off
GenerateASAP2	off
MultiInstanceErrorCode	Error

Table 6.16. Induction_motor Configuration
Set.Components(11).Components(1)

Property	Value
Description	
Components	
Name	SimscapeMultibody1G
WarnOnRedundantConstraints	on
WarnOnSingularInitialAssembly	off
ShowCutJoints	off
VisOnUpdateDiagram	off

VisDuringSimulation	off
EnableVisSimulationTime	on
VisSampleTime	0
DisableBodyVisControl	off
ShowCG	on
ShowCS	on
ShowOnlyPortCS	off
HighlightModel	on
FramesToBeSkipped	0
AnimationDelay	3
RecordAVI	off
CompressAVI	on
AviFileName	
AutoFitVis	off
EnableSelection	on
LastVizWinPosition	[-1 -1 -1 -1]
CamPosition	[0 0 0]
CamTarget	[0 0 -1]
CamUpVector	[0 1 0]
CamHeight	-1
CamViewAngle	0
VisBackgroundColor	[0.9 0.9 0.95]
DefaultBodyColor	[1 0 0]
MDLBodyVisualizationType	Convex hull from body CS locations
OVRRIDBodyVisualizationType	NONE
VisConfigFile	

Table 6.17. Induction_motor Configuration Set.Components(11).Components(2)

Property	Value
Description	Simscape Multibody
Components	[Induction_motor Configuration Set.Components(11).Components(2).Components(1)], Induction_motor Configuration Set.Components(11).Components(2).Components(2)]
Name	SimscapeMultibody

Induction_motor Configuration Set.Components(11).Listener(1) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(2) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(3) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(4) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(5) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(6) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(7) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(8) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(9) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(10) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(11) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(12) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(13) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(14) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(15) (handle.listener,)

Note: this object has no unfiltered properties.

Induction_motor Configuration Set.Components(11).Listener(16) (handle.listener,)

Note: this object has no unfiltered properties.

**Table 6.18. Induction motor Configuration
Set.Components(11).Components(2).Components(1)**

Property	Value
Description	Diagnostics
Components	
Name	DiagnosticsConfigSet
SimMechanicsInvalidVisualProperty	warning
SimMechanicsCrossSectionNullEdge	warning
SimMechanicsUnconnectedFramePorts	warning
SimMechanicsUnconnectedGeometryPorts	warning
SimMechanicsRedundantBlock	warning
SimMechanicsConflictingReferenceFrames	warning
SimMechanicsRigidlyBoundBlock	error
SimMechanicsUnsatisfiedHighPriorityTargets	warning
SimMechanicsJointTargetOverSpecification	error

**Table 6.19. Induction motor Configuration
Set.Components(11).Components(2).Components(2)**

Property	Value
Description	Explorer
Components	
Name	ExplorerConfigSet
SimMechanicsOpenEditorOnUpdate	on
InternalSimMechanicsExplorerSettings	

Table 6.20. HDL Coder

Property	Value
HDLSubsystem	Induction_motor
Workflow	Generic ASIC/FPGA
TargetPlatform	
ReferenceDesign	

Chapter 6. System Model Configuration

ReferenceDesignPath	
CoeffPrefix	coeff
InputType	std_logic_vector
OutputType	Same as input type
ScalarizePorts	off
CoeffMultipliers	Multiplier
ResetType	Asynchronous
FIRAdderStyle	linear
MultiplierInputPipeline	0
MultiplierOutputPipeline	0
FoldingFactor	1
NumMultipliers	-1
OptimizeForHDL	off
TimingControllerPostfix	_tc
OptimizeTimingController	on
TimingControllerArch	default
CastBeforeSum	on
CheckHDL	off
EnablePrefix	enb
ClockEnableInputPort	clk_enable
ClockEnableOutputPort	ce_out
ClockInputPort	clk
ClockEdge	Rising
ResetInputPort	reset
SimulatorFlags	
HDLCompileFilePostfix	_compile.do
HDLCompileInit	vlib %s\n
HDLCompileTerm	
HDLCompileVerilogCmd	vlog %s %s\n
HDLCompileVHDLCmd	vcom %s %s\n
EnableForGenerateLoops	on
HDLMapFilePostfix	_map.txt
HDLMapSeparator	
HDLSimCmd	vsim -novopt %s.%s\n
HDLSimFilePostfix	_sim.do
HDLSimProjectFilePostfix	_init.do
HDLSimInit	onbreak resume\nnonerror resume\n
HDLSimProjectCmd	project addfile %s\n

