Automated State Model Generator

Application Notes and Examples

Purdue University
School of Electrical and Computer Engineering
West Lafayette, IN 47907-1285

Prepared by

Juri Jatskevich
Oleg Wasynczuk

December 9, 2001
## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1. Modeling Concept</td>
<td>1</td>
</tr>
<tr>
<td>1.2. Built-in Switch Types</td>
<td>2</td>
</tr>
<tr>
<td>1.3. Interface with Simulink</td>
<td>3</td>
</tr>
<tr>
<td>2. MODELING OF STATIONARY ELECTRIC CIRCUITS</td>
<td>7</td>
</tr>
<tr>
<td>2.1. Second-Order RLC Circuit</td>
<td>7</td>
</tr>
<tr>
<td>2.2. Buck Converter</td>
<td>9</td>
</tr>
<tr>
<td>2.3. Boost Converter</td>
<td>12</td>
</tr>
<tr>
<td>2.4. Buck-Boost Converter</td>
<td>14</td>
</tr>
<tr>
<td>2.5. Cuk Converter</td>
<td>16</td>
</tr>
<tr>
<td>2.6. Single-Phase Diode Rectifier</td>
<td>17</td>
</tr>
<tr>
<td>2.7. Single-Phase Thyristor Rectifier</td>
<td>18</td>
</tr>
<tr>
<td>2.8. Single-Phase Triac Commutator</td>
<td>20</td>
</tr>
<tr>
<td>2.9. Single-Phase Diode H-Bridge</td>
<td>22</td>
</tr>
<tr>
<td>2.10. Single-Phase Thyristor H-Bridge</td>
<td>24</td>
</tr>
<tr>
<td>2.11. Three-Phase Diode Rectifier</td>
<td>25</td>
</tr>
<tr>
<td>2.12. Three-Phase Thyristor Rectifier</td>
<td>27</td>
</tr>
<tr>
<td>2.13. Three-Phase System with D-Y Transformer</td>
<td>28</td>
</tr>
<tr>
<td>2.14. Twelve-Pulse Diode Rectifier with D/Y-D Transformer</td>
<td>30</td>
</tr>
<tr>
<td>3. SYSTEMS WITH ELECTRICAL MACHINES</td>
<td>35</td>
</tr>
<tr>
<td>3.1. Three-Phase Synchronous Machine Rectifier Circuit Model</td>
<td>35</td>
</tr>
<tr>
<td>3.2. Six-Phase Synchronous Machine Rectifier in VBR Form</td>
<td>37</td>
</tr>
<tr>
<td>4. ZONAL ELECTRICAL DISTRIBUTION SYSTEM COMPONENTS</td>
<td>43</td>
</tr>
<tr>
<td>4.1. 15-kW Power Supply</td>
<td>43</td>
</tr>
<tr>
<td>4.2. Ship Service Converter Module</td>
<td>47</td>
</tr>
<tr>
<td>4.3. Constant Power Load</td>
<td>50</td>
</tr>
<tr>
<td>4.4. Ship Service Inverter Module</td>
<td>53</td>
</tr>
<tr>
<td>4.5. Motor Controller</td>
<td>58</td>
</tr>
<tr>
<td>5. REFERENCES</td>
<td>63</td>
</tr>
<tr>
<td>Appendix A: M-File for the Second-Order RLC Circuit</td>
<td>65</td>
</tr>
<tr>
<td>Appendix B: M-File for the Buck Converter</td>
<td>66</td>
</tr>
<tr>
<td>Appendix C: C-File for the Buck Converter</td>
<td>67</td>
</tr>
<tr>
<td>Appendix D: M-File for the Boost Converter</td>
<td>68</td>
</tr>
<tr>
<td>Appendix E: M-File for the Buck-Boost Converter</td>
<td>69</td>
</tr>
<tr>
<td>Appendix F: M-File for the Cuk Converter</td>
<td>70</td>
</tr>
<tr>
<td>Appendix G: M-File for the Single-Phase Diode Rectifier</td>
<td>71</td>
</tr>
<tr>
<td>Appendix H: M-File for the Single-Phase Thyristor Rectifier</td>
<td>72</td>
</tr>
<tr>
<td>Appendix I:</td>
<td>M-File for the Single-Phase Triac-Commutator</td>
</tr>
<tr>
<td>Appendix J:</td>
<td>M-File for the Single-Phase Diode H-Bridge.</td>
</tr>
<tr>
<td>Appendix K:</td>
<td>M-File for the Single-Phase Thyristor H-Bridge</td>
</tr>
<tr>
<td>Appendix L:</td>
<td>M-File for the Three-Phase Diode Rectifier</td>
</tr>
<tr>
<td>Appendix M:</td>
<td>M-File for the Single-Phase Thyristor Rectifier</td>
</tr>
<tr>
<td>Appendix N:</td>
<td>M-File for the Three-Phase Y-D Transformer</td>
</tr>
<tr>
<td>Appendix O:</td>
<td>M-File for the 12-Pulse Diode Rectifier with D-Y/D Transformer</td>
</tr>
<tr>
<td>Appendix P:</td>
<td>M File for the Three-Phase Synchronous Machine</td>
</tr>
<tr>
<td>Appendix Q:</td>
<td>C File for the Three-Phase Synchronous Machine</td>
</tr>
<tr>
<td>Appendix R:</td>
<td>M File for the Six-Phase Synchronous Machine Rectifier in VBR Form</td>
</tr>
<tr>
<td>Appendix S:</td>
<td>M-File for the 15-kW Power Supply</td>
</tr>
<tr>
<td>Appendix T:</td>
<td>M-File for the Ship Service Converter Module</td>
</tr>
<tr>
<td>Appendix U:</td>
<td>M-File for the Constant Power Load</td>
</tr>
<tr>
<td>Appendix V:</td>
<td>M-File for the Ship Service Inverter Module</td>
</tr>
<tr>
<td>Appendix W:</td>
<td>M-File for the Motor Controller Inverter</td>
</tr>
<tr>
<td>Appendix X:</td>
<td>M-file Containing Induction Motor Parameters</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

The Automated State Model Generator (ASMG) [1] is a powerful tool for modeling and analyzing power-electronic-based circuits and energy conversion devices. The ASMG enables the rapid development of models of switched circuits and systems without requiring the user to derive the corresponding differential equations. In this approach, a state-space model is established automatically based upon a branch list that defines the circuit. For systems with switches and/or variable parameters, the state model is automatically re-assembled and/or updated as necessary. The time-domain response is obtained by numerical integrating the state equation using appropriate ODE solvers. The resulting models can be used to study and analyze the dynamic characteristics of a wide variety of electrical systems.

In these application notes, the ASMG is applied to numerous power-electronic-based components and subsystems to serve as an example for model developers and to provide an indication of the capabilities of the ASMG. Each section includes a description of a power-electronic-based component or subsystem. The ASMG implementation of that component is then set forth and sample studies are provided. All computer files, which can be executed “as is” or modified to suit other applications, are provided in a companion CD. Prior to describing the example systems, it is instructive to briefly described the underlying concepts behind the ASMG.

1.1. Modeling Concept

The approach underlaying the ASMG consists of grouping circuit elements into structurally compact branches as those shown in Fig. 1.1-1. The two branches in Fig. 1.1-1 (a) and (b) are topological duals. Therein, $v_{br}$ is the branch voltage, $i_{br}$ is the branch current, $r_{br}$ is the resistance, $L_{br}$ is the inductance, $C_{br}$ is the capacitance, $e_{br}$ is the branch voltage source, and $j_{br}$ is the current source. Simple resistors, inductors, or sources can also be represented by setting the appropriate parameters of the elementary branch types to zero. All parameters can be constant, time varying, or depend upon other variables (inductive branch currents or capacitive branch voltages). In order to model switched electric circuits, the ASMG provides a switch branch as shown in Fig. 1.1-2.
Therein, \( u_{br} \) is a control variable that is used to open or close the switch. With appropriate switching logic, more complicated circuit elements such as transistors, diodes, thyristors, etc., can be represented by appropriately combining elementary branch models with possibly variable parameters and dependent sources.

### 1.2. Built-in Switch Types

Four built-in switch types, each implementing specific switching logic, are implemented in the present ASMG release. These switch types can represent the idealized characteristics of all of the common solid-state switching devices such as diodes, thyristors, transistors (MOSFET, BJT, IGBT, etc.), triacs, etc. In general, the built-in switching logic does not permit the opening of switches that would cause discontinuities of currents in inductors and/or current sources, as well as closing of switches that would cause discontinuities of capacitor voltages and/or voltage sources. Such attempts would violate Kirchhoff’s current law (KCL) and/or Kirchhoff’s voltage law (KVL), and therefore are not allowed. If the violation of KCL, KVL, and/or energy conservation principles is detected, the ASMG provides an appropriate error message. The logic of the built-in switch types is described below. Additional information on built-in switch types is provided in the ASMG Reference Manual [1].

---

**Fig. 1.1-1.** Branch models: (a) inductive, (b) capacitive.

**Fig. 1.1-2.** Switch branch.
Type-1: Unlatched bidirectional switch (UBS)

A switch of this type can conduct current in either direction when the switching control signal \( u_{br} > 0 \), and block positive or negative voltages otherwise. The switch can be closed or opened at any instant of time by setting the control variable \( u_{br} \) to a positive or negative value, respectively, subject to KCL, KVL, and energy conservation principles.

Type-2: Unlatched Unidirectional Switch (UUS)

A switch of this type is similar to the UBS with the exception that it can conduct current only in the positive direction. In any case, \( i_{br} \geq 0 \). The switch closes when both \( u_{br} \) and \( v_{br} \) are positive. The switch opens when \( u_{br} \) or \( i_{br} \) become zero or negative.

Type-3: Latched Bidirectional Switch (LBS)

A switch of this type is similar to the UBS with the exception that it can be opened only at current-zero-crossings. That is, the switch can be closed by setting the control signal \( u_{br} > 0 \). The switch opens when the control signal is removed (set negative) and the current crosses zero.

Type-4: Latched Unidirectional Switch (LUS)

A switch of this type is similar to the LBS with the exception that it can conduct current only in the positive direction. That is, the switch can be closed only when \( v_{br} > 0 \) by setting the control signal \( u_{br} \) to a positive value. The switch opens when the control signal \( u_{br} \) is removed and the branch current \( i_{br} \) crosses zero going from positive-to-negative.

1.3. Interface with Simulink

The ASMG has a flexible textual interface convenient for modeling complex power-electronic-based circuits and systems. The ASMG can be readily interfaced with other differential-equation-based simulation languages such as Simulink [3], which is a GUI-based language for dynamic system simulation. For use with Simulink, the ASMG is supplied to the user in the form of a masked CMEX S-function (files
asmg_system.mdl, asmgsfun.c), a file asmg_var_par.c for implementing variable parameters, a collection of supporting M-files that implement the branch and initialization statements, and the library asmgsim.lib. Additional information regarding installation of the ASMG and its organization within the Matlab/Simulink environment is provided in the ASMG reference manual [1].

After the ASMG is installed, it appears as a toolbox that is visible from the Simulink Library Browser as shown in Fig. 1.3-1. Thereafter, the ASMG block can be dragged from the Library Browser and dropped into a new model. An example of a new Simulink model is shown in Fig. 1.3-2. The corresponding dialog box, which can be displayed by double-clicking the ASMG_System block, is shown in Fig. 1.3-3. As shown in Fig. 1.3-2, the ASMG block has the three input and two output ports:

- \( E_s \) - vector of external voltage sources
- \( J_s \) - vector of external current sources
- \( U \) - vector of switching control signals
- \( I_{br} \) - vector of branch currents
- \( V_{br} \) - vector of branch voltages

The variables in each port appear ordered in the same way as they are declared in the branch list. For example, the output branch currents and voltages have the same order as the branch numbering in initial file. However, since not all branches are expected to

![Simulink Library Browser](image_url)

Fig. 1.3-1. Simulink Library Browser.
have external sources, the variables in \( \textbf{E} \) and \( \textbf{J} \) appear in the same relative order of occurrence as their respective branches within the branch list. Similarly, the control signals \( \textbf{U} \) appear in the same relative order as the switch branches.
In the dialog box, the user must provide an initial M-file that describes circuit topology and defines its parameters. The initial file must be a Matlab function that returns the ASMG handle - an integer. The return value should be 1 for the first ASMG block, and respectively increase for all subsequent blocks. If the Simulink model includes more than one ASMG block, each should have a unique handle. Inside the initial file, the circuit is defined in terms of branch and declaration statements that are available as Matlab functions. The user also has a choice of state variables (currents/fluxes, voltages/charges). The verbose level allows the user to specify the type of messages that are displayed during the simulation. For models with variable parameters, those parameters must be computed and updated at run-time inside the C-file `asmg_var_par.c`. If this option is used, the model should be recompiled each time a change to `asmg_var_par.c` is made.
2. MODELING OF STATIONARY ELECTRIC CIRCUITS

In this Chapter, example Simulink models are described that demonstrate the use of ASMG for modeling stationary electrical circuits.

2.1. Second-Order RLC Circuit

As a first example, let us consider a simple RLC circuit shown in Fig. 2.1-1. In the given example, it is assumed that a signal generator is producing a square-wave output voltage with frequency \( f_s = 15 \text{ Hz} \) and \( V_{\text{peak}} = 10 \text{ V} \). Other parameters are: \( L = 4.0 \text{ mH} \), \( R_L = 0.002 \Omega \), \( C = 200 \mu\text{F} \), and \( R_C = 20 \Omega \).

![Fig. 2.1-1. Second-order RLC circuit.](image)

In order to implement this circuit, it is sufficient to use only two branches Fig. 1.1-1 (a) and (b). The user can go to the Matlab window and under option FILE, open a new M-file (the same new file may also be created using any text editor). This M-file defines an M-function that returns an integer corresponding the ASMG instance handle. This function also defines the parameters and topology of the circuit in Fig. 2.1-1. The corresponding file is given in Appendix A. It should be noted that the file and the function must be given the same name. In this case, since the function is named \texttt{rlc
circuit}, the file is saved as \texttt{rlc_circuit.m}.

At this point, the user should create a new Simulink model and drag-and-drop the \texttt{ASMG\_System} block into the new model. After that, the signal generator and scope may be added from the standard Simulink Library. The overall model should look similar to that shown in Fig. 2.1-2. Since in the given model there are no current sources and/or switches, the input ports \texttt{Js} and \texttt{U} can be connected to a dummy constant in order to avoid Simulink warnings. After the model is wired-up, the user can double-click on the \texttt{ASMG\_System} block and enter the name of initial file and other model parameters into
appropriate fields of the dialog box. Since, in the given example, there is only one inductor and one capacitor, only one state variables is expected in each of the inductive and capacitive subnetworks. If the project directory does not include the *asmgsfun.dll* file, it is necessary to re-compile the model by checking the appropriate box in the dialog window. After the *asmgsfun.dll* is created, the user can execute the model. Running the model Fig. 2.1-2 for 0.1 s produces the results shown in Fig. 2.1-3.

![ASMG model of the RLC example circuit.](image)

![Computer study for the RLC circuit.](image)
2.2. Buck Converter

As a next example, let us consider a buck converter composed of ideal circuit elements. The corresponding circuit diagram is shown in Fig. 2.2-1. This type of converter, depending on the duty cycle \( d \), produces a lower average output voltage than the input voltage \( V_{dc} \). In the given example, it is assumed that the switching frequency is \( f_{sw} = 20 \) kHz and converter is operating with the fixed duty cycle \( d = 0.2 \). The other converter parameters are: \( V_{dc} = 500 \) V, \( L = 1.0 \) mH, \( C_f = 50 \) \( \mu \)F, and the resistive load \( R_{load} = 200 \) \( \Omega \).

Before modeling a circuit using the ASMG, it is useful to redraw the circuit diagram using branches that are given in Fig. 1.1-1 and Fig. 1.1-2. In doing so, one should attempt to minimize the overall number of branches, as well as pay attention to the polarity of switches and branches with external sources. In the given converter, \( C_f \) and \( R_{load} \) may be represented by a single capacitive branch Fig. 1.1-1 (b). The final circuit diagram that illustrates the branch and node numbering is shown in Fig. 2.2-2. The UU switch Type-2 is used to represent the diode as well as the controllable switch. The corresponding M-file is given in Appendix B. It should be noted that in order to allow a change in load, the resistance of branch 5 is declared variable. Thereafter, the value of this resistance can

![Fig. 2.2-1. Buck converter.](image)

![Fig. 2.2-2. Buck converter circuit diagram showing branch and node numbering.](image)
be changed in the `asmg_var_par.c` file. At this point, the user can build the model using the `ASMG_System` block. In the associated dialog box, the user has an option of specifying a non-zero initial conditions for the states. By default, the state variables are inductor currents and capacitor voltages. If desired, the state variables may be fluxes and charges (see [1]). The final Simulink model is shown in Fig. 2.2-3.

In the following computer study, it is assumed that initial inductor current is zero and initial capacitor voltage is 180 Volts. The model is started with the given initial conditions. At $t = 0.0005 \text{ s}$, the load resistance is changed to 10 $\Omega$ and the model is continued to run until $t = 0.004 \text{ s}$. The change of load is implemented inside `asmg_var_par.c` file that is given in Appendix C. The calculated inductor current $I_{br}(4)$ and load voltage $V_{br}(5)$ are plotted in Fig. 2.2-4. Using the Simulink Scope, the user can plot any other branch voltage and current. Based on the results in Fig. 2.2-4, it can be noted that prior to the step change in load, the converter operates in a discontinuous-conduction mode. The corresponding sequence of topological states may be identified.

![Simulink model of the buck converter.](image-url)
based on the messages in the ASMG window when the verbose level is set to 3. For example, during the discontinuous-conduction mode, the sequence of topologies includes a switching cycle $S=10$, $S=01$, and $S=00$, which indicates the existence of time interval where neither the diode or switch conduct current. The ASMG window with corresponding messages is shown in Fig. 2.2-5. Based on these messages, all topological states as well as corresponding time intervals can be readily identified. For the continuous-conduction mode, the messages are shown in Fig. 2.2-6, from which it can be seen that the topology cycle contains only two states $S=10$ and $S=01$, and the respective time intervals correspond to the given switching frequency and duty cycle $d$. Other useful information that is provided is the number of state variables in the minimal state model of inductive and capacitive sub-circuits for the given topological state. For example, $NLCA(0,1,0)$ implies that inductive subnetwork currently does not have a state, whereas capacitive subnetwork has one state variable. The third number tells how many branches form a so-called algebraic subnetwork that does not contain inductors or capacitors.

