Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240

APPLICATION REPORT: SPRA282

Mohammed S Arefeen
Automotive/Industrial Applications

Digital Signal Processing Solutions
September 1998
IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain application using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage (“Critical Applications”).

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

Copyright © 1998, Texas Instruments Incorporated
TRADEMARKS

TI is a trademark of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.
CONTACT INFORMATION

US TMS320 HOTLINE    (281) 274-2320
US TMS320 FAX        (281) 274-2324
US TMS320 BBS        (281) 274-2323
US TMS320 email      dsph@ti.com
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>8</td>
</tr>
<tr>
<td>Switched Reluctance Motor Configuration</td>
<td>9</td>
</tr>
<tr>
<td>Basic Principles of Operation</td>
<td>11</td>
</tr>
<tr>
<td>Converter Topologies</td>
<td>13</td>
</tr>
<tr>
<td>Control Strategies</td>
<td>15</td>
</tr>
<tr>
<td>Voltage Controlled Drive</td>
<td>15</td>
</tr>
<tr>
<td>Current Controlled Drive</td>
<td>16</td>
</tr>
<tr>
<td>Controlling an SR motor using TMS320F240</td>
<td>17</td>
</tr>
<tr>
<td>Hardware Setup</td>
<td>17</td>
</tr>
<tr>
<td>Current and Voltage Sensing</td>
<td>19</td>
</tr>
<tr>
<td>Position Sensing</td>
<td>20</td>
</tr>
<tr>
<td>Commutation Sequence and Instants</td>
<td>21</td>
</tr>
<tr>
<td>Bi-directional Operation</td>
<td>22</td>
</tr>
<tr>
<td>Changing Conduction Angle</td>
<td>23</td>
</tr>
<tr>
<td>Current Regulation</td>
<td>24</td>
</tr>
<tr>
<td>Startup Operation</td>
<td>24</td>
</tr>
<tr>
<td>Brief Software Description</td>
<td>26</td>
</tr>
<tr>
<td>Experimental Results</td>
<td>30</td>
</tr>
<tr>
<td>References</td>
<td>34</td>
</tr>
<tr>
<td>Appendix A. Motor Specifications</td>
<td>35</td>
</tr>
<tr>
<td>Appendix B. Switched Reluctance Motor Drive code</td>
<td>36</td>
</tr>
<tr>
<td>Appendix C. Header Files</td>
<td>64</td>
</tr>
<tr>
<td>C240APP.H</td>
<td>64</td>
</tr>
<tr>
<td>SD_init.H</td>
<td>69</td>
</tr>
<tr>
<td>Appendix D. LINKER COMMAND FILE</td>
<td>71</td>
</tr>
<tr>
<td>Appendix E. C240tbed.cmd</td>
<td>72</td>
</tr>
</tbody>
</table>
### Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Stator and rotor configurations of SR motor (top) 4 phase 8/6 SR motor (bottom) 3 phase 6/4 SR motor.</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Inductance profile of an SR motor. (a) Variation of inductance with rotor position. (b) Phase current for motoring operation. (c) Phase current for generating operation.</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Power Converter for a Current Controlled Switched Reluctance Motor Drive</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Voltage Controlled Switched Reluctance Motor Drive</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Current controlled Switched Reluctance Motor Drive</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Top view of TMS320F240 Evaluation Module.</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Top view of Motor Interface Board for TMS320F240-EVM</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Top view of classical converter that plugs on to EVM motor interface board shown in Figure 6.</td>
</tr>
<tr>
<td>Figure 9</td>
<td>Counting four edges for every cycle of channels A/B.</td>
</tr>
<tr>
<td>Figure 10</td>
<td>Rotor rotates 15 mechanical degrees from nonaligned to aligned position.</td>
</tr>
<tr>
<td>Figure 11</td>
<td>QEP configuration for bi-directional operation of the motor.</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Flowchart for startup operation.</td>
</tr>
<tr>
<td>Figure 13</td>
<td>Current loop and QEP loop</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Flowchart for ADC measurement.</td>
</tr>
<tr>
<td>Figure 15</td>
<td>Phase A current and voltage for a voltage controlled SR drive.</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Phase A current is regulated 25 kHz PWM (2A/div.)</td>
</tr>
<tr>
<td>Figure 17</td>
<td>Expanded view of phase A current shown in Figure 16.</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Phase A current during high speed operation of the motor.</td>
</tr>
</tbody>
</table>
Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240

Abstract

The DSP controller TMS320F240 from Texas Instruments is suitable for a wide range of motor drives and power electronics applications. TMS320F240 reduces chip count of a system by integrating most of the important power electronics peripherals on chip and thus increases system reliability and reduces system cost. This application note describes complete implementation of a current controlled Switched Reluctance Motor Drive using TMS320F240.
Introduction

The Switched Reluctance motor (SR) drive is receiving renewed attention as a viable candidate for various adjustable speed and torque applications. Combining the unique features of an SR motor with its relatively simple and efficient power converter results in a motor drive that may be preferable for many applications compared to other ac or dc drive systems. The following are the main advantages of an SR motor drive:

- Simple and low-cost machine construction due to the absence of rotor winding and permanent magnets.
- No shoot-through faults between the dc buses in the SR motor drive converter since each stator winding is connected in series with converter switching elements.
- Unidirectional current required by the motor drive makes power electronics drive circuitry simple and reliable.
- Low rotor inertia and high torque/inertia ratio.
- Stator phases can be controlled independently. This makes SR drive more robust than other motor drives. Failure or loss of one of the stator phases does not prevent drive operation.
- SR motor can be operated in a harsh environment.

The main disadvantages of SR motor drives are the following:

- High torque ripple.
- Higher acoustic noise level than other motors.

However, advanced motor design techniques and high performance algorithms are successfully addressing the above mentioned disadvantages and SR motor drives are becoming more and more suitable for a wide range of applications.
Switched Reluctance Motor Configuration

The switched reluctance motor (SR) is a doubly salient machine with independent phase windings on the stator and a solid laminated rotor. The stator windings on diametrically opposite poles are connected in series to form one phase of the motor. Figure 1 (a) and Figure 1 (b) shows a four phase 8/6 switched reluctance motor and a three phase 6/4 switched reluctance motor respectively. When a stator phase is energized, the most adjacent rotor pole-pair is attracted towards the energized stator in order to minimize the reluctance of the magnetic path. Therefore, by energizing consecutive phases in succession it is possible to develop constant torque in either direction of rotation.

Figure 1. Stator and rotor configurations of SR motor (top) 4 phase 8/6 SR motor (bottom) 3 phase 6/4 SR motor.
Theoretically it is possible to have a number of stator and rotor pole combinations. However, certain combinations, such as 4/4 or 2/2, will have problems during start up operation. With a combinations like 2/2 or 4/4 it will be impossible to develop a starting torque with rotor and stator pole exactly aligned. Although, the configurations with higher number of stator/rotor pole combinations have less torque ripple and do not have the problem of starting torque, 6/4 or 8/6 combinations are typically used.

A well designed Switched reluctance motor will minimize the core losses, will offer good starting capability, and will also minimize the unwanted effects due to varying flux distributions and saturation and to eliminate mutual coupling. The choice of the number of phases is open but increasing the number of phases would increase the number of power devices needed for power converter. Moreover, higher number of poles will decrease the maximum inductance ratio obtainable for a good torque per ampere. These practical issues limit stator and pole ratio to 6/4 or 8/6 in most applications of Switched Reluctance motor drives.
Basic Principles of Operation

The phase voltage equation in a Switched reluctance motor can be written as:

$$V = iR + \frac{d\lambda}{dt}$$  \hspace{1cm} \text{Equation 1}

where, $V$ is the dc bus voltage, $i$ is the instantaneous phase current, $R$ is the phase winding resistance and $\lambda$ is the flux linking the phase coil. Ignoring stator resistance, Equation 1 can also be written as:

$$V = L(\theta) \frac{di}{dt} + i \frac{dL(\theta)}{d\theta} \omega$$  \hspace{1cm} \text{Equation 2}

where, $\omega$ is the rotor speed, $\theta$ is the rotor angular position, and $L(\theta)$ is the instantaneous phase inductance. The rate of flow of energy can be obtained by multiplying the voltage with current and can be written as:

$$Vi = L_i \frac{di}{dt} + i^2 \frac{dL_i}{d\theta} \omega$$  \hspace{1cm} \text{Equation 3}

or

$$P = \frac{d}{dt} \left( \frac{1}{2} Li^2 \right) + \frac{1}{2} i^2 \frac{dL_i}{d\theta} \omega$$  \hspace{1cm} \text{Equation 4}

The first term of the above equation represents the rate of increase in the stored magnetic field energy while the second term is the mechanical output. Thus, the instantaneous torque can be written as:

$$T(\theta, i) = \frac{1}{2} i^2 \frac{dL_i}{d\theta}$$  \hspace{1cm} \text{Equation 5}

$$T(\theta, i) = \frac{1}{2} i^2 \frac{dL_i}{d\theta}$$  \hspace{1cm} \text{Equation 6}
Figure 2 shows an idealized inductance profile of an SR motor. In order to obtain motoring torque, phase current is switched on during the rising period of phase inductance. Generating operation or braking torque can be obtained by switching phase current during the decreasing period of phase inductance. It is obvious from Figure 2 that in order to obtain optimum performance, switching of phase currents must be done accurately. This is why rotor position information (which can be obtained either by using position sensors or the position can be estimated by implementing position sensorless schemes) is always necessary to operate an SR motor drive. Rotor position information is fed back to the controller to determine the phase commutation sequence and instants.