Chapter 6. System Model Configuration

HDLSimProjectTerm	project compileall\n
HDLSimProjectInit	project new . %s work\n
HDLSimTerm	run -all\n
HDLSimViewWaveCmd	add wave sim:%s\n
HDLSynthTool	None
HDLSynthCmd	
HDLSynthFilePostfix	
HDLSynthInit	
HDLSynthLibCmd	
HDLSynthLibSpec	
HDLSynthTerm	
ReservedWordPostfix	_rsvd
BlockGenerateLabel	_gen
VHDLLibraryName	work
UseSingleLibrary	off
VHDLArchitectureName	rtl
ClockProcessPostfix	_process
ComplexImagPostfix	_im
ComplexRealPostfix	_re
EntityConflictPostfix	_block
InstancePrefix	u_
InstancePostfix	
InstanceGenerateLabel	_gen
OutputGenerateLabel	outputgen
PackagePostfix	_pkg
SplitEntityArch	off
SplitEntityFilePostfix	_entity
SplitArchFilePostfix	_arch
VectorPrefix	vector_of_
ClockInputs	Single
TriggerAsClock	off
ConditionalizePipeline	off
InferControlPorts	off
UseRisingEdge	off
TargetDirectory	hdlsrc
TargetSubdirectory	Model
EDAScriptGeneration	on
AddInputRegister	on

Chapter 6. System Model Configuration

AddOutputRegister	on
AddPipelineRegisters	off
PipelinePostfix	_pipe
InputPort	filter_in
OutputPort	filter_out
FracDelayPort	filter_fd
Name	filter
RemoveResetFrom	None
ResetAssertedLevel	Active-high
ReuseAccum	off
ScaleWarnBits	3
SerialPartition	-1
DALUTPartition	-1
DARadix	2
CoefficientSource	Internal
CoefficientMemory	Registers
InputComplex	off
AddRatePort	off
InputDataType	
GenerateHDLCode	on
GenerateModel	on
GenerateTB	off
GenerateCEGenModel	off
Traceability	off
ResourceReport	off
OptimizationReport	off
ErrorCheckReport	on
HDLGenerateWebview	off
IPCoreReport	off
Recommendations	off
RequirementComments	on
Backannotation	off
HierarchicalDistPipelining	off
PreserveDesignDelays	off
AcquireDesignDelaysForEMLOptimizations	off
ClockRatePipelining	on
CRPWithoutFlattening	on
UseCRPAlternativeStrategy	off

Chapter 6. System Model Configuration

IncreaseCRPBudget	on
AdaptivePipelining	on
MinDelaysRequiredAtLocalMultirateOutput	1
ClockRatePipelineOutputPorts	off
CriticalPathEstimation	off
optimizeserializer	on
shareequalwl	on
sharedmulsign	Signed
MultiplierPromotionThreshold	0
RoutingFudgeFactor	0.5000
OptimizationCompatibilityCheck	off
NumCriticalPathsEstimated	1
CriticalPathEstimationFile	criticalPathEstimated
HardwarePipeliningCharacterizationFile	
HighlightFeedbackLoops	on
HighlightFeedbackLoopsFile	highlightFeedbackLoop
HighlightClockRatePipeliningDiagnostic	on
HighlightClockRatePipeliningFile	highlightClockRatePipelining
DistributedPipeliningBarriers	on
DistributedPipeliningBarriersFile	highlightDistributedPipeliningBarriers
BlocksWithNoCharacterizationFile	highlightCriticalPathEstimationOffendingBlocks
AXIStreamingTransformFeatureControl	off
SerializerRatioThreshold	8192
RetimingCP	off
RetimingCPFile	highlightRetimingCP
ClearHighlightingFile	clearhighlighting
FunctionallyEquivalentRetiming	on
DistributedPipeliningPriority	NumericalIntegrity
RetimingDetails	on
CriticalPathDetails	off
SignalNamesMangling	off
GuidedRetiming	off
LatencyConstraint	0
ReduceMatchingDelays	on
OptimizationData	
CPGuidanceFile	
CPAnnotationFile	
HandleAtomicSubsystem	on