Fig. 2.2-4. Buck converter inductor current and output voltage transients.
2.3. Boost Converter

The boost converter is capable of producing an average output voltage larger than the input voltage $V_{dc}$. A circuit diagram of the boost converter composed of ideal circuit elements is shown in Fig. 2.3-1. In this example, it is assumed that the switching frequency is $f_{sw} = 20$ kHz and converter is operating with the fixed duty cycle $d = 0.2$. Other converter parameters are: $V_{dc} = 500$ V, $L = 1.0$ mH, $C_f = 50$ µF, and $R_{load} = 700$ Ω. The numbering of nodes and branches is shown in Fig. 2.3-2.

As before, the UU switch type is used to represent the diode as well as the controllable switch. The corresponding M-file is given in Appendix D. In order to implement a change in load, the resistor of branch 5 is declared as a variable parameter in the M-file; whereas, the actual value is changed in the corresponding `asmg_var_par.c` file.
Although the branch circuits Fig. 2.2-2 and Fig. 2.3-2 are similar, the position of inductor and switch branches is different. In both cases, it is important to make sure that the polarity of switches and branches with sources are in respective order.

In the following computer study, it is assumed that initial inductor current is zero and initial capacitor voltage is 735 Volts. At $t = 0.0005$ s, the load resistance is changed to 100 Ω and the model is continued to run until $t = 0.004$ s. The resulting inductor current $I_{br}(4)$ and load voltage $V_{br}(5)$ are plotted in Fig. 2.3-3. Similar observations regarding the mode of operation can be made based on the plot of $I_{br}(4)$ as well as the messages in ASMG window. In particular, during light load conditions, the converter operates in a discontinuous mode. In this mode, there exists a time interval corresponding to the topological state $S=00$ during which both diode and switch are open and the inductor current is zero. On the other hand, after the load is increased, the converter goes into a continuous conduction mode that is composed of only two topologies $S=10$ and $S=01$ for which the time intervals directly correspond to the duty cycle $d$. 

---

Fig. 2.3-1. Boost converter.

Fig. 2.3-2. Boost converter circuit diagram showing branch and node numbering.
2.4. Buck-Boost Converter

A circuit diagram of a buck-boost converter is shown in Fig. 2.4-1. As the name implies, depending on the duty cycle, the average output voltage of this converter may be lower or higher than the respective input voltage. In the model considered herein, it is assumed that the switching frequency is \( f_{sw} = 20 \text{ kHz} \) and that converter is operating with the fixed duty cycle \( d = 0.2 \). Other converter parameters are: \( V_{dc} = 500 \text{ V} \), \( L = 1.0 \text{ mH} \), \( C_f = 50 \mu\text{F} \), and \( R_{\text{load}} = 200 \Omega \).

The numbering of nodes and branches is shown in Fig. 2.4-2. As before, it is important to make sure that the polarity of switches and branches with sources are in respective order. Also, since the output voltage of this converter is negative, the polarity of the branch representing the load is reversed. The UU switch type is used to represent the

![Fig. 2.4-1. Buck-boost converter.](image)
diode and controllable switch. The M-file is given in Appendix E. The load change is implemented in the corresponding `asmg_var_par.c` file, which is similar to the C-files in the two previous models (Appendix C).

In the following computer study, it is assumed that initial inductor current is zero and initial capacitor voltage is 223 Volts. At $t = 0.0005$ s, the load resistance is changed to $20 \Omega$ and the model is continued to run until $t = 0.004$ s. The resulting inductor current $I_{br}(3)$ and load voltage $V_{br}(5)$ are plotted in Fig. 2.4-3. From the messages in ASMG window and from the plot of $I_{br}(3)$ in Fig. 2.4-3, the converter changes mode from discontinuous to continuous.

Fig. 2.4-2. Buck-Boost converter circuit diagram showing branch and node numbering.

![Buck-Boost converter circuit diagram](image)

**Fig. 2.4-3.** Buck-boost converter inductor current and output voltage transients.
2.5. Cuk Converter

A circuit diagram of a Cuk converter is shown in Fig. 2.5-1. Depending on the duty cycle, the average output voltage of this converter may be lower or higher than the respective input voltage. In the model considered herein, it is assumed that the switching frequency is $f_{sw} = 20$ kHz, and the other converter parameters are: $V_{dc} = 500$ V, $L_1 = L_2 = 1.0$ mH, $C_1 = C_2 = 50$ μF, and $R_{load} = 200$ Ω. The numbering of nodes and branches is shown in Fig. 2.5-2. As before, it is important to make sure that the polarity of switches and branches with sources are in respective order. The corresponding M-file is given in Appendix F. The UU switch type is used to represent the diode and controllable switch.

![Fig. 2.5-1. Cuk converter.](image)

![Fig. 2.5-2. Cuk converter circuit diagram showing branch and node numbering.](image)

In the following computer study, it is assumed that the model is started with zero initial conditions for all inductor currents and capacitor voltages, and the converter duty cycle is $d = 0.2$. At $t = 0.1$ s, the duty cycle is changed to 0.5, and the model is continued to run until $t = 0.2$ s. The resulting capacitor voltages $V_{br}(4)$ and $V_{br}(7)$ are plotted in Fig. 2.5-2. The user can view and plot any other branch current or voltage. As
can be expected, due to the additional inductor and capacitor with no direct damping, the converter exhibits oscillatory behavior. Another fact about the Cuk converter is that the voltage on capacitor \( V_{br}(4) \), is greater than the output voltage \( V_{br}(7) \).

![Graph showing the capacitor voltages transients](image1.png)

![Graph showing the capacitor voltages transients](image2.png)

Fig. 2.5-3. Cuk converter capacitor voltages transients.

### 2.6. Single-Phase Diode Rectifier

A circuit diagram of the single-phase diode rectifier is shown in Fig. 2.6-1. The magnitude of the output voltage ripple depends on the filtering capacitor and the load. In the example considered, a sinusoidal source with \( f_s = 60 \text{ Hz} \) and \( V_a = 500 \text{ V}_{\text{peak}} \) is assumed. The other rectifier parameters are: \( R_s = 1.0 \text{ \Omega} \), \( L_s = 0.1 \text{ mH} \), \( C_f = 500 \mu\text{F} \), and \( R_{\text{load}} = 200 \text{ \Omega} \). It can be noted that only three branches are needed in order to represent the given rectifier. The corresponding M-file is given in Appendix G. The change in load is implemented similar to that shown in Appendix C.

![Circuit diagram of the single-phase diode rectifier](image3.png)

Fig. 2.6-1. Single-phase diode rectifier.
In the following computer study, it is assumed that the model is started with zero initial conditions for inductor current and capacitor voltage. At $t = 0.1$ s, the load resistor is step-changed to $R_{\text{load}} = 50 \, \Omega$, after which the model is continued to run until $t = 0.2$ s. The resulting traces of the diode current $I_{br\,(2)}$, diode voltage $V_{br\,(2)}$, and the load voltage $V_{br\,(3)}$ are plotted in Fig. 2.6-2. As can be expected, the output voltage ripple increases with the load current.

2.7. Single-Phase Thyristor Rectifier

The single-phase thyristor-controlled rectifier considered herein is very similar to the diode rectifier considered previously, except that $R_{\text{load}} = 100 \, \Omega$ and that instead of a diode a thyristor is used. Other circuit parameters are assumed to be the same as in the previous example of a single-phase diode rectifier. The corresponding circuit diagram is shown in Fig. 2.7-1. By controlling the thyristor firing angle, it is possible to regulate the average output voltage. As in the previous case, the output voltage ripple depends on the filtering capacitor and load current. Depending upon the implementation of firing signal, it is possible to use built-in UU or LU switch types (Chapter 1). If the UU switch is used, it is important to maintain a positive value of control signal all the way until the current
zero-crossing. On the other hand, if the LU switch is used, the control signal can be removed right after the switch is closed. In this case, the built-in logic will open the switch at the next current zero-crossing. The corresponding M-file is given in Appendix H. In the given implementation, the firing signal is generated using standard Simulink blocks. Thus, by changing the firing angle $\alpha$, the negative-to-positive zero crossing of the variable $u$ is spaced in time from the corresponding zero crossing of the variable $V_{br(2)}$.

In the following computer study, it is assumed that the model is started with zero initial conditions and no delay in thyristor firing angle. In this mode, the thyristor operates similar to diode in the previous rectifier circuit. At $t = 0.07\ s$, the delay angle $\alpha$ is step-changed to $90^\circ$, after which the model is continued to run until $t = 0.2\ s$. The resulting thyristor current $I_{br(2)}$, thyristor voltage $V_{br(2)}$, and load voltage $V_{br(3)}$ are plotted in Fig. 2.7-2. As can be expected, the average output voltage decreases. Using this control, the average output voltage may be regulated from full to zero by controlling the delay angle $\alpha$. 

Fig. 2.7-1. Single-phase thyristor rectifier.
2.8. Single-Phase Triac Commutator

The single-phase triac commutator is somewhat similar to the thyristor rectifier considered previously, except that instead of a thyristor, a triac is used. The corresponding circuit diagram is shown in Fig. 2.8-1. The difference between triac and thyristor is that the triac can conduct current in either direction. Indeed, the triac could be implemented by two thyristors connected in parallel in opposite directions. Since the load current is allowed to be positive as well as negative, the capacitor is not needed. All other circuit parameters are assumed to be the same as those in the previous case of a single-phase thyristor rectifier. The triac may be used to regulate the effective or the rms voltage or current in AC systems. As in the previous case, the firing signal is required only to activate the device. The triac becomes open at the next current zero crossing. Thus, in order to implement the triac, the LB switch Type-3 (Chapter 1) is used. The corresponding M-file is given in Appendix I. In the given implementation, the firing signal is generated using the Pulse Generator in combination with Variable Transport Delay, which are standard Simulink blocks. By changing \textit{alpha} from zero to 180 degrees, the firing of the triac is delayed by the corresponding angle.
In the following computer study, it is assumed that the model is started with zero initial conditions and 5 electrical degrees delay in firing angle \( \alpha \). At \( t = 0.05 \) s, the firing angle \( \alpha \) is step-changed to 90 degrees, and the model is continued to run until \( t = 0.1 \) s. At \( t = 0.1 \) s, the firing angle \( \alpha \) is step-changed to 175 degrees, and the model is continued to run until \( t = 0.15 \) s. The resulting triac current \( I_{br}(2) \), triac voltage \( V_{br}(2) \), and load voltage \( V_{br}(3) \) are plotted in Fig. 2.8-2. As shown, when the triac operates with a small firing delay, it acts almost like a closed switch supplying nearly full voltage to the load. When \( \alpha \) is changed to 90 degrees, about half of the voltage \( V_{br}(3) \) and current \( I_{br}(2) \) is chopped. The effective output voltage can be reduced even further by increasing the delay in firing. When \( \alpha \) is set to 175 degrees, the output voltage becomes very small.

Fig. 2.8-2. Voltage and current waveforms for the triac commutator.
2.9. Single-Phase Diode H-Bridge

The circuit diagram of a single-phase diode H-bridge is shown in Fig. 2.9-1. For comparison, the circuit parameters are assumed to be the same as in the case of a single-phase thyristor rectifier. That is \( f_s = 60 \text{ Hz} \), \( V_a = 500 \text{ V}_{\text{peak}} \), \( R_s = 1.0 \text{ } \Omega \), \( L_s = 0.1 \text{ mH} \), \( C_f = 500 \mu\text{F} \), and \( R_{\text{load}} = 200 \text{ } \Omega \).

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{Fig2.9-1.png}
\caption{Single-phase diode H-bridge.}
\end{figure}

The numbering of nodes and branches is shown in Fig. 2.9-2. As before, it is important to make sure that the polarity of switch branches are in respective order. The UU switch Type-2 is used to represent all diodes. The corresponding M-file is given in Appendix J. The load change is implemented in the corresponding `asmg_var_par.c` file that is similar to that shown in Appendix C.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{Fig2.9-2.png}
\caption{Single-phase diode H-bridge diagram showing branch and node numbering.}
\end{figure}

In the following computer study, it is assumed that the model is started with zero initial conditions. At \( t = 0.1 \text{ s} \), the load resistor is step-changed to \( R_{\text{load}} = 50 \text{ } \Omega \), after which the model is continued to run until \( t = 0.2 \text{ s} \). In order to see the effect of filtering, the same computer study was implemented with and without the capacitor. The resulting source current \( I_{br}(1) \), load current \( I_{br}(6) \), and load voltage \( V_{br}(6) \) are plotted in
Fig. 2.9-3 and Fig. 2.9-4, respectively. As it can be observed in Fig. 2.9-3, both positive and negative half-cycles are rectified. Because of this, the frequency of the output voltage ripple is twice that of the source frequency. When the capacitor is used, the magnitude of the voltage ripple in Fig. 2.9-4 is significantly less as compared to the one diode rectifier with the same capacitor.

Fig. 2.9-3. Current and voltage waveforms for the diode H-bridge without capacitor.

Fig. 2.9-4. Current and voltage waveforms for the diode H-bridge with capacitor.
2.10. Single-Phase Thyristor H-Bridge

The thyristor H-bridge is very similar to the corresponding diode H-bridge presented in the previous example. For comparison of the two circuits, the parameters are assumed to be the same. For consistency, the corresponding circuit diagram is shown in Fig. 2.10-1. The numbering of nodes and branches is the same as that in Fig. 2.9-2. However, in this case the LU switch Type-4 is used to represent the thyristors. The corresponding M-file is given in Appendix K. The firing signals are implemented using standard Simulink blocks.

In the following computer study, it is assumed that the model is started with zero initial conditions and that the thyristors are fired with a 10-degree delay. At $t = 0.05 \text{s}$, the firing angle $\alpha$ is set to 150 degrees, after which the model is continued to run until $t = 0.2 \text{s}$. The resulting traces of the source current $I_{br}(1)$, the load current $I_{br}(6)$, and load voltage $V_{br}(6)$ are plotted in Fig. 2.10-2. As shown in Fig. 2.10-2, the output voltage can be controlled by adjusting the firing angle.
2.11. Three-Phase Diode Rectifier

A circuit diagram of the three-phase diode rectifier is shown in Fig. 2.11-1. Due to the three-phase source, the output voltage ripple is significantly smaller than that in the single-phase diode H-bridge. In the example considered, a balanced three-phase source with $f_s = 60$ Hz and $500 \text{ V}_{\text{peak}}$ is assumed. The other parameters are: $R_s = 1.0 \Omega$, $L_s = 1.0 \text{ mH}$, $R_{\text{load}} = 10 \Omega$, and $L_{\text{load}} = 4.0 \text{ mH}$. The numbering of nodes and branches is shown in Fig. 2.11-2. The corresponding M-file showing the branch list is given in Appendix L. Similar to previous models, the diodes are represented by UU switch Type-2 branches with a positive control signal that is held constant.
In the following computer study, it is assumed that the model is started with zero initial conditions. At $t = 0.05 \text{s}$, the load resistance is step-changed to $0.5 \, \Omega$, after which the model is continued to run until $t = 0.1 \, \text{s}$. The resulting phase $a$ source voltage $V_{br(1)}$, source current $I_{br(1)}$, load voltage $V_{br(10)}$, and load current $I_{br(10)}$ are plotted in Fig. 2.11-3. As shown in Fig. 2.11-3, after the change in load, the source current becomes almost continuous, which indicates a different operational mode. The change in operational mode can also be detected by setting the verbose level to 3 and observing the messages that are printed to the ASMG window.

Fig. 2.11-2. Branch and node numbering for the three-phase diode rectifier.

Fig. 2.11-3. Voltage and current waveforms for the three-phase diode rectifier.
2.12. Three-Phase Thyristor Rectifier

A circuit diagram of the three-phase thyristor rectifier is shown in Fig. 2.12-1. Due to the three-phase source, the output voltage ripple is also significantly smaller than that in the case of a single-phase thyristor H-bridge. In the example considered, a balanced three-phase source with $f_s = 60$ Hz and $V_{\text{peak}} = 500$ V is assumed. The other parameters are:

\[ R_s = 1.0 \, \Omega, \quad L_s = 1.0 \, \text{mH}, \quad R_{\text{load}} = 20 \, \Omega, \quad \text{and} \quad C_f = 500 \, \mu\text{F}. \]

![Fig. 2.12-1. Three-phase thyristor rectifier.](image)

The numbering of nodes and branches is the same as in Fig. 2.11-2. The corresponding M-file is given in Appendix M. Similar to previous models, the thyristors are represented by LU switch Type-4 branches. The thyristor firing signals are implemented using standard Simulink blocks that produce cosine waves with the appropriate shifts in phase.

In the following computer study, it is assumed that the model is started with zero initial conditions and no delay firing angle. At $t = 0.05$ s, the delay angle $\alpha$ is step-changed to 60 degrees, after which the model is continued to run until $t = 0.1$ s. The resulting phase $a$ source voltage $V_{\text{br}(1)}$, source current $I_{\text{br}(1)}$, load voltage $V_{\text{br}(10)}$, and load current $I_{\text{br}(10)}$ are plotted in Fig. 2.12-2. As shown in Fig. 2.12-2, the output voltage can be regulated by adjusting the delay angle $\alpha$. However, at the same time, the harmonic content of the source currents and voltages also changes.
2.13. Three-Phase System with Δ-Y Transformer

The ASMG can be used to model circuits with coupling of arbitrary complexity. Examples of such circuits may be systems with transformers where the windings are coupled magnetically. Similarly, circuits in which mutual capacitance among certain elements must be taken into consideration can also be readily modeled using the ASMG. An example three-phase system with Δ-Y transformer is shown in Fig. 2.13-1. A symmetrical three-phase source with \( f_s = 60 \text{ Hz}, V = 500 \text{ V}_{\text{peak}}, R_s = 1.0 \Omega, \) and \( X_s = 2.0 \Omega \) is assumed. The transformer parameters are summarized in Table 2.13-1. The unbalanced three-phase load \( R_{la} = 200.0 \Omega, X_{la} = 1.0 \Omega, R_{lb} = 300.0 \Omega, X_{lb} = 10.0 \Omega, R_{lc} = 150.0 \Omega, \) and \( X_{lc} = 15.0 \Omega \) is considered. In addition, a phase-to-ground fault with \( R_{ft} = 1.0 \Omega \) and \( X_{ft} = 1.0 \Omega \) may be applied to the secondary side. The fault is implemented as a separate branch that may be connected in parallel with the load in phase.
The branch and node numbering is shown in Fig. 2.13-2. The corresponding M-file is given in Appendix N. Note that all reactances are converted into inductances, which are then entered using appropriate branch statements.

