MOTOROLA
intelligence everywhere™

digital dna™

3-Phase SR Motor Sensorless Control Reference Design

Designer Reference Manual

56800 Hybrid Controller

DRM030//D
Rev. 0, 03/2003

MOTOROLA.COM/SEMICONDUCTORS

3-Phase SR Motor Sensorless Control Reference Design

Designer Reference Manual — Rev 0

by: Radim Visinka, Jaroslav Musil
Motorola Czech Systems Laboratories
Roznov pod Radhostem, Czech Republic

Revision history

To provide the most up-to-date information, the revision of our documents on the World Wide Web will be the most current. Your printed copy may be an earlier revision. To verify you have the latest information available, refer to:

http://www.motorola.com/semiconductors

The following revision history table summarizes changes contained in this document. For your convenience, the page number designators have been linked to the appropriate location.

Revision history

Date	Revision Level	Description	Page Number(s)
February 2003	1	Initial release	N/A

List of Sections

Table of Contents

Section 1. Introductuion

Section 2. Control Theory

Section 3. System Concept

Section 4. Hardware

Section 5. Software Design

Section 6. Application Setup

Appendix A. References

Appendix B. Glossary

List of Figures

List of Tables

List of Tables

Section 1. Introductuion

1.1 Contents

1.2 Introduction

This paper describes the design of a sensorless 3-Phase SR (Switched Reluctance) motor drive. It is based on the Motorola DSP56F805. The software design takes advantage of Quick_Start developed by Motorola.

SR motors are gaining wider popularity among variable speed drives. This is due to their simple low-cost construction characterized by an absence of magnets and rotor winding, high level of performance over a wide range of speeds, and fault tolerant power stage design. Availability and the moderate cost of the necessary electronic components make SR drives a viable alternative to other commonly used motors like AC, BLDC, PM Synchronous or universal motors for numerous applications.

The concept of this application is that of a sensorless speed closed loop SR drive with an inner current loop using flux linkage position estimation. The change in phase resistance during motor operation due to its temperature dependency creates errors in the position estimation and significantly affects the performance of the drive. Therefore, a novel algorithm for on-the-fly estimation of the phase resistance is included. This application demonstrates the sensorless SR motor drive and serves as an example of a system design using a Motorola DSP. It also illustrates the usage of dedicated motor control algorithm libraries. The application helps start the development of the sensorless SR drive dedicated to the targeted application.

This paper includes a description of Motorola DSP features, basic SR motor theory, system design concept, hardware implementation, and software design including the use of the PC master software visualization tool.

1.3 Motorola DSP Advantages and Features

The Motorola DSP56F805 is well suited for digital motor control, combining a DSP's computational ability with an MCU's controller features on a single chip. These DSP's offer many dedicated peripherals like a Pulse Width Modulation (PWM) unit, Analog-to-Digital Converter (ADC), timers, communications peripherals (SCI, SPI, CAN), on-board Flash and RAM. Generally, all family members are well-suited for Switched Reluctance motor control.

The DSP56F805 provides the following peripheral blocks:

- Two Pulse Width Modulator modules (PWMA & PWMB), each with six PWM outputs, three Current Sense inputs, and four Fault inputs; fault tolerant design with dead time insertion; supports both center- and edge-aligned modes
- Twelve-bit, Analog-to-Digital Converters (ADCs), supporting two simultaneous conversions with dual 4-pin multiplexed inputs; the ADC can be synchronized by the PWM
- Two Quadrature Decoders (Quad Dec0 & Quad Dec1), each with four inputs, or two additional Quad Timers A & B
- Two dedicated General Purpose Quad Timers totaling 6 pins: Timer C with 2 pins and Timer D with 4 pins
- CAN 2.0 A/B Module with 2-pin ports used to transmit and receive
- Two Serial Communication Interfaces (SCI0 & SCI1), each with two pins, or four additional GPIO lines
- A Serial Peripheral Interface (SPI), with a configurable 4-pin port, or four additional GPIO lines
- Computer Operating Properly (COP) Watchdog Timer
- Two dedicated external interrupt pins

- Fourteen dedicated General Purpose I/O (GPIO) pins, 18 multiplexed GPIO pins
- An external reset pin for hardware reset
- JTAG/On-Chip Emulation (OnCE)
- A software-programmable, phase lock loop-based frequency synthesizer for the DSP core clock

Table 1-1. Memory Configuration

	DSP56F801	DSP56F803	DSP56F805	DSP56F807
Program Flash	8188 x 16-bit	32252 x 16-bit	32252 x 16-bit	61436 x 16-bit
Data Flash	2K x 16-bit	4K x 16-bit	4K x 16-bit	8K x 16-bit
Program RAM	1K x 16-bit	512 x 16-bit	512 x 16-bit	2K x 16-bit
Data RAM	1K x 16-bit	2K x 16-bit	2K x 16-bit	4K x 16-bit
Boot Flash	2K x 16-bit	2K x 16-bit	2K x 16-bit	2K x 16-bit

From the switched reluctance motor control point of view, the most interesting peripherals are the fast Analog-to-Digital Converter (ADC) and the Pulse-Width-Modulation (PWM) on-chip modules. They offer a lot of freedom of configuration, enabling efficient sensorless control of SR motors.

The **PWM module** incorporates a PWM generator, enabling the generation of control signals for the motor power stage. The module has the following features:

- Three complementary PWM signal pairs, or six independent PWM signals
- Complementary channel operation
- Deadtime insertion
- Separate top and bottom pulse width correction via current status inputs or software
- Separate top and bottom polarity control
- Edge- or center-aligned PWM signals

- 15 bits of resolution
- Integral reload rates from one to 16 with a half-cycle reload capability
- Individual software-controlled PWM output
- Programmable fault protection
- Polarity control
- 20-mA current sink capability on PWM pins
- Write-protectable registers

The SR motor control application utilizes the PWM module set in independent PWM mode, permitting fully independent generation of control signals for all switches of the power stage. In addition to the PWM generators, the PWM outputs can be controlled separately by software, allowing the setting of the control signal to logical 0 or 1. Thus, the state of the control signals can be changed instantly at a given rotor position (phase commutation) without changing the contents of the PWM value registers. This change can be made asynchronously with the PWM duty cycle update.

The **Analog-to-Digital Converter** (ADC) consists of a digital control module and two analog sample and hold (S/H) circuits. It has the following features:

- 12-bit resolution
- Maximum ADC clock frequency of 5 MHz with a 200ns period
- Single conversion time of 8.5 ADC clock cycles (8.5 x 200 ns = 1.7µs)
- Additional conversion time of 6 ADC clock cycles (6 x 200 ns = 1.2µs)
- Eight conversions in 26.5 ADC clock cycles (26.5 x 200 ns = 5.3µs) using simultaneous mode
- ADC can be synchronized to the PWM via the SYNC signal
- Simultaneous or sequential sampling
- Internal multiplexer to select two of eight inputs

- Ability to sequentially scan and store up to eight measurements
- Ability to simultaneously sample and hold two inputs
- Optional interrupts at end of scan, at zero crossing or if an out-of-range limit is exceeded
- Optional sample correction by subtracting a pre-programmed offset value
- Signed or unsigned result
- Single-ended or differential inputs

The application utilizes the ADC on-chip module in simultaneous mode and sequential scan. The sampling is synchronized with the PWM pulses for precise sampling and reconstruction of phase currents. Such a configuration allows instant conversion of the desired analog values of all phase currents, voltages and temperatures.

Section 2. Control Theory

2.1 Contents

2.2 Target Motor Theory

2.2.1 Switched Reluctance Motor

A Switched Reluctance (SR) motor is a rotating electric machine where both stator and rotor have salient poles. The stator winding is comprised of a set of coils, each of which is wound on one pole. The rotor is created from lamination in order to minimize the eddy-current losses.

SR motors differ in the number of phases wound on the stator. Each of them has a certain number of suitable combinations of stator and rotor poles. **Figure 2-1** illustrates a typical 3-Phase SR motor with a 6/4 (stator/rotor) pole configuration.

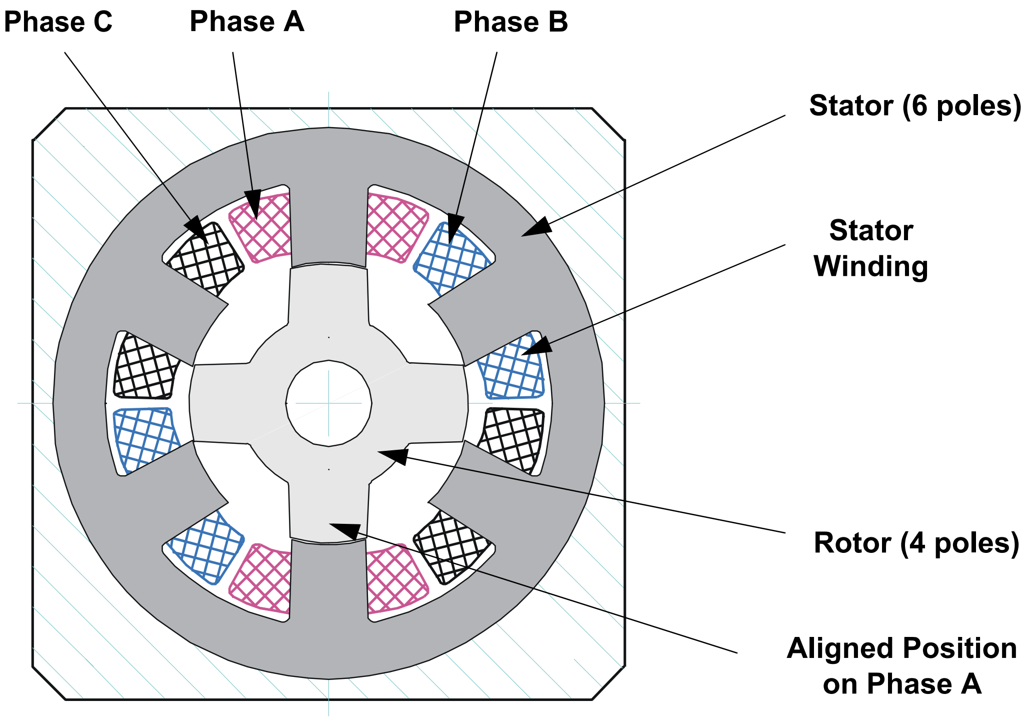


Figure 2-1. 3-Phase 6/4 SR Motor

The motor is excited by a sequence of current pulses applied at each phase. The individual phases are consequently excited, forcing the motor to rotate. The current pulses need to be applied to the respective phase at the exact rotor position relative to the excited phase. When any pair of rotor poles is exactly in line with the stator poles of the selected phase, the phase is said to be in an aligned position, i.e., the rotor is in the position of maximal stator inductance (see **Figure 2-1**). If the interpolar axis of the rotor is in line with the stator poles of the selected phase, the phase is said to be in an unaligned position, i.e., the rotor is in a position of minimal stator inductance. The inductance profile of SR motors is triangular shaped, with maximum inductance when it is in an aligned position and minimum inductance when unaligned. **Figure 2-2** illustrates the idealized triangular-like inductance profile of all three phases of an SR motor with phase A highlighted. The individual Phases A, B, and C are shifted electrically by 120^{o} relative to each other. When the respective phase is powered, the interval is called the dwell angle - θ_{dwell}. It is defined by the turn-on θ_{on} and the turn-off θ_{off} angles.

When the voltage is applied to the stator phase, the motor creates torque in the direction of increasing inductance. When the phase is energized in its minimum inductance position, the rotor moves to the forthcoming position of maximal inductance. The movement is defined by the magnetization characteristics of the motor. A typical current profile for a constant phase voltage is shown in **Figure 2-2**. For a constant phase voltage the phase current has its maximum in the position when the inductance starts to increase. This corresponds to the position where the rotor and the stator poles start to overlap. When the phase is turned off, the phase current falls to zero. The phase current present in the region of decreasing inductance generates negative torque. The torque generated by the motor is controlled by the applied phase voltage and by the appropriate definition of switching turn-on and turn-off angles.

As is apparent from the description, the SR motor requires position feedback for motor phase commutation. In many cases, this requirement is addressed by using position sensors, like encoders, Hall sensors, etc. The result is that the implementation of mechanical sensors increases costs and decreases system reliability. Traditionally, developers of motion control products have attempted to lower system costs by reducing the number of sensors. A variety of algorithms for sensorless control have been developed, most of which involve evaluation of the variation of magnetic circuit parameters that are dependent on the rotor position.