Chapter 6. System Model Configuration

OptimizeMdlGen	on
MulticyclePathInfo	off
MulticyclePathConstraints	off
FloatingPointTargetConfiguration	
GenerateTargetComps	on
NativeFloatingPoint	off
FPToleranceValue	1.0000e-07
FPToleranceStrategy	DEFAULT
nfpLatency	DEFAULT
nfpDenormals	DEFAULT
AlteraBackwardIncompatibleSinCosPipeline	off
FamilyDevicePackageSpeed	
ToolName	
SynthesisToolChipFamily	
SynthesisToolDeviceName	
SynthesisToolPackageName	
SynthesisToolSpeedValue	
SynthesisTool	
SynthesisProjectAdditionalFiles	
SimulationLibPath	
XilinxSimulatorLibPath	
AdderSharingMinimumBitwidth	0
MultiplierSharingMinimumBitwidth	0
MultiplyAddSharingMinimumBitwidth	0
ShareAdders	off
ShareMultipliers	on
ShareMultiplyAdds	on
ShareMATLABBlocks	on
ShareAtomicSubsystems	on
ShareFloatingPointIPs	on
PipelinedSharing	on
OptimizeCRPSSharingRegisters	on
ClockRatePipeliningBudgetCheck	off
EnableFPGAWorkflow	off
FPGAWorkflowParameters	
GainMultipliers	Multiplier
ProductOfElementsStyle	linear
UserComment	

Chapter 6. System Model Configuration

CustomFileHeaderComment	
CustomFileFooterComment	
DateComment	on
SafeZeroConcat	on
SumOfElementsStyle	linear
TargetLanguage	VHDL
Oversampling	1
ClockRatePipeliningFraction	1
Verbosity	1
TestBenchName	filter_tb
MultifileTestBench	off
IgnoreDataChecking	0
TestBenchPostfix	_tb
TestBenchDataPostfix	_data
TestBenchStimulus	
TestBenchUserStimulus	
TestBenchFracDelayStimulus	
TestBenchCoeffStimulus	
TestBenchRateStimulus	
ForceClockEnable	on
MinimizeClockEnables	off
MinimizeGlobalResets	off
NoResetInitializationMode	InsideModule
NoResetInitScript	noresetinitscript.tcl
ComplexMulElaboration	MultiplyAddBlock
FlattenBus	off
TestBenchClockEnableDelay	1
ForceClock	on
ClockHighTime	5
ClockLowTime	5
HoldTime	2
InputDataInterval	0
ForceReset	on
ErrorMargin	4
HoldInputDataBetweenSamples	on
InitializeTestBenchInputs	off
ResetLength	2
TestBenchReferencePostFix	_ref

Chapter 6. System Model Configuration

GenerateValidationModel	off
RAMMappingThreshold	256
MapPipelineDelaysToRAM	off
RemoveRedundantCounters	on
ReplaceUnitDelayWithIntegerDelay	on
ConcatenateDelays	on
MergeDelaysOnFanouts	on
FoldDelaysToConstant	on
RAMArchitecture	WithClockEnable
InlineMATLABBlockCode	off
InlineHDLCode	off
MaskParameterAsGeneric	off
FlattenSharedSubsystems	off
StringTypeSupport	off
BalanceDelays	on
TargetFrequency	0
ExtraEffortMargin	1
MaxOversampling	Inf
MaxComputationLatency	1
MultiplierPartitioningThreshold	Inf
TreatDelayBalancingFailureAs	Error
TransformDelaysWithControlLogic	on
TransformNonZeroInitValDelay	on
DelayElaborationLimit	20
GenerateCoSimBlock	off
HDLCodeCoverage	off
GenerateHDLTestBench	on
GenerateCoSimModel	None
GenerateSVDPIBench	None
SimulationTool	Mentor Graphics Modelsim
CoSimModelSetup	CosimBlockAndDut
SynthesisOnDirective	
SynthesisOffDirective	
LoopUnrolling	off
InlineConfigurations	on
UseAggregatesForConst	off
UseVerilogTimescale	on
Timescale	`timescale 1 ns / 1 ns