In the following computer study, it is assumed that the model is started with zero initial conditions. At $t = 0.05\,\text{s}$, the fault is applied to by closing the switch. At $t = 0.1\,\text{s}$, the fault is removed, after which the model is continued to run until $t = 0.15\,\text{s}$. The resulting phase $a$ load current $\text{Ibr(10)}$, fault current $\text{Ibr(14)}$, load voltages $\text{Vbr(10, 11, 12)}$, and source voltages $\text{Vbr(1, 2, 3)}$ are plotted in Fig. 2.13-3. As shown in Fig. 2.13-3, when the fault is applied, the load voltage in phase $a$ and load current $\text{Ibr(10)}$ drop significantly. Moreover, due to the Δ-Y connection, the appropriate source voltages also decrease for the duration of the fault.
2.14. Twelve-Pulse Diode Rectifier with ∆/Y-∆ Transformer

A circuit diagram of a 12-pulse diode rectifier is shown in Fig. 2.14-1. The main purpose of using two three-phase rectifiers is to reduce the voltage and current ripple on the dc side. From previous examples, a single three-phase diode rectifier produces an output voltage with six pulses per ac cycle, each pulse lasting 60°. In order to make the second rectifier produce a 6-pulse ripple that is out-of-phase with respect to the first rectifier, the ac voltages of the second rectifier must be shifted by 30°. In the rectifier design shown in Fig. 2.14-1, the two sets of secondary voltages are spaced 30° apart due to the ∆/Y-∆ transformer, which results in the 12-pulse output ripple. In the given example system, a symmetrical three-phase source with \( f_s = 60 \text{ Hz}, \ V = 500 \text{ V}_{\text{peak}}, \ R_s = 1.0 \ \Omega, \) and \( X_s = 2.0 \ \Omega \) is assumed. The transformer parameters are summarized in Table 2.14-1. The output of the rectifier is connected to an inductive load with \( R_l = 20.0 \ \Omega \) and \( X_l = 1.0 \ \Omega \). As before, it is important to make sure that the polarity of switch branches

![Fig. 2.13-3. Voltage and current waveforms for the three-phase ∆-Y transformer system.](image)
Table 2.13-1: Transformer parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Branch(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>primary side leakage reactance</td>
<td>$X_{dl}$</td>
<td>3.0 Ω</td>
<td>4, 5, 6</td>
</tr>
<tr>
<td>primary side magnetizing reactance</td>
<td>$X_{dm}$</td>
<td>700.0 Ω</td>
<td>4, 5, 6</td>
</tr>
<tr>
<td>primary side mutual reactance between phases</td>
<td>$X_{dd}$</td>
<td>-350.0 Ω</td>
<td>4-5, 5-6, 6-4</td>
</tr>
<tr>
<td>primary side winding resistance</td>
<td>$R_d$</td>
<td>1.0 Ω</td>
<td>4, 5, 6</td>
</tr>
<tr>
<td>secondary side leakage reactance</td>
<td>$X_{yl}$</td>
<td>2.0 Ω</td>
<td>7, 8, 9</td>
</tr>
<tr>
<td>secondary side magnetizing reactance</td>
<td>$X_{ym}$</td>
<td>700.0 Ω</td>
<td>7, 8, 9</td>
</tr>
<tr>
<td>secondary side mutual reactance between phases</td>
<td>$X_{yy}$</td>
<td>-350.0 Ω</td>
<td>7-8, 8-9, 9-7</td>
</tr>
<tr>
<td>secondary side winding resistance</td>
<td>$R_y$</td>
<td>1.0 Ω</td>
<td>7, 8, 9</td>
</tr>
<tr>
<td>primary-to-secondary mutual reactance between windings of the same phase</td>
<td>$X_{dym}$</td>
<td>700.0 Ω</td>
<td>4-7, 5-8, 6-9</td>
</tr>
<tr>
<td>primary-to-secondary mutual reactance between windings of different phases</td>
<td>$X_{dy}$</td>
<td>-350.0 Ω</td>
<td>4-8, 4-9, 5-7, 5-9, 6-7, 6-8</td>
</tr>
</tbody>
</table>

are in respective order. The UU switch Type-2 is used to represent all diodes. The corresponding M-file is given in Appendix O. The load change is implemented in an `asmg_var_par.c` file similar to that shown in Appendix C.

In the following computer study, it is assumed that the model is started with zero initial conditions. At $t = 0.05$ s, the load resistor is step-changed to $R_{load} = 0.5$ Ω, after which the model is continued to run until $t = 0.1$ s. The resulting load current $I_{br}(25)$, load voltage $V_{br}(25)$, phase $a$ source current $I_{br}(1)$, and phase $a$ source voltage $I_{br}(1)$ are plotted in Fig. 2.14-3. The secondary Y-side current $I_{br}(7)$, voltage $V_{br}(7)$, and Δ-side current $I_{br}(10)$, voltage $V_{br}(10)$ are plotted in Fig. 2.14-2. As shown in Fig. 2.14-3, the ripple on the dc side has 12 pulses per cycle and relatively low magnitude.
Fig. 2.14-1. Branch and node numbering for twelve pulse rectifier system.

Fig. 2.14-2. Secondary side currents and voltages.
Fig. 2.14-3. Load and source currents and voltages transients.
### Table 2.14-1: Transformer parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>primary Δ-side leakage reactance</td>
<td>$X_{d1}$</td>
<td>9.0 Ω</td>
</tr>
<tr>
<td>primary Δ-side magnetizing reactance</td>
<td>$X_{d1m}$</td>
<td>600.0 Ω</td>
</tr>
<tr>
<td>primary Δ-side mutual reactance between phases</td>
<td>$X_{d1d1}$</td>
<td>-300.0 Ω</td>
</tr>
<tr>
<td>primary Δ-side winding resistance</td>
<td>$R_{d1}$</td>
<td>1.0 Ω</td>
</tr>
<tr>
<td>secondary Y-side leakage reactance</td>
<td>$X_{y2}$</td>
<td>3.0 Ω</td>
</tr>
<tr>
<td>secondary Y-side magnetizing reactance</td>
<td>$X_{y2m}$</td>
<td>200.0 Ω</td>
</tr>
<tr>
<td>secondary Y-side mutual reactance between phases</td>
<td>$X_{y2y2}$</td>
<td>-100.0 Ω</td>
</tr>
<tr>
<td>secondary Y-side winding resistance</td>
<td>$R_{y2}$</td>
<td>0.35 Ω</td>
</tr>
<tr>
<td>primary Δ-side leakage reactance</td>
<td>$X_{d2}$</td>
<td>9.0 Ω</td>
</tr>
<tr>
<td>primary Δ-side magnetizing reactance</td>
<td>$X_{d2m}$</td>
<td>600.0 Ω</td>
</tr>
<tr>
<td>primary Δ-side mutual reactance between phases</td>
<td>$X_{d2d2}$</td>
<td>-300.0 Ω</td>
</tr>
<tr>
<td>primary Δ-side winding resistance</td>
<td>$R_{d2}$</td>
<td>1.0 Ω</td>
</tr>
<tr>
<td>primary Δ-to-secondary-Δ mutual reactance between windings of the same phase</td>
<td>$X_{d1d2m}$</td>
<td>700.0 Ω</td>
</tr>
<tr>
<td>primary Δ-to-secondary-Δ mutual reactance between windings of different phases</td>
<td>$X_{d1d2}$</td>
<td>-350.0 Ω</td>
</tr>
<tr>
<td>primary Δ-to-secondary-Y mutual reactance between windings of the same phase</td>
<td>$X_{d1y2m}$</td>
<td>346.4 Ω</td>
</tr>
<tr>
<td>primary Δ-to-secondary-Y mutual reactance between windings of different phases</td>
<td>$X_{d1y2}$</td>
<td>-173.2 Ω</td>
</tr>
<tr>
<td>secondary Δ-to-secondary-Y mutual reactance between windings of the same phase</td>
<td>$X_{d2y2m}$</td>
<td>346.4 Ω</td>
</tr>
<tr>
<td>secondary Δ-to-secondary-Y mutual reactance between windings of different phases</td>
<td>$X_{d2y2}$</td>
<td>-173.2 Ω</td>
</tr>
</tbody>
</table>
3. SYSTEMS WITH ELECTRICAL MACHINES

3.1. Three-Phase Synchronous Machine Rectifier Circuit Model

An electrical machine can be modeled as a circuit with magnetic coupling among respective branches. In general, in such representation, the mutual inductances will depend on the rotor position. The resulting model of the machine can be readily interconnected with power electronic circuits and drives. The circuit diagram of the synchronous machine rectifier system considered showing the branch and node numbering is depicted in Fig. 3.1-1. This system represents a switched network with time-varying inductive parameters, and has been previously used in [6], [7]. The system parameters correspond to a 125-kW synchronous machine [6] and are given in Table 3.1-1. All impedances are normalized with respect to \( Z_B = 127 \, \Omega \). The UU switch Type-2 is used to represent all diodes. The corresponding M-file is given in Appendix P. The load change is implemented in the `asmg_var_par.c` file that is given in Appendix Q.

In the following computer study, it is assumed that the system starts-up with initial conditions selected close to steady-state operation with a normalized load resistance \( R_{\text{load}} = 4.0 \, \Omega \) (branch 19) and a normalized excitation voltage of \( e_{e\text{fd}} = 1.0e^{-3} \). All voltages are assumed to be normalized with respect to a base voltage of 2300 V. At \( t = 0.05 \, \text{s} \), the load resistance is changed to 0.36 \( \Omega \), after which the model is continued to run until \( t = 0.2 \, \text{s} \). The computer-generated transient responses of the load current \( I_{\text{br}}(18) \), the field winding current \( I_{\text{br}}(6) \), the phase \( a \) generator current \( I_{\text{br}}(9) \), and the phase \( a \) generator voltage \( V_{\text{br}}(9) \) are shown in Fig. 3.1-2. As it can be noted from the messages in the ASMG window as well as trace of generator current \( I_{\text{br}}(9) \) in Fig. 3.1-2, the rectifier initially operates in a light mode with discontinuous currents on the ac side. In this mode, the number of conducting diodes alternates between 2 and 3. However, when the load is increased, the operational mode also changes. In particular, in the heavy mode, there are 3 conducting diodes at any switching interval and the phase currents on the ac side are continuous.
Table 3.1-1: Synchronous machine rectifier system parameters.

<table>
<thead>
<tr>
<th>Comments</th>
<th>Symbol</th>
<th>Value, pu</th>
<th>Branch(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base frequency</td>
<td>$\omega_b$</td>
<td>377π/s</td>
<td></td>
</tr>
<tr>
<td>Stator, phase winding, resistance</td>
<td>$r_s$</td>
<td>0.00515</td>
<td>9, 10, 11</td>
</tr>
<tr>
<td>Stator, phase winding, leakage reactance</td>
<td>$x_{ls}$</td>
<td>0.08</td>
<td>9, 10, 11</td>
</tr>
<tr>
<td>Magnetizing reactance in $q$-axis</td>
<td>$x_{mq}$</td>
<td>1.00</td>
<td>1, 3, 9, 10, 11</td>
</tr>
<tr>
<td>Damper winding in $q$-axis, leakage reactance</td>
<td>$x_{kq1}$</td>
<td>0.146</td>
<td>1</td>
</tr>
<tr>
<td>Damper winding in $q$-axis, leakage reactance</td>
<td>$x_{kq2}$</td>
<td>0.330</td>
<td>3</td>
</tr>
<tr>
<td>Rotor, damper winding in $q$-axis, resistance</td>
<td>$r_{kq1}$</td>
<td>10.440</td>
<td>1</td>
</tr>
<tr>
<td>Rotor, damper winding in $q$-axis, resistance</td>
<td>$r_{kq2}$</td>
<td>0.061</td>
<td>3</td>
</tr>
<tr>
<td>Magnetizing reactance in $d$-axis</td>
<td>$x_{md}$</td>
<td>1.770</td>
<td>5, 7, 9, 10, 11</td>
</tr>
<tr>
<td>Rotor, field winding in $d$-axis, leakage reactance</td>
<td>$x_{lfd}$</td>
<td>0.137</td>
<td>5</td>
</tr>
<tr>
<td>Rotor, damper winding in $d$-axis, leakage reactance</td>
<td>$x_{lkd}$</td>
<td>0.334</td>
<td>7</td>
</tr>
<tr>
<td>Rotor, field winding in $d$-axis, resistance</td>
<td>$r_{fd}$</td>
<td>0.00111</td>
<td>5</td>
</tr>
<tr>
<td>Rotor, damper winding in $d$-axis, resistance</td>
<td>$r_{kd}$</td>
<td>0.024</td>
<td>7</td>
</tr>
<tr>
<td>Rectifier diodes</td>
<td>$r_v$</td>
<td>0.0001</td>
<td>12 - 17</td>
</tr>
<tr>
<td>Filter inductance</td>
<td>$x_{fl}$</td>
<td>0.1</td>
<td>18</td>
</tr>
<tr>
<td>Filter capacitance</td>
<td>$x_{fc}$</td>
<td>0.377</td>
<td>19</td>
</tr>
<tr>
<td>Constant load</td>
<td>$r_{load}$</td>
<td>4.0</td>
<td>19</td>
</tr>
</tbody>
</table>
3.2. Six-Phase Synchronous Machine Rectifier in VBR Form

Electrical machines can also be modeled in a so called voltage behind reactance (VBR) form depicted in Fig. 3.2-1. In this representation, the reactances represent the inductances of the stator, and the voltage sources represents the back emf due to the rotor. The advantage of representing an electrical machine in VBR form is that the resulting stator network model can be readily combined with models of inverters and converters. In order to illustrate how the ASMG can be used to implement VBR models, an example system comprised of a 6-phase synchronous generator connected to two rectifiers and an interphase transformer is considered [8], [9].

The circuit diagram of the stator network illustrating branch and node numbering is shown in Fig. 3.2-2. The stator windings are grouped as two sets of Y-connected 3-phase windings that are displaced from each other by 60 electrical degrees. The corresponding neutral points can be isolated or connected to each other by closing switch branch 22 as shown in Fig. 3.2-2. The system parameters, summarized in Table 3.2-1, correspond to a 240-Hz 210-kW synchronous machine rated 355-V line-to-neutral. The synchronous machine is modeled in VBR form with dynamic saliency neglected [10]. This modeling technique results in a constant stator inductance matrix, which provides a signif-
significant computational advantage, and is equivalent to the Park’s representation over a wide range of frequencies. The relevant equations can be found in [10] and, therefore, are not repeated here. The standard Simulink library blocks were used to implement the rotor state model, whereas the ASMG was used to implement the stator network and rectifiers shown in Fig. 3.2-2. The UU switch Type-2 is used to represent all diodes. The branch 22 is represented by LB switch Type-3 that can be closed at any time and open at the current-zero-crossing. The corresponding M-file is given in Appendix R. Since in the given model formulation the inductances are constant, the file asmg_var_par.c is not needed.

In the following computer study, it is assumed that the system starts-up with initial conditions selected close to steady-state operation with a constant excitation voltage of $e_{sfd} = 502$ V and disconnected generator neutrals (branch 22). At $t = 0.01$ s, the switch branch 22 is closed after which the model is continued to run until $t = 0.02$ s. The computer-generated transient responses of the phase $a1$ generator voltage $V_{br(16)}$, generator current $I_{br(16)}$, and the output dc voltage $V_{br(15)}$ are shown in Fig. 3.2-3. The voltage between the generator neutrals $V_{br(22)}$, current flowing through the switch $I_{br(22)}$, and the rectifier currents $I_{br(13)}$ and $I_{br(14)}$ are shown in Fig. 3.2-4.

When the generator neutrals are disconnected, the upper and lower rectifiers essentially operate as two independent 3-phase rectifiers connected in parallel and shifted by 60 degrees; thus producing dc voltages with the same in-phase ripple. Since the dc currents

![Fig. 3.2-1. VBR representation of an electrical machine.](image-url)
**Ibr(13) and Ibr(14)** have in-phase ripple as shown in Fig. 3.2-4, the interphase transformer does not produce the desired harmonic cancellation. Also, since the rectifiers operate in parallel, the currents are essentially equally divided between the two sets of stator windings. Therefore, each diode conducts only approximately one-half of the load current during its conduction period, which results in symmetric phase current shown in Fig. 3.2-3.

---

**Fig. 3.2-2.** Six-phase synchronous machine rectifier system.

**Fig. 3.2-3.** Voltages and currents at the generator terminal and at the load.
Fig. 3.2-4. Voltages and currents at the generator neutrals and interface transformer.