Figure 2. Inductance profile of an SR motor. (a) Variation of inductance with rotor position. (b) Phase current for motoring operation. (c) Phase current for generating operation.
Converter Topologies

Switched reluctance motors do not require bi-directional current like other common ac motors. Therefore, unipolar converters are used as the power converter for SR motor drives. The converter topology for any specific SR motor depends on the motor construction as well as on the application. There are many types of converters available for different types of applications. Discussing all those converters is beyond the scope of this application note.

Figure 3 shows the converter that is used in this application note. This particular converter is a modified version of a classical SR motor converter. The switches and diodes are rated for the supply voltage with any required safety factor. During motoring operation, both power switching devices are turned on for a particular phase. Pulse Width Modulation (PWM) can be obtained either by switching both or single power devices. At commutation, both power devices are turned off, and the phase gets defluxed quickly through the freewheeling power diodes. This converter provides independent control for each phase and consequently phase overlap operation can be implemented easily. Two current sensors are available for phase current measurement. DC bus voltage and DC bus current can also be measured in order to implement advanced control techniques. These two variables (DC bus voltage and DC link current) are directly interfaced with TMS320F240 as shown in Figure 3.
Figure 3. Power Converter for a Current Controlled Switched Reluctance Motor Drive
Control Strategies

An SR motor drive is controlled by proper positioning of the phase current pulses relative to the rotor position. The turn on timing and the total conduction period determines torque, efficiency and other performance characteristics.

At low speed, phase current builds up very quickly after turn-on due to the negligible back-EMF, and the current must be limited by either controlling the average voltage or regulating the current level.

Voltage Controlled Drive

In low-performance drives, where precise torque control is not a critical issue, fixed-frequency PWM voltage control with variable duty-cycle provides the simplest means of controlling an SR motor. A highly efficient, variable speed drive having a wide speed range can be achieved with SR motor by using voltage PWM with closed loop position control only. Figure 4 shows a block diagram of such a system. The angle controller generates the turn-on and turn-off angles for a phase, i.e., determines the conduction period, depending on the instantaneous rotor position. The duty-cycle is changed according to the voltage command signal. A speed feedback loop can be added on the outside as shown in the figure for a good speed control.

*Figure 4. Voltage Controlled Switched Reluctance Motor Drive*
Current Controlled Drive

Figure 5 shows a torque controlled switched reluctance drive. The torque command is executed by regulating the current in the inner loop as shown in the Figure 5. The reference current $I^*$ for a given operating point is determined from the load characteristics. The controller needs current feedback information from each of the motor phases. Current controlled SR drive provides fast motor response.

Figure 5. Current controlled Switched Reluctance Motor Drive
Controlling an SR motor using TMS320F240

The DSP controller TMS320F240 of Texas Instruments is capable of controlling an SR motor. This application note presents a complete current controlled SR motor drive implemented using TMS320F240. TMS320F240 has nine independent PWM channels, providing maximum flexibility for SR motor control. Built-in Quadrature Encoder Pulse (QEP) module makes encoder interfacing to TMS320F240 simple. Quadrature pulses A and B can directly be connected to a TMS320F240 by configuring Capture pins 1 and 2 as QEP input pins. TMS320F240 as dual ADC module with eight channels in each ADC. Therefore, phase currents can be read simultaneously. Two variables can be converted in 6.6 μS.

A current controlled SR motor drive is implemented by using Evaluation Module (EVM) for TMS320F240 interfaced with a motor interface board and power converter board. Motor interface and power converter boards are designed by Spectrum Digital. Figure 5 shows the converter used SR drive. The following sections describe all major components of the above mentioned hardware setup.

Hardware Setup

The SR drive was implemented using three boards. The first board, as shown in Figure 6, is the evaluation module for TMS320F240 introduced by Texas Instruments. The second board, the motor interface board, introduced by Spectrum Digital, plugs into TI’s EVM board. This motor interface board has all the drivers for power MOSFETs, current/voltage sensing circuitry, QEP input port, over current protection circuitry, user interface LEDs and user input DIP switch. Figure 7 shows the top view of motor interface card. The power converter board, introduced by Spectrum Digital, plugs into the interface board. The converter board, as shown in Figure 8, has all power MOSFETs, current sensing resisters, and heat sink. The following sections describe major blocks of this hardware setup.
Figure 6. Top view of TMS320F240 Evaluation Module.

Figure 7. Top view of Motor Interface Board for TMS320F240-EVM
Current and Voltage Sensing

Different currents and voltages can be sensed for control purposes. In this particular implementation, phase currents are sensed to implement current controlled SR motor drive. TMS320F240 has a dual ADC that enables a user to simultaneously sample/hold and convert two variables. Total time required for this sample/hold and conversion is 6.6μS. Two current sensing resistors are used to sense phase currents. The equation for current sensing circuit is:

\[
\text{current A} = \frac{0.005 \times 12}{2} = \frac{0.06}{2} = 0.03 \text{ A}
\]

The current rating of the converter is ±40A. The ADC reading for a +40A phase current will be 0 and for a -40A will 3FFh. Zero phase current is read as 1FFh. Channel 3 of ADC 1 is connected to the current sense resister of phases B&D. Channel 2 of ADC 2 is connected to the current sense resister of phases A&C.

DC bus voltage is also sensed and can be used to implement any advanced features. Channel 4 of ADC 1 is connected to DC bus voltage sensing circuit. The equation for DC bus voltage sensing is:

\[
\left(\text{DC bus voltage} \times \frac{1}{151}\right) \times 12 = A / D \text{ volts}
\]
The phase currents are sensed at every 40uS to implement a 25kHz current loop. The new PWM duty cycles obtained from current information is loaded at the beginning of a PWM cycle. This is achieved by programming ACTR and SACTR control registers of TMS320F240. ACTR and SACTR registers are programmed to load compare (CMPRx, x=1,2,3) and simple compare (SCMPRx, x=1,2,3) registers at T1 counter = 0. This synchronizes PWM operation.

**Position Sensing**

Information of phase inductance variation is always needed for proper operation of an SR motor drive. In a typical SR motor drive, this phase inductance variation is sensed by sensing rotor position information. Rotor position information can be obtained by using position encoders. Phase commutation sequence and instants are determined from position information.

TMS320F240 has a QEP (Quadrature Encoder Pulse) circuit that enables a user to directly interface A/B channels of an encoder. QEP circuit can be associated with timer 2 or 3 of TMS320F240 (if needed timers 2&3 can be cascaded for 32 bit operation). Timer count value will increase or decrease (depending on the direction of rotation) with every edge of channels A and B of encoder. Therefore, TMS320F240 counts four ticks for every cycle of channel A and B as shown in Figure 9.

*Figure 9. Counting four edges for every cycle of channels A/B.*

A 360 line encoder is used for rotor position information feedback. Timer T3 of TMS320F240 is associated with QEP operation. Timer T3 will count $\frac{360 \times 4}{1440}$ ticks for one mechanical revolution of the rotor.
Commutation Sequence and Instants

Rotor position information is used to generate correct commutation sequence and instants. A 8/6 SR motor is used for this drive implementation. Therefore, from an aligned position, the rotor will move 15 mechanical degrees before getting aligned with the next phase. This situation is shown in Figure 10. For this particular case, the rotor is aligned with phase B and, in order to have a rotation in the clock wise direction, phase C needs to be energized. After energizing phase C, the rotor will rotate $15^\circ$ mechanical degrees before getting aligned with phase C when phase D will have to be energized to sustain rotor rotation in the same direction. QEP counter in TMS320F240 will count $15 \times 4 = 60$ ticks for one stroke of rotation. In order to identify each commutation instant, timer3 is programmed to interrupt at every 60 counts. The interrupt service routine then determines the commutation sequence depending on the present state and directional input from user. For example, after energizing phase B, QEP counter will be reset. The timer will interrupt after 60 counts and the program will enter the appropriate interrupt service routine (ISR). In that ISR, the directional input will be checked and if the directional input stays unchanged then the program will commutate to phase C. In case of directional change, the commutation will be made to phase A.
Figure 10. Rotor rotates 15 mechanical degrees from nonaligned to aligned position.