Chapter 6. System Model Configuration

VerilogFileExtension	.v
SystemVerilogFileExtension	.sv
VHDLFileExtension	.vhd
CodeGenerationOutput	GenerateHDLCode
GeneratedModelName	
GeneratedModelNamePrefix	gm_
UseDotLayout	off
ShowCodeGenPIR	off
SerializeModel	0
SerializeIO	0
UseSLAutoRoute	on
UseAutoPlace	on
CustomDotPath	
HighlightAncestors	on
HighlightColor	cyan
InitializeBlockRAM	on
InitializeRealPort	off
MapVectorPortToStream	off
UseFileIOInTestBench	on
TurnkeyWorkflow	off
AlteraWorkflow	off
GenerateFILBlock	off
CoSimLibPostfix	_cosim
TestBenchInitializeInputs	off
MinimizeIntermediateSignals	off
GenerateCodeInfo	off
GatewayoutWithDTC	off
IncrementalCodeGenForTopModel	off
HDLWFSmartbuild	on
HDLCodingStandard	None
HDLCodingStandardCustomizations	
ReferenceDesignParameter	
HDLLintTool	None
HDLLintInit	
HDLLintTerm	
HDLLintCmd	
ModulePrefix	
DetectBlackBoxNameCollision	Warning

Chapter 6. System Model Configuration

PIRTB	on
PIRTC	off
EmitNetlist	off
UsePipelinedToolboxFunctions	on
savepirtoscript	off
ConcatenateHDLModules	off
AMS	off
ML2PIR	off
OptimBetweenMATLABAndSimulink	off
EnableTestpoints	off
TraceabilityStyle	Line Level
TreatRealsInGeneratedCodeAs	Error
EnumEncodingScheme	default
BuildToProtectModel	off
OptimizeConstants	on
StreamingMatrix	off
HDLDTO	off

Chapter 7. Glossary

Atomic Subsystem. A subsystem treated as a unit by an implementation of the design documented in this report. The implementation computes the outputs of all the blocks in the atomic subsystem before computing the next block in the parent system's block execution order (sorted list).

Block Diagram. A Simulink block diagram represents a set of simultaneous equations that relate a system or subsystem's inputs to its outputs as a function of time. Each block in the diagram represents an equation of the form $y = f(t, x, u)$ where t is the current time, u is a block input, y is a block output, and x is a system state (see the Simulink documentation for information on the functions represented by the various types of blocks that make up the diagram). Lines connecting the blocks represent dependencies among the blocks, i.e., inputs whose current values are the outputs of other blocks. An implementation of a design described in this document computes a root or atomic system's outputs at each time step by computing the outputs of the blocks in an order determined by block input/output dependencies.

Block Parameter. A variable that determines the output of a block along with its inputs, for example, the gain parameter of a Gain block.

Block Execution Order. The order in which Simulink evaluates blocks during simulation of a model. The block execution order determined by Simulink ensures that a block executes only after all blocks on whose outputs it depends are executed.

Checksum. A number that indicates whether different versions of a model or atomic subsystem differ functionally or only cosmetically. Different checksums for different versions of the same model or subsystem indicate that the versions differ functionally.

Design Variable. A symbolic (MATLAB) variable or expression used as the value of a block parameter. Design variables allow the behavior of the model to be altered by altering the value of the design variable.

Signal. A block output, so-called because block outputs typically vary with time.

Virtual Subsystem. A subsystem that is purely graphical, i.e., is intended to reduce the visual complexity of the block diagram of which it is a subsystem. An implementation of the design treats the blocks in the subsystem as part of the first nonvirtual ancestor of the virtual subsystem (see Atomic Subsystem).