### Table 3.2-1: System parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Branch(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base frequency</td>
<td>$\omega_b$</td>
<td>1507 rad/sec</td>
<td>-</td>
</tr>
<tr>
<td>Stator phase winding resistance</td>
<td>$r_s$</td>
<td>0.114 Ω</td>
<td>16-21</td>
</tr>
<tr>
<td>Stator phase winding leakage inductance</td>
<td>$L_{ls}$</td>
<td>0.135 mH</td>
<td>16-21</td>
</tr>
<tr>
<td>Magnetizing inductance q-axis</td>
<td>$L_{mq}$</td>
<td>1.60 mH</td>
<td>16-21</td>
</tr>
<tr>
<td>Damper winding q-axis leakage inductance</td>
<td>$L_{lkq}$</td>
<td>0.06 mH</td>
<td>-</td>
</tr>
<tr>
<td>Damper winding q-axis, resistance</td>
<td>$r_{kq}$</td>
<td>0.110 Ω</td>
<td>-</td>
</tr>
<tr>
<td>Magnetizing inductance d-axis</td>
<td>$L_{md}$</td>
<td>1.82 mH</td>
<td>16-21</td>
</tr>
<tr>
<td>Field winding d-axis leakage inductance</td>
<td>$L_{lfd}$</td>
<td>0.255 mH</td>
<td>-</td>
</tr>
<tr>
<td>Damper winding d-axis leakage inductance</td>
<td>$L_{lkd}$</td>
<td>0.650 mH</td>
<td>-</td>
</tr>
<tr>
<td>Field winding d-axis resistance</td>
<td>$r_{fd}$</td>
<td>0.015 Ω</td>
<td>-</td>
</tr>
<tr>
<td>Damper winding d-axis resistance</td>
<td>$r_{kd}$</td>
<td>0.118 Ω</td>
<td>-</td>
</tr>
<tr>
<td>Interphase transformer winding resistance</td>
<td>$r_{tr}$</td>
<td>0.05 Ω</td>
<td>13, 14</td>
</tr>
<tr>
<td>Interphase transformer leakage inductance</td>
<td>$L_{ltr}$</td>
<td>0.25 mH</td>
<td>13, 14</td>
</tr>
<tr>
<td>Interphase transformer mutual inductance</td>
<td>$L_{mtr}$</td>
<td>1.0 mH</td>
<td>13, 14</td>
</tr>
<tr>
<td>Load resistance</td>
<td>$r_L$</td>
<td>12.0 Ω</td>
<td>15</td>
</tr>
</tbody>
</table>
The system performance changes significantly when the generator neutrals are connected to one another. In particular, since the negative rails of both rectifiers are common, the load current $V_{br(15)}$ can return through any of the six negative rail diodes. However, the positive rails are separate, resulting in the splitting of the load current between the two bridges. Also, because of the 60-degree shift of the stator windings, the current ripple produced by each bridge is out-of-phase. Therefore, the dc currents $I_{br(13)}$ and $I_{br(14)}$ have out-of-phase ripple as shown in Fig. 3.2-4 after the neutrals are connected.
4. ZONAL ELECTRICAL DISTRIBUTION SYSTEM COMPONENTS

In this chapter, the ASMG/Simulink models of the DC ZEDS Testbed components are described briefly. Detailed descriptions of each component and the corresponding ACSL model can be found in the appropriate Component Report [11]-[15]; therefore, only a brief description of each component is included here. The ASMG/Simulink models are provided on the compact disk (CD) that accompanies this document.

4.1. 15-kW Power Supply

A circuit diagram of the 15-kW Power Supply (PS) proposed by the University of Missouri-Rolla for the Integrated Propulsion System (IPS) is shown in Fig. 4.1-1. The power supply accepts a 3-phase input voltage that may vary between 480-560 V line-line rms, and is designed to regulate the output to 500 V dc for loads up to 15 kW. In the given topology, the three-phase ac source is connected to an uncontrolled diode-rectifier. The output of the rectifier provides the input to a buck converter that is used for output voltage regulation. The proposed architecture differs from that of the ac/dc converter presently used in the IPS. In particular, presently the conversion is accomplished through the use of an active thyristor bridge rectifier. Although an active bridge rectifier eliminates the need for the buck converter shown in Fig. 4.1-1, the proposed circuit has the advantage that there is no need for synchronization with the ac supply, which simplifies the control design. In addition, the buck converter has a larger regulation bandwidth than a thyristor-controlled rectifier.

In order to develop an ASMG model of the PS, the circuit of Fig. 4.1-1 is represented in terms of the fundamental branches given in Fig. 1.1-1 and Fig. 1.1-2. The circuit

Fig. 4.1-1. Topology and branch layout of the 15-kW Power Supply.
parameters are summarized in Table 4.1-1. The rectifier circuit consists of commutating inductors \( b_1 - b_3 \), six diodes \( b_4 - b_9 \), filtering capacitor \( b_{12} \), an auxiliary starting resistor \( b_{10} \) that is used to limit inrush current through the filtering capacitor, and a switch \( b_{11} \) that is used to short the starting resistor once the capacitor \( b_{12} \) is sufficiently charged. The converter circuit consists of an IGBT that is represented by a controllable switch \( b_{13} \), diode \( b_{14} \), and the dc inductor \( b_{15} \). The output capacitor \( C_2 \) is modeled outside of the ASMG using standard Simulink integrator. Using this arrangement, the inductor current \( i_L \) is supplied to the capacitor, and the capacitor voltage \( V_{out} \) is fed back into branch \( b_{15} \) as a external voltage source. The corresponding M-file is given in Appendix S. Since in the given model the inductances are constant and no other parameters are changed at run time, the file \texttt{asmg_var_par.c} is left blank.

The block diagram of the PS control is shown in Fig. 4.1-2. The corresponding control parameters used in the model are summarized in Table 4.1-2. In the given control, the measured output voltage \( V_{out} \) is compared with the reference voltage \( V_{ref} \), and the error provides an input to a proportional-plus-integral (PI) controller. The feedforward path provides a fast response to changes in load. The difference between the output of the

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Branch(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base frequency</td>
<td>( \omega_b )</td>
<td>377 rad/sec</td>
<td>-</td>
</tr>
<tr>
<td>Commutating resistance</td>
<td>( r_1 )</td>
<td>0.01 ( \Omega )</td>
<td>1-3</td>
</tr>
<tr>
<td>Commutating inductance</td>
<td>( L_1 )</td>
<td>1.0 mH</td>
<td>1-3</td>
</tr>
<tr>
<td>Starting resistor</td>
<td>( r_{st} )</td>
<td>20.0 ( \Omega )</td>
<td>10</td>
</tr>
<tr>
<td>Rectifier output capacitor</td>
<td>( C_1 )</td>
<td>500.0 ( \mu F )</td>
<td>12</td>
</tr>
<tr>
<td>DC inductor</td>
<td>( L_2 )</td>
<td>3.0 mH</td>
<td>15</td>
</tr>
<tr>
<td>Resistance of the DC inductor</td>
<td>( r_2 )</td>
<td>0.03 ( \Omega )</td>
<td>15</td>
</tr>
<tr>
<td>Buck converter output capacitor</td>
<td>( C_2 )</td>
<td>500.0 ( \mu F )</td>
<td>Implemented in Simulink</td>
</tr>
<tr>
<td>Nominal load resistance</td>
<td>( r_{load} )</td>
<td>16.7 ( \Omega )</td>
<td>Implemented in Simulink</td>
</tr>
</tbody>
</table>
PI controller and the measured output current is multiplied by the output of nonlinear stabilizing control (NSC). The bounded result is passed to the current regulator. The output of the current regulator is used for the hysteresis current controller. The control is implemented using standard Simulink blocks. Additional information regarding the control design is given in Component Report [11]. Details of the Simulink implementation can be seen by opening the model `ps_model.mdl` that is provided in the companion CD.

In the following computer study, it is assumed that the model starts up with zero initial conditions and a nominal resistive load. At \( t = 0.04 \) s, the switch \( b_{11} \) is closed shorting the starting resistor \( b_{10} \). At \( t = 0.08 \) s, the load is stepped from nominal to 36.7 \( \Omega \). Then, at \( t = 0.12 \) s the load is restored to 16.7 \( \Omega \) and the model is continued to

---

**Table 4.1-2: Power supply control parameters.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference voltage (command output voltage)</td>
<td>( V_{\text{ref}} )</td>
<td>500.0 V</td>
</tr>
<tr>
<td>Maximum command current</td>
<td>( i_{\text{max}} )</td>
<td>40.0 A</td>
</tr>
<tr>
<td>PI controller, proportional gain</td>
<td>( K_p )</td>
<td>1.0</td>
</tr>
<tr>
<td>PI controller, integral gain</td>
<td>( K_i )</td>
<td>100.0</td>
</tr>
<tr>
<td>Current regulator, integral gain</td>
<td>( K_{ih} )</td>
<td>100.0</td>
</tr>
<tr>
<td>NSC exponent</td>
<td>( n )</td>
<td>1.0</td>
</tr>
<tr>
<td>NSC time constant</td>
<td>( t )</td>
<td>5.0e-3</td>
</tr>
<tr>
<td>Hysteresis error bandwidth</td>
<td>( h )</td>
<td>1.0 A</td>
</tr>
</tbody>
</table>
run until $t = 0.15 \text{ s}$. The simulation results are shown in Fig. 4.1-3. As can be observed in $V_{br(1)}$ and $I_{br(1)}$, when the starting resistor is not shorted, the high frequency harmonics propagate to the ac side. However, when the capacitor $b_{12}$ is sufficiently charged and the switch $b_{11}$ is closed, the ac waveforms $V_{br(1)}$ and $I_{br(1)}$ look much cleaner. The purpose of the PI controller is to regulate the output voltage to 500 Volts, which is achieved approximately after $t = 0.05 \text{ s}$. The step changes in load resistance are reflected in the current $I_{out}$. However, due to the feedforward control path, the output voltage $V_{out}$ remains at 500 V essentially unchanged.

Fig. 4.1-3. Simulation results for the PS.
4.2. Ship Service Converter Module

A circuit diagram of the Ship Service Converter Module (SSCM) proposed by the Naval Post Graduate School for the Integrated Propulsion System (IPS) is shown in Fig. 4.2-1. The converter accepts 500 V dc, that may vary, and regulates the output voltage to 400 V dc for load currents up to 20 A. The rated output power is 8 kW. The SSCM topology is based on the classical buck converter discussed in Section 2.2. The circuit parameters are summarized in Table 4.2-1.

![Circuit Diagram](https://via.placeholder.com/150)

**Fig. 4.2-1.** Topology and branch layout of SSCM.

**Table 4.2-1: SSCM circuit parameters.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Branch(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switching frequency</td>
<td>$f_{sw}$</td>
<td>20.0 kH</td>
<td>-</td>
</tr>
<tr>
<td>Rated dc input voltage</td>
<td>$V_{in}$</td>
<td>500.0 V</td>
<td>1</td>
</tr>
<tr>
<td>Rated dc output voltage</td>
<td>$V_{out}$</td>
<td>400.0 V</td>
<td>7</td>
</tr>
<tr>
<td>Rated output power</td>
<td>$P_{out}$</td>
<td>8.0 kW</td>
<td>-</td>
</tr>
<tr>
<td>Input filter inductor</td>
<td>$L_f$</td>
<td>357.0 µH</td>
<td>1</td>
</tr>
<tr>
<td>Input inductor resistance</td>
<td>$R_{fL}$</td>
<td>0.2 Ω</td>
<td>1</td>
</tr>
<tr>
<td>Input filter capacitor</td>
<td>$C_{f1}$</td>
<td>500.0 µF</td>
<td>2</td>
</tr>
<tr>
<td>Input filter series resistor</td>
<td>$R_{fC}$</td>
<td>1.0 Ω</td>
<td>3</td>
</tr>
<tr>
<td>Input filter additional capacitor</td>
<td>$C_{f2}$</td>
<td>45.0 µF</td>
<td>4</td>
</tr>
<tr>
<td>Buck converter dc inductor</td>
<td>$L$</td>
<td>3.0 mH</td>
<td>7</td>
</tr>
<tr>
<td>Resistance of the dc inductor</td>
<td>$R_L$</td>
<td>0.5 Ω</td>
<td>7</td>
</tr>
<tr>
<td>Buck converter output capacitor</td>
<td>$C$</td>
<td>500.0 µF</td>
<td>Implemented in Simulink</td>
</tr>
<tr>
<td>Nominal load resistance</td>
<td>$r_{load}$</td>
<td>20.0 Ω</td>
<td>Implemented in Simulink</td>
</tr>
</tbody>
</table>
In order to develop an ASMG model of the SSCM, the circuit Fig. 4.2-1 is represented in terms of the fundamental branches given in Fig. 1.1-1 and Fig. 1.1-2. The input filter consists of inductor \( b_1 \), capacitor \( b_3 \) in series with damping resistor \( b_2 \), and an additional filtering capacitor \( b_4 \). The input voltage is implemented as an external voltage source of branch \( b_1 \). The converter circuit consists of an IGBT that is represented by a controllable switch \( b_5 \), a diode \( b_6 \), and a dc inductor \( b_7 \). The output capacitor \( C \) is modeled using a standard Simulink integrator. Using this arrangement, the inductor current \( i_L \) is supplied to the capacitor, and the capacitor voltage \( V_{out} \) is fed back into branch \( b_7 \) as a external voltage source. The corresponding M-file is given in Appendix T. Since all circuit parameters are constant in the given model, the file `asmg_var_par.c` is left blank.

The block diagram of the SSCM control is given in Fig. 4.2-2. The control parameters used in the model are summarized in Table 4.2-2. In the given control, the measured output voltage \( V_{out} \) is compared with the reference voltage \( V_{ref} \), and the error provides an input to a PI controller. The PI controller ensures the long-term regulation of the output voltage to the reference level \( V_{ref} \). The output current and the inductor current are compared, and the error provides a feedforward path that ensures a fast response to changes in load. The bounded result is passed to the switching signal generator that implements pulses corresponding to the specified duty cycle. The control is implemented using standard Simulink blocks. Additional information regarding the control design is given in Component Report [12]. Details of the Simulink implementation can be seen by opening the model `sscm_study.mdl` that is provided in the companion CD.

In the following computer study, the SSCM model starts with a resistive load of 200.0 Ω. The initial conditions of the model are set to correspond to steady-state operation with the given load. In particular, capacitors \( C_{f1} \) and \( C_{f2} \) are each pre-charged to 500.0 V, and the output capacitor \( C \) to 400.0 V. The current in the input filter inductor \( L_f \) is initialized to 1.6 Amps. Since, initially, the converter is lightly loaded and is expected to operate in the discontinuous mode, the current in the dc inductor \( L \) is initialized to zero.
Finally, the initial condition of the integrator in PI controller is set to 0.83. The model is started with these initial conditions. At $t = 0.02$ s, the load is stepped to the nominal value of 20.0 $\Omega$. Then, at $t = 0.05$ s, the load is switched back to 200.0 $\Omega$ and the model is continued to run until $t = 0.08$ s. The simulated response is shown in Fig. 4.2-3. As shown, the current $i_{\text{out}}$ follows the changes in load resistance; however, the output voltage $V_{\text{out}}$ undergoes small transients and remains within ±5 V of the reference level. Also, based on the waveform of dc inductor current $I_L$, it is concluded that with a nominal load of 20.0 $\Omega$, the converter operates in the continuous conduction mode.
4.3. Constant Power Load

A circuit diagram of the Constant Power Load (CPL) [13] module proposed by the University of Missouri-Rolla (UMR) for the Integrated Propulsion System (IPS) is shown in Fig. 4.3-1. The topology of CPL is based on the classical buck converter discussed in Section 2.2. The CPL accepts dc voltage that may vary between 120-600 V dc, and regulates the output voltage to 100 V dc. The converter is loaded with a 2.0 Ω resistor, which at the given output voltage, dissipates 5 kW of power. Since the output voltage is tightly regulated to a constant level, the dissipated power is also constant. The circuit parameters are summarized in Table 4.3-1.

In order to develop an ASMG model of the CPL, the circuit Fig. 4.3-1 is represented in terms of fundamental branches. The input filter consists of resistor $b_1$ and capacitor $b_2$. The input voltage is implemented as an external voltage source of branch $b_1$. The converter circuit consists of an IGBT that is represented by a controllable switch $b_3$, a diode $b_4$, and a dc inductor $b_5$. The output capacitor $C$ and load resistor $R_{load}$ are
modeled using standard Simulink blocks. Using this arrangement, the inductor current $i_L$ is supplied to the capacitor, and the capacitor voltage $V_{out}$ is fed back into branch $b_5$ as an external voltage source. The corresponding M-file is given in Appendix U. Since all circuit parameters are constant, the file `asmg_var_par.c` is left blank.

The block diagram of the CPL control is given in Fig. 4.3-2. The control parameters are summarized in Table 4.3-2. In the given control, the inductor current $i_L$ is controlled using a hysteresis current control scheme so as to regulate the output power at the specified level $P_{ref}$. In particular, the reference power $P_{ref}$ is first modulated by the nonlinear stabilizing control (NSC) and is then divided by the measured output voltage $V_{out}$. The resulting signal is passed through the limits to the current regulator that produces the command current, which is then used by the hysteresis current control. The entire control

---

**Table 4.3-1: CPL circuit parameters.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Branch(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated dc input voltage</td>
<td>$V_{in}$</td>
<td>120-600.0 V</td>
<td>1</td>
</tr>
<tr>
<td>Rated dc output voltage</td>
<td>$V_{out}$</td>
<td>100.0 V</td>
<td>5</td>
</tr>
<tr>
<td>Rated output power</td>
<td>$P_{out}$</td>
<td>5.0 kW</td>
<td>-</td>
</tr>
<tr>
<td>Input filter resistance</td>
<td>$R_{in}$</td>
<td>1.0 mΩ</td>
<td>1</td>
</tr>
<tr>
<td>Input filter capacitor</td>
<td>$C_{in}$</td>
<td>470.0 µF</td>
<td>2</td>
</tr>
<tr>
<td>Buck converter dc inductor</td>
<td>$L$</td>
<td>2.0 mH</td>
<td>5</td>
</tr>
<tr>
<td>Resistance of the dc inductor</td>
<td>$R_L$</td>
<td>0.01 Ω</td>
<td>5</td>
</tr>
<tr>
<td>Buck converter output capacitor</td>
<td>$C$</td>
<td>470.0 µF</td>
<td>Implemented in Simulink</td>
</tr>
<tr>
<td>Nominal load resistance</td>
<td>$R_{load}$</td>
<td>2.0 Ω</td>
<td>Implemented in Simulink</td>
</tr>
</tbody>
</table>
is implemented using standard Simulink blocks. Additional information regarding the control design is given in the Component Report [13]. Details of the Simulink implementation can be seen by opening the model `cpl_model.mdl` that is provided on the companion CD.