Bi-directional Operation

Bi-directional operation of the motor can be easily accommodated using QEP circuit of TMS320F240. For this particular SR motor, phase commutation needs to happen at every 15 mechanical degrees, which is equivalent to 60 counts of QEP counter, T3. Initially timer T3 is initialized at the middle of its full range, i.e. at 0x7FFFh. The period register of T3 is loaded with 0x803Bh (0x7FFFh + 0x3Ch). The compare register of T3 is loaded with 0x7FC3h (0x7FFFh - 0x3Ch). For clockwise rotation, T3 will generate period interrupt after every 60 counts indicating commutation instant. The interrupt service routine will then determine new commutation sequence and will reset the count to 0x7FFFh and will wait for next interrupt. In case of counter clockwise rotation, T3 will generate timer compare interrupt after every 60 counts indicating commutation instant. Interrupt service routine will then take care of necessary commutation sequence. This operation is shown in Figure 11. In TMS320F240, period and compare interrupts of timer T3 belong to the same group (Group B). Therefore, both period and compare interrupts of T3 can be handled with one ISR.
At every T3 interrupt, the software looks for directional input from user. The input DIP switch available in TMS320F240 EVM is used to provide this input. This DIP switch is located in I/O space at the location 0x000Ch. The ON position of switch one indicates clock wise rotation and OFF position indicates anti clock wise rotation. As mentioned earlier, commutation sequence A-B-C-D is needed for CW rotation and D-C-B-A is needed for CCW. Directional input can be changed any time of operation, however, the drive is implemented to change direction only from commutation points. Therefore, in the worst case, the directional command will have an error of 15 mechanical degrees.

**Changing Conduction Angle**

In this implementation, the SR drive uses a fixed 15-degree stroke angle. Often, users may want to vary this angle to accommodate different conduction angles and more sophisticated current control scheme. The variation of conduction angle can be achieved by loading appropriate periods and compare values into the timer’s period and compare registers. The shadow feature of compare and period registers, which allows the register values to be modified “on the fly”, makes this update procedure convenient.
Current Regulation

Current regulation is achieved by fixed frequency Pulse Width Modulation signals with varying duty cycles. During energizing of a particular phase, free wheeling of phase current is done by pulsing one switch while the other switch is kept ON for whole PWM period. During commutation both switches are turned OFF in order to achieve maximum rate of current decay (with both switches OFF, phase current has to flow against DC bus). PWM width is determined by comparing the measured phase current with the desired reference current as shown below -

\[ I_{error} = I_{ref} - I_{actual} \]

\[ duty\_cycle_{new} = duty\_cycle_{old} + error \times K \]

if \( duty\_cycle_{new} \geq Timer\_period \)
then
\[ duty\_cycle_{new} = Timer\_period \]

if \( duty\_cycle_{new} \leq 0 \)
then
\[ duty\_cycle_{new} = 0 \]

where \( K \) is the proportional gain and depends on motor parameters and also on DC bus voltage and currents. The proportional gain \( K \) can be determined using the following procedure.

Let ‘S’ be the number of steps allowed in one PWM cycle. Let \( \Delta i \) be the change in phase current for 100% change in PWM duty cycle. The proportional gain, \( K \), can be defined as:

\[ K = \frac{S}{\Delta i} \]

The parameter \( \Delta i \) depends on motor and converter types.

Startup Operation

In most of the applications incremental position encoders are used over absolute encoders in order to save system cost. However, using incremental encoder for position information requires some sort of startup technique to determine starting position of the rotor.

There are a number of techniques available for this startup operation. These startup techniques are dependent on system and load types.
The switched reluctance drive implemented for this application note energizes a particular phase pair to align the rotor with that phase. This gives the controller a starting reference. Here, phase A is first energized for couple of PWM periods, so that that rotor gets aligned with phase A. The controller then comes out of that starting subroutine and goes in normal operation knowing phase A is the starting position.

The controller then checks the rotational input from the user. For CW rotation, commutation is made to phase B and for CCW rotation commutation is made to phase D.

Although startup technique used in this application note is simple, specific load configuration may not allow this technique to be used. For example, this technique cannot be applied to any load that cannot tolerate any reverse rotation during startup. This is because, when phase A is energized, the rotor will get aligned with phase A but the rotational direction cannot be guaranteed in doing so. (See the flowchart in Figure 12.)
Brief Software Description

The software written on TMS320F240 EVM implements all the control blocks described in previous sections. There two loops in the software. First, the current loop, which runs at 25kHz and checks and regulates phase current according to the reference. Second the QEP loop that checks the position and determines the proper rotation direction and proper commutation sequence. QEP loop generates an interrupt after every 15 mechanical degree rotation of rotor. For current regulation, PWM always starts with 100% duty cycle. This helps to buildup phase current quickly. (See the flowcharts in Figure 13 and Figure 14.)
Figure 12. Flowchart for startup operation

If the user wants to obtain a specific rotational direction during startup, then other available techniques have to be applied.
Figure 13. Current loop and QEP loop

T1 Interrupt (Current Loop)

Commutation Happened?

Yes

Direction?

FORWARD

Increment Commutation by 1

Commutation > D

Yes

Commutate to A

Decrement Commutation by 1

Commutation < A

Yes

Commutate to D

Call DATAINT

Load Appropriate Address from Branch Table and Branch

PHASE_x
x = A, B, C or D

Load Phase Current CURRxx
xx = AC or BD

Load Command CMDxx
xx = AC or BD

CMDxx >= CURRxx

Yes

Increase Current COMxx = COMxx - K
xx = AC or BD

COMxx <= 0
xx = AC or BD

Yes

COMxx = MIN
xx = AC or BD

Decrease Current COMxx = COMxx + K
xx = AC or BD

COMxx >= PER
xx = AC or BD

Yes

COMxx = MAX
xx = AC or BD

Load Appropriate Action Control Registers

Current Loop is Complete RETURN

T3 Interrupt (Period or Compare)

Initialize T3 Counter T3CNT = 0x7FFF

GPR3 = 0

T3 Interrupt Complete RETURN
Figure 14. Flowchart for ADC measurement.

1. DATAINT
2. Start a Conversion
3. Read results from previous conversion
4. Load Appropriate Registers with ADC results
   - CURRAC=CH2 of AD2
   - CURRBD=CH3 of AD1
5. RETURN
Experimental Results

Figure 15 shows phase A current and phase A voltage of an SR motor with voltage control. It is obvious that the response with voltage control is slow as phase current builds up slowly as seen in Figure 15.

*Figure 15. Phase A current and voltage for a voltage controlled SR drive.*
Figure 16. Phase A current is regulated 25 kHz PWM (2A/div.)
Figure 17. Expanded view of phase A current shown in Figure 16.
Figure 18. Phase A current during high speed operation of the motor.

Figure 16 shows phase A current of SR motor. Phase current is regulated at 3A. Figure 17 shows the expanded view of the same phase current. It is obvious that the phase current is regulated at the reference current. Figure 18 shows phase A current during high speed operation. Note: no PWM is present during high speed operation. Phase current does not get sufficient time to reach the reference value during this mode of operation.
References


Appendix A. Motor Specifications

1) Number of phases: 4
2) Number of stator poles: 8
3) Number of rotor poles: 6
4) Average phase resistance: 445 mΩ
5) Average inductance (aligned): 3.56 mH
6) Average inductance (unaligned): 0.64 mH
7) Voltage: 12VDC
8) Current: 5ADC
9) Encoder: 360 lines
Appendix B. Switched Reluctance Motor Drive code

;*******************************************************************************
; File Name : SR.asm
; Project : x240 silicon debug
; Originator : M. S. Arefeen (Texas Instruments)
;
; Target System : F240 EVM
;
; A/C UPPER : PWM 5 - CMPR3
; B/D UPPER : PWM 8 - SCMPR2
; A LOWER : PWM 3 - CMPR2
; B LOWER : PWM 7 - SCMPR1
; C LOWER : PWM 9 - SCMPR3
; D LOWER : PWM 1 - CMPR1
;
;*******************************************************************************

; Status : complete
; Last update : 03/20/97
; Started on : 12/17/96
________________________________________________________
| Date of Mod | Description
|---------------------------------------------------------|
| 12/17/96 | Created SR.asm file
| 01/29/97 | Variable speed voltage control is added
| 02/13/97 | SR motor is connected. Stroke angle is 15 degrees, however, encoder line is 92
| 02/17/97 | Simple current control mode is added
| 02/24/97 | Speed reversal is added
| 02/27/97 | Current loop is added with P control

; Debug derivatives


.def T1CNT
.def T2CNT
.def T3CNT
.def T1PER
.def T2PER
.def T3PER
.def T1CMP
.def T2CMP
.def T3CMP
.def GPTCON
.def T1CON
.def T2CON
.def T3CON
.def CMPR1
.def CMPR2
.def CMPR3
.def SCMPR1
.def SCMPR2
.def SCMPR3
.def ACTR
.def SACTR
.def COMCON
.def CAPCON
.def CAPFIFO
.def FIFO1
.def FIFO2
.def FIFO3
.def FIFO4
.def IFRA
.def IFRB
.def IFRC
.def IMRA
.def IMRB
.def IMRC
.def IVRA
.def IVRB
.def IVRC

; Peripheral Registers
; Initialize the SD (System) Module for shared pins

.include "c240app.h"
.include "ev_test.h"
.include "SD_init.h"

; Macro definitions

KICK_DOG .macro ; Watchdog reset macro
    LDP #00E0h
    SPLK #05555h, WD_KEY
    SPLK #0AAAAh, WD_KEY
    LDP #0h
.endm