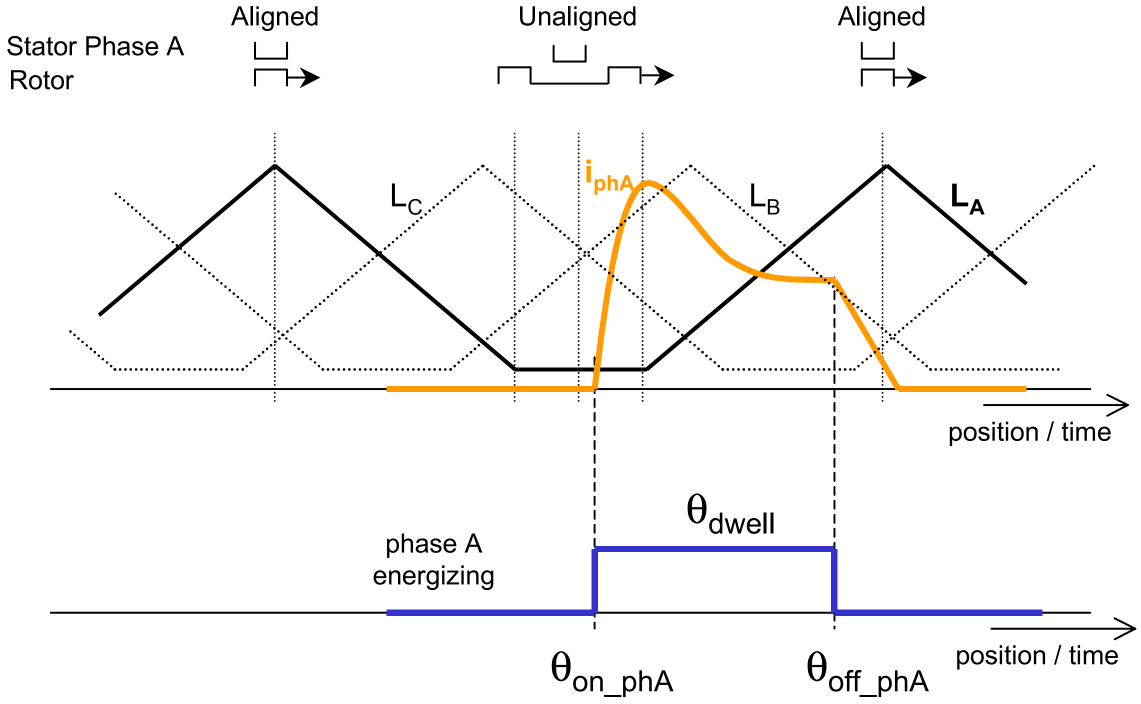


Figure 2-2. Phase Energizing

The motor itself is a low cost machine of simple construction. Since high-speed operation is possible, the motor is suitable for high speed applications, like vacuum cleaners, fans, white goods, etc. As discussed above, the disadvantage of the SR motor is the need for shaft-position information for the proper switching of individual phases. Also, the motor structure causes noise and torque ripple. The greater the number of poles, the smoother the torque ripple, but motor construction and control electronics become more expensive. Torque ripple can also be reduced by advanced control techniques such as phase current profiling.

2.2.2 Mathematical Description of an SR Motor

An SR motor is a highly non-linear system, so a non-linear theory describing the behavior of the motor was developed. Based on this theory, a mathematical model can be created. On one hand it enables the simulation of SR motor systems and on the other hand, it makes the development and implementation of sophisticated algorithms for controlling the SR motor easier.

The electromagnetic circuit of the SR motor is characterized by non-linear magnetization. **Figure 2-3** illustrates a magnetization characteristic for a specific SR motor. It is a function between the magnetic flux ψ, the phase current *i* and the motor position θ. The influence of the phase current is mostly apparent in the aligned position, where saturation effects can be observed.

The magnetization characteristic curve defines the non-linearity of the motor. The torque generated by the motor phase is a function of the magnetic flux, therefore the phase torque is not constant for a constant phase current for different motor positions. This creates torque ripple and noise in the SR motor.

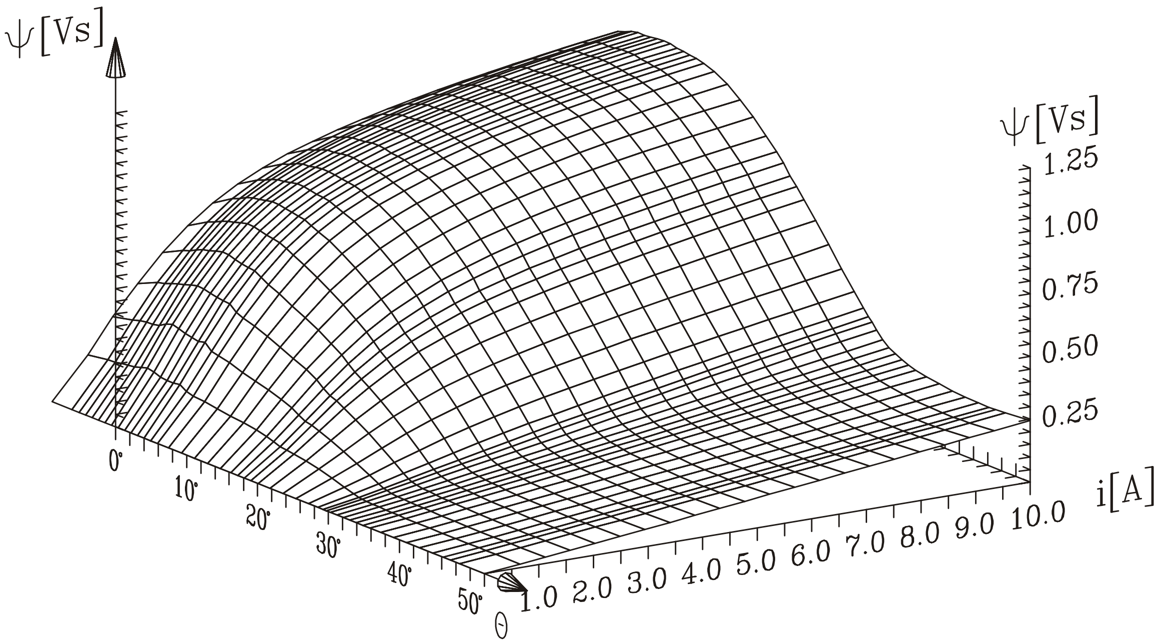


Figure 2-3. Magnetization Characteristics of the SR Motor

A mathematical model of an SR motor can be developed. The model is based on the electrical diagram of the motor, incorporating phase resistance and phase inductance. The diagram for one phase is illustrated in **Figure 2-4**.

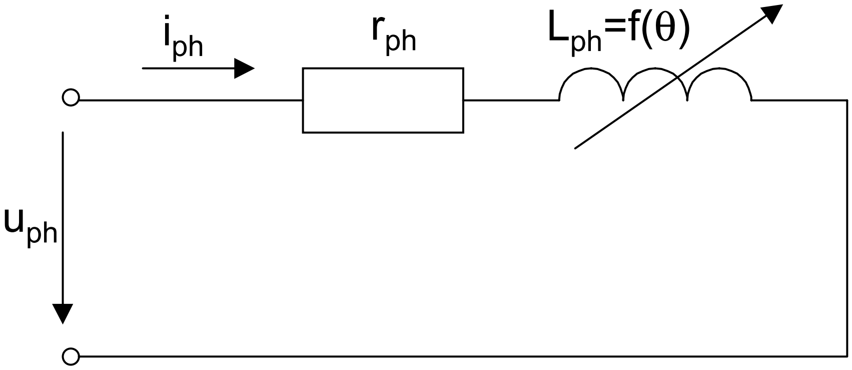


Figure 2-4. Electrical Diagram of One SR Motor Phase

According to the diagram, any voltage applied to a phase of the SR motor can be described as a sum of voltage drops in the phase resistance and induced voltages on the phase inductance:

$$u_{ph}(t) = r_{ph} \cdot i_{ph}(t) + u_{Lph}(t) \tag{2-1}$$

where:

u_{ph} is the applied phase voltage

r_{ph} is the phase resistance

i_{ph} is the phase current

u_{Lph} is the induced voltage on the phase inductance

The equation **(2-1)** supposes that all the phases are independent and have no mutual influence.

The induced voltage u_{Lph} is defined by the magnetic flux linkage Ψ_{ph}, that is a function of the phase current i_{ph} and rotor position θ_{ph}. So the induced voltage can be expressed as:

$$u_{Lph}(t) = \frac{d\Psi_{ph}(i_{ph}, \theta_{ph})}{dt} = \frac{\partial\Psi_{ph}(i_{ph}, \theta_{ph})}{\partial i_{ph}} \cdot \frac{di_{ph}}{dt} + \frac{\partial\Psi_{ph}(i_{ph}, \theta_{ph})}{\partial\theta_{ph}} \cdot \frac{d\theta_{ph}}{dt} \tag{2-2}$$

Then the phase voltage can be expressed as:

$$u_{ph}(t) = r_{ph} \cdot i_{ph}(t) + \frac{d\Psi_{ph}(i_{ph}, \theta_{ph})}{dt} \tag{2-3}$$

or:

$$u_{ph}(t) = r_{ph} \cdot i_{ph}(t) + \frac{\partial\Psi_{ph}(i_{ph}, \theta_{ph})}{\partial i_{ph}} \cdot \frac{di_{ph}}{dt} + \frac{\partial\Psi_{ph}(i_{ph}, \theta_{ph})}{\partial\theta_{ph}} \cdot \omega \tag{2-4}$$

where:

ω is the electrical speed of the motor.

The torque M_{ph} generated by one phase can be expressed as:

$$M_{ph} = \int_0^{I_{ph}} \frac{\partial\Psi_{ph}(i_{ph}, \theta_{ph})}{\partial\theta_{ph}} di_{ph} \tag{2-5}$$

The mathematical model of an SR motor is then represented by a system of equations, describing the conversion of electromechanical energy.

For 3-Phase SR motors the equation **(2-4)** can be expanded as follows:

$$u_a(t) = r_a \cdot i_a(t) + \frac{\partial\Psi_a(i_a, \theta_a)}{\partial i_a} \cdot \frac{di_a}{dt} + \frac{\partial\Psi_a(i_a, \theta_a)}{\partial\theta_a} \cdot \omega \tag{2-6}$$

$$u_b(t) = r_b \cdot i_b(t) + \frac{\partial\Psi_b(i_b, \theta_b)}{\partial i_b} \cdot \frac{di_b}{dt} + \frac{\partial\Psi_b(i_b, \theta_b)}{\partial\theta_b} \cdot \omega \tag{2-7}$$

$$u_c(t) = r_c \cdot i_c(t) + \frac{\partial\Psi_c(i_c, \theta_c)}{\partial i_c} \cdot \frac{di_c}{dt} + \frac{\partial\Psi_c(i_c, \theta_c)}{\partial\theta_c} \cdot \omega \tag{2-8}$$

where *a, b* and *c* index the individual phases.

2.2.3 Digital Control of an SR Motor

The SR motor is driven by voltage strokes coupled with the given rotor position. The profile of the phase current together with the magnetization characteristics define the generated torque and thus the speed of the motor. Due to this fact, the motor requires electronic control for operation. Several power stage topologies are being implemented, according to the number of motor phases and the desired control algorithm. The particular structure of the SR power stage structure defines the freedom of control for an individual phase.

A power stage with two independent power switches per motor phase is the most used topology. Such a power stage for 3-Phase SR motors is illustrated in **Figure 2-5**. It enables control of the individual phases fully independent of each other and thus permits the widest freedom of control. Other power stage topologies share some of the power devices for several phases, thus saving on power stage cost, but with these the phases cannot be fully independently controlled. Note that this particular topology of SR power stage is fault tolerant -- in contrast to power stages of AC induction motors -- because it eliminates the possibility of a rail-to-rail short circuit.

During normal operation, the electromagnetic flux in an SR motor is not constant and must be built for every stroke. In the motoring period, these strokes correspond to the rotor position when the rotor poles are approaching the corresponding stator pole of the excited phase. In the case of Phase A, shown in **Figure 2-1**, the stroke can be established by activating the switches Q1 and Q2. At low-speed operation the Pulse Width Modulation (PWM), applied to the corresponding switches, modulates the voltage level.

Two basic switching techniques can be applied:

- Soft switching - where one transistor is left turned-on during the whole commutation period and PWM is applied to the other one
- Hard switching - where PWM is applied to both transistors simultaneously

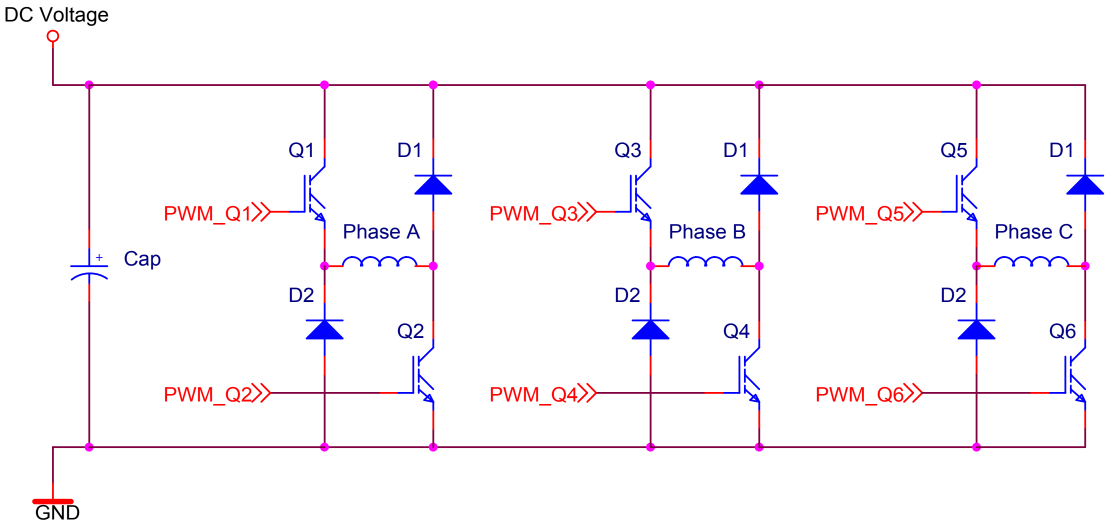


Figure 2-5. 3-Phase SR Power Stage

Figure 2-6 illustrates both soft and hard switching PWM techniques. The control signals for the upper and the lower switches of the above-described power stage define the phase voltage and thus the phase current. The soft switching technique generates lower current ripple compared to the hard switching technique. Also, it produces lower acoustic noise and less EMI. Therefore, soft switching techniques are often preferred for motoring operation.