In the following computer study, it is assumed that the CPL is connected to a 400 V dc source with the capacitor $C_{in}$ initially pre-charged to the same voltage. The CPL starts up connected to a resistive load of 2.0 $\Omega$ with $C$ discharged. At $t = 0.02$ s, the input voltage is stepped down to 200 V. Then, at $t = 0.03$ s, the input voltage is increased to 600 V. Finally, at $t = 0.04$ s, the input voltage is stepped back to 400 V, and the model is continued to run until $t = 0.05$ s. The simulated response is shown in Fig. 4.3-3. As

Table 4.3-2: CPL control parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSC time constant</td>
<td>$\tau$</td>
<td>5e-4 sec</td>
</tr>
<tr>
<td>NSC exponent</td>
<td>$n$</td>
<td>1</td>
</tr>
<tr>
<td>Lower bound on expected input voltage</td>
<td>$V_{in_min}$</td>
<td>120 Volts</td>
</tr>
<tr>
<td>Upper bound on expected input voltage</td>
<td>$V_{in_max}$</td>
<td>1000 Volts</td>
</tr>
<tr>
<td>Lower bound on expected output voltage</td>
<td>$V_{out_min}$</td>
<td>50 Volts</td>
</tr>
<tr>
<td>Upper bound on expected output voltage</td>
<td>$V_{out_max}$</td>
<td>1000 Volts</td>
</tr>
<tr>
<td>Upper bound on command current</td>
<td>$i_{\text{max}}$</td>
<td>50 Amps</td>
</tr>
<tr>
<td>Current regulator time constant</td>
<td>$K_i$</td>
<td>100</td>
</tr>
<tr>
<td>Hysteresis bound on current control</td>
<td>$h$</td>
<td>1 Amps</td>
</tr>
</tbody>
</table>

**Fig. 4.3-2. Control for the CPL.**
shown, the voltage $V_{\text{in}}$ changes abruptly; however, the inductor current $I_L$ and output voltage $V_{\text{out}}$ undergo small transients and quickly return to the levels that correspond to an output power of 5 kW. These transients are attributed primary to the NSC that limits the responsiveness of the hysteresis current controller and thus reduces the effect of the negative incremental impedance. The duration of these transients as well as the control responsiveness can be adjusted by changing the NSC time constant $\tau$.

![Figure 4.3-3. Simulation results for the CPL.](image)

### 4.4. Ship Service Inverter Module

The topology of the Ship Service Inverter Module (SSIM) [14] proposed by Purdue University for the Integrated Propulsion System (IPS) is shown in Fig. 4.4-1. The SSIM accepts a dc voltage that may vary between 380-440 V dc, and provides balanced three-phase ac voltages of the specified frequency and magnitude for loads up to 10 kW. As can be seen in Fig. 4.4-1, the SSIM topology is represented in terms of fundamental branches. In particular, the SSIM consists of an input filter composed of branches $b_1 - b_3$, a three-phase four-quadrant inverter represented by $b_4 - b_{15}$, an output filter $b_{16} - b_{24}$, and a $\Delta$-connected load $b_{28} - b_{30}$. The load is connected to the filter through the LB
switches $b_{25} - b_{27}$. The circuit parameters are summarized in Table 4.4-1. The corresponding M-file is given in Appendix V. Since all circuit parameters are constant, the file `asmg_var_par.c` is left blank.

A block diagram of the SSIM control is given in Fig. 4.4-2. The control parameters are summarized in Table 4.4-2. In the given control, the output filter inductor currents $i_{aL}$, $i_{bL}$, and $i_{cL}$ are controlled using a delta-hysteresis current control scheme so as to regulate the output voltages $v_{ab}$, $v_{bc}$, and $v_{ca}$ at the specified level and frequency. In
Table 4.4-2: SSIM control parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSC time constant</td>
<td>$\tau$</td>
<td>$5 \times 10^{-4}$ sec</td>
</tr>
<tr>
<td>NSC exponent</td>
<td>$n$</td>
<td>1</td>
</tr>
<tr>
<td>Lower bound on expected input voltage</td>
<td>$V_{in_min}$</td>
<td>120 Volts</td>
</tr>
<tr>
<td>Upper bound on expected input voltage</td>
<td>$V_{in_max}$</td>
<td>1000 Volts</td>
</tr>
<tr>
<td>Lower bound on expected output voltage</td>
<td>$V_{out_min}$</td>
<td>50 Volts</td>
</tr>
<tr>
<td>Upper bound on expected output voltage</td>
<td>$V_{out_max}$</td>
<td>1000 Volts</td>
</tr>
<tr>
<td>Upper bound on command current</td>
<td>$i_{\text{max}}$</td>
<td>50 Amps</td>
</tr>
<tr>
<td>Current regulator time constant</td>
<td>$K_i$</td>
<td>100</td>
</tr>
<tr>
<td>Sampling frequency</td>
<td>$f_s$</td>
<td>100 kHz</td>
</tr>
<tr>
<td>Hysteresis bound on current control</td>
<td>$h$</td>
<td>1 Amps</td>
</tr>
</tbody>
</table>

Fig. 4.4-2. Control for the SSIM.

In particular, the commanded $q$ and $d$ voltages $v_{qd\_ref}$ are first compared to the measured voltages transformed into the synchronous reference frame $v_{qd}$, and the resulting error is fed to the PI controller. The measured load currents $i_a$, $i_b$, and $i_c$ are also transformed into the synchronous reference frame $i_{qd}$, which after adding the decoupling term, repre-
sents a feedforward control path. The feedforward current signals are added to the output of the PI controller, and the result is passed through a limiter to the current regulator. The output of the current regulator $i_{qdl}^*$ is transformed back to $abc$ variables $i_{abc}^*$, which, in turn, provides reference currents to the delta-hysteresis modulation strategy. All reference frame transformations are performed with respect to the angle $\Theta_e$, which sets the frequency of the output voltage at $\omega_{e\_ref}$. The entire control is implemented using standard Simulink blocks. Additional information regarding the control design is given in the Component Report [14]. Details of the Simulink implementation can be seen by opening the model `ssim_model.mdl` that is provided on the companion CD.

In the following computer study, it is assumed that the SSIM is connected to a 400 V dc source with the capacitor $C_{in}$ initially pre-charged to 400 V. The SSIM is initially off. At $t = 0.05$ s, the SSIM is turned on under no-load conditions with the following reference settings: $\omega_{e\_ref} = 377 \text{ r/ s}, \quad v_{q\_ref} = 200.0 \text{ V}, \quad$ and $\quad v_{d\_ref} = 0 \text{ V}$. At $t = 0.1$ s, the switches $b_{25} - b_{27}$ connecting the RL load are closed. Then, at $t = 0.15$ s, the control signals for the switches $b_{25} - b_{27}$ are removed and the load is disconnected at subsequent current-zero-crossings. The simulation results are shown in Fig. 4.4-3. The output filter inductor current and the output voltage are shown for one phase as $I_{aL}$ and $V_{ab}$, respectively. As shown in Fig. 4.4-3, after the SSIM is turned on, the voltages $V_q$ and $V_d$ reach the specified reference settings and the corresponding line-to-line voltage $V_{ab}$ becomes sinusoidal. At the same time, the inductor current $I_{aL}$ also becomes sinusoidal with some switching noise attributed to the delta-hysteresis modulation. Based on the plots of $I_{qL}$ and $I_{dL}$, it can be noted that for no-load conditions, the current supplied by inverter to the output filter is predominantly reactive. When the load is connected, the output voltages $V_q$ and $V_d$ undergo very small transients and quickly return to their previous levels. At the same time, the load current is quickly picked-up by the inverter as can be seen in the plots of $I_{qL}$ and $I_{aL}$. It should also be noted that, despite of high-frequency switching, the load voltages $V_{load}$ and currents $I_{load}$ are very clean. After the control signals to the switches $b_{25} - b_{27}$ have been removed, the
Fig. 4.4-3. Simulation results for the example study with SSIM.
load disconnects at the subsequent current-zero-crossings, and the SSIM quickly returns to the no-load state of operation.

4.5. Motor Controller

The inverter topology of the Motor Controller (MC) [15] proposed by Purdue University for the Integrated Propulsion System (IPS) is shown in Fig. 4.5-1. The MC inverter accepts a dc voltage that may vary between 300-420 V dc. The output of the inverter is connected to an induction motor. In order to implement the MC inverter using the ASMG, the inverter circuit is represented in terms of fundamental branches as depicted in Fig. 4.5-1. In particular, the MC inverter consists of an input filter composed of branches \( b_1 - b_2 \), a three-phase four-quadrant power inverter represented by branches \( b_3 - b_{14} \), and Y-connected load branches \( b_{15} - b_{17} \) that represent the stator windings of the induction motor. It is assumed that the motor model accepts voltages as inputs and returns the corresponding stator currents. Therefore, in order to accommodate the interconnection of the inverter and the induction motor, branches \( b_{15} - b_{17} \) must be of the type Fig. 1.1-1(b). The branch parameters are summarized in Table 4.5-1. The corresponding M-file is given in Appendix W. Since all circuit parameters are constant, the file `asmg_var_par.c` is left blank.

In the MC model (`mc_model.mdl`) that is provided on the companion CD, the induction motor is represented using a traditional \( qd \)-model [16] and implemented using

![Fig. 4.5-1. MC inverter topology and branch layout.](image)
standard Simulink blocks. Parameters for the 5-hp motor considered in [15] were estimated based on parameters of the motors with similar ratings given in [16]. For consistency, the motor parameters are summarized in Table 4.5-2. If necessary, a more advanced induction machine model such as that used with the MC model described in [15], can be used with the ASMG/Simulink model discussed herein. In any case, the ASMG can be used to effectively implement the circuit part of the overall system, and the rest of the model including the machine and its controls can be implemented using standard library blocks available in Simulink. The induction machine parameters are summarized in the Matlab script file mc_impar.m that is given in Appendix X. In order for the Simulink model to access these parameters, they must be loaded into Matlab workspace, which can be achieved by executing the script mc_impar from the Matlab command line.

Table 4.5-1: MC circuit parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Branch(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated dc input voltage</td>
<td>$V_{in}$</td>
<td>300-400 V</td>
<td>1</td>
</tr>
<tr>
<td>Input filter, series resistance</td>
<td>$R_{in}$</td>
<td>10.0 mΩ</td>
<td>1</td>
</tr>
<tr>
<td>Input filter, capacitor</td>
<td>$C_{in}$</td>
<td>1000.0 µF</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 4.5-2: Induction motor parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated dc input voltage</td>
<td>$V_{in}$</td>
<td>230 V (line-to-line, rms)</td>
</tr>
<tr>
<td>Base frequency</td>
<td>$\omega_b$</td>
<td>377 r/s</td>
</tr>
<tr>
<td>Number of poles</td>
<td>$P$</td>
<td>4</td>
</tr>
<tr>
<td>Mechanical inertia</td>
<td>$J$</td>
<td>0.089 kg · m²</td>
</tr>
<tr>
<td>Stator winding resistance</td>
<td>$R_s$</td>
<td>0.315 Ω</td>
</tr>
<tr>
<td>Stator winding leakage reactance</td>
<td>$X_{ls}$</td>
<td>0.546 Ω</td>
</tr>
<tr>
<td>Magnetizing inductance</td>
<td>$X_m$</td>
<td>18.92 Ω</td>
</tr>
<tr>
<td>Rotor winding resistance</td>
<td>$R_r$</td>
<td>0.591 Ω</td>
</tr>
<tr>
<td>Rotor winding leakage reactance</td>
<td>$X_{lr}$</td>
<td>0.546 Ω</td>
</tr>
</tbody>
</table>

The MC is capable of operating in two modes, namely the speed and torque modes. A block diagrams of the Drive Control is given in Fig. 4.5-2. Depending upon the selected mode of operation, the Drive Control determines the necessary commanded torque $T_{e\_com}$. In particular, if the speed mode is selected, the Drive Control produces the com-
manded torque signal so as to drive the motor speed \( \omega_r \) equal to the specified reference speed \( \omega_{r,ref} \). On the other hand, in torque mode, the commanded torque \( T_{e,com} \) is directly set to the specified reference torque \( T_{e,ref} \). The resulting torque command \( T_{e,com} \) is fed into the Torque Control that is depicted in Fig. 4.5-3. The purpose of the Torque Control is to establish the commanded stator currents \( i_{abc}^* \) that are used by the delta-hysteresis modulation to control the inverter switches \( b_3, b_4, b_7, b_8, b_{11}, b_1 \).

---

**Fig. 4.5-2. Drive Control.**

**Fig. 4.5-3. Torque Control.**
order to establish $i_{abc}^*$, first the stator voltages and currents are transformed into the synchronous $qd$ reference frame. The resulting $v_{qds}$ and $i_{qds}$ are used to compute the estimated torque $T_{e\_hat}$. The Torque Estimator is shown in Fig. 4.5-4 and is based on the induction machine $qd$-model given in [16]. The Torque Trim compensates for the error that may exist between $T_{e\_com}$ and $T_{e\_hat}$, and the resulting signal, after being subjected to limits, is passed to the Maximum Torque Per-Ampere (MTPA) control block. Based on the given input, the MTPA computes the current $i_{qd}^*$ and slip frequency $\omega_s$ that supply the required torque with minimum current. The MTPA is implemented as a rational function of polynomials and is described in more detail in [15]. The output of MTPA is modulated by the NSC and the result is passed to the Current Regulator. Finally, the commanded currents $i_{qd\_com}^*$ are transformed back into $abc$ variables, and $i_{abc\_com}^*$ is passed to the delta-hysteresis modulator. The electrical frequency $\omega_e$ that is used to determine the transformation angle $\Theta_e$ is computed as $\omega_e = \omega_r + \omega_s$. The corresponding control parameters are summarized in Table 4.5-3.

In the following computer study, it is assumed that the MC is supplied from an ideal dc source of 400 V, and that induction machine has a mechanical load with the following characteristic $T_l = \omega_{rm}^2 \cdot 2.8384e^{-4}$. It is also assumed that initially the motor is accelerated to $\omega_{rm} = 188.5 \text{ r/ s}$, and that the capacitor $C_m$ is pre-charged to 400 V. The MC is started in the torque mode with $T_{e\_ref} = 10 \text{ N m}$. At $t = 0.2 \text{ s}$, the reference
torque is stepped to $T_{e_{\text{ref}}} = 15 \text{ N} \cdot \text{m}$. The simulation results are shown in Fig. 4.5-5. As shown, after $t = 0.2 \text{ s}$, the electromagnetic torque follows the change in command and the motor begins to accelerate.

![Graphs showing simulation results](image)

**Table 4.5-3: MC control parameters.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed control proportional gain</td>
<td>$k_{sc}$</td>
<td>5 Nms</td>
</tr>
<tr>
<td>Speed control time constant</td>
<td>$\tau_{sc}$</td>
<td>1 s</td>
</tr>
<tr>
<td>Limit on torque</td>
<td>$T_{e\text{max}}$</td>
<td>25 Nm</td>
</tr>
<tr>
<td>NSC time constant</td>
<td>$\tau_{nsc}$</td>
<td>1 sec.</td>
</tr>
<tr>
<td>NSC exponent</td>
<td>$n_{nsc}$</td>
<td>0</td>
</tr>
<tr>
<td>Lower bound on expected input voltage</td>
<td>$V_{dc\text{min}}$</td>
<td>300 Volts</td>
</tr>
<tr>
<td>Upper bound on expected input voltage</td>
<td>$V_{dc\text{max}}$</td>
<td>420 Volts</td>
</tr>
<tr>
<td>Torque trim time constant</td>
<td>$\tau_{tt}$</td>
<td>5.0e-3 s</td>
</tr>
<tr>
<td>Synchronous current regulator time constant</td>
<td>$\tau_{scr}$</td>
<td>5.0e-4 s</td>
</tr>
<tr>
<td>Synchronous current regulator turn-off time constant</td>
<td>$\tau_{scr\text{off}}$</td>
<td>1.0e-2 s</td>
</tr>
<tr>
<td>Feedback control current limit</td>
<td>$i_{fcl}$</td>
<td>2.5 Amps</td>
</tr>
<tr>
<td>Hysteresis bound on current control</td>
<td>$h$</td>
<td>1 Amps</td>
</tr>
</tbody>
</table>

Fig. 4.5-5. MC example study.
5. REFERENCES


Load,” Component Report for the Naval Combat Survivability Tasks 1, 3, and 4, *University of Missouri-Rolla and Purdue University, May 3, 2001. (Draft)*


Appendix A: M-File for the Second-Order RLC Circuit

function h = rlc_circuit
% This file implements a second-order RLC circuit
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
h = 1; % Define a model handle
start_br_list(h); % Initialize ASMG instance

% System parameters are
L = 4e-3; % Inductance, 4mH
rl = 0.002; % Resistance of an inductor, 0.002 Ohms
C = 200e-6; % Capacitor, 200mcF
rc = 20; % Resistance in parallel with capacitor, 20 Ohms

% Syntax of use the BRANCH function
% # hdl br p n r L/C E/J
% # 1 2 3 4 5 6 7

L_branch(h, 1, 2, 1, rl, L, 1); % inductor
C_branch(h, 2, 2, 1, rc, C, 0); % capacitor
Appendix B: M-File for the Buck Converter

function h = buck
% This file implements a BUCK converter [1].
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 06, 2001.
%
% Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
% 1995.
%
h = 1; % Define a model handle
start_br_list(h);

% System parameters are
fs = 20e3; Ts = 1/fs; % Switching frequency, 20 kHz
Vdc = 500; % DC voltage, 500-V
L = 1e-3; % Inductance, 1mH
C = 50e-6; % Capacitor, 50mcF
rl = 200.0; % Load resistance, 200 Ohms

% Syntax of use the BRANCH function.
% #   h  br  p  n  r/sw   L/C  E/J
L_branch(h, 1,  2, 1,   0,    0,   1); % source branch
S_branch(h, 2,  2, 3,   2,    1);      % controlled switch
S_branch(h, 3,  1, 3,   2,    0);      % diode
L_branch(h, 4,  3, 4,   0,    L,   0); % inductor branch
C_branch(h, 5,  4, 1,  rl,    C,   0); % load and capacitor

r_varmut(h, 5,  5,  rl); % declare variable resistance to
% do the step-change in load
% See the ASMG_VAR_PAR.C file
Appendix C: C-File for the Buck Converter

/*************************** ASMG_VAR_PAR ***************************/
void __stdcall asmg_var_par(int *h, double *t)
{
/* This function implements time-varying parameters for the ASMG
 * system defined by the S-function ASMGFUN.C.
*/
static int ncalls = 0;
static double pi = 3.14159265358979;

/* Initialize parameters and store them as static variables */
if (!ncalls) {
    ncalls = 1;
}

/* Check the handle to make sure that the variable parameters are
 * implemented for the correct instance of the ASMG
*/
if (*h == 1) {
    // First Instance

    // Do a step change in load
    if (*t>0.0005){
        set_rvm(h, 10.0, 1);
    }
}

// End of First Instance

return;