; Reset & interrupt vectors

.rsvect B START ; PM 0 Reset Vector
.int1 B PHANTOM ; PM 2 Int level 1
.int2 B T1PERIOD_ISR ; PM 4 Int level 2
.int3 B T3PERIOD_ISR ; PM 6 Int level 3
.int4 B PHANTOM ; PM 8 Int level 4
.int5 B PHANTOM ; PM A Int level 5
.int6 B PHANTOM ; PM C Int level 6
.reserved B PHANTOM ; PM E (Analysis Int)
.sw_int8 B PHANTOM ; PM 10 User S/W int
.sw_int9 B PHANTOM ; PM 12 User S/W int
.sw_int10 B PHANTOM ; PM 14 User S/W int
.sw_int11 B PHANTOM ; PM 16 User S/W int
SW_INT12 B PHANTOM ; PM 18 User S/W int
SW_INT13 B PHANTOM ; PM 1A User S/W int
SW_INT14 B PHANTOM ; PM 1C User S/W int
SW_INT15 B PHANTOM ; PM 1E User S/W int
SW_INT16 B PHANTOM ; PM 20 User S/W int
TRAP B PHANTOM ; PM 22 Trap vector
NMI B PHANTOM ; PM 24 Non maskable Int
EMU_TRAP B PHANTOM ; PM 26 Emulator Trap
SW_INT20 B PHANTOM ; PM 28 User S/W int
SW_INT21 B PHANTOM ; PM 2A User S/W int
SW_INT22 B PHANTOM ; PM 2C User S/W int
SW_INT23 B PHANTOM ; PM 2E User S/W int

.def ADDRESS
.def GPR0
.def GPR1
.def RESULT1
.def RESULT2
.def CMD
.def CMDAC,1
.def CMDBD,1
.def MAX,1
.def MIN,1
.def COMAC,1
.def COMBD,1
.def CURRAC,1
.def CURRBD,1

.bss ADDRESS,1
.bss GPR0,1
.bss GPR1,1
.bss GPR2,1
.bss GPR3,1
.bss GPR5,1
.bss RESULT1,1
Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240

```
.bss RESULT2,1
.bss CMD,1
.bss CMDAC,1
.bss CMDBD,1
.bss MAX,1
.bss MIN,1
.bss COMAC,1
.bss COMBD,1
.bss CURRAC,1
.bss CURRBD,1
.bss NUM,1

AC2CNTL0 .set 7032h
AC2CNTL1 .set 7034h
AC2DR0 .set 7036h
AC2DR1 .set 7038h
LEDS .set 000Ch
SWITCH .set 0008h

;*******************************************************************
; MAIN CODE - starts here
;*******************************************************************
.text
START
; SPLK #PSA_FB_OFF,ABRPT ; Turn PSA and FEEB off
SETC INTM ; Set global interrupt mask
LDP #00E0h
SPLK #006Fh, WD_CNTL ;Disable WD if VCCP=5V
KICK_DOG

;*******************************************************************
; Set clock control register to x2 clock mode                      *
;*******************************************************************
LDP #00E0h
SPLK #00CCh, CKCR1 ;CLKIN(XTAL)=8MHz,CPUCLK=20MHz
SPLK #00C3h, CKCR0 ;CLKMD=PLL,SYCLK=CPUCLK/2
```
SPLK #40C0h, SYSCR ;CLKOUT=CPUCLK
LACL SYSCR

LACL SYSCR
OR #000Fh
SACL SYSCR
LACL SYSCR
ZAC
LACL SYSSR

;*******************************************************************
; Enable all core interrupts                                  *
;*******************************************************************
LDPK #0
SPLK #003fh, IMR ; core interrupts
SPLK #000ffh, IFR ; Clear all core interrupt flags
CLRC INTM

;*******************************************************************
; Point to EV register date page                            *
;*******************************************************************
LDPK #DP_EV ; DP => EV Registers

;*******************************************************************
; Mask all EV interrupts                                    *
;*******************************************************************
LACL IFRA
SACL IFRA
LACL IFRB
SACL IFRB
SPLK #00000h, IMRA ; Mask all Group C interrupt flags
SPLK #00000h, IMRB ; Mask all Group C interrupt flags
SPLK #00000h, IMRC ; Mask all Group C interrupt flags
LDPK #DP_EV

;***************************************************************
; Startup routine
;***************************************************************

CALL START2

;***************************************************************

LACL IFRA
SACL IFRA
LACL IFRB
SACL IFRB

SPLK #0080h,IMRA ; T1 period int & PDPINT is enabled
SPLK #0030h,IMRB ; T3 period int is enabled

;***************************************************************
; Clear all EV control registers
;***************************************************************

LDP #DP_PF1
SPLK #000000000000011b, AC2CNTL1

;             FEDCBA9876543210
;
; bit 0-2 011 Prescaler
; bit 3-4 FIFO index for channel 0
; bit 5   Reserved
; bit 6-7 FIFO index for channel 1
; bit 8   Reserved
; bit 9 0 Mask external SOC input
; bit A 0 Mask EV SOC input
; bit B-F Reserved
; Reference current is stored in CMD register
; Current conversion formula -
; (current * .005ohms * 12) + 2.5V = A/D volts
LDP    #0H
LACC   #0218H
SACL   CMD

; Same current command (CMD) is taken for all phases
LACC   CMD
SACL   CMDAC
SACL   CMDBD

; Proportional constant to be used later with PI controller
LACC   #50
LACC   #10
SACL   NUM
LT     NUM

; Set up EV registers
LDPK   #DP_EV

; Configure CAPCON and enable QEP operation
SPLK   #1110001011110000b, CAPCON; Capture control
* bits 0-1  00: No detection for Capture U 4
* bits 2-3  00: No detection for Capture U 3
* bits 4-5  11: Capture U 2 detects both edges
* bits 6-7  11: Capture U 1 detects both edges
* bit 8  0: No Capture U 4 event starts ADC
* bit 9  1: GP Timer 3 is time base for Cap Us 1&2
* bit 10  0: GP Timer 2 is time base for Cap Us 3&4
* bit 11  0: Disable Capture U 4
* bit 12  0: Disable Capture U 3
* bit13-14  11: Enable QEP
* bit 15  1: No action

;*******************************************************************
; Initialize counter registers                                       *
;*******************************************************************
SPLK #0000H, T1CNT        ; GP Timer 3 counter
SPLK #0000H, T3CNT        ; GP Timer 3 counter

;*******************************************************************
; Initialize T1 period register
; This timer defines PWM frequency
;*******************************************************************
SPLK #0800, T1PER         ; GP Timer 1 period 25kHz PWM
SPLK #2000, T1PER         ; GP Timer 1 period 12.5kHz PWM
LACC T1PER

;*******************************************************************
; MAX and MIN variable is used to obtain 0% and 100% PWM
;*******************************************************************
LDP    #0
SACL  MAX                 ; MAX = T1PER
LACC  #00
SACL  MIN                 ; MIN = 0
LACC  MIN
; COMAC = Compare register for phases A and C
; COMBD = Compare register for phases B and D
; These registers are initialized to zero before starting any loop
; This is done to start any loop with 100% PWM

;*******************************************************************
SACL COMAC
SACL COMBD
;*******************************************************************
; CURRAC = Register for measured current of phases A or C
; CURRBD = Register for measured current of phases B or D
; These registers are initialized to zero before starting any loop
;*******************************************************************
ZAC
SACL CURRAC
SACL CURRBD

LDP #DP_EV
;*******************************************************************
; T3 will count this value when rotor travels from non aligned to
; aligned position. Therefore, T3 interrupt provides the
; commutation instance.
;*******************************************************************
SPLK #0803Bh, T3PER ; T3 is configured for
; bi-directional operation
SPLK #07FC3h, T3CMP ;

SPLK #07FFFh, T3CNT ;

;*******************************************************************
; Clear all EV interrupts before operation starts
;*******************************************************************
SPLK #0ffffh,IFRA ; Clear all Group A interrupt flags
SPLK #0ffffh,IFRB ; Clear all Group B interrupt flags
SPLK #0ffffh,IFRC ; Clear all Group C interrupt flags
;******************************************************************************
; Configure COMCON registers
;******************************************************************************

SPLK #0000001100000111b, COMCON ;
SPLK #1000001100000111b, COMCON ;

; FEDCBA9876543210

* bit 0 1: CMP1/PWM1 & CMP2/PWM2 are PWM outputs
* bit 1 1: CMP3/PWM3 & CMP4/PWM4 are PWM outputs
* bit 2 1: CMP5/PWM5 & CMP6/PWM6 are PWM outputs
* bit 3-4 00: Load SACTR when TCNT1=0
* bit 5-6 00: Load SCMPR when TCNT1=0
* bit 7 0: T1 is used for Simple Compares
* bit 8 1: SACTR has control
* bit 9 1: ACTR has control
* bit 10-11 00: Load ACTR when TCNT1=0
* bit 12 0: Space vector disabled
* bit 13-14 00: Load CMPR when TCNT1=0
* bit 15 1: Enable PWM