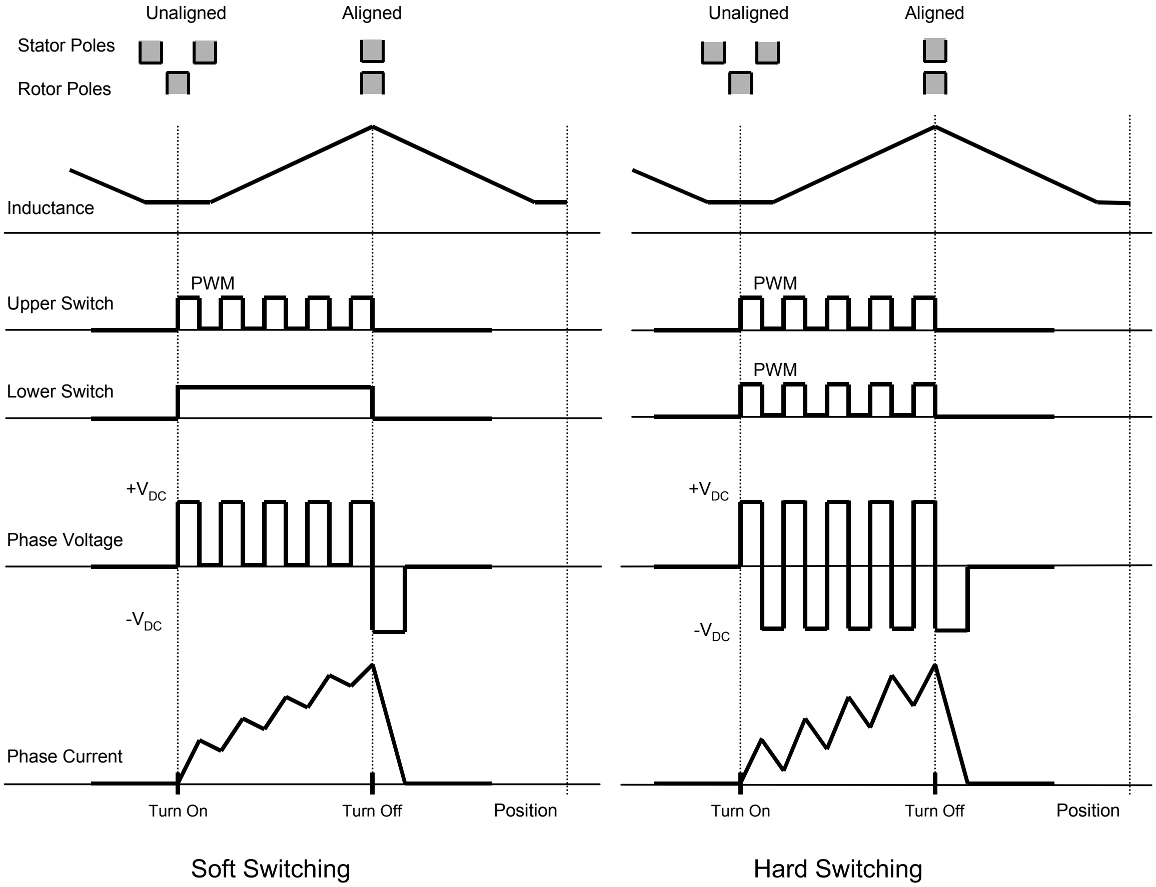


Figure 2-6. Soft Switching and Hard Switching

2.2.4 Voltage and Current Control of SR Motors

A number of control techniques for SR motors exist. They differ in the structure of the control algorithm and in position evaluation. Two basic techniques for controlling SR motors can be distinguished, according to the motor variables that are being controlled:

- Voltage control - where phase voltage is a controlled variable
- Current control - where phase current is a controlled variable

2.2.4.1 Voltage Control of an SR Motor

In voltage control techniques, the voltage applied to the motor phases is constant during the complete sampling period of the speed control loop. The commutation of the phases is linked to the position of the rotor.

The voltage applied to the phase is directly controlled by a speed controller. The speed controller processes the speed error -- the difference between the desired speed and the actual speed -- and generates the desired phase voltage. The phase voltage is defined by a PWM duty cycle implemented at the DC-Bus voltage of the SR inverter. The phase voltage is constant during a complete dwell angle. The technique is illustrated in **Figure 2-7**. The current and the voltage profiles can be seen in **Figure 2-8**. The phase current is at its peak at the position when the inductance starts to increase (stator and rotor poles start to overlap) due to the change in the inductance profile.

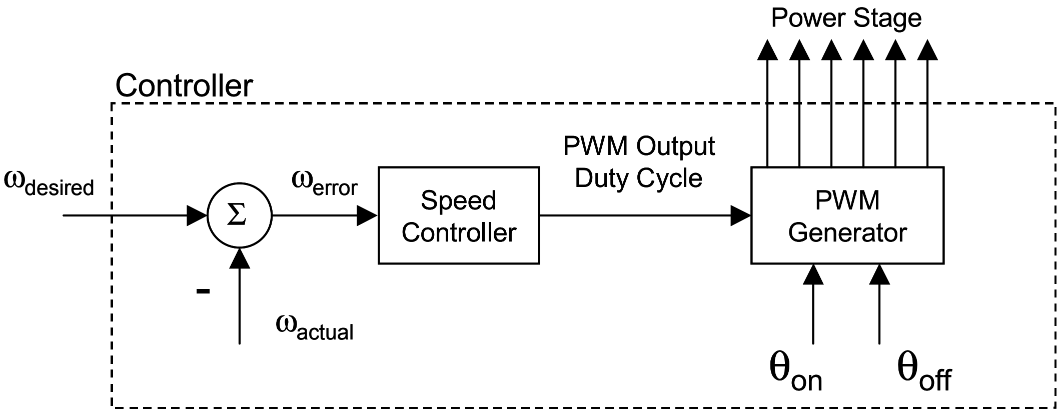


Figure 2-7. Voltage Control Technique

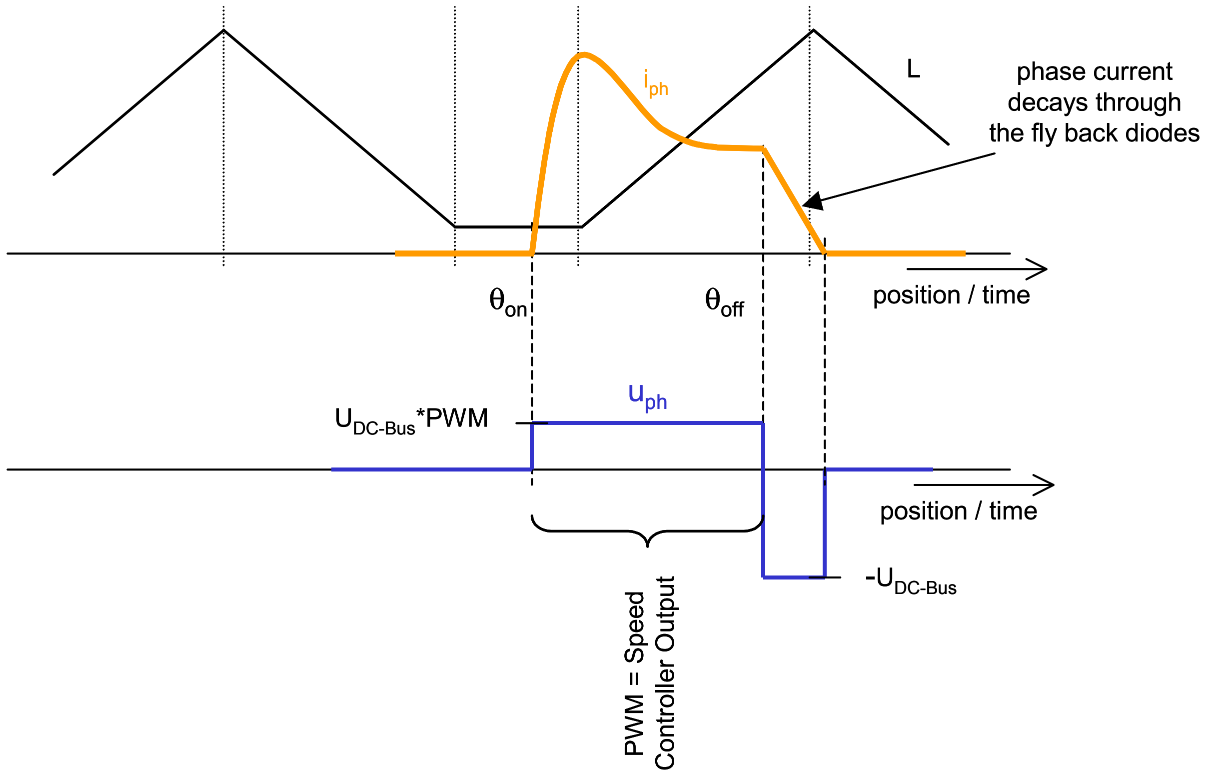


Figure 2-8. Voltage Control Technique - Voltage and Current Profiles

2.2.4.2 Current Control of an SR Motor

In current control techniques the voltage applied to the motor phases is modulated to reach the desired current at the powered phase. For most applications, the desired current is constant during the complete sampling period of the speed control loop. The commutation of the phases is linked to the position of the rotor.

The voltage applied to the phase is controlled by a current controller with an external speed control loop. The speed controller processes the speed error - the difference between the desired speed and the actual speed - and generates the desired phase current. The current controller evaluates the difference between actual and desired phase current and calculates the appropriate PWM duty cycle. The phase voltage is defined by a PWM duty cycle implemented at the DC-Bus voltage of the

SR inverter. Thus, the phase voltage is modulated at the rate of the current control loop. This technique is illustrated in **Figure 2-9**.

The processing of the current controller needs to be linked to the commutation of the phases. When the phase is turned on (commutated), a duty cycle of 100% is applied to the phase. The increasing actual phase current is regularly compared to the desired current. As soon as the actual current slightly exceeds the desired current, the current controller is turned on. The current controller controls the output of the duty cycle until the phase is turned off (following commutation). The procedure is repeated for each commutation cycle of the motor. The current and the voltage profiles can be seen in **Figure 2-10**. In ideal cases the phase current is controlled to follow the desired current.

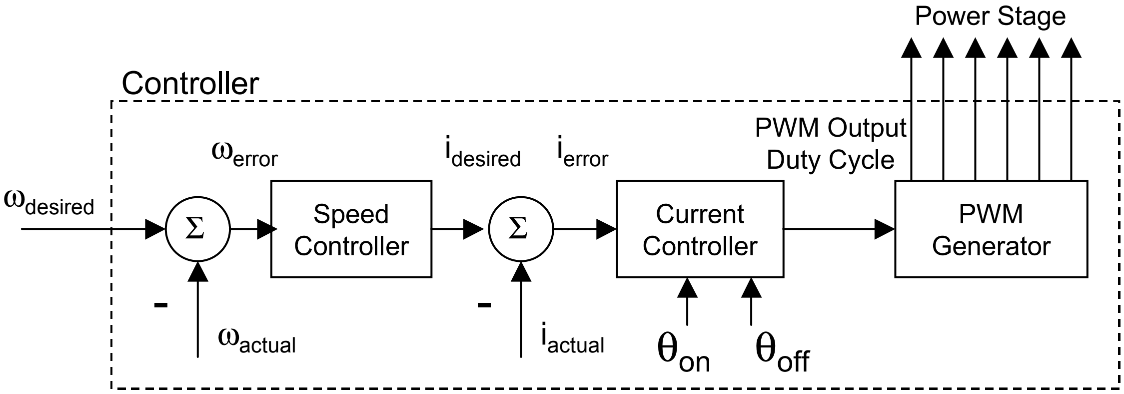


Figure 2-9. Current Control Technique

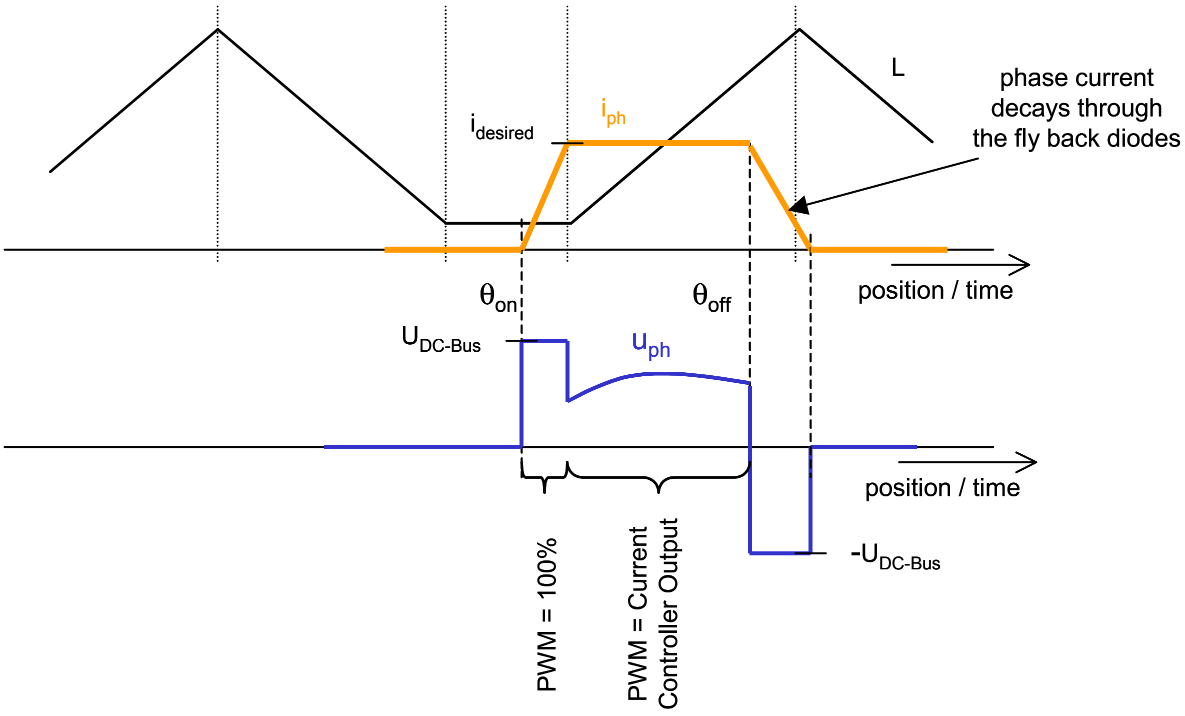


Figure 2-10. Current Control Technique - Voltage and Current Profiles

The individual phases of an SR motor need to be turned on at such a position that the phase current is able to rise to the desired level. The basic condition specifies that the phase current needs to achieve at least the desired level at a position where the stator and the rotor phases start to overlap. After the overlap position, the phase current starts to decrease due to the positive change in the inductance. So, if the phase is turned on late, the phase current is not able to reach the desired level for the commutation stroke.