}/***************** end of ASMG_VAR_PAR *******************/
Appendix D: M-File for the Boost Converter

function  h = boost
% This file implements a Boost converter.
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynszczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 06, 2001.
%
% Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
%

h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

fs = 20e3; Ts = 1/fs; % Switching frequency, 20 kHz
L = 1e-3;    % Inductance, 1mH
C = 50e-6;   % Capacitor, 50mcF
Vdc = 500;   % DC voltage, 500V
rl = 700.0;  % Load resistance, 700 Ohms

% Syntax of use the BRANCH function. Here goes the branch list
%
% # hdl br p n r L/C E/J
% # 1 2 3 4 5 6 7

L_branch(h, 1, 2, 1, 0, 0, 1);  % source branch
L_branch(h, 2, 2, 3, 0, L, 0);  % stator branch
S_branch(h, 3, 3, 1, 2, 1);    % controllable switch
S_branch(h, 4, 3, 4, 2, 0);    % diode
C_branch(h, 5, 4, 1, rl, C, 0); % load branches

r_varmut(h, 5, 5, rl);  % declare variable resistance to
                        % do the step-change in load
                        % See the ASMG_VAR_PAR.C file
Appendix E: M-File for the Buck-Boost Converter

function h = buckboost
% This file implements a Buck-Boost converter.
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 9, 2001.
%
%     Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
%

h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

fs = 20e3; Ts = 1/fs;  % Switching frequency, 20 kHz
Vdc = 500;  % DC voltage, 500V
L = 1e-3;   % Inductance, 1mH
C = 50e-6;  % Capacitor, 50mcF
rl = 200.0; % Load resistance, 200 Ohms

% Syntax of use the BRANCH function. Here goes the branch list
%
%    # hdl br   p  n    r    L/C  E/J
%    #   1  2   3  4    5     6    7
L_branch(h, 1,  2, 1,   0,    0,   1); % source branch
S_branch(h, 2,  2, 3,   2,    0); % controlled switch
L_branch(h, 3,  3, 1,   0,    L,   0); % inductor branch
S_branch(h, 4,  4, 3,   2,    0); % diode
C_branch(h, 5,  1, 4,  rl,    C,   0); % load
r_varmut(h, 5,  5,  rl); % declare variable resistance to
% do the step-change in load
% See the ASMG_VAR_PAR.C file
Appendix F: M-File for the Cuk Converter

function h = cuk
% This file implements a Cuk converter.
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 9, 2001.
%
%    Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
%    1995., pp-186, example 7-3/problem 7-17.
%

h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

fs = 20e3; Ts = 1/fs; % Switching frequency, 20 kHz
Vdc = 500;  % DC voltage, 500V

L1 = 1e-3; % Input inductance, 1mH
L2 = 1e-3; % DC inductance, 1mH
C1 = 50e-6; % Capacitor, 50mcF
C2 = 50e-6; % Load capacitor, 50mcF
rl = 200.0; % Load resistance, 200 Ohms

% Syntax of use the BRANCH function. Here goes the branch list
%
% # hdl br p n r L/C E/J
%    1 2 3 4 5 6 7

L_branch(h, 1, 2, 3, 0, 0, 1);  % Vdc source branch
L_branch(h, 2, 2, 3, 0, L1, 0);  % inductor branch
S_branch(h, 3, 3, 1, 2, 1);  % controlled switch
C_branch(h, 4, 3, 4, 0, C1, 0); % capacitor C1
S_branch(h, 5, 4, 1, 2, 0);  % diode
L_branch(h, 6, 4, 5, 0, L2, 0); % inductor branch
C_branch(h, 7, 1, 5, rl, C2, 0); % load
Appendix G: M-File for the Single-Phase Diode Rectifier

function  h = sp_drec
% This file implements a Single-Phase Diode Rectifier [1].
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
% Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
% 1995.
%

h = 1;  % Define a model handle
start_br_list(h);
%
% System parameters are

fs = 60;     % Switching frequency, 60-Hz
Vdc = 500;   % DC voltage, 500-V
Rs = 1.0;    % Source resistance
Ls = 1e-4;   % Source inductance, 0.1-mH
C = 500e-6;  % Capacitor, 50mcF
rl = 200.0;  % Load resistance, 200 Ohms
%
% Syntax of use the BRANCH function.
% #   1  2   3  4    5     6    7
% #   h  br  p  n  r/sw   L/C  E/J
L_branch(h, 1,  2, 1,  Rs,   Ls,   1); % source branch
S_branch(h, 2,  2, 3,   2,    1);      % diode
C_branch(h, 3,  3, 1,  rl,    C,   0); % load and capacitor
r_varmut(h, 3,  3,  rl); % declare variable resistance to
% do the step-change in load
% See the ASMG_VAR_PAR.C file
function  h = sp_thyrec
% This file implements a Single-Phase Thyristor Rectifier [1].
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
%    Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
%    1995.
%
h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

fs = 60;     % Switching frequency, 60-Hz
Vdc = 500;   % DC voltage, 500-V
Rs = 1.0;    % Source resistance
Ls = 1e-4;   % Source inductance, 0.1-mH
C = 500e-6;  % Capacitor, 50mcF
rl = 100.0;  % Load resistance, 200 Ohms

% Syntax of use the BRANCH function.
% #   1  2   3  4    5     6    7
% #   h  br  p  n  r/sw   L/C  E/J

L_branch(h, 1,  2, 1,  Rs,   Ls,   1); % source branch
S_branch(h, 2,  2, 3, 4,   0);        % thyristor
C_branch(h, 3,  3, 1, rl,   C,   0);  % load and capacitor
Appendix I: M-File for the Single-Phase Triac-Commutator

function h = sp_trcom
% This file implements a Single-Phase Triac Commutator [1].
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
%     Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
%     1995.
%
h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

fs = 60;     % Switching frequency, 60-Hz
Vdc = 500;   % DC voltage, 500-V
Rs = 1.0;    % Source resistance
Ls = 1e-4;   % Source inductance, 0.1-mH
rl = 100.0;  % Load resistance, 100 Ohms

% Syntax of use the BRANCH function.
% #   1  2   3  4    5     6    7
% #   h  br  p  n  r/sw   L/C  E/J
L_branch(h, 1,  2, 1,  Rs,   Ls,   1); % source branch
S_branch(h, 2,  2, 3,   3,    0);      % triac commutator
L_branch(h, 3,  3, 1,  rl,    0,   0); % load
function h = spdhbr
% This file implements a Single-Phase Diode H-Bridge [1].
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 9, 2001.
%
% Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
% 1995.
%
h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

wb=377.0;  % 60 Hz base frequency
Ls = 0.1e-3;  %--- Source Inductance, 0.1-mH
rs = 1.0;  %--- Source Resistance, 1.0-Ohms
rl = 200.0;  %--- Load resistance, 200 Ohms
Cf = 500e-6;  %--- Capacitor filter, 500-mcF

% Syntax of use the BRANCH function
%  # hdl br   p  n    r    L/C  E/J
%     #  1  2   3  4    5     6    7
L_branch(h, 1,  2, 1,  rs,   Ls,   1); % source branch
S_branch(h, 2,  2, 3,   2,   0); % diode 1
S_branch(h, 3,  1, 3,   2,   0); % diode 2
S_branch(h, 4,  4, 2,   2,   0); % diode 3
S_branch(h, 5,  4, 1,   2,   0); % diode 4
C_branch(h, 6,  3, 4,  rl,   Cf,   0); % load branch
R_varmut(h, 6,  6, rl); % declare variable resistance to
% do the step-change in load
% See the ASMG_VAR_PAR.C file
Appendix K: M-File for the Single-Phase Thyristor H-Bridge

function  h = spthyhbr
% This file implements a Single-Phase Thyristor H-Bridge [1].
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 9, 2001.
%
%   Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
%   1995.
%

h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

wb=377.0;          %  60 Hz base frequency
Ls = 0.1e-3;          %--- Source Inductance, 0.1-mH
rs = 1.0;             %--- Source Resistance, 1.0-Ohms
rl = 200.0;           %--- Load resistance, 200 Ohms
Cf = 500e-6;          %--- Capacitor filter, 500-mcF

% Syntax of use the BRANCH function
%    # hdl br   p  n    r    L/C  E/J
%    #   1  2   3  4    5     6    7
L_branch(h, 1,  2, 1,  rs,   Ls,   1); % source branch
S_branch(h, 2,  2, 3,   4,   0); % Thyristor 1
S_branch(h, 3,  1, 3,   4,   0); % Thyristor 2
S_branch(h, 4,  4, 2,   4,   0); % Thyristor 3
S_branch(h, 5,  4, 1,   4,   0); % Thyristor 4
C_branch(h, 6,  3, 4,  rl,   Cf,   0); % load branch
Appendix L: M-File for the Three-Phase Diode Rectifier

function h = tpdrec
% This file implements a Three-Phase Diode Rectifier [1].
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 9, 2001.
% Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
% 1995.

h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

wb=377.0;             % 60 Hz base frequency
Ls = 1.0e-3;          %--- Source Inductance, 1.0 mH
rs = 1.0;             %--- Source Resistance, 1.0 Ohms
rl = 10.0;            %--- Load resistance, 10 Ohms
Ll = 4.0e-3;          %--- Load Inductance, 4.0 mH

% Syntax of use the BRANCH function. Here goes the branch list

L_branch(h, 1,  2, 1,  rs,   Ls,   1); % source branch, Eas 
L_branch(h, 2,  3, 1,  rs,   Ls,   1); % source branch, Ebs 
L_branch(h, 3,  4, 1,  rs,   Ls,   1); % source branch, Ecs 
S_branch(h, 4,  2, 5,   2,   0); % diode 1 
S_branch(h, 5,  3, 5,   2,   0); % diode 2 
S_branch(h, 6,  4, 5,   2,   0); % diode 3 
S_branch(h, 7,  6, 2,   2,   0); % diode 4 
S_branch(h, 8,  6, 3,   2,   0); % diode 5 
S_branch(h, 9,  6, 4,   2,   0); % diode 6 
L_branch(h,10,  5, 6,  rl,   Ll,   0); % load branch

R_varmut(h, 10, 10, rl); % declare variable resistance to 
% do the step-change in load 
% See the ASMG_VAR_PAR.C file
Appendix M: M-File for the Three-Phase Thyristor Rectifier

function  h = tpthyrec
% This file implements a Three-Phase Thyristor Rectifier [1].
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
%       Converters, Applications, and Design, JOHN WILEY & SONS, Inc.
%       1995.
%
h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

wb=377.0;             %  60 Hz base frequency
Ls = 1.0e-3;          %--- Source Inductance,  1.0 mH
rs = 1.0;             %--- Source Resistance,  1.0 Ohms
rl = 20.0;            %--- Load resistance, 20 Ohms
Cf = 500.0e-6;        %--- Filter capacitance,  500.0-mcF

% Syntax of use the BRANCH function. Here goes the branch list
%
%    # hdl br   p  n    r    L/C  E/J
%    #   1  2   3  4    5     6    7
L_branch(h, 1,  2, 1,  rs,   Ls,   1); % source branch, Eas
L_branch(h, 2,  3, 1,  rs,   Ls,   1); % source branch, Ebs
L_branch(h, 3,  4, 1,  rs,   Ls,   1); % source branch, Ecs
S_branch(h, 4,  2, 5,   2,   0); % Thyristor 1
S_branch(h, 5,  3, 5,   2,   0); % Thyristor 2
S_branch(h, 6,  4, 5,   2,   0); % Thyristor 3
S_branch(h, 7,  6, 2,   2,   0); % Thyristor 4
S_branch(h, 8,  6, 3,   2,   0); % Thyristor 5
S_branch(h, 9,  6, 4,   2,   0); % Thyristor 6
C_branch(h,10,  5, 6,  rl,   Cf,   0); % load branch
Appendix N: M-File for the Three-Phase Y-Δ Transformer

function h = tpdytr
% This file implements a Three-Phase D-Y transformer system.
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%

h = 1;  % Define a model handle
start_br_list(h);
%
% System parameters are

wb=377.0;          %  60 Hz base frequency
Xs = 2.0; Ls = Xs/wb; %--- Source Inductance, 2.0 Ohms
rs = 1.0;             %--- Source Resistance, 1.0 Ohms
rla = 200.0;        %--- Phase A, Load resistance, 200 Ohms
rpb = 300.0;        %--- Phase B, Load resistance, 300 Ohms
rlc = 150.0;        %--- Phase C, Load resistance, 150 Ohms
rft = 1.0;          %--- Phase A, Short to ground, 1 Ohms
Xft = 1.0; Lft = Xft/wb; %--- Phase A, Short to ground, 1 Ohms
Xla = 1.0; Lla = Xla/wb; %--- Phase A, Load reactance, 1 Ohms
Xlb = 10; Llb = Xlb/wb; %--- Phase B, Load reactance, 10 Ohms
Xlc = 15; Llc = Xlc/wb; %--- Phase C, Load reactance, 15 Ohms
Xdl = 3.0; Ldl = Xdl/wb; % Y - leakage inductance
Xdm = 700; Ldm = Xdm/wb; % Y - self mutual inductance
Rds = 1.0;          % D - resistance
Lds = Ldm + Ldl;    % D - total self inductance
Ldd =-Ldm/2;        % D-to-D - mutual inductance
Xyl = 2.0; Lyl = Xyl/wb; % Y - leakage inductance
Xym = 700; Lym = Xym/wb; % Y - self mutual inductance
Rys = 1.0;          % Y - resistance
Lys = Lym + Lyl;    % Y - total self inductance
Lyy =-Lym/2;        % Y-to-Y - mutual inductance
Ldym = Lym;         % Y-to-D - mutual inductance in the same phase
Ldy = -Lym/2;       % Y-to-D - mutual inductance in the different phase
%

% Syntax of use the BRANCH function. Here goes the branch list
%

Appendix N: M-File for the Three-Phase Y-D Transformer

```matlab
% # hdl br p n r L/C E/J
% 1 2 3 4 5 6 7

L_branch(h, 1, 2, 1, rs, Ls, 1); % source branch, Eas
L_branch(h, 2, 3, 1, rs, Ls, 1); % source branch, Ebs
L_branch(h, 3, 4, 1, rs, Ls, 1); % source branch, Ecs

L_branch(h, 4, 2, 3, Rds, Lds, 0); % transformer, Dab
L_branch(h, 5, 3, 4, Rds, Lds, 0); % transformer, Dbc
L_branch(h, 6, 4, 2, Rds, Lds, 0); % transformer, Dca

L_branch(h, 7, 6, 5, Rys, Lys, 0); % transformer, Yas
L_branch(h, 8, 7, 5, Rys, Lys, 0); % transformer, Ybs
L_branch(h, 9, 8, 5, Rys, Lys, 0); % transformer, Ycs

L_branch(h, 10, 6, 5, rla, Lla, 0); % Phase A, Load
L_branch(h, 11, 7, 5, rlb, Llb, 0); % Phase B, Load
L_branch(h, 12, 8, 5, rlc, Llc, 0); % Phase C, Load

L_branch(h, 13, 6, 9, rft, Lft, 0); % Phase A, fault
S_branch(h, 14, 9, 5, 3, 0); % short to ground

%--- Define mutual inductances Y-to-Y
L_smutual(h, 4, 5, Ldd);
L_smutual(h, 5, 6, Ldd);
L_smutual(h, 6, 4, Ldd);

%--- Define mutual inductances D-to-D
L_smutual(h, 7, 8, Lyy);
L_smutual(h, 8, 9, Lyy);
L_smutual(h, 9, 7, Lyy);

%--- Define mutual inductances y-to-D
L_smutual(h, 4, 7, Ldym);
L_smutual(h, 4, 8, Ldy);
L_smutual(h, 4, 9, Ldy);
L_smutual(h, 5, 7, Ldy);
L_smutual(h, 5, 8, Ldym);
L_smutual(h, 5, 9, Ldy);
L_smutual(h, 6, 7, Ldy);
L_smutual(h, 6, 8, Ldy);
L_smutual(h, 6, 9, Ldym);
```
Appendix O: M-File for the 12-Pulse Diode Rectifier with ∆-Y/∆ Transformer

function h = p12drec
% This file implements a 12-Pulse Diode Rectifier System.
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.

h = 1;  % Define a model handle
start_br_list(h);

% System parameters are
wb=377.0;             %--- 60 Hz base frequency
Xs = 2.0; Ls = Xs/wb; %--- Source Inductance, 2.0 Ohms
rs = 1.0;             %--- Source Resistance, 1.0 Ohms
rl = 20.0;            %--- Load resistance, 20 Ohms
Xl = 1.0; Ll = Xl/wb; %--- Load reactance, 1 Ohms

Xd1 = 9.0; Ld1 = Xd1/wb; % D - leakage inductance
Xdm = 600; Ldm = Xdm/wb; % D - self mutual inductance
Rd1 = 1.0;             % D - resistance
Ld1 = Ldm + Ld1;       % D - total self inductance
Ld1d1 = Ldm/2;         % D-to-D - mutual inductance

Rd2 = 1.0;             % D - resistance
Ld2 = Ldm + Ld1;       % D - total self inductance
Ld2d2 = Ldm/2;         % D-to-D - mutual inductance
Ld1d2 = Ldm/2;         % D-to-D - mutual inductance
Ld1d2m = Ldm;          % D-to-D - mutual inductance

Xyl = 3.0; Lyl = Xyl/wb; % Y - leakage inductance
Xym = 200; Lym = Xym/wb; % Y - self mutual inductance
Ry2 = 0.35;            % Y - resistance
Ly2 = Lym + Lyl;       % Y - total self inductance
Ly2y2 = Lym/2;         % Y-to-Y - mutual inductance

Ld1y2m = Lym*sqrt(3); % D-to-Y - mutual inductance in the same phase
Ld1y2 = -Ld1y2m/2;    % D-to-Y - mutual inductance in the different phase
Ld2y2m = Lym*sqrt(3); % D-to-Y - mutual inductance in the same phase
Ld2y2 = -Ld1y2m/2;    % D-to-Y - mutual inductance in the different phase