;****************************************************************************
; Determines the initial direction of rotation.
; SWITCH (IO location 0x0008)= 1 means CW
; SWITCH (IO location 0x0008)= 0 means CCW
;****************************************************************************

LDPK #0
IN  GPR2, SWITCH
LACC GPR2
SUB  #1
BCND X3, EQ
LACC #4
SACL GPR0
B   XX
X3  LACC  #2h
    SACL  GPR0
XX  LACC  #1
    SACL  GPR3
    LDPK  #DP_EV

;*******************************************************************
; Configure T1CON and start GP Timer 1
;*******************************************************************
SPLK #1001000001000000b, T1CON ; Set GP Timer 2 control

    * bit 0  0: Use own Period
    * bit 1  0: GP Timer compare disabled
    * bits 2-3 00: Load GP Timer comp register on underflow
    * bits 4-5 00: Select internal clock
    * bit 6  1: Timer (counting operation) enabled
    * bit 7  0: Use own Timer ENABLE
    * bits 8-10 000: Prescaler = /1
    * bits 11-13 010: continuous up-mode.
    * bit 14  0: SOFT = 0
    * bit 15  1: FREE = 1

;*******************************************************************
; Configure T3CON and start GP Timer 3
;*******************************************************************
SPLK #1001000101000010b, T3CON ; Set GP Timer 3 control
SPLK #1001100001110010b, T3CON ; Set GP Timer 3 control

    * bit 0  0: Use own Period
    * bit 1  1: GP Timer compare enabled
    * bits 2-3 00: Load GP Timer comp register on underflow
    * bits 4-5 11: QEP
* bit 6  1: Timer (counting operation) enabled
* bit 7  0: Use own Timer ENABLE
* bits 8-10 000: Prescaler = \(/1\)
* bits 11-13 011: Directional up/down
* bit 14  0: SOFT = 0
* bit 15  1: FREE = 1

WAIT      B      WAIT

;***************************************************************
; MAIN CODE - ends here
;***************************************************************

;***************************************************************
;***************************************************************

T1PERIOD_ISR
  LDPK #DP_EV ; DP => EV Registers
  LACC IVRA
  LDP #0
  LACC GPR3
  SUB #1
  BCND NOCOM, EQ

;***************************************************************
  LDP #0
  IN GPR2, SWITCH
  LACC GPR2
  SUB #1H
  BCND FORWARD, EQ

REVERSE  LACC GPR0
  SUB #1
  SACL GPR0
BCND  X1, EQ
B    GOOUT

X1
LACC  #4
SACL  GPR0
B    GOOUT

FORWARD
LDP   #0
LACC  GPR2
SUB   #0
BCND  REVERSE, EQ
LACC  GPR0
ADD   #1
SACL  GPR0
SUB   #5
BCND  X2, EQ
B    GOOUT

X2
LACC  #1
SACL  GPR0

GOOUT
LDP   #0
LACC  #1
SACL  GPR3

;*******************************************************************
NOCOM  CALL DATAINT

LDPK  #0
OUT   GPR0, LEDS
LACC  #SECTOR_TABLE
ADD   GPR0
OUT   GPR0, LEDS
TBLR  ADDRESS
LACC  ADDRESS
BACC

PHASE_00:  LACC  #0DEAdh
PHASE_A: LDP #0
   LACC CMDAC
   SUB CURRAC
   BCND DEC_A, LT

;***********************************************************************
   LACC COMAC
   SUB NUM
   SACL COMAC

;***********************************************************************
   LACC MIN
   SUB COMAC
   BCND SET_A, GEQ
   B GOON_A

DEC_A

;***********************************************************************
   LACC COMAC
   ADD NUM
   SACL COMAC

;***********************************************************************
   LACC MAX
   SUB COMAC
   BCND GOON_A, GEQ
   LACC MAX
   SACL COMAC
   B GOON_A

SET_A LACC MIN
ACL COMAC

GOON_A LACC COMAC

LDPK #DP_EV
SACL CMPR2
SACL CMPR3

;*******************************************************************************
; Configure ACTR for phase A PWM registers
; PWM - 5 --> SWITCH A/C+  *
; pwm - 3 --> SWITCH A-      *
;*******************************************************************************
SPLK #0000000000000000B, SACTR
SPLK #0000000000000100B, ACTR
;

; ACTR bits description
; bit 15 : 0 - Space vector rotation direction
; bit 14-12: 000 - Spacc vector bits
; bit 11-10 : 00 - PWM6 forced low
; bit 9-8 : 11 - PWM5 (A/C+) forced high
; bit 7-6 : 00 - PWM4 forced low
; bit 5-4 : 10 - PWM3 (A-) active high
; bit 3-2 : 00 - PWM2 forced low
; bit 1-0 : 00 - PWM1 forced low
;
LDP #0
LACC MIN
SACL COMBD
ZAC
SACL CURRBD
B PPP

;*******************************************************************************
; Configure ACTR AND SACTR for phase B PWM registers
; PWM - 8 --> SWITCH B/D+  *
; PWM - 7 --> SWITCH B-      *
;*******************************************************************************
PHASE_B:

```assembly
LDP  #0
LACC COMBD
SUB CURRBD
BCND DEC_B, LT
; BCND DEC_B, GEQ
;******************************************************************************
LACC COMBD
SUB NUM
SACL COMBD

;******************************************************************************
LACC MIN
SUB COMBD
BCND SET_B, GEQ
B GOON_B

DEC_B
;******************************************************************************
LACC COMBD
ADD NUM
SACL COMBD

;******************************************************************************
LACC MAX
SUB COMBD
BCND GOON_B, GEQ
LACC MAX
SACL COMBD
B GOON_B

SET_B LACC MIN
SACL COMBD

GOON_B LACC COMBD
```
LDP  #DP_EV
SACL  SCMPR1
SACL  SCMPR2
SPLK  #0000000000000000B, ACTR
SPLK  #0000000000001010B, SACTR
;
    |||||||||||||||
    FEDCBA9876543210
;
*******************************************************************
; SACTR bits description
; bit 15-6 : 0000000000 reserved
; bit 5-4 : 00 - PWM 9 forced low
; bit 3-2 : 11 - PWM 8 (B/D+) forced high
; bit 1-0 : 10 - PWM 7 active high
*******************************************************************
LDP  #0
LACC  MIN
SACL  COMAC
ZAC
SACL  CURRAC
B  PPP

PHASE_C:
LDP  #0
LACC  CMDAC
SUB  CURRAC
BCND  DEC_C, LT
;     BCND  DEC_C, GEQ
;*******************************************************************
LACC  COMAC
SUB  NUM
SACL  COMAC

;*******************************************************************
LACC  MIN
SUB  COMAC
BCND  SET_C, GEQ
  B  GOON_C

DEC_C

;*******************************************************************
LACC  COMAC
ADD  NUM
SACL  COMAC

;*******************************************************************
LACC  MAX
SUB  COMAC
BCND  GOON_C, GEQ
LACC  MAX
SACL  COMAC
B  GOON_C
SET_C  LACC  MIN
SACL  COMAC

GOON_C  LACC  COMAC

LDPK  #DP_EV
SACL  CMPR3  ;
SACL  SCMPR3

;*******************************************************************
; Configure ACTR for phase A PWM registers  *
; PWM - 5 --> SWITCH A/C+  *
; PWM - 9 --> SWITCH C-  *
;*******************************************************************
SPLK  #0000001000000000B, ACTR
;  |||||||||||||||||
;  FEDCBA9876543210
;  ACTR bits description
;  bit 15  : 0  -  Space vector rotation direction
; bit 14-12 : 000 - Spacc vector bits
; bit 11-10 : 00 - PWM6 forced low
; bit 9-8 : 11 - PWM5 (A/C+) forced high
; bit 7-6 : 00 - PWM4 forced low
; bit 5-4 : 00 - PWM3 forced low
; bit 3-2 : 00 - PWM2 forced low
; bit 1-0 : 00 - PWM1 forced low

SPLK #000000000100000B, SACTR
;
    |||||||||||||||||
;
    FEDCBA9876543210
;
;*******************************************************************
; SACTR bits description
; bit 15-6 : 0000000000 reserved
; bit 5-4 : 10 - PWM 9 (C-) active high
; bit 3-2 : 00 - PWM 8 (B2) forced low
; bit 1-0 : 00 - PWM 7 (A2) forced low
;*******************************************************************

LDP #0
LACC MIN
SACL COMBD
ZAC
SACL CURRBD
B PPP

PHASE_D:
LDP #0
LACC CMDBD
SUB CURRBD
BCND DEC_D, LT
;
BCND DEC_D, GEQ
;
;*******************************************************************
LACC COMBD
SUB NUM
SACL COMBD

Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240  55
LACC MIN
SUB COMBD
BCND SET_D, GEQ
B GOON_D
DEC_D

LACC COMBD
ADD NUM
SACL COMBD

LACC MAX
SUB COMBD
BCND GOON_D, GEQ
LACC MAX
SACL COMBD
B GOON_D
SET_D LACC MIN
SACL COMBD

GOON_D LACC COMBD

LDPK #DP_EV
SACL CMPR1 ;
SACL SCMPR2

; Configure ACTR for phase A PWM registers *
; PWM - 8 --> SWITCH B/D+ *
; pwm - 1 --> SWITCH D- *
;******************************************************************************

SPLK #000000000000010B, ACTR
; ........................................................................
; FEDCBA9876543210
; ........................................................................