The turn-on position needs to be determined according to the applied phase voltage, the actual motor speed and the inductance profile of the motor. The phase is turned on at the position of minimal inductance, so the inductance can be considered a constant until the position where the stator and rotor poles start to overlap.

For constant inductance, the phase current may be considered as linearly rising. Then the time required to achieve the desired current is determined from (2-3) as:

$$\Delta t = \frac{L_U \cdot i_{desired}}{u_{phase} \cdot \gamma} \tag{2-9}$$

where:

Δt is the required time to achieve the desired current
$i_{desired}$ is the desired current to be achieved
L_u is the unaligned inductance
u_{DC_Bus} is the DC-Bus voltage
γ is the PWM duty cycle

The electrical angle corresponding to the time required to reach the desired current can be determined as:

$$\Delta\vartheta = \omega_{actual} \cdot \Delta t \tag{2-10}$$

where:

ω_{actual} is the actual speed.

2.3 Techniques for Sensorless Control of SR Motors

2.3.1 Sensorless Pos. Estimation using Flux Linkage Estimation

The flux linkage estimation method belongs among the most popular sensorless SR position estimation techniques. A number of methods that use the flux linkage calculation have been developed. These methods calculate the actual phase flux linkage and use its relation to the reference flux linkage for position estimation.

The method implemented in this application is based on the comparison of the estimated flux linkage and the reference flux linkage, defined for the turn-off (commutation) position. When the estimated flux linkage reaches the desired reference flux linkage it indicates that the commutation position was reached. The actual phase is turned off and the following phase is turned on.

The **reference flux linkage** is derived from the magnetization characteristic as a function of phase current for the desired commutation position (see **Figure 2-11**).

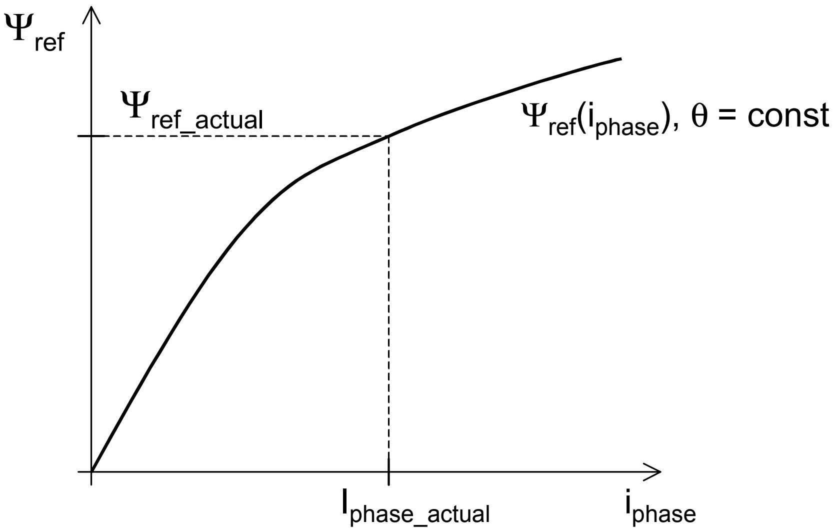


Figure 2-11. Reference Magnetization Curve for Constant Position

In order to simplify the determination of the reference flux linkage, we can assume that for a constant current, the flux linkage rises linearly in the interval between the unaligned and the aligned positions. This assumption can be considered in the region of the expected commutation. Then the reference flux linkage can be derived from the flux linkage in the aligned position as:

$$\Psi_{\theta_{off}}(i_{ph}) = k(\theta_{off}) \cdot \Psi_{\theta_{Aligned}} i_{ph}) \qquad \textbf{(2-11)}$$

where $k(\theta_{off})$ is a linear function corresponding to the commutation angle. It can reach a value in the interval <0, 1>, (0 corresponds to the unaligned position, 1 corresponds to the aligned position).

The reference magnetization curve $\psi(i_{ph})$ for the aligned position $\theta_{Aligned}$ is stored in controller memory.

The **estimated flux linkage** Ψ_{ph} of the turned-on phase is calculated using the following equation:

$$\Psi_{ph} = \int_{ton}^{t} (u_{ph} - R \cdot i_{ph})dt \qquad \textbf{(2-12)}$$

where:

u_{ph}	is the voltage applied to the motor phase (coil) winding
i_{ph}	is the actual phase current
R	is the phase resistance

The flux linkage estimation starts when the phase is turned on. The simultaneously sampled phase current and phase voltage are measured periodically at predetermined intervals and the flux linkage is estimated. Each time the flux linkage is calculated, it is compared with the reference level taken from the reference magnetization curve as a function of the actual phase current. When the estimated flux linkage exceeds the reference flux linkage, it indicates that the switching position has been reached and the commutation can be performed. The method is illustrated in **Figure 2-12**.

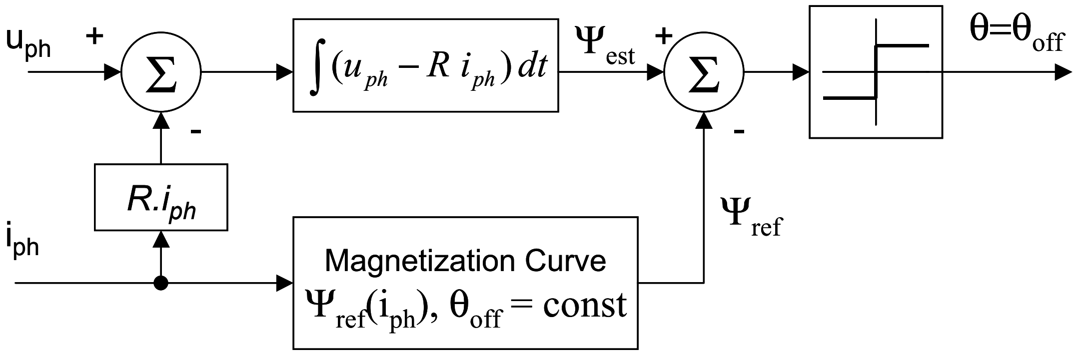


Figure 2-12. Pos. Estimation using One Reference Flux Linkage Function

The advantage of the flux linkage estimation methods is that they are usable over wide speed ranges, from start-up to high speeds. The position can accurately be estimated if the phase resistance is determined correctly. Four-quadrant operation is possible.

The main disadvantage of all these methods is that the estimation of the flux linkage is based on a precise knowledge of the phase resistance. The phase resistance varies significantly with temperature which yields to unwanted integration errors, especially at low speed. The integration error creates a significant position estimation error. Note that powerful DSP-based controllers (like the DSP56F80x) can easily perform all the needed calculations of the sensorless flux linkage algorithm.

2.3.2 Flux Linkage Calculation in a Discrete Time Domain

The introduced algorithm for the flux linkage estimation can be used for both analog and digital controllers. Digital control is preferred today for reasons of cost, flexibility and performance. For digital systems, the flux linkage calculation based on **(2-12)** needs to be converted at the discrete time domain.

The flux linkage estimation is performed regularly at the sampling frequency of the measurements of phase voltage and phase current. Equation **(2-12)** can be converted to:

$$\Psi_N = \sum_{k=1}^{N} [u_k - i_k r_k] \cdot T \quad , \qquad \textbf{(2-13)}$$

where:

T is the sampling period
u_k is the sampled phase voltage
i_k is the sampled phase current
r_k is the sampled phase resistance
Ψ_N is the calculated flux linkage at sample N

The flux linkage Ψ_N is calculated regularly at each sampling cycle from the beginning of the commutation stroke (t1). The sampling period T is constant. Equation **(2-13)** can be transformed to the following form:

$$\Psi_N = [u_N - i_N r_k] \cdot T + \Psi_{N-1} \quad , \qquad \textbf{(2-14)}$$

where:

Ψ_{N-1} - calculated flux linkage
for the previous measuring cycle (N-1).

In order to decrease the computational requirements, equation (2-14) can be transferred to:

$$\frac{\Psi_N}{T} = [u_N - i_N r_k] + \frac{\Psi_{N-1}}{T} \qquad \textbf{(2-15)}$$

So, instead of the pure flux linkage, the flux linkage divided by the sampling period is calculated. Because the sampling period is kept constant, the division can be considered a scaling factor. For proper functionality of the position estimation algorithm, the reference flux linkage has to be scaled in the same way.

2.3.3 Sensorless On-the-fly Resistance Estimation

The resistance of the phase winding is one of the most decisive factors in the magnetic flux linkage estimation (2-12). During motor operation, the variation of the resistance can exceed 30% of the nominal value because the phase resistance depends strongly on temperature. The effect of the phase resistance drift is more significant at low- and middle-speed ranges, where the voltage drop on the winding is comparable to the phase supply voltage u_{ph}. This variation causes an inaccurate estimation of the flux linkage, hence it generates position estimation errors and, based on such magnetic flux estimations, the sensorless techniques do not give satisfactory results. Therefore, in the case of an accurate and robust sensorless control algorithm, the actual value of the winding resistance must be accurately measured or estimated during motor operation.

In order to improve the behavior of the sensorless flux linkage estimation algorithm, an on-the-fly phase resistance estimator has been invented. The resistance estimation algorithm was patented as No. 6,366,865 at the US Patent Office.

The development of the phase resistance estimation was based on the flux linkage estimator (2-12). The flux linkage estimator calculates the flux linkage Ψ_{Est} at time t using the following formula:

$$\Psi_{Est} = \int_{t1}^{t} (u_{ph} - R^* \cdot i_{ph}) dt \qquad \textbf{(2-16)}$$

Where:

u_{ph}	is the voltage applied to the motor phase (coil) winding
i_{ph}	is the phase current
R^*	is the assumed phase winding resistance
$t1$	is the time when the motor phase winding starts to be energized

The assumed phase winding resistance *R** is the sum of the actual phase winding resistance *R* and the resistance error ΔR. The resistance error can be caused by temperature drift, an inaccurately obtained value, etc.

$$R^* = R + \Delta R \qquad \textbf{(2-17)}$$

Figure 2-13 illustrates the flux linkage waveforms calculated by the flux linkage estimator during a typical working cycle of one phase of an SR motor. Unlike the sensorless flux linkage estimation method, where the flux linkage is calculated up to the phase commutation angle θ_{off}, the flux linkage is calculated the whole time during which the current is flowing through the phase. The phase current and the shape of the flux linkage are defined by the control strategy, rotor position, and magnetization characteristic. SR motors are driven in a way that the motor phases are energized sequentially and the phase current therefore rises from zero, at the beginning of the cycle where the phase is turned on ($t1 \approx \theta_{on}$), up to θ_{off}, where the phase is disconnected and then falls down to zero again at the end of the cycle (t2). As can be seen, the flux linkage rises during the interval between the turn-on (t1) and the turn-off angles of the phase. When the phase is turned off, the flux linkage decreases until the phase current disappears. If all the parameters in **(2-16)** are obtained correctly, and the resistance error ΔR is zero, then the flux linkage is equal to zero at *t2,* as can be seen in **Figure 2-13**.

$$\Psi_{t2} = 0 \qquad \textbf{(2-18)}$$

For the influence of the resistance error, let’s assume that:

- The phase voltage and the phase current were measured correctly and the measurement error can be ignored
- The resistance error ΔR is not equal to zero, but it affects the estimation of the flux linkage.

Because the flux estimation is the result of an integration (see **Figure 2-13**), the total flux estimation error at the end of the working cycle (t2) can be quite significant.