% Syntax of use the BRANCH function. Here goes the branch list
Appendix O: M-File for the 12-Pulse Diode Rectifier with D-Y/D Transformer

```
% # hdl br p n r L/C E/J
% # 1 2 3 4 5 6 7

L_branch(h, 1, 2, 1, rs, Ls, 1); % source branch, Eas
L_branch(h, 2, 3, 1, rs, Ls, 1); % source branch, Ebs
L_branch(h, 3, 4, 1, rs, Ls, 1); % source branch, Ecs

L_branch(h, 4, 2, 3, Rd1, Ld1, 0); % transformer, D1ab
L_branch(h, 5, 3, 4, Rd1, Ld1, 0); % transformer, D1bc
L_branch(h, 6, 4, 2, Rd1, Ld1, 0); % transformer, D1ca

L_branch(h, 7, 6, 5, Ry2, Ly2, 0); % transformer, Y2as
L_branch(h, 8, 7, 5, Ry2, Ly2, 0); % transformer, Y2bs
L_branch(h, 9, 8, 5, Ry2, Ly2, 0); % transformer, Y2cs

L_branch(h, 10, 9, 10, Rd2, Ld2, 0); % transformer, D2as
L_branch(h, 11, 10, 11, Rd2, Ld2, 0); % transformer, D2bs
L_branch(h, 12, 11, 9, Rd2, Ld2, 0); % transformer, D2cs

S_branch(h, 13, 6, 12, 2, 0); % diode 1
S_branch(h, 14, 7, 12, 2, 0); % diode 2
S_branch(h, 15, 8, 12, 2, 0); % diode 3
S_branch(h, 16, 13, 6, 2, 0); % diode 4
S_branch(h, 17, 13, 7, 2, 0); % diode 5
S_branch(h, 18, 13, 8, 2, 0); % diode 6

S_branch(h, 19, 9, 12, 2, 0); % diode 7
S_branch(h, 20, 10, 12, 2, 0); % diode 8
S_branch(h, 21, 11, 12, 2, 0); % diode 9
S_branch(h, 22, 13, 9, 2, 0); % diode 10
S_branch(h, 23, 13, 10, 2, 0); % diode 11
S_branch(h, 24, 13, 11, 2, 0); % diode 12

L_branch(h, 25, 12, 13, rl, Ll, 0); % load branch
r_varmut(h, 25, 25, rl); % declare variable resistance to
% do the step-change in load
% See the ASMG_VAR_PAR.C file

%--- Define mutual inductances D1-to-D1
L_smutual(h, 4, 5, Ld1d1);
L_smutual(h, 5, 6, Ld1d1);
L_smutual(h, 6, 4, Ld1d1);

%--- Define mutual inductances Y2-to-Y2
L_smutual(h, 7, 8, Ly2y2);
L_smutual(h, 8, 9, Ly2y2);
L_smutual(h, 9, 7, Ly2y2);

%--- Define mutual inductances D2-to-D2
L_smutual(h, 10, 11, Ld2d2);
L_smutual(h, 11, 12, Ld2d2);
```
L_smutable(h, 12, 10, Ld2d2);

%--- Define mutual inductances D1-to-Y2
L_smutable(h, 4, 7, Ld1y2m);
L_smutable(h, 4, 8, Ld1y2);
L_smutable(h, 4, 9, Ld1y2);
L_smutable(h, 5, 7, Ld1y2);
L_smutable(h, 5, 8, Ld1y2m);
L_smutable(h, 5, 9, Ld1y2);
L_smutable(h, 6, 7, Ld1y2);
L_smutable(h, 6, 8, Ld1y2);
L_smutable(h, 6, 9, Ld1y2m);

%--- Define mutual inductances D1-to-D2
L_smutable(h, 4, 10, Ld1d2m);
L_smutable(h, 4, 11, Ld1d2);
L_smutable(h, 4, 12, Ld1d2);
L_smutable(h, 5, 10, Ld1d2);
L_smutable(h, 5, 11, Ld1d2m);
L_smutable(h, 5, 12, Ld1d2);
L_smutable(h, 6, 10, Ld1d2);
L_smutable(h, 6, 11, Ld1d2);
L_smutable(h, 6, 12, Ld1d2m);

%--- Define mutual inductances D2-to-Y2
L_smutable(h, 7, 10, Ld2y2m);
L_smutable(h, 7, 11, Ld2y2);
L_smutable(h, 7, 12, Ld2y2);
L_smutable(h, 8, 10, Ld2y2);
L_smutable(h, 8, 11, Ld2y2m);
L_smutable(h, 8, 12, Ld2y2);
L_smutable(h, 9, 10, Ld2y2);
L_smutable(h, 9, 11, Ld2y2);
L_smutable(h, 9, 12, Ld2y2m);
Appendix P: M File for the Three-Phase Synchronous Machine

function  h = tpsmrec
% This Matlab function implements a Three-Phase Synchronous Machine Rectifier
% System in Circuit Form [1]. The system parameters are taken from [2].
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% July 24, 2001.
% [1] O. Wasynczuk, S. D. Sudhoff, "Automated State Model Generation
%     Algorithm for Power Circuits and Systems", IEEE Transactions
% saliency of synchronous machine in power electronic based systems,"
% IEEE Transactions on Energy Conversion. Vol. 14, No. 4, 1999,
% p. 1177-1183.

h = 1;  % Define a model handle
start_br_list(h);

% System parameters are

wb = 377;  % Base frequency, rad/sec.
xls = 0.08;  % Stator
Lls = xls/wb;
rs = 5.15e-3;

xmq = 1.0;  % Lmq = xmq/wb;
xlkq1 = 0.146;  % Llkq1 = xlkq1/wb;
rkq1 = 10.44;
xlkq2 = 0.330;  % Llkq2 = xlkq2/wb;
rkq2 = 0.061;

xmd = 1.77;  % Lmd = xmd/wb;
xlf = 0.137;  % Llf = xlf/wb;
rfd = 1.11e-3;
xld = 0.334;  % Lld = xld/wb;
rkd = 0.024;

xl = 0.1;  % Ll = xl/wb;
rl = 4.0;
Cf = 1e-3;

% Start the branch list

% Syntax of use the BRANCH function. Here goes the branch list
% # hdl br  p n  r  L/C  E/J
% # 1 2 3 4 5 6 7
Appendix P: M File for the Three-Phase Synchronous Machine

L_branch(h, 1, 1, 2, rkq1, Llkq1+Lmq, 0); % rotor kq1 winding
L_branch(h, 2, 1, 2, 0, 0, 0); % short
L_branch(h, 3, 3, 4, rkq2, Llkq2+Lmq, 0); % rotor kq2 winding
L_branch(h, 4, 3, 4, 0, 0, 0); % short
L_branch(h, 5, 5, 6, rfd/2, Llfd+Lmd, 0); % rotor kf field winding
L_branch(h, 6, 5, 6, 0, 1); % source branch
L_branch(h, 7, 7, 8, rkd, Llkd+Lmd, 0); % rotor kd winding
L_branch(h, 8, 7, 8, 0, 0, 0); % short
L_branch(h, 9, 9, 10, rs, Lls, 0); % stator branch
L_branch(h, 10, 9, 11, rs, Lls, 0); % stator branch
L_branch(h, 11, 9, 12, rs, Lls, 0); % stator branch
S_branch(h, 12, 10, 13, 2, 1); % diode inst 1
S_branch(h, 13, 11, 13, 2, 0); % diode inst 0
S_branch(h, 14, 12, 13, 2, 0); % diode inst 0
S_branch(h, 15, 14, 10, 2, 0); % diode inst 0
S_branch(h, 16, 14, 11, 2, 0); % diode inst 0
S_branch(h, 17, 14, 12, 2, 0); % diode inst 0
L_branch(h, 18, 13, 15, 0, Ll, 0); % load branches
C_branch(h, 19, 15, 14, rl, Cf, 0); % load branches
R_varmut(h, 19, 19, rl); % declare variable resistance to
% do the step-change in load
% See the ASMG_VAR_PAR.C file

% # BR1 BR2 Lmut
% # 1 2 3

% Rotor constant mutual inductances
L_mutual(h, 3, 1, Lmq);
L_mutual(h, 1, 3, Lmq);
L_mutual(h, 7, 5, Lmd);
L_mutual(h, 5, 7, Lmd);

% Stator variable mutual inductances
% (initial values can be anything positive,
% the actual values are given in the ASMG_VAR_PAR.C file)
L_varmut(h, 9, 9, Lls);
L_varmut(h, 10, 9, Lls);
L_varmut(h, 11, 9, Lls);
L_varmut(h, 9, 10, Lls);
L_varmut(h, 10, 10, Lls);
L_varmut(h, 11, 10, Lls);
L_varmut(h, 9, 11, Lls);
L_varmut(h, 10, 11, Lls);
L_varmut(h, 11, 11, Lls);

% Mutual between stator and rotor
% (initial values can be anything positive,
% the actual values are given in the ASMG_VAR_PAR.C file)
L_varmut(h, 9, 1, Lls);
L_varmut(h, 10, 1, Lls);
L_varmut(h, 11, 1, Lls);

L_varmut(h, 9, 3, Lls);
L_varmut(h, 10, 3, Lls);
L_varmut(h, 11, 3, Lls);

L_varmut(h, 9, 5, Lls);
L_varmut(h, 10, 5, Lls);
L_varmut(h, 11, 5, Lls);

L_varmut(h, 9, 7, Lls);
L_varmut(h, 10, 7, Lls);
L_varmut(h, 11, 7, Lls);

L_varmut(h, 1, 9, Lls);
L_varmut(h, 3, 9, Lls);
L_varmut(h, 5, 9, Lls);
L_varmut(h, 7, 9, Lls);

L_varmut(h, 1, 10, Lls);
L_varmut(h, 3, 10, Lls);
L_varmut(h, 5, 10, Lls);
L_varmut(h, 7, 10, Lls);

L_varmut(h, 1, 11, Lls);
L_varmut(h, 3, 11, Lls);
L_varmut(h, 5, 11, Lls);
L_varmut(h, 7, 11, Lls);
Appendix Q: C File for the Three-Phase Synchronous Machine

/*************************** ASMG_VAR_PAR ***************************/
void __stdcall asmg_var_par(int *h, double *t)
{
    /* This function implements time-varying parameters for the ASMG
     * system defined by the S-function ASMGSFUN.C.
     */
    static int ncalls = 0;
    static double pi = 3.14159265358979, wb = 377.0;
    static double xls, Lls, rs;
    static double xmq, Lmq, xlkq1, Llkq1, rkq1, xlkq2, Llkq2, rkq2;
    static double xmd, Lmd, xlfd, Llfd, rfd, xlkd, Llkd, rkd;
    static double La, Lb;
    double theta;

    /* Initialize parameters and store them as static variables */
    if (!ncalls) {
        ncalls = 1;
        xls = 0.08;
        Lls = xls/wb;
        rs = 5.15e-3;
        xmq = 1.0;
        Lmq = xmq/wb;
        xlkq1 = 0.146;
        Llkq1 = xlkq1/wb;
        rkq1 = 10.44;
        xlkq2 = 0.330;
        Llkq2 = xlkq2/wb;
        rkq2 = 0.061;
        xmd = 1.77;
        Lmd = xmd/wb;
        xlfd = 0.137;
        Llfd = xlfd/wb;
        rfd = 1.11e-3;
        xlkd = 0.334;
        Llkd = xlkd/wb;
        rkd = 0.024;

        /* Additional parameters */
        La = (Lmd + Lmq)/3.0;
        Lb = (Lmd - Lmq)/3.0;
    }

    /* Check the handle to make sure that the variable parameters are
* implemented for the correct instance of the ASMG */

if (*h == 1) {
    // First instance, Synchronous Machine
    theta = *t * wb;

    set_lvm(h, Lls+La-Lb*cos(2.0*theta), 1);
    set_lvm(h, -(La/2.0)-Lb*cos(2.0*theta-2.0*pi/3.0), 2);
    set_lvm(h, -(La/2.0)-Lb*cos(2.0*theta+2.0*pi/3.0), 3);
    set_lvm(h, -(La/2.0)-Lb*cos(2.0*theta-2.0*pi/3.0), 4);
    set_lvm(h, Lls+La-Lb*cos(2.0*theta-4.0*pi/3.0), 5);
    set_lvm(h, -(La/2.0)-Lb*cos(2.0*theta+2.0*pi), 6);
    set_lvm(h, -(La/2.0)-Lb*cos(2.0*theta+2.0*pi/3.0), 7);
    set_lvm(h, -(La/2.0)-Lb*cos(2.0*theta+2.0*pi), 8);
    set_lvm(h, Lls+La-Lb*cos(2.0*theta+4.0*pi/3.0), 9);

    /* Mutual between stator and rotor */

    set_lvm(h, Lmq*cos(theta), 10);
    set_lvm(h, Lmq*cos(theta-2.0*pi/3.0), 11);
    set_lvm(h, Lmq*cos(theta+2.0*pi/3.0), 12);
    set_lvm(h, Lmq*cos(theta), 13);
    set_lvm(h, Lmq*cos(theta-2.0*pi/3.0), 14);
    set_lvm(h, Lmq*cos(theta+2.0*pi/3.0), 15);
    set_lvm(h, Lmd*sin(theta), 16);
    set_lvm(h, Lmd*sin(theta-2.0*pi/3.0), 17);
    set_lvm(h, Lmd*sin(theta+2.0*pi/3.0), 18);
    set_lvm(h, Lmd*sin(theta), 19);
    set_lvm(h, Lmd*sin(theta-2.0*pi/3.0), 20);
    set_lvm(h, Lmd*sin(theta+2.0*pi/3.0), 21);
    set_lvm(h, (2.0/3.0)*Lmq*cos(theta), 22);
    set_lvm(h, (2.0/3.0)*Lmq*cos(theta), 23);
    set_lvm(h, (2.0/3.0)*Lmd*sin(theta), 24);
    set_lvm(h, (2.0/3.0)*Lmd*sin(theta), 25);
    set_lvm(h, (2.0/3.0)*Lmq*cos(theta-2.0*pi/3.0), 26);
    set_lvm(h, (2.0/3.0)*Lmq*cos(theta-2.0*pi/3.0), 27);
    set_lvm(h, (2.0/3.0)*Lmd*sin(theta-2.0*pi/3.0), 28);
    set_lvm(h, (2.0/3.0)*Lmd*sin(theta-2.0*pi/3.0), 29);
    set_lvm(h, (2.0/3.0)*Lmq*cos(theta+2.0*pi/3.0), 30);
    set_lvm(h, (2.0/3.0)*Lmq*cos(theta+2.0*pi/3.0), 31);
    set_lvm(h, (2.0/3.0)*Lmd*sin(theta+2.0*pi/3.0), 32);
    set_lvm(h, (2.0/3.0)*Lmd*sin(theta+2.0*pi/3.0), 33);

    /* Compute derivatives of variable inductances and write them into LVAR */
/* Stator self inductances */

set_dlv(h, 2.0*wb*Lb*sin(2.0*theta), 1);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta-2.0*pi/3.0), 2);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta+2.0*pi/3.0), 3);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta-2.0*pi/3.0), 4);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta-4.0*pi/3.0), 5);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta+2.0*pi), 6);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta+2.0*pi/3.0), 7);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta+2.0*pi), 8);
set_dlv(h, 2.0*wb*Lb*sin(2.0*theta+4.0*pi/3.0), 9);

/* Mutual between stator and rotor */

set_dlv(h, -wb*Lmq*sin(theta), 10);
set_dlv(h, -wb*Lmq*sin(theta-2.0*pi/3.0), 11);
set_dlv(h, -wb*Lmq*sin(theta+2.0*pi/3.0), 12);
set_dlv(h, -wb*Lmq*sin(theta), 13);
set_dlv(h, -wb*Lmq*sin(theta-2.0*pi/3.0), 14);
set_dlv(h, -wb*Lmq*sin(theta+2.0*pi/3.0), 15);
set_dlv(h, wb*Lmd*cos(theta), 16);
set_dlv(h, wb*Lmd*cos(theta-2.0*pi/3.0), 17);
set_dlv(h, wb*Lmd*cos(theta+2.0*pi/3.0), 18);
set_dlv(h, wb*Lmd*cos(theta), 19);
set_dlv(h, wb*Lmd*cos(theta-2.0*pi/3.0), 20);
set_dlv(h, wb*Lmd*cos(theta+2.0*pi/3.0), 21);
set_dlv(h, -wb*(2.0/3.0)*Lmq*sin(theta), 22);
set_dlv(h, -wb*(2.0/3.0)*Lmq*sin(theta), 23);
set_dlv(h, wb*(2.0/3.0)*Lmd*cos(theta), 24);
set_dlv(h, wb*(2.0/3.0)*Lmd*cos(theta), 25);
set_dlv(h, -wb*(2.0/3.0)*Lmq*sin(theta-2.0*pi/3.0), 26);
set_dlv(h, -wb*(2.0/3.0)*Lmq*sin(theta-2.0*pi/3.0), 27);
set_dlv(h, wb*(2.0/3.0)*Lmd*cos(theta-2.0*pi/3.0), 28);
set_dlv(h, wb*(2.0/3.0)*Lmd*cos(theta-2.0*pi/3.0), 29);
set_dlv(h, -wb*(2.0/3.0)*Lmq*sin(theta+2.0*pi/3.0), 30);
set_dlv(h, -wb*(2.0/3.0)*Lmq*sin(theta+2.0*pi/3.0), 31);
set_dlv(h, wb*(2.0/3.0)*Lmd*cos(theta+2.0*pi/3.0), 32);
set_dlv(h, wb*(2.0/3.0)*Lmd*cos(theta+2.0*pi/3.0), 33);

// Do a step change in load
if (*t>0.05){
    set_rvm(h, 0.36, 1);
}

}
return;
}

/************************ end of ASMG_VAR_PAR ***************************/
Appendix R: M File for the Six-Phase Synchronous Machine Rectifier in VBR Form

function h = spsmrec
% This file implements a Six-Phase Synchronous Machine Rectifier
% System [1], [2]. The synchronous machine is modeled in a so called
% voltage-behind-reactance (VBR) form with dynamic saliency neglected
% This modeling technique results in a constant stator inductance matrix,
% which provides a significant computational advantage, and is equiva-
% to the Park's representation over a wide range of frequencies.
% The relevant equations can be found in [3].
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Waszynzuk
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% E. A. Walters, "Analysis and simulation of a six-phase
% generator/rectifier system," Proceedings of the 31st IECEC-96,
% Vol. 3, p. 1804-1808.
% [2] H. J. Hegner, P. C. Krause, O. Waszynzuk, E. Walters, S. Pekrek,
% "Parameter measurement of a six-phase synchronous machine for
% simulation of machine/converter systems," Proceedings of the
% dynamic saliency of synchronous machines in power electronic
% based systems," IEEE Transactions on Energy Conversion, Vol. 4,

h = 1; % Define a model handle
start_br_list(h);

%--- System parameters are

%--- Useful constants
pio3 = pi/3.0;
T03 = 2.0/3.0;
rad30 = pi/6;
rad90 = pi/2;
rad120 = 2*pi/3;
rad150 = 5*pi/6;

%--- System parameters are
wb = 1507.0; %--- Base frequency, (rad/sec) for 240 Hz machine.