; ACTR bits description
; bit 15 : 0 - Space vector rotation direction
; bit 14-12 : 000 - Spacc vector bits
; bit 11-10 : 00 - PWM6 forced low
; bit 9-8 : 00 - PWM5 forced low
; bit 7-6 : 00 - PWM4 forced low
; bit 5-4 : 00 - PWM3 forced low
; bit 3-2 : 00 - PWM2 forced low
; bit 1-0 : 10 - PWM1 (D-) active high

; ........................................................................

SPLK #000000000001000B, SACTR
; ........................................................................
; FEDCBA9876543210
; ........................................................................

; SACTR bits description
; bit 15-6 : 0000000000 reserved
; bit 5-4 : 00 - PWM 9 forced low
; bit 3-2 : 11 - PWM 8 (B/D+) forced high
; bit 1-0 : 00 - PWM 7 forced low

; ........................................................................

LDP #0
LACC MIN
SACL COMAC
ZAC
SACL CURRAC
PPP CLRC INTM
RET

; ........................................................................

; SUBROUTINE: DATAINIT TYPE: Call-return
; Description: This routine updates the compare values.

Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240  57
DATAINT

   LDP   #DP_PF1
   SPLK  #1101100110100111b, AC2CNTL0

   FEDCBA9876543210

; bit 0   0  Start of conversion
; bit 1-3 010  Channel 0 address
; bit 4-6 010  Channel 1 address
; bit 7   1  End of convert
; bit 8   1  Interrupt flag - write 1 to clear
; bit 9   0  Interrupt mask - enable with 1, mask 0
; bit A   0  Continuous run mode disabled
; bit B   1  Enable channel 0
; bit C   1  Enable channel 1
; bit D   1  Immediate start - 0 = no action
; bit E   1  Free run - ignore suspend
; bit F   1  Soft - Not applicable with bit E = 1

   LACC  AC2DR0
   SFR   SFR
   SFR   SFR
   SFR   SFR
   SFR
   AND   #0000001111111111B

   LDP   #0
   SAACL CURRBD
   LDP   #DP_PF1
   LACC  AC2DR1
   SFR   SFR
   SFR   SFR
SFR
SFR
SFR
AND  #0000011111111111B

LDP #0
SACL CURRAC
RET

;*******************************************************************
; ISR:  PHANTOM   TYPE: ISR
;
; Description: ISR used to trap spurious interrupts.
;      Reads System Interrupt Vector register to capture
;      vector of module that caused the interrupt.
;
; Last Update: 10/22/96
;*******************************************************************

PHANTOM LACC #0DEAdh
    B   PHANTOM

;*******************************************************************
; ISR:  T3PERIOD_ISR   TYPE: ISR
;
; Description: This ISR is used to generate appropriate commutation
;    sequence. QEP will generate interrupt at every T3 period. T3
;    period is equal to 15 electrical degrees. Therefore, T3 period
;    interrupt will indicate that the rotor has rotated from one phase
;    to the next and it is time to turn off the present phase and
;    commutate to next phase.
;*******************************************************************

T3PERIOD_ISR
LDPK #DP_EV ; DP => EV Registers
LACC IVRB
SPLK #07FFh, T3CNT
LDP #0
LACC #0
SACL GPR3
LDPK #DP_EV ; DP => EV Registers
ZAC
CLRC INTM
RET

;********************************************************************
; This is a simple startup routine. Phase A is energized with
; 50 % duty cycle to get the rotor aligned with phase A.
; This routine is applicable only when a reverse rotation is
; tolerated by the load. If reverse rotation is unacceptable
; then some other direct or indirect method has to be applied
; for proper rotational direction.
;********************************************************************

START2 NOP
;********************************************************************
; Configure ACTR and SACTR operation
;********************************************************************
SPLK #000000000000010110b, SACTR ; Energizing Phase B
SPLK #000000000000000010b, ACTR

;********************************************************************
; Configure COMCON and enable PWM operation
;********************************************************************
SPLK #0100101101010110b, COMCON; Compare control
SPLK #1100101101010110b, COMCON; Compare control

;********************************************************************
; Configure Compare registers
  ;**********************************************************
  SPLK #01000h, SCMPR1
  SPLK #01000h, SCMPR2
  SPLK #01000h, SCMPR3
  SPLK #01000h, CMPR1
  SPLK #01000h, CMPR2
  SPLK #01000h, CMPR3

;**********************************************************
; Initialize counter and period registers
;**********************************************************
  SPLK #00000H,T1CNT ; GP Timer 3 counter
  SPLK #04E20h,T1PER ; GP Timer 1 period

;**********************************************************
; Clear all EV interrupts before operation starts
;**********************************************************
  SPLK #0xffffh,IFRA ; Group A interrupt flags
  SPLK #0xffffh,IFRB ; Group B interrupt flags
  SPLK #0xffffh,IFRC ; Group C interrupt flags

;**********************************************************
; Configure T1CON and start GP Timer 1
;**********************************************************
  SPLK #1001000001000000b,T1CON

* bit 0  0:  Use own PR
* bit 1  0:  GP Timer compare disabled
* bits 2-3 00:  Load GP Timer comp register on under flow
* bits 4-5 00:  Select internal CLK
* bit 6  1:  Timer (counting operation) enabled
* bit 7  0:  Use own Timer ENABLE
* bits 8-10 000:  Prescaler = /1
* bits 11-13 010: continuous-up count mode
* bit 14 0: SOFT = 0
* bit 15 1: FREE = 1

;**********************************************************
; Loop: Providing some time for the rotor to get aligned
; with phase A
;**********************************************************
LAR AR6, #307h
LARP AR6
LACC #0FFFFh
SACL *

LOOP
LACL *
SUB #1
SACL *
BCND LOOP, NEQ

LACC #0FFFFh
SACL *

LOOP2
LACL *
SUB #1
SACL *
BCND LOOP2, NEQ

;**********************************************************
; All PWM channels will be turned off before returning to
; the main routine. Timer T1 is also turned off
;**********************************************************
SPLK #0000000000000000B, SACTR
SPLK #0000000000000000B, ACTR

RET
SECTOR_TABLE:

PH_00 .WORD PHASE_00
PH_A .WORD PHASE_A
PH_B .WORD PHASE_B
PH_C .WORD PHASE_C
PH_D .WORD PHASE_D
Appendix C. Header Files

C240APP.H

;*******************************************************************
; File Name: C240APP.H
; Project: C240 EVM Apps S/W suite
; Originator/s: Auto/Ind DSP Apps group - Texas
; (Houston)
;
; Description: C240 Header file with all Peripheral Register
; declarations other useful defines.
;
; Last Update: Oct 22, 96
;
; Date of Mod | DESCRIPTION
; ------------|-----------------------------------------------------
; 3 Oct 96   | Added Event Manager Register declarations.
; 18 Oct 96  | Modified SPI register declarations to match the U.G.
; | and added some definitions for peripheral data
; | pages and analysis breakpoint register.
; Oct 22, 96 | dropped "_abs" from all the register names.
; | added DP_EV, EV register data memory page
;
;*******************************************************************

; On Chip Peripheral Register Definitions (DATA SPACE)
;--------------------------------------------------------
; C2xx Core Registers
;--------------------
IMR .set 0004h ;Int Mask Register
GREG .set 0005h ;Global memory allocation reg
IFR .set 0006h ;Int Flag Register
; System Module Registers
; ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
SYSCR .set 07018h ; System Module Control Register
SYSSR .set 0701Ah ; System Module Status Register
DIN .set 0701Ch ; Device Identification Register
SYSIVR .set 0701Eh ; System Interrupt Vector Register

XINT1_CNTL .set 07070h ; Int1 (type A) Control reg
NMI_CNTL .set 07072h ; Non maskable Int (type A) Cntl reg
XINT2_CNTL .set 07078h ; Int2 (type C) Control reg
XINT3_CNTL .set 0707Ah ; Int3 (type C) Control reg
PDPINT_CNTL .set 0742Ch ; Power Drive Protection Int cntl reg

; Digital I/O
; ~~~~~~~~~
OPCRA .set 07090h ; Output Control Reg A
OPCRB .set 07092h ; Output Control Reg B
IPSRA .set 07094h ; Input Status Reg A
IPSRB .set 07096h ; Input Status Reg B
IOPA_DDR .set 07098h ; I/O port A Data & Direction reg.
IOPB_DDR .set 0709Ah ; I/O port B Data & Direction reg.
IOPC_DDR .set 0709Ch ; I/O port C Data & Direction reg.
IOPD_DDR .set 0709Eh ; I/O port D Data & Direction reg.