Chapter 8. About this Report

Report Overview

This report describes the design of the Induction_motor system. The report was generated automatically from a Simulink model used to validate the design. It contains the following sections:

Model Version. Specifies information about the version of the model from which this design description was generated. Includes the model checksum, a number that indicates whether different versions of the model differ functionally or only cosmetically. Different checksums for different versions indicate that the versions differ functionally.

Root System. Describes the design's root system.

Subsystems. Describes each of the design's subsystems.

Design Variables. Describes system design variables, i.e., MATLAB variables and expressions used as block parameter values.

System Model Configuration. Lists the configuration parameters, e.g., start and stop time, of the model used to simulate the system described by this report.

Requirements. Shows design requirements associated with elements of the design model. This section appears only if the design model contains requirements links.

Glossary. Defines Simulink terms used in this report.

Root System Description

This section describes a design's root system. It contains the following sections:

Diagram. Simulink block diagram that represents the algorithm used to compute the root system's outputs.

Description. Description of the root system. This section appears only if the model's root system has a Documentation property or a Doc block.

Interface. Name, data type, width, and other properties of the root system's input and output signals. The number of the block port that outputs the signal appears in angle brackets appended to the signal name. This section appears only if the root system has input or output ports.

Blocks. This section has two subsections:

- **Parameters.** Describes key parameters of blocks in the root system. This section also includes graphical and/or tabular representations of lookup table data used by lookup table blocks, i.e., blocks that use lookup tables to compute their outputs.
- **Block Execution Order.** Order in which blocks must be executed at each time step in order to ensure that each block's inputs are available when it executes.

State Charts. Describes state charts used in the root system. This section appears only if the root system contains Stateflow blocks.

Subsystem Descriptions

This section describes a design's subsystems. Each subsystem description contains the following sections:

Checksum. This section appears only if the subsystem is an atomic subsystem. The checksum indicates whether the version of the model subsystem used to generate this report differs functionally from other versions of the model subsystem. If two model checksums differ, the corresponding versions of the model differ functionally.

Diagram. Simulink block diagram that graphically represents the algorithm used to compute the subsystem's outputs.

Description. Description of the subsystem. This section appears only if the subsystem has a Documentation property or contains a Doc block.

Interface. Name, data type, width, and other properties of the subsystem's input and output signals. The number of the block port that outputs the signal appears in angle brackets appended to the signal name. This section appears only if the subsystem is atomic and has input or output ports.

Blocks. Blocks that this subsystem contains. This section has two subsections:

- **Parameters.** Key parameters of blocks in the subsystem. This section also includes graphical and/or tabular representations of lookup table data used by lookup table blocks, blocks that use lookup tables to compute their outputs.
- **Block Execution Order.** Order in which the subsystem's blocks must be executed at each time step in order to ensure that each block's inputs are available when the block executes. This section appears only if the subsystem is atomic. Note: in Acrobat(PDF) reports, the number in square brackets next to the block name is a hyperlink to the block parameter table. The number has no model significance.

State Charts. Describes state charts used in the subsystem. This section appears only if the root system contains Stateflow blocks.

State Chart Descriptions

This section describes the state machines used by Stateflow blocks to compute their outputs, i.e., Stateflow blocks. Each state machine description contains the following sections:

Chart. Diagram representing the state machine.

States. Describes the state machine's states. Each state description includes the state's diagram and diagrams and/or descriptions of graphical functions, Simulink functions, truth tables, and MATLAB functions parented by the state.

Transitions. Transitions between the state machine's states. Each transition description specifies the values of key transition properties. Appears only if a transition has properties that do not appear on the chart.

Junctions. Transition junctions. Each junction description specifies the values of key junction properties. Appears only if a junction has properties that do not appear on the chart.

Events. Events that trigger state transitions. Each event description specifies the values of key event properties.

Data. Data types and other properties of the Stateflow block's inputs, outputs, and other state machine data.

Targets. Executable implementations of the state machine used to compute the outputs of the corresponding Stateflow block.

MATLAB Supporting Functions. List of functions invoked by MATLAB functions defined in the chart.