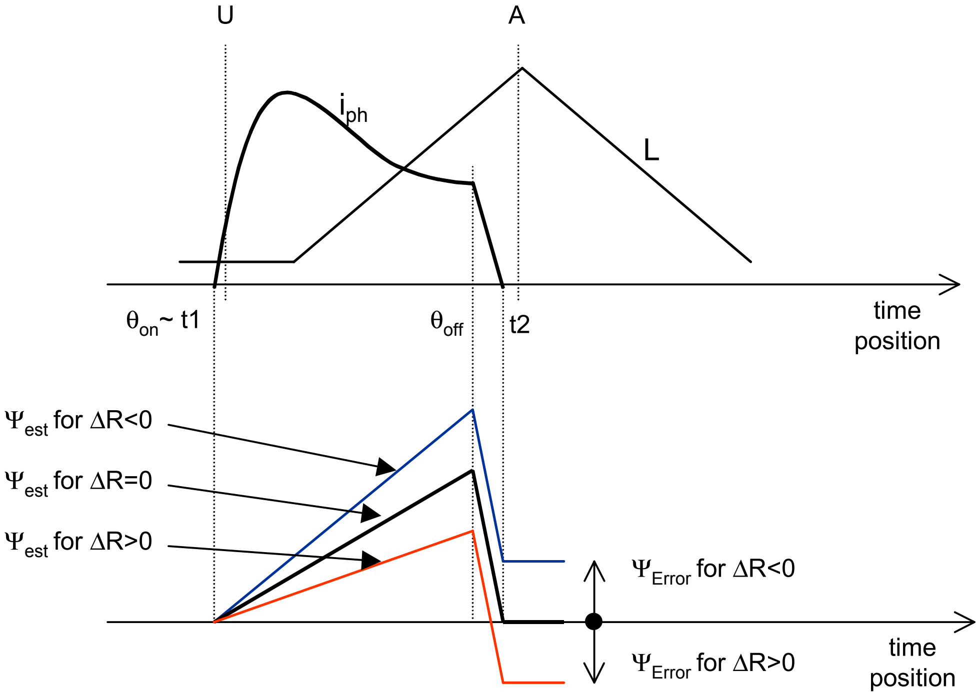


Figure 2-13. Flux Linkage and Phase Current

The **resistance estimation algorithm** is based on the fact that if the phase current is zero, then the magnetic flux must be zero as well. Resistance error leads to flux estimation error (see **Figure 2-13**). Thus, it enables us to calculate the flux estimation error at the point in time (t2) when the phase current falls to zero.

$$\Psi_{phEstim(t2)} = \int_{t1}^{t2} (u_{ph} - R \cdot i_{ph} - \Delta R \cdot i_{ph}) dt = \Psi_{ph(t2)} + \Psi_{Error(t2)} \tag{2-19}$$

Because the flux linkage at time t2 is equal to zero (2-18), the estimation error is equal to:

$$\Psi_{phEstim(t2)} = \Psi_{Error(t2)} = -\int_{t1}^{t2} \Delta R \cdot i_{ph} dt \tag{2-20}$$

Based on equation 4-10, it is apparent that if the flux linkage estimation error is positive, the resistance error is negative; and if the flux linkage estimation error is negative, the resistance error is positive.

$$\Psi_{Error(t2)} > 0 \qquad \Rightarrow \qquad \Delta R < 0 \tag{2-21}$$

$$\Psi_{Error(t2)} < 0 \qquad \Rightarrow \qquad \Delta R > 0 \tag{2-22}$$

Let us assume that the rate of change of the phase resistance is small during one commutation of the SR motor (this is valid for temperature drift):

$$\frac{\Delta R}{t2 - t1} \cong 0 \tag{2-23}$$

Using the above assumption, equation **(2-20)** can be rewritten as the following:

$$\Psi_{EstErr(t2)} = -\Delta R \int_{t1}^{t2} i_{ph} dt \tag{2-24}$$

Then the resistance error can be expressed as:

$$\Delta R = -\frac{\Psi_{EstErr(t2)}}{\int_{t1}^{t2} i_{ph} dt} \tag{2-25}$$

Equation 4-15 illustrates that the resistance error can be expressed as the ratio between the calculated flux linkage error at time t2, where the phase current decreases to zero, and the integral of the phase current, both of which are calculated over the complete phase current pulse.

Section 3. System Concept

3.1 Contents

3.2 System Outline

This system is designed to drive a 3-Phase SR motor. The application meets the following performance specifications:

- Sensorless speed control of an SR motor using a flux linkage estimation technique with an inner-current closed loop
- Targeted for DSP56F805EVM
- Running on a 3-Phase SR HV Motor Control Development Platform at a variable line voltage of between 115V AC and 230V AC (voltage range -15% ... +10%)
- The control technique incorporates:
  - current SRM control with a speed-closed loop
  - phase resistance measurement during start-up
  - phase resistance estimation at low speeds
  - motor starts from any position with rotor alignment

  - one direction of rotation
  - motoring mode
  - minimal speed 600 RPM
  - maximal speed 2600RPM at input power line 230V AC
  - maximal speed 1600RPM at input power line 115V AC
- Encoder position reference for evaluation of position estimation - visualized by PC master software (not used for SR control technique)
- Manual interface (start/stop switch, up/down push button control, LED indicator)
- PC master software control interface (motor start/stop, speed set-up)
- PC master software monitor
  - graphical control page (required speed, actual motor speed, operational mode PC/manual, start/stop status, drive fault status, DC-Bus voltage level, identified power stage boards, system status)
  - speed scope (observes actual and desired speeds and desired current)
  - start-up recorder (observes start-up phase current, flux linkage, output duty cycle and encoder position reference with fine resolution)
  - flux linkage recorder (observes phase current, estimated flux linkage, reference flux linkage and encoder position reference with fine resolution)
  - current controller recorder (observes actual and desired phase current, output duty cycle and encoder position reference with fine resolution)
- Power stage identification
- DC-Bus over-voltage, DC-Bus under-voltage, DC-Bus over-current and over-heating fault protection

3.3 Application Description

3.3.1 Application Concept

For the drive, a standard system concept was chosen (see **Figure 3-1**). The system incorporates the following hardware parts:

- 3-Phase SR high-voltage development platform (power stage with optoisolation board, SR motor with attached brake)
- Feedback sensors: DC-Bus voltage, DC-Bus current, phase currents, temperature
- DSP56F805 controller

The DSP runs the main control algorithm. It generates 3-Phase PWM output signals for the SR motor power stage according to the user interface input and feedback signals.

The drive can be controlled in two different ways (or operational modes):

- In Manual operational mode, the required speed is set by a Start/Stop switch and Up and Down push buttons.
- In PC master software operational mode, the required speed is set by the PC master software.

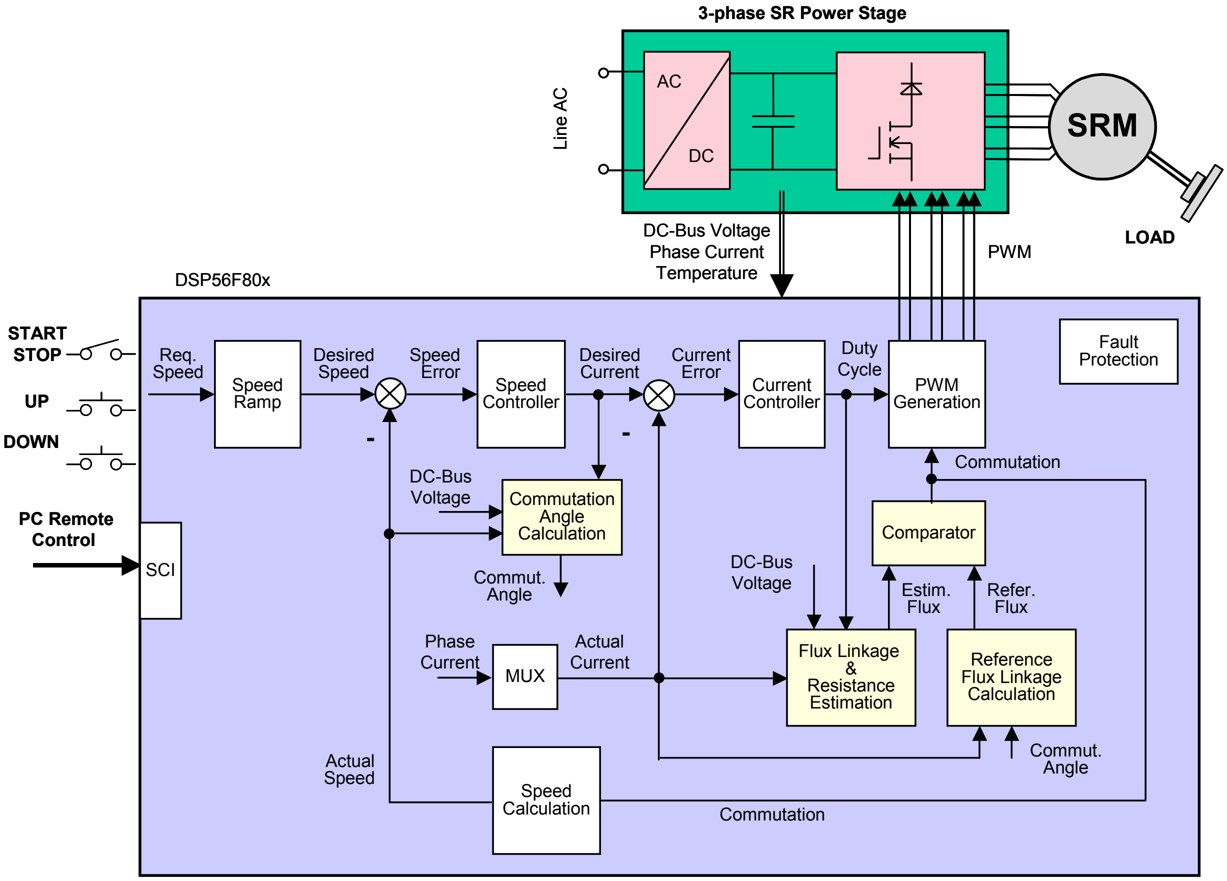


Figure 3-1. System Concept

After RESET the drive is initialized and automatically enters MANUAL operational mode. Note, PC master software can only take over control when the motor is stopped. When the Start command is detected (using the Start/Stop switch or the PC master software button “Start”) and while no fault is pending, the application can be started.

Rotor position is evaluated using the sensorless flux linkage estimation algorithm. The actual flux linkage is calculated at the rate of the PWM frequency and is compared with the reference flux linkage for a given commutation angle. The commutation angle is calculated according to the desired speed, the desired current and the actual DC-Bus voltage. When the actual flux linkage exceeds the reference, the commutation of the phases in the desired direction of rotation is done; the actual phase is turned off and the following phase is turned on. Flux linkage error is used for estimation of the phase resistance at low speeds (US Patent No.: 6,366,865).

The actual speed of the motor is determined using the commutation instances. The reference speed is calculated according to the control signals (start/stop switch, up/down push buttons) and PC master software commands (when controlled by PC master software). The acceleration/deceleration ramp is implemented. The comparison between the reference speed and the measured speed gives a speed error. Based on the speed error, the speed controller generates the desired phase current. When the phase is commutated, it is turned on with a duty cycle of 100%. Then, during each PWM cycle, the actual phase current is compared with the desired current. As soon as the actual current exceeds the desired current, the current controller is turned on. The current controller controls the output duty cycle until the phase is turned off (following commutation). Finally, the 3-Phase PWM control signals are generated. The procedure is repeated for each commutation cycle of the motor.

DC-Bus voltage, DC-Bus current, and power stage temperature are measured during the control process. The measurements are used for DC-Bus over-voltage, DC-Bus under-voltage, DC-Bus over-current and over-temperature protection of the drive. DC-Bus under-voltage and over-temperature protection are performed by software, while DC-Bus over-current and the DC-Bus over-voltage fault signals utilize the Fault inputs of the DSP on-chip PWM module. The line voltage is measured during initialization of the application. According to the detected level, the 115VAC or 230VAC mains are recognized. If the line voltage is detected outside -15% ... +10% of the nominal voltage, the fault “Out of the Mains Limit” disables drive operation. If any of the above mentioned faults occur, the motor control PWM outputs are disabled in order to protect the drive. The fault status can only be exited when the fault conditions have disappeared and the Start/Stop switch is moved to the STOP position. The fault state is indicated by the on-board LED.

The SR power stage uses a unique configuration of power devices, different than AC or BLDC configuration. SR software would cause the destruction of AC or BLDC power stages due to the simultaneous switching of the power devices. Since the application software could be accidentally loaded into an AC or BLDC drive, the software incorporates a protection feature to prevent this. Each power stage contains a simple module which generates a logic signal sequence that is unique for that type of power stage. During the initialization of the chip, this sequence is read and evaluated according to the decoding table. If the correct SR power stage is not identified, the “Wrong Power Stage” fault disables drive operation.

3.3.2 Initialization and Start-Up

Before the motor can be started, rotor alignment and initialization of the control algorithms must be performed (see **Figure 3-2**). Initialization of the control algorithm includes the measurement of the actual start-up phase resistance.

First, the rotor needs to be aligned to a known position to be able to start the motor in the desired direction of rotation. This is done in the following steps:

1. Two phases (Phases B & C) are turned on simultaneously
2. After 50msec one phase (Phase C) is turned off, the other phase (Phase B) stays powered
3. After an additional 550 msec, the rotor is stabilized enough in the aligned position with respect to the powered phase (Phase B).

Step 1 provides the initial impulse to the rotor. If Phase B is exactly in an unaligned position and thus does not generate any torque, Phase C provides the initial movement. Then, Phase C is disconnected and Phase B stays powered (Step 2). The stabilization pulse to Phase B must be long enough to stabilize the rotor in the aligned position with respect to that phase.