; Watch-Dog (WD) / Real Time Int (RTI) / Phase Lock Loop (PLL) Registers
; ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240
;RTI Counter reg
RTI_CNTR .set 07021h

;WD Counter reg
WD_CNTR .set 07023h

;WD Key reg
WD_KEY .set 07025h

;RTI Control reg
RTI_CNTL .set 07027h

;WD Control reg
WD_CNTL .set 07029h

;PLL Clock Control Register 0
CKCR0 .set 0702Bh

;PLL Clock Control Register 1
CKCR1 .set 0702Dh

;SPI Configuration Control Register
SPICCR .set 07040h

;SPI Operation Control Register
SPICTL .set 07041h

;SPI Status Register
SPISTS .set 07042h

;SPI Baud Rate Register
SPIBRR .set 07044h

;SPI Emulation Buffer Register
SPIEMU .set 07046h

;SPI Serial Input Buffer Register
SPIBUF .set 07047h

;SPI Serial Data Register
SPIDAT .set 07049h

;SPI Port Control Register #1
SPIPC1 .set 0704Dh

;SPI Port Control Register #2
SPIPC2 .set 0704Eh

;SPI Priority Register
SPIPRI .set 0704Fh

;SCI Comms Control Reg
SCI_CCNTL .set 07050h

;SCI Control Reg 1
SCI_CNTL1 .set 07051h

;SCI Baud rate control
SCI_HBAUD .set 07052h

;SCI Baud rate control
SCI_LBAUD .set 07053h

;SCI Control Reg 2
SCI_CNTL2 .set 07054h

;SCI Receive status reg
SCI_RX_STAT .set 07055h

;SCI EMU data buffer
SCI_RX_EMU .set 07056h

;SCI Receive data buffer
SCI_RX_BUF .set 07057h

;SCI Transmit data buffer
SCI_TX_BUF .set 07059h

;SCI Port control reg1
SCI_PORT_C1 .set 0705Dh

;SCI Port control reg2
SCI_PORT_C2 .set 0705 Eh

;SCI Priority control reg
SCI_PRI .set 0705Fh

Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240
Implementation of a Current Controlled Switched Reluctance Motor Drive Using TMS320F240

; Event Manager (EV) - EV Base Address = 0x7400

; ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

EV_BASE .set 7400h ; Event Manager Base Address.

; DP 232)

GPTCON .set 00h + EV_BASE ; General Timer Controls
T1CNT .set 01h + EV_BASE ; T1 Counter Register
T1CMP .set 02h + EV_BASE ; T1 Compare Register
T1PER .set 03h + EV_BASE ; T1 Period Register
T1CON .set 04h + EV_BASE ; T1 Control Register
T2CNT .set 05h + EV_BASE ; T2 Counter Register
T2CMP .set 06h + EV_BASE ; T2 Compare Register
T2PER .set 07h + EV_BASE ; T2 Period Register
T2CON .set 08h + EV_BASE ; T2 Control Register
T3CNT .set 09h + EV_BASE ; T3 Counter Register
T3CMP .set 0ah + EV_BASE ; T3 Compare Register
T3PER .set 0bh + EV_BASE ; T3 Period Register
T3CON .set 0ch + EV_BASE ; T3 Control Register
COMCON .set 11h + EV_BASE ; Compare Unit Control
ACTR .set 13h + EV_BASE ; Compare Unit Output Action Control
SACTR .set 14h + EV_BASE ; S Comp Unit Output Action Control
DBTCON .set 15h + EV_BASE ; Dead Band Timer Control
CMPR1 .set 17h + EV_BASE ; Compare Channel 1 Threshold
CMPR2 .set 18h + EV_BASE ; Compare Channel 2 Threshold
CMPR3 .set 19h + EV_BASE ; Compare Channel 3 Threshold
SCMPR1 .set 1ah + EV_BASE ; S Comp Channel 1 Threshold
SCMPR2 .set 1bh + EV_BASE ; S Comp Channel 2 Threshold
SCMPR3 .set 1ch + EV_BASE ; S Comp Channel 3 Threshold
CAPCON .set 20h + EV_BASE ; Capture Unit Control
CAPFIFO .set 22h + EV_BASE ; FIFO1-4 Status Register
FIFO1 .set 23h + EV_BASE ; Capture Channel 1 FIFO Top
FIFO2 .set 24h + EV_BASE ; Capture Channel 2 FIFO Top
FIFO3 .set 25h + EV_BASE ; Capture Channel 3 FIFO Top
FIFO4 .set 26h + EV_BASE ; Capture Channel 4 FIFO Top
IMRA .set 2ch + EV_BASE ; Group A Interrupt Mask Register
IMRB .set 2dh + EV_BASE ; Group B Interrupt Mask Register
IMRC .set 2eh + EV_BASE ; Group C Interrupt Mask Register
IFRA .set 2fh + EV_BASE ; Group A Interrupt Flag Register
IFRB .set 30h + EV_BASE ; Group B Interrupt Flag Register
IFRC .set 31h + EV_BASE ; Group C Interrupt Flag Register
IVRA .set 32h + EV_BASE ; Group A Int. Vector Offset (AD dec; only)
IVRB .set 33h + EV_BASE ; Group B Int. Vector Offset (AD dec; only)
IVRC .set 34h + EV_BASE ; Group C Int. Vector Offset (AD dec; only)

;-------------------------------------------------------------------
; Constant defines

;-------------------------------------------------------------------
B0_SADDR .set 00100h ; Block B0 start address
B0_EADDR .set 001FFh ; Block B0 end address
B1_SADDR .set 00300h ; Block B1 start address
B1_EADDR .set 003FFh ; Block B1 end address
B2_SADDR .set 00060h ; Block B2 start address
B2_EADDR .set 0007Fh ; Block B2 end address

DP_PF1 .set 224 ; page 1 of peripheral file
; (7000h/80h)

DP_PF2 .set 225 ; page 2 of peripheral file
; (7080h/80h)

DP_PF3 .set 226 ; page 3 of peripheral file
; (7100h/80h)

DP_EV .set 232 ; EV register data memory page
; (7400h)

ABRPT .set 01fh ; Analysis Break Point Register
PSA_ON .set 03A1h ; Turn PSA and FEEDB on
PSA_OFF .set 0121h ; Turn PSA and FEEDB off
; Bit codes for Test bit instruction (BIT)
BIT15 .set* 0000h ; Bit Code for 15
BIT14 .set 0001h ; Bit Code for 14
BIT13 .set 0002h ; Bit Code for 13
BIT12 .set 0003h ; Bit Code for 12
BIT11 .set 0004h ; Bit Code for 11
BIT10 .set 0005h ; Bit Code for 10
BIT9 .set 0006h ; Bit Code for 9
BIT8 .set 0007h ; Bit Code for 8
BIT7 .set 0008h ; Bit Code for 7
BIT6 .set 0009h ; Bit Code for 6
BIT5 .set 000Ah ; Bit Code for 5
BIT4 .set 000Bh ; Bit Code for 4
BIT3 .set 000Ch ; Bit Code for 3
BIT2 .set 000Dh ; Bit Code for 2
BIT1 .set 000Eh ; Bit Code for 1
BIT0 .set 000Fh ; Bit Code for 0

SD_init.H

********************************************************************
* File Name: SD_init.h
* Created on 3/28/96 to configure relevant shared pins for usage
* by Event Manager.
* It assumes that c240.h or c240a.h has been included to define
* CDPORT_FSL.
*
* Changed to include Port A and B
* Updated register names (OPCRA,B). 10/22/96
********************************************************************
LDPK #225
SPLK #0FFFFH,OPCRA
SPLK #0FFF3H,OPCRB
Appendix D. LINKER COMMAND FILE

/*----------------------------------------------------------------*/
/* LINKER COMMAND FILE - MEMORY SPECIFICATION for C240 */
/* Last update 18 Oct 96 */
/* removed reference to filenames for object, map and out. */
/* Use c240lnk.bat to call this file. */
/*----------------------------------------------------------------------*/
MEMORY
{
    PAGE 0 : VECS : origin = 0h , length = 040h /* PROGRAM */
    PROG : origin = 40h , length = 3FC0h /* EMIF */

    PAGE 1 : MMRS: origin = 0h , length = 05Fh /* MMRS */
    B2 : origin = 0060h , length = 020h /* DARAM */
    B0 : origin = 0100h , length = 0200h /* DARAM */
    B1 : origin = 0300h , length = 0200h /* DARAM */
    DATA : origin = 4000h , length = 4000h /* EMIF */
}