% transformer parameters
Lipm = 1.0e-3; %--- interphase transformer magnetizing
Lipl = 0.25e-3; %--- interphase transformer leakage
rip = 0.05; %--- interphase transformer resistance

% load parameters
C1 = 1.0e-3;
rl = 12.0; % 0.2 nim

Lls = 13.5e-5; %--- stator leakage inductance
Llm = 4.5e-5; %--- stator leakage inductance
rs = 0.114; %--- stator resistance

Lmq = 1.6e-3; %--- magnetizing inductance q-axis
Lmd = 1.82e-3; %--- magnetizing inductance d-axis

Llfd = 0.255e-3; %--- rotor field leakage inductance
rfd = 0.015; %--- rotor field resistance

rkq1 = -1.69; %--- re-fitted rotor q-axis resistance
rkq2 = 0.1033; %--- re-fitted rotor q-axis resistance
Llkq1 = -7.62e-5; %--- re-fitted rotor leakage q-axis inductance
Llkq2 = 5.4e-5; %--- re-fitted rotor leakage q-axis inductance

rkd1 = 0.118; %--- rotor d-axis resistance
Llkdl = 0.650e-3; %--- rotor leakage d-axis inductance

%--- sub-transient inductances
Lmqpp = 1.0/(1.0/Lmq + 1.0/Llkq1 + 1.0/Llkq2);
Lmdpp = 1.0/(1.0/Lmd + 1.0/Llkdl + 1.0/Llfd);

La = (Lmqpp + Lmdpp)/3.0;
Lb = (Lmdpp - Lmqpp)/3.0;

La11 = Lls + Llm + La;
Lbb11 = Lls + Llm + La;
Lcc11 = Lls + Llm + La;
Lab11 = - La/2.0;
Lac11 = - La/2.0;
Lbc11 = - La/2.0;

Laa22 = Lls + Llm + La;
Lbb22 = Lls + Llm + La;
Lcc22 = Lls + Llm + La;
Lab22 = - La/2.0;
Lac22 = - La/2.0;
Lbc22 = - La/2.0;

Laa12 = (La+T03*Llm)*cos(pio3);
Lbb12 = (La+T03*Llm)*cos(pio3);
Lcc12 = (La+T03*Llm)*cos(pio3);
Lab12 = (La+TO3*Llm)*cos(pi);
Lac12 = (La+TO3*Llm)*cos(pio3);
Lba12 = (La+TO3*Llm)*cos(pio3);
Lbc12 = (La+TO3*Llm)*cos(pi);
Lca12 = (La+TO3*Llm)*cos(pi);
Lcb12 = (La+TO3*Llm)*cos(pio3);

% Syntax of use the BRANCH function.
%   #  h  br  p  n  r/sw   L/C  E/J

S_branch(h, 1,  1,  7,   2,  0); % diode inst 0
S_branch(h, 2,  9,  3,   2,  0); % diode inst 0
S_branch(h, 3,  2,  7,   2,  0); % diode inst 0
S_branch(h, 4,  9,  1,   2,  0); % diode inst 0
S_branch(h, 5,  3,  7,   2,  0); % diode inst 0
S_branch(h, 6,  9,  2,   2,  1); % diode inst 0
S_branch(h, 7,  4,  8,   2,  0); % diode inst 0
S_branch(h, 8,  9,  6,   2,  0); % diode inst 0
S_branch(h, 9,  5,  8,   2,  0); % diode inst 0
S_branch(h, 10, 9,  4,   2,  0); % diode inst 0
S_branch(h, 11, 6,  8,   2,  0); % diode inst 0
S_branch(h, 12, 9,  5,   2,  1); % diode inst 0

L_branch(h, 13, 7, 10, rip, Lipl+Lipm,   0); % interface transformer
L_branch(h, 14, 8, 10, rip, Lipl+Lipm,   0); % interface transformer
L_branch(h, 15, 10, 9, r1,   0.0,   0); % load branches

L_branch(h, 16, 1, 11,   rs,   Laa11,   1); % stator branch
L_branch(h, 17, 2, 11,   rs,   Lbb11,   1); % stator branch
L_branch(h, 18, 3, 11,   rs,   Lcc11,   1); % stator branch
L_branch(h, 19, 4, 12,   rs,   Laa22,   1); % stator branch
L_branch(h, 20, 5, 12,   rs,   Lbb22,   1); % stator branch
L_branch(h, 21, 6, 12,   rs,   Lcc22,   1); % stator branch
S_branch(h, 22, 12, 11,   3,   0); % connecting neutrals

% Define interface transformer mutual inductance
L_smutual(h, 13, 14, -Lipm);
L_smutual(h, 16, 17, Lab11);
L_smutual(h, 16, 18, Lac11);
L_smutual(h, 17, 18, Lbc11);
L_smutual(h, 19, 20, Lab22);
L_smutual(h, 19, 21, Lac22);
L_smutual(h, 20, 21, Lbc22);
L_smutual(h, 16, 19, Laa12);
L_smutual(h, 16, 20, Lab12);
L_smutual(h, 16, 21, Lac12);
L_smutual(h, 17, 19, Lba12);
L_smutual(h, 17, 20, Lbb12);
L_smutual(h, 17, 21, Lbc12);
L_smutual(h, 18, 19, Lca12);
L_smutual(h, 18, 20, Lcb12);
L_smutual(h, 18, 21, Lcc12);
Appendix S: M-File for the 15-kW Power Supply

function h = ps
% This file implements a 15-kW power supply (PS) that is
% composed of three-phase diode rectifier connected
% to DC-to-DC BUCK converter.
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
%

h = 1; % Define a model handle and initialize the ASMG
start_br_list(h);

% System parameters are
wb=377;
L1 = 1e-3; % Source inductance, 1-mH
r1 = 1e-2; % Source inductance resistance, 0.01-Ohms
L2 = 3e-3; % DC inductance, 3-mH
r2 = 3e-2; % DC inductance resistance, 0.03-Ohms
rst = 20.0; % Starting resistor, 20.0-Ohms
C1 = 500e-6; % Rectifier output capacitor, 500-mcF
C2 = 500e-6; % Converter output capacitor, 500-mcF, Implemented outside
r1 = 16.7; % Load resistance, 16-335-Ohms, Implemented outside

% Syntax
%    # hdl br   p  n    r    L/C  E/J
%    #   1  2   3  4    5     6    7
L_branch(h, 1,  1, 2,  r1,   L1,   1); % source branch, Eas
L_branch(h, 2,  1, 3,  r1,   L1,   1); % source branch, Ebs
L_branch(h, 3,  1, 4,  r1,   L1,   1); % source branch, Ecs
S_branch(h, 4,  2, 5,   2,    1); % diode inst 1
S_branch(h, 5,  3, 5,   2,    0); % diode inst 0
S_branch(h, 6,  4, 5,   2,    0); % diode inst 0
S_branch(h, 7,  6, 2,   2,    0); % diode inst 0
S_branch(h, 8,  6, 3,   2,    0); % diode inst 0
S_branch(h, 9,  6, 4,   2,    0); % diode inst 0
L_branch(h,10,  5, 7,  rst,    0,   0); % starting resistor
S_branch(h,11,  5, 7,   1,    0); % switch inst 0
C_branch(h,12,  7, 6,   0,   C1,   0); % Rectifier output capacitor
S_branch(h,13,  5, 8,   2,    0); % controllable switch
S_branch(h, 14, 6, 8, 2, 0); % diode inst 0
L_branch(h, 15, 8, 6, r2, L2, 1); % DC inductor
Appendix T: M-File for the Ship Service Converter Module

function h = sscm
% This file implements a Ship Service Converter Module (SSCM)
% that was built by NPS.
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
%
% Define a handle and initialize the ASMG instance
h = 1; start_br_list(h);

% System parameters are
fs = 20e3;   % Switching frequency, 20 kHz
Vdc = 500;   % DC voltage, 500-V
Vo = 400;    % Output 400-V
Lft = 357e-6; % Input Filter Inductor, 357.0-mkH
RLft = 0.2;  % Input Inductor Resistance, 0.2-Ohms
Cft = 500e-6; % Input Filter Capacitor, 500.0-mcF
RCft = 1.0;  % Series with capacitor Resistance, 1.0-Ohms
Cft2 = 45e-6; % Additional Input Filter Capacitor, 45.0-mcF
L = 1e-3;    % Main Inductance, 1mH
RL = 0.5;    % Inductor Resistance, 0.5-Ohms
C = 500e-6;  % Output Capacitor, 500.0-mcF, implemented externally
rl = 20.0;   % Load resistance, 20-200 Ohms, implemented externally

% Syntax of use the BRANCH function.
% # 1 2 3 4 5 6 7
% # h br p n r/sw L/C E/J
L_branch(h, 1, 2, 1, RLft, Lft, 1); % source and input inductor
L_branch(h, 2, 3, RCft, 0, 0); % series resistor
C_branch(h, 3, 1, 0, Cft, 0); % input capacitor
C_branch(h, 4, 2, 0, Cft2, 0); % additional input capacitor
S_branch(h, 5, 2, 4, 2, 1); % controlled switch
S_branch(h, 6, 1, 4, 2, 0); % diode
L_branch(h, 7, 4, 1, RL, L, 1); % inductor branch

% The output capacitor and load are implemented in Simulink,
% outside of the ASMG block
Appendix U: M-File for the Constant Power Load

function h = cpl
% This file implements a 5-kW Constant Power Load (CPL)
% that is composed of DC-to-DC BUCK converter connected
% to resistive load.
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
% June 12, 2001.
%
    h = 1;  % Define a model handle and initialize the ASMG
    start_br_list(h);

    % System parameters are
    % Vin = 120 - 600-Volts
    % Vout = 100-Volts

    Ls   = 0.0e-6; % Input inductance, 1-mcH
    rs   = 1e-3;  % Input resistance, 1-mOhms
    Lb   = 2e-3;  % DC inductance, 2-mH
    rlb  = 1e-2;  % DC inductance resistance, 0.01-Ohms
    Cin  = 470e-6; % Rectifier output capacitor, 470-mcF
    Cout = 470e-6; % Converter output capacitor, 470-mcF, Implemented outside
    rl   = 20.0;  % Load resistance, 16-335-Ohms, Implemented outside

    % Syntax
    % # hdl br p n r L/C E/J
    % #  1 2  3  4  5     6    7
    %
    % L_branch(h, 1,  2, 1, rs,   Ls,   1); % source branch, Vin
    % C_branch(h, 2,  2, 1,   0,  Cin,   0); % Input capacitor, Cin
    %
    % S_branch(h, 3,  2, 3,    2,    1); % controlled switch
    % S_branch(h, 4,  1, 3,    2,    0); % diode
    %
    % L_branch(h, 5,  3, 1,  rlb,  Lb,   1); % DC inductor branch

    % The output capacitor and load are implemented in Simulink,
    % outside of the ASMG block
Appendix V: M-File for the Ship Service Inverter Module

function h = ssim
% This file implements a circuit part of the Ship Service Inverter
% Module (SSIM) for the DC Zonal Electrical Distribution System (ZEDS)
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% November 16, 2001.
%
% Define a model handle and initialize the ASMG
h = 1;
start_br_list(h);

% System parameters are

% Vin = 300-600-Volts
Ls = 0.0e-6; % Input series inductance, 1-mcH
Rs = 10.0e-3; % Input series resistance, 10-mOhms
Rcin = 127.0e-3; % Input capacitor series resistance, 127-mOhms
Cin = 590e-6; % Input capacitor, 590-mcF
Lac = 550.0e-6; % AC filter inductance, 550-mcH
RLac = 38.0e-3; % AC filter inductor resistance, 38-mOhms
Cac = 50.0e-6; % AC filter capacitor, 50-mcF
RCac = 8.0e-3; % AC filter capacitor series resistance, 8-mOhms
Rld = 15.9; % AC load resistance, 15.9-Ohms
Lld = 4.0e-6; % AC load inductance, 4-mcH

% Syntax
% # hdl br p n r L/C E/J
% # 1 2 3 4 5 6 7
L_branch(h, 1, 3, 1, Rs, Ls, 1); % source branch, Vin
L_branch(h, 2, 3, 2, Rcin, 0, 0); % series resistor, Rcin
C_branch(h, 3, 2, 1, 0, Cin, 0); % Input capacitor, Cin
S_branch(h, 4, 3, 4, 2, 1); % controlled switch
S_branch(h, 5, 4, 1, 2, 0); % controlled switch
S_branch(h, 6, 4, 3, 2, 0); % blocking diode
S_branch(h, 7, 1, 4, 2, 0); % blocking diode
S_branch(h, 8, 3, 5, 2, 1); % controlled switch
S_branch(h, 9, 5, 1, 2, 0); % controlled switch
S_branch(h, 10, 5, 3, 2, 0); % blocking diode
S_branch(h, 11, 1, 5, 2, 0); % blocking diode
Appendix V: M-File for the Ship Service Inverter Module

S_branch(h,12,  3, 6, 2, 1);  % controlled switch
S_branch(h,13,  6, 1, 2, 0);  % controlled switch
S_branch(h,14,  6, 3, 2, 0);  % blocking diode
S_branch(h,15,  1, 6, 2, 0);  % blocking diode

L_branch(h,16,  4, 7,RLac, Lac, 0);  % AC filter inductor branch
L_branch(h,17,  5, 8,RLac, Lac, 0);  % AC filter inductor branch
L_branch(h,18,  6, 9,RLac, Lac, 0);  % AC filter inductor branch

C_branch(h,19,  7, 10, 0, Cac, 0);  % AC filter capacitor, Cac
L_branch(h,20, 10, 8,RCac, 0, 0);  % Capacitor series resistance, rl

C_branch(h,21,  8, 11, 0, Cac, 0);  % AC filter capacitor, Cac
L_branch(h,22, 11, 9,RCac, 0, 0);  % Capacitor series resistance, rl

C_branch(h,23,  9, 12, 0, Cac, 0);  % AC filter capacitor, Cac
L_branch(h,24, 12, 7,RCac, 0, 0);  % Capacitor series resistance, rl

S_branch(h,25,  7, 13, 3, 1);  % LB switch, connecting load
S_branch(h,26,  8, 14, 3, 1);  % LB switch, connecting load
S_branch(h,27,  9, 15, 3, 1);  % LB switch, connecting load

L_branch(h,28, 13, 14, Rld, Lld, 0);  % AC Load, phase ab
L_branch(h,29, 14, 15, Rld, Lld, 0);  % AC Load, phase bc
L_branch(h,30, 15, 13, Rld, Lld, 0);  % AC Load, phase ca
Appendix W: M-File for the Motor Controller Inverter

function h = mc_inverter
% This file implements an inverter part of the Motor Controller (MC)
% for the DC Zonal Electrical Distribution System (ZEDS)
%
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
%
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
%
% November 26, 2001.
%
% Define a model handle and initialize the ASMG
start_br_list(h);
%
% System parameters are
%
% Vin = 300-600-Volts
Lin  = 0.0e-6; % Input series inductance, 1-mcH
Rin  = 10.0e-3; % Input series resistance, 10-mOhms
Cin  = 1.0e-3; % Rectifier output capacitor, 1-mF
Rld  = 1.0e4; % AC side series resistance, 10-kOhms
%
% Syntax
%    # hdl br   p n  r  L/C  E/J
%    #   1  2   3  4    5     6    7
%
L_branch(h, 1,  2, 1, Rin,  Lin,   1); % source branch, Vin
C_branch(h, 2,  2, 1,   0,  Cin,   0); % input capacitor, Cin
S_branch(h, 3,  2, 3,   2,    0);      % controlled switch
S_branch(h, 4,  3, 1,   2,    0);      % controlled switch
S_branch(h, 5,  3, 2,   2,    0);      % blocking diode
S_branch(h, 6,  1, 3,   2,    0);      % blocking diode
S_branch(h, 7,  2, 4,   2,    0);      % controlled switch
S_branch(h, 8,  4, 1,   2,    0);      % controlled switch
S_branch(h, 9,  4, 2,   2,    0);      % blocking diode
S_branch(h,10,  1, 4,   2,    0);      % blocking diode
S_branch(h,11,  2, 5,   2,    0);      % controlled switch
S_branch(h,12,  5, 1,   2,    0);      % controlled switch
S_branch(h,13,  5, 2,   2,    0);      % blocking diode
S_branch(h,14,  1, 5,   2,    0);      % blocking diode
C_branch(h,15,  3, 6, Rld,    0,   1); % AC side, connection for phase a
C_branch(h,16,  4, 6, Rld,    0,   1); % AC side, connection for phase b
C_branch(h,17,  5, 6, Rld,    0,   1); % AC side, connection for phase c
Appendix X: M-file Containing Induction Motor Parameters

% This script file contains parameters for the SIMULINK
% simulation of a 5 Hp Induction Machine. The parameters
% have been estimated based on information in [1, page 190]
% for machines of similar ratings.
% 
% Developed for Naval Combat Survivability Effort
% by J. Jatskevich and O. Wasynczuk
% 
% Purdue University, School of Electrical and Computer Engineering
% West Lafayette, Indiana 47907-1285.
% 
% November 26, 2001.
% 
% [1] P.C. Krause, O. Wasynczuk, S. D. Sudhoff,
%

% Initial conditions set for the start-up study.

wric    = 377;   % Initial speed
siqsic  = 0.0;
siqric  = 0.0;
sidsic  = 0.0;
sidric  = 0.0;
thetaic = 0.0;

wb    = 377;          % 60 Hz base frequency
Tb    = 19.83;
J     = 0.089;
P     = 4;
H     = 0.5*(2/P)*J*wb/Tb;         % Mechanical time constant, sec.
rs  = 0.315;
xls = 0.546;
xm  = 18.92;
xlr = 0.546;
rr  = 0.591;
 xmm = 1/((1/xm)+(1/xls)+(1/xlr));
Tour = (xlr + xm)/(wb*rr);         % Rotor electrical time constant
xrr = (xlr + xm);