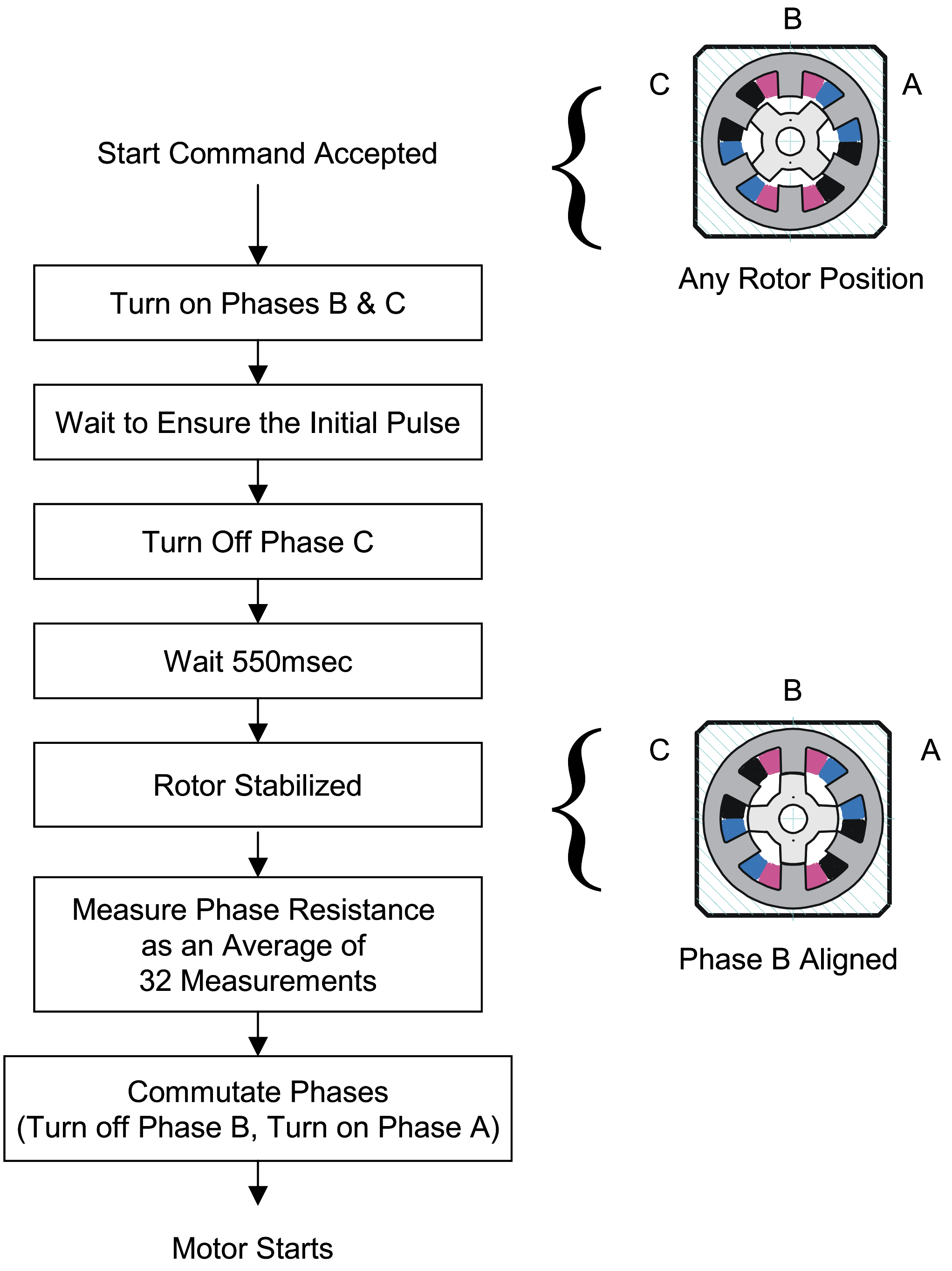


Figure 3-2. Start-Up Sequence

When the rotor is stabilized at the known position, measurement of the phase resistance of the powered phase can be performed. Phase resistance is calculated from the measured phase current i_{ph}, DC-Bus voltage $U_{DC\text{-}Bus}$ and the applied PWM duty cycle γ. It is assumed that the resistance of all three phases is identical. The phase resistance R_0 is calculated as:

$$R_0 = \frac{\sum(\gamma \cdot U_{DCBus})}{\sum i_{ph}} \qquad \textbf{(3-1)}$$

In total, stabilization and the resistance measurement take 1 sec. After this time, the rotor is stable enough to reliably start the motor in the desired direction of rotation. When the phase resistance has been measured, the motor can be started by commutation of the phases (turning off the stabilization of Phase B and applying power to the start-up Phase A).

This starting sequence is followed for every start-up of the motor because neither the initial rotor position nor the actual phase resistance is known.

3.3.3 Commutation Algorithm and Resistance Estimation

The core of the control algorithm includes the calculation of the commutation angle, the flux linkage, the reference flux, the commutation of phases and an estimation of the phase resistance.

Calculation of the commutation angle calculation is performed regularly during motor operation according to **(2-1)** and **(2-2)**.

Flux linkage is estimated during a complete current stroke of the powered phase, from the moment the phase is turned on until the moment the phase current disappears. It serves for both position estimation (determination of the commutation instance) and for resistance estimation. Commutation of the motor phases is based on a comparison of the actual estimated flux linkage and the reference flux linkage for the required commutation angle (see **Section 2.3.1**). Phase resistance is estimated according to the flux linkage error, which is captured the moment the phase current disappears (see **Section 2.3.3**). A detailed block diagram of the control algorithm is shown in **Figure 3-3**.

The control process starts at the moment the given phase is turned on. It can be either during start-up (after the rotor is aligned and commutated.

When the phase is turned on (θ_{on}), the phase current and the phase voltage are measured simultaneously at the center of the PWM pulses. The phase current, i_{ph} is measured directly using the phase current sensing circuitry with s/w noise elimination implemented, while phase voltage, u_{ph}, is calculated according to the measured DC-Bus voltage and the actual PWM duty cycle γ:

$$u_{ph} = \gamma \cdot U_{DCBus} \tag{3-2}$$

The measured phase current and DC-Bus voltage are used for calculating the actual flux linkage Ψ_{actual} **(2-7)**.

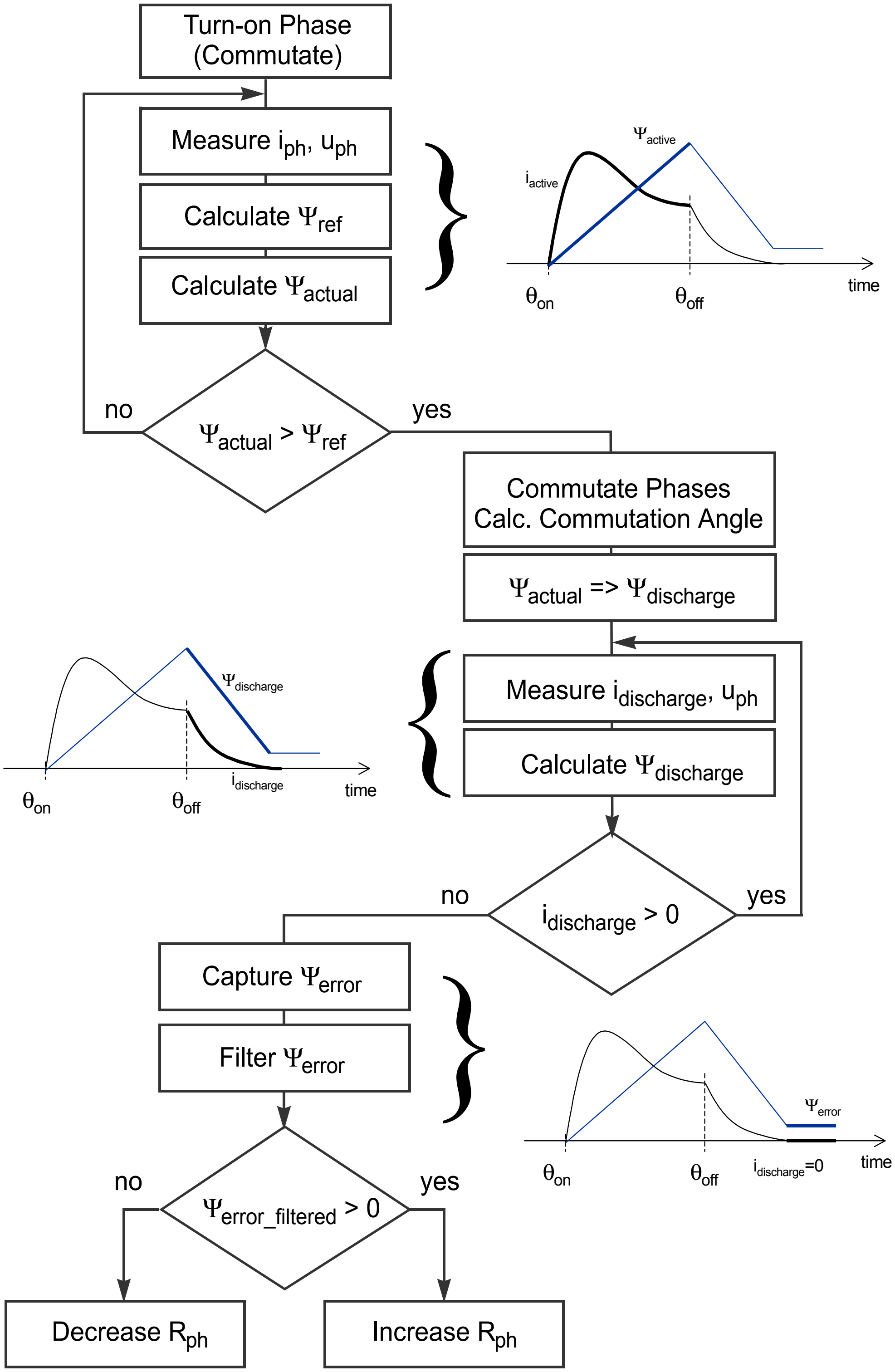


Figure 3-3. Control Flow Diagram

The reference flux linkage Ψ_{ref} for a given commutation angle θ_{off} is a function of the phase current i_{ph}, $\Psi_{ref} = f(i_{ph}, \theta_{off})$. The reference flux linkage characteristic for the aligned position needs to be derived from the motor magnetization characteristic. Such a characteristic for the tested motor is shown in **Figure 3-4**. Compare it with **Figure 2-11** which illustrates the general magnetization curve. As can be seen, the measured characteristic is linear -- we work in the linear part of the magnetization characteristic. For other positions, the reference flux linkage is calculated according to **(2-3)**.

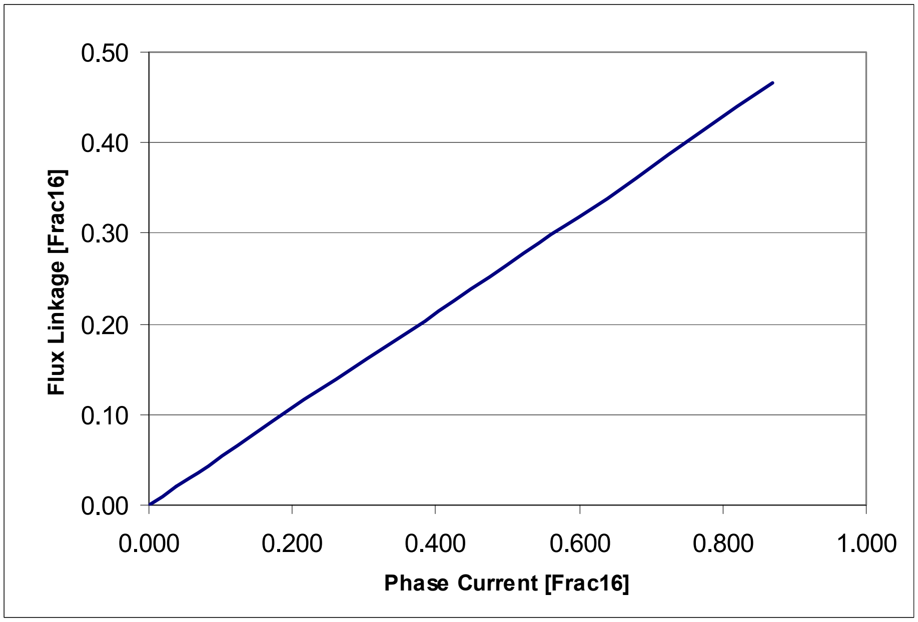


Figure 3-4. Flux Linkage as a Func. of Phase Current for the Aligned Pos.

The estimated flux linkage Ψ_{actual} is compared with the reference flux linkage Ψ_{ref}. If the estimated value is lower than the reference value, the estimation continues regularly at the sampling frequency. When the estimated value reaches the reference value, this indicates that the desired position θ_{off} is achieved. At that moment, commutation of the phases is performed - the powered phase is turned off and the following phase, in the direction of the rotation, is turned on. The flux linkage calculation for determining the following commutation event starts again at an initial values of zero.

When the phase is turned off, the phase current starts to decrease -- the phase is discharged. The flux linkage $\Psi_{discharge}$ continues to be calculated regularly at the rate of the sampling period (PWM frequency) during the phase current discharge. The discharge phase current $i_{discharge}$ is monitored. As soon as the phase current approaches zero, the flux linkage error Ψ_{Error} is captured. The flux linkage error corresponds to the phase resistance error used for the flux linkage calculation.

The flux linkage error is then filtered through several samples in order to eliminate calculation, measurement, and noise error.

The filtered value is used for evaluation of phase resistance according to **(2-21)** and **(2-22)**. If the filtered flux linkage error is greater than zero, the estimated phase resistance is increased by a small amount (0.1%). In the opposite case, the estimated phase resistance is decreased by a small amount (0.1%). The corrected resistance value is then used during the next flux linkage estimation process. In this way, phase resistance is tracked throughout operation.

3.3.4 Current and Voltage Measurement

Precise phase current and DC-Bus voltage measurement is a key factor in the implementation of sensorless flux linkage estimation. Any inaccuracy in the measurement leads to flux linkage estimation error and thus to position estimation error and resistance estimation error.

3.3.5 Current Sensing

Current measurement needs to be investigated according to the current sensors used and the influence of noise on the measurement.

The quality of current measurement depends heavily on the type of current sensors used. The most useful are Hall effect sensors. Unfortunately, these sensors are expensive and thus are not suitable for most cost-sensitive applications. Therefore, current shunt resistors inserted into the current path of the phase are often used (see **Figure 3-5**). The phase current is sensed as a voltage drop across the sense resistor.