/*----------------------------------------------------------------*/
/* SECTIONS ALLOCATION */
/*----------------------------------------------------------------*/
SECTIONS
{
    .vectors : { } > VECS PAGE 0 /* INTERRUPT VECTOR TABLE */
    .text    : { } > PROG PAGE 0 /* CODE */
    .data    : { } > PROG PAGE 0 /* INITIALIZATION DATA */
    /* TABLES */
    .mmrs    : { } > MMRS PAGE 1 /* Memory Mapped Registers */
    .blk0    : { } > B0 PAGE 1 /* Block B0 - page 4 */
    .bss     : { } > B2 PAGE 1 /* Block B2 - page 0 */
    .blk1    : { } > B1 PAGE 1 /* Block B1 - page ? */
    .blk3    : { } > DATA PAGE 1 /* External Data Space */
}
Appendix E.  C240tbed.cmd

echo C240tbed.CMD for C240 Test Bed

;Reset Memory Map
mr

;DATA MEMORY
ma 0x00000,1,0x005F,ram ;MMRs
ma 0x00060,1,0x0020,ram ;On-Chip RAM B2
ma 0x00100,1,0x0100,ram ;On-Chip RAM B0 if CNF=0
ma 0x00300,1,0x0100,ram ;On-Chip RAM B1
ma 0x07010,1,0x000F,ram ;Peripheral - System Configure & Control
ma 0x07020,1,0x0010,ram ;Peripheral - WDT / RTI
ma 0x07030,1,0x0010,ram ;Peripheral - ADC
ma 0x07040,1,0x0010,ram ;Peripheral - SPI
ma 0x07050,1,0x0010,ram ;Peripheral - SCI
ma 0x07070,1,0x0010,ram ;Peripheral - External Interrupts
ma 0x07090,1,0x0010,ram ;Peripheral - Digital I/O
ma 0x07400,1,0x001D,ram ;Peripheral - Event Manager
ma 0x0742C,1,0x0009,ram ;Peripheral - Event Manager Int cntl

; for test bed, external RAM is in memory map
ma 0x08000,1,0x02000,ram ;External SRAM

;PROGRAM MEMORY
ma 0x00000,0,0x03FFF,ram ;Internal Prog memory (Flash EEPROM or ROM)

;Available if MPNMC=0.

ma 0xfe000,0,0x00100, ram ;Available if CNF=1 i.e. B0

;I/O MEMORY
ma 0x06000,2,0x02000,ram ;External I/O memory.
; Enable program execution from B0 RAM
; e st1 |= 0x1000 ; Set CNF = 1
; ?psaint=0 ; Enter test mode
; reset ; Make test mode visible to code
; Enable internal program and data RAM
; Set CNF = 1 and PC to 0xfe00
;
; e st1 |= 0x1000
;
; map on
; dasm pc
; e pc = 0x0fe00
; e st0 = 0x300
; e 0x8029 = 0x6F
; sconfig screen.fmt

meml 0x0300

; System Module Registers
; -------------------------------
; wa *0x07018, SYSCR ; System Module Control Register
; wa *0x0701A, SYSSR ; System Module Status Register
; wa *0x0701E, SYSIVR ; System Interrupt Vector Register
; wa *0x07070, XINT1_CR ; Int1 (type A) Control reg
; wa *0x07072, NMI_CR ; Non maskable Int (type A) Control reg
; wa *0x07078, XINT2_CR ; Int2 (type C) Control reg
; wa *0x0707A, XINT3_CR ; Int3 (type C) Control reg
; wa *0x0742C, PDPINT_CR ; Power Drive Protection Int cntl reg

; Clock Registers
; -------------------------------
; wa *0x0702B, CKCR0
; wa *0x0702D, CKCR1
;Digital I/O
;~~~~~~~~~~~~~~~~
;wa *0x07090, OPCRA ;Output Control Reg A
;wa *0x07092, OPCRB ;Output Control Reg B
;wa *0x07094, IPSRA ;Input Status Reg A
;wa *0x07096, IPSRB ;Input Status Reg B
;wa *0x07098, IOPA_DDR ;I/O port A Data & Direction reg.
;wa *0x0709A, IOPB_DDR ;I/O port B Data & Direction reg.
;wa *0x0709C, IOPC_DDR ;I/O port C Data & Direction reg.
;wa *0x0709E, IOPD_DDR ;I/O port D Data & Direction reg.

;Watch-Dog (WD) / Real Time Int (RTI) / Phase Lock Loop (PLL) Registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07021, RTI_CNT ;RTI Counter reg
;wa *0x07023, WD_CNT ;WD Counter reg
;wa *0x07025, WD_KEY ;WD Key reg
;wa *0x07027, RTI_CR ;RTI Control reg
;wa *0x07029, WD_CR ;WD Control reg
;wa *0x0702B, PLL_CR1 ;PLL control reg 1
;wa *0x0702D, PLL_CR2 ;PLL control reg 2

;Analog-to-Digital Converter (ADC) registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07032, ADCNTL0 ;ADC Control0
;wa *0x07034, ADCNTL1 ;ADC Control1
;wa *0x0701E, SYSIVR ;ADC Channel Select reg
;wa *0x07036, ADCDATA0 ;ADC Channel 0 Result Data
;wa *0x07038, ADCDATA1 ;ADC Channel 1 Result Data

;Serial Peripheral Interface (SPI) Registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07040, SPI_CR1 ;SPI Configure Control Reg 1
;wa *0x07041, SPI_CR2 ;SPI Operation Control Reg 2
;wa *0x07042, SPI_SR ;SPI Status Reg
;wa *0x07044, SPI_BAUD ;SPI Baud rate control reg
;wa *0x07046, SPI_EMU ;SPI Emulation buffer reg
;wa *0x07047, SPI_BUF ;SPI Serial Input buffer reg
;wa *0x07049, SPI_DAT ;SPI Serial Data reg
;wa *0x0704D, SPI_PORT_CR1 ;SPI Port control reg1
;wa *0x0704E, SPI_PORT_CR2 ;SPI Port control reg2
;wa *0x0704F, SPI_PRI_CR ;SPI Priority control reg

;Serial Communications Interface (SCI) Registers
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07050, SCI_COM_CR ;SCI Comms Control Reg
;wa *0x07051, SCI_CR1 ;SCI Control Reg 1
;wa *0x07052, SCI_HBAUD ;SCI Baud rate control
;wa *0x07053, SCI_LBAUD ;SCI Baud rate control
;wa *0x07054, SCI_CR2 ;SCI Control Reg 2
;wa *0x07055, SCI_RX_STAT ;SCI Receive status reg
;wa *0x07056, SCI_RX_EMU ;SCI EMU data buffer
;wa *0x07057, SCI_RX_BUF ;SCI Receive data buffer
;wa *0x07059, SCI_TX_BUF ;SCI Transmit data buffer
;wa *0x0705D, SCI_PORT_CR1 ;SCI Port control reg1
;wa *0x0705E, SCI_PORT_CR2 ;SCI Port control reg2
;wa *0x0705F, SCI_PRI_CR ;SCI Priority control reg

;Event Manager (EV)
;~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
;wa *0x07400, GPTCON ; General Timer Controls
;wa *0x07401, T1CNT ; T1 Counter Register
;wa *0x07402, T1CMP ; T1 Compare Register
;wa *0x07403, T1PER ; T1 Period Register
;wa *0x07404, T1CON ; T1 Control Register
;wa *0x07405, T2CNT ; T2 Counter Register
;wa *0x07406, T2CMP ; T2 Compare Register
;wa *0x07407, T2PER ; T2 Period Register
;wa *0x07408, T2CON ; T2 Control Register
;wa *0x07409, T3CNT ; T3 Counter Register
;wa *0x0740a, T3CMP ; T3 Compare Register
;wa *0x0740b, T3PER ; T3 Period Register
;wa *0x0740c, T3CON ; T3 Control Register
;wa *0x07411, COMCON ; Compare Unit Control
;wa *0x07413, ACTR ; Full Compare Unit Output Action
;wa *0x07414, SACTR ; Simple Comp Unit Output Action
;wa *0x07415, DBTCON ; Dead Band Timer Control
;wa *0x07417, CMPR1 ; Full Compare Channel 1 Threshold
;wa *0x07418, CMPR2 ; Full Compare Channel 2 Threshold
;wa *0x07419, CMPR3 ; Full Compare Channel 3 Threshold
;wa *0x0741a, SCMPR1 ; Simple Comp Channel 1 Threshold
;wa *0x0741b, SCMPR2 ; Simple Comp Channel 2 Threshold
;wa *0x0741c, SCMPR3 ; Simple Comp Channel 3 Threshold
;wa *0x07420, CAPCON ; Capture Unit Control
;wa *0x07422, CAPFIFO ; FIFO1-4 Status Register
;wa *0x07423, FIFO1 ; Capture Channel 1 FIFO Top
;wa *0x07424, FIFO2 ; Capture Channel 2 FIFO Top
;wa *0x07425, FIFO3 ; Capture Channel 3 FIFO Top
;wa *0x07426, FIFO4 ; Capture Channel 4 FIFO Top
;wa *0x0742c, IMRA ; Group A Interrupt Mask Register
;wa *0x0742d, IMRB ; Group B Interrupt Mask Register
;wa *0x0742e, IMRC ; Group C Interrupt Mask Register
;wa *0x0742f, IFRA ; Group A Interrupt Flag Register
;wa *0x07430, IFRB ; Group B Interrupt Flag Register
;wa *0x07431, IFRC ; Group C Interrupt Flag Register
;wa *0x07432, IVRA ; Group A Int. Vector Offset Register
;wa *0x07433, IVRB ; Group B Int. Vector Offset Register
;wa *0x07434, IVRC ; Group C Int. Vector Offset Register

reset
load c:\asm\sr.out
reset
sconfig dis_aref.cmd
echo   C240tbed.CMD HAS BEEN LOADED