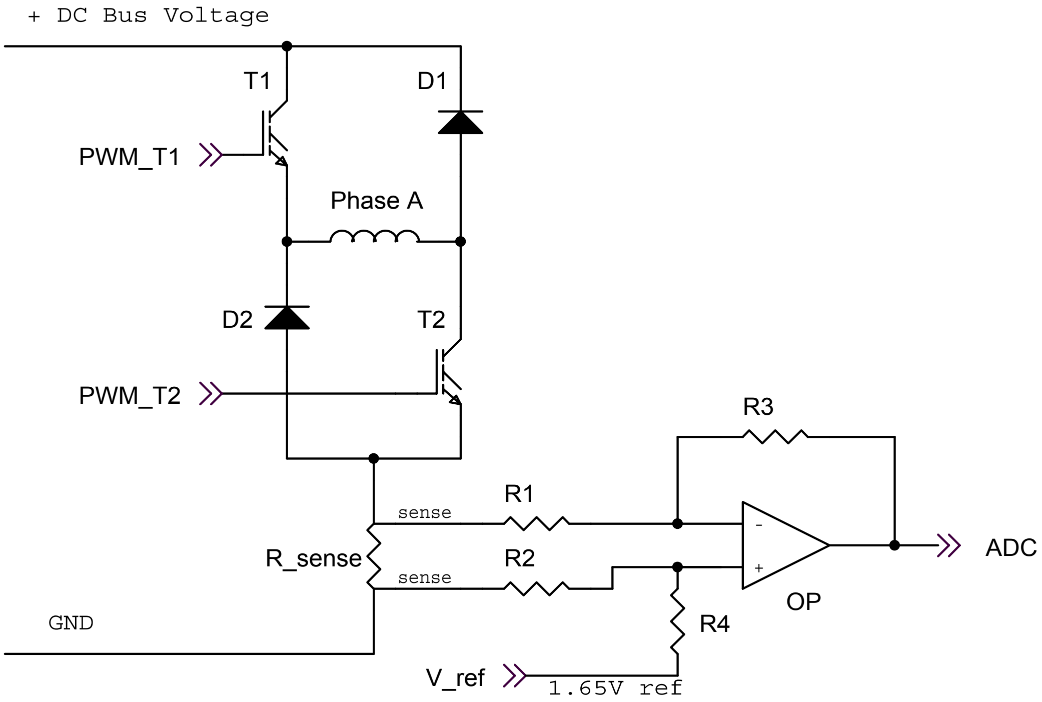


Figure 3-5. Shunt Resistors Current Sensors

When the power switches' soft switching is used (the lower switch is left ON during a complete commutation period, while the upper switch is modulated by the PWM), the current is not visible on the shunt resistor all the time. The soft switching phase current, measured at the shunt resistor, is shown in **Figure 3-6**. The phase current is visible only when both switches are turned on (the phase current flows through switches and the sensing resistor) or when both switches are turned off (phase current flows through the freewheeling diodes and the sensing resistor). When both switches of the phase are turned on, the measured current is negative, so it needs to be inverted. The diagram shows that for a reliable current shape reconstruction, the sensing needs to be synchronized with the PWM frequency at the center of the PWM pulse and both positive and the negative voltage drop polarities should be measured. The zero current may be set to half of the ADC range, so both the positive and the negative voltage drops on the phase current shunt resistors can be measured. The voltage drop is then amplified according to the ADC range. Proceeding like this, the current can be read with accuracy and credibility.

Figure 3-7 illustrates the actual phase currents of a 3-Phase motor, measured on the shunt resistors as described above.

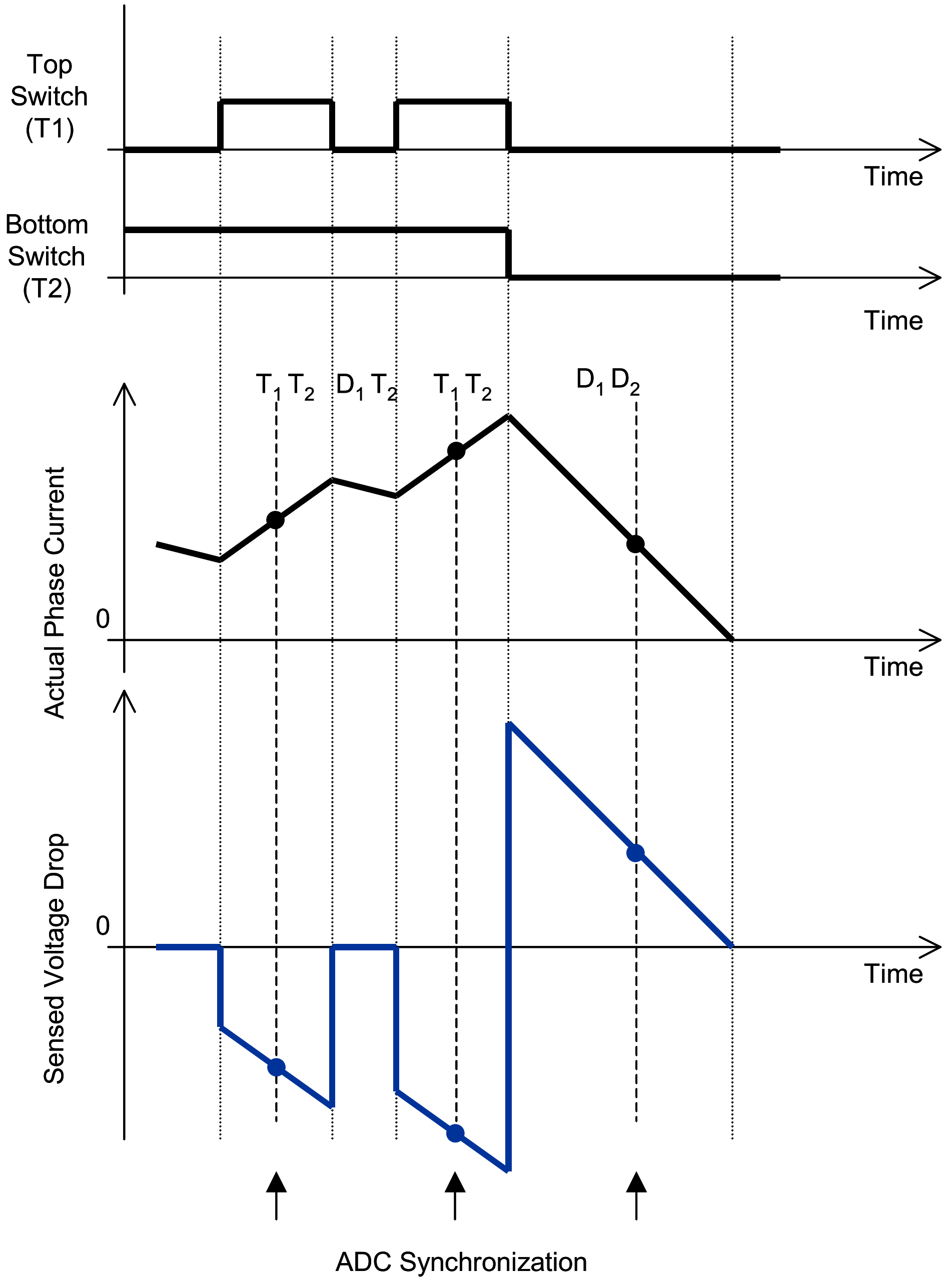


Figure 3-6. Soft Switching Current on Shunt Resistors

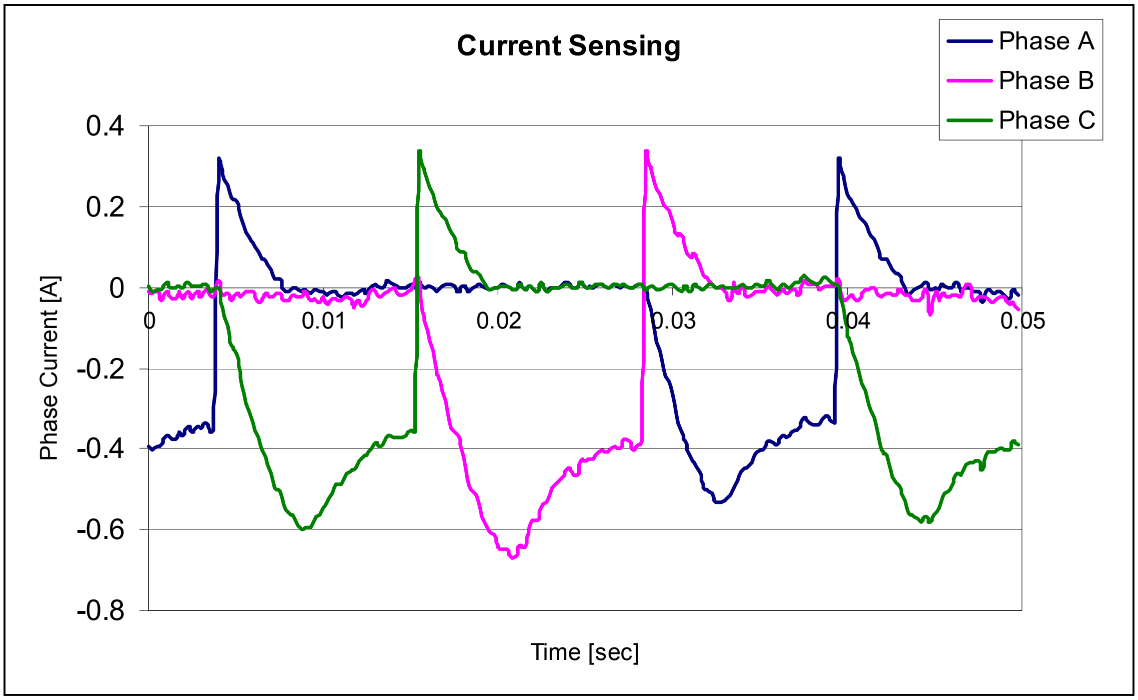


Figure 3-7. Phase Current Measured at Current Shunt Resistors

The low cost shunt resistor sensors bring one serious issue. Due to the low-voltage drop sensed across the shunt current resistors, the measured signals are susceptible to noise.

Based on the assumption that the same noise is induced simultaneously on all measured signals, a technique for noise elimination has been developed and successfully implemented. The method supposes the measurement of two signals simultaneously -- one known signal (a reference) and one signal to be measured. Then the reference signal consists of a known signal and noise, while the measured signal consists of an actual signal and the same noise.

$$MeasuredSignal = ActualSignal + Noise \tag{3-3}$$

$$ReferenceSignal = KnownSignal + Noise \tag{3-4}$$

If the noise is the same, it can be eliminated by subtraction of the reference signal from the measured signal. As described above, the necessary condition is the simultaneous sampling of both signals, ensuring that the noise on both signals is identical.

$$ActualSignal = MeasuredSignal - (ReferenceSignal - KnownSignal) \quad (3-5)$$

This technique has been implemented for phase current sensing. The SR motor is controlled in a way in which the phases are commutated sequentially, which means that as the working phase is turned off, the following phase, in the direction of rotation, is turned on. Thus one phase of the motor is never powered during a complete commutation interval. This phase is considered as a reference. Because the reference phase is not powered, the reference phase current should be equal to zero. The measured value of the reference current can be then considered as noise for a given commutation interval. The actual phase current is equal to the difference between the measured current and the reference current:

$$I_{ph} = I_{measured} - I_{reference} \quad (3-6)$$

The reference signal needs to be commutated together with the commutation of the phases. **Table 3-1** defines the active, discharge and reference phases for the commutation sequence C - B - A - C. It is derived from **Figure 3-7**.

Table 3-1. Commutation Sequence of the Reference Phase

Step	Active Phase	Discharge Phase	Reference Phase
1	C	A	B
2	B	C	A
3	A	B	C
1	C	A	B

The efficiency of the current sensing noise reduction technique is illustrated in **Figure 3-8**. The figures illustrate the phase current as it is measured (the active phase current is inverted compared to **Figure 3-7**), and the same current with the implemented noise reduction technique. As can be seen, the implemented technique improves current sensing significantly. It eliminates not only the noise on the current sensors, but also the noise induced on the sensing cables and the noise of the ADC reference power supply. Thus, position estimation and resistance evaluation are improved as well.

3.3.6 Voltage Sensing

The DC-Bus voltage sensor is represented by a simple voltage divider. DC-Bus voltage does not change rapidly. It is nearly constant with the ripple given by the power supply structure. If a bridge rectifier for rectification of AC line voltage is used, the ripple frequency is two times the AC line frequency.If the power stage is designed correctly, the ripple amplitude should not exceed 10% of the nominal DC-Bus value.

The measured DC-Bus voltage needs to be filtered in order to eliminate noise. One of the most useful techniques is a moving average filter that calculates an average value from the last N samples:

$$u_{DCBus} = \sum_{n=1}^{-N} u_{DCBus}(n) \qquad \textbf{(3-7)}$$

In order to increase the precision of the voltage sensing, the voltage drop on the power switches and on the diodes of the power stage can be incorporated into the determination of the actual voltage present in the motor phase.

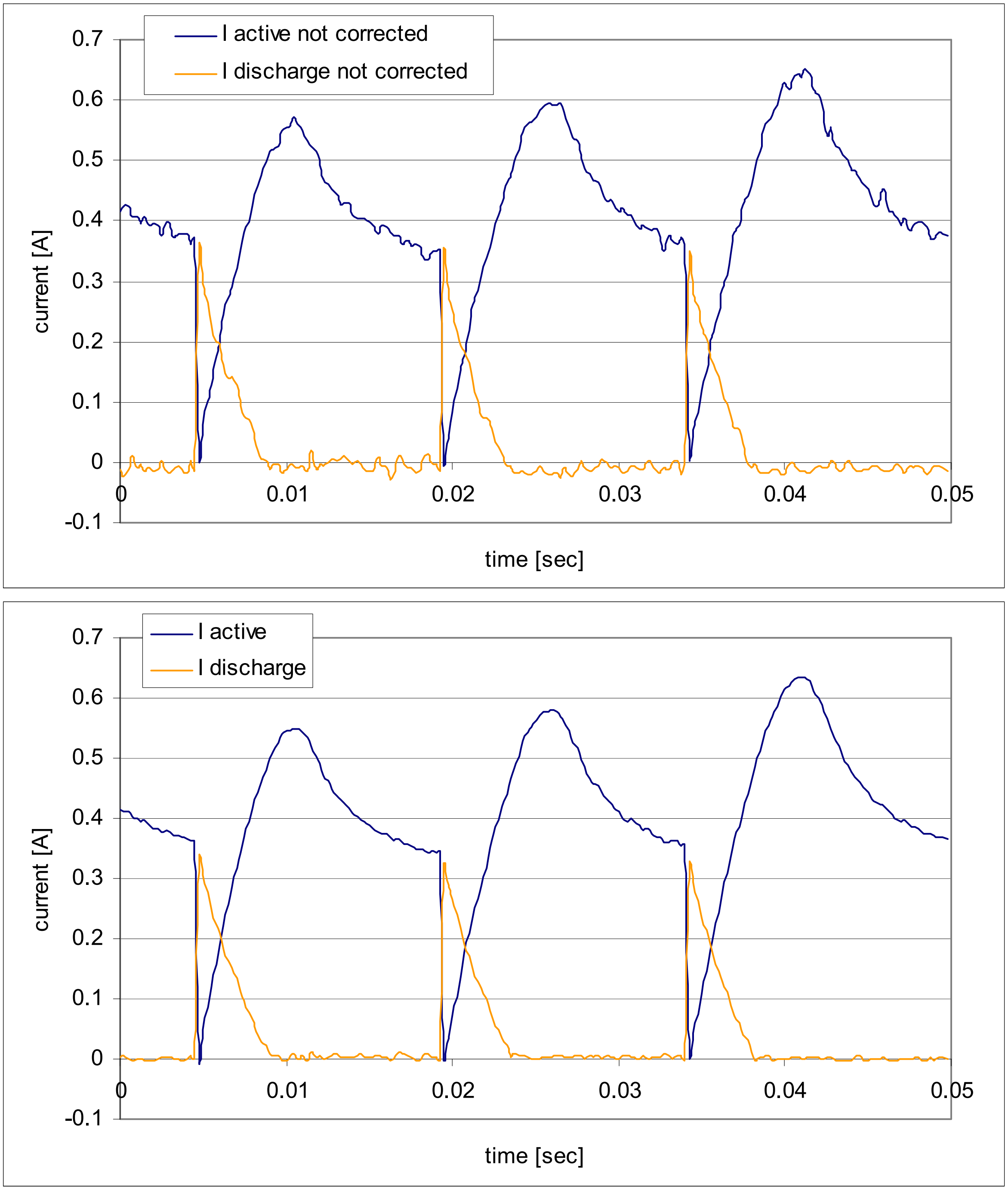


Figure 3-8. Measured 3-Phase Currents with & without Noise Correction Implemented

3.3.7 Power Module Temperature Sensing

The measured power module temperature is used for thermal protection The hardware realization is shown in **Figure 3-9.** The circuit consists of four diodes connected in series, a bias resistor, and a noise suppression capacitor. The four diodes have a combined temperature coefficient of 8.8 mV/°C. The resulting signal, *Temp_sense*, is fed back to an A/D input where software can be used to set safe operating limits. In the presented application, the temperature in degrees Celsius is calculated according to the conversion equation:

$$\text{temp} = \frac{\text{Temp\_sense - b}}{\text{a}} \tag{3-8}$$

where:

temp is the power module temperature in degrees Celsius

*Temp_sense*is the voltage drop on the diodes which is measured by ADC

a is the diode-dependent conversion constant (a = -0.0073738)

b is the diode-dependent conversion constant (b = 2.4596)

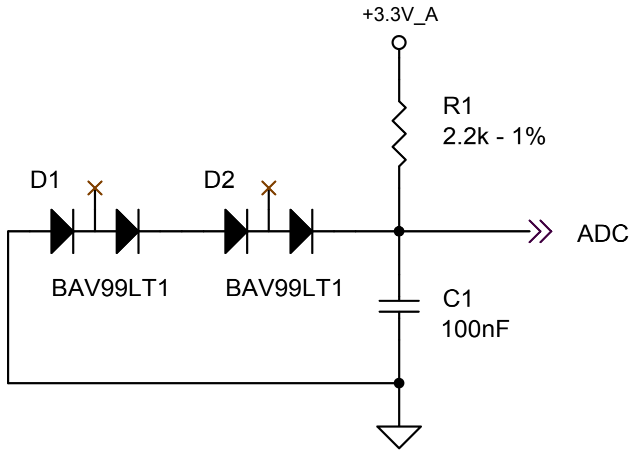


Figure 3-9. Temperature Sensor Topology

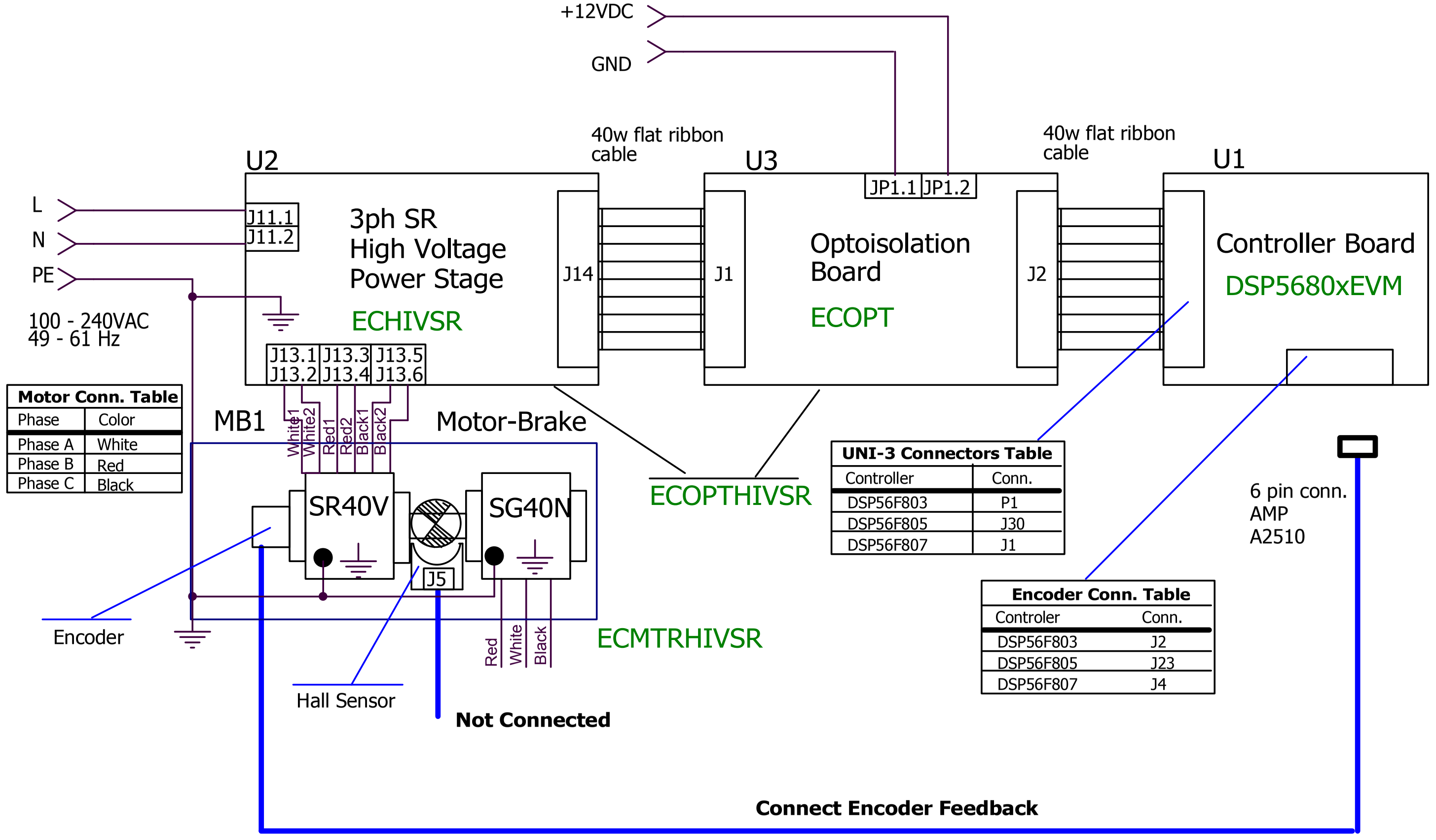
Section 4. Hardware

4.1 Contents

4.2 System Configuration

The application is designed to drive the 3-phase SR motor. The application is controlled by the Motorola DSP56F805 motor control DSP. It consists of the following modules (see **Figure 4-1**):

- DSP56F805EVM Control Board
- 3 ph SR High Voltage Power Stage
- Optoisolation Board
- 3-phase Switched Reluctance Motor



Motor Conn. Table

Phase	Color
Phase A	White
Phase B	Red
Phase C	Black

UNI-3 Connectors Table

Controller	Conn.
DSP56F803	P1
DSP56F805	J30
DSP56F807	J1

Encoder Conn. Table

Controler	Conn.
DSP56F803	J2
DSP56F805	J23
DSP56F807	J4

Figure 4-1. 3-Phase SR High Voltage Platform Configuration

4.3 DSP56F805EVM Control Board

The DSP56F805EVM facilitates the evaluation of various features present in the DSP56F805 part. The DSP56F805EVM can be used to develop real-time software and hardware products based on the DSP56F805. The DSP56F805EVM provides the features necessary for a user to write and debug software, demonstrate the functionality of that software and interface with the customer's application-specific device(s). The DSP56F805EVM is flexible enough to allow a user to fully exploit the DSP56F805's features to optimize the performance of their product, as shown in **Figure 4-2**.

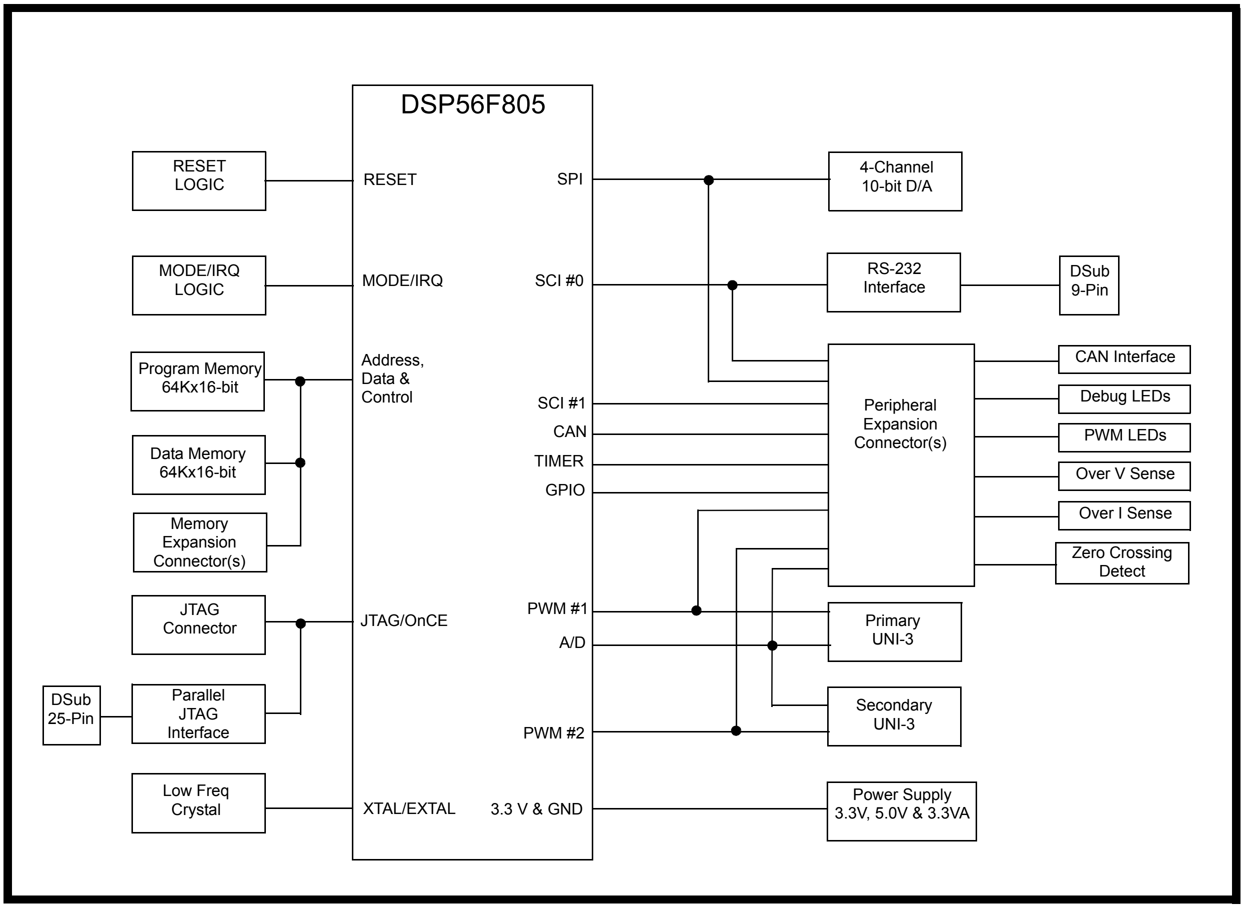


Figure 4-2. Block Diagram of the DSP56F805EVM

4.3.1 DSP56F805EVM Configuration Jumpers

Eighteen jumper groups, (JG1-JG18), shown in **Figure 4-3**, are used to configure various features on the DSP56F805EVM board. **Table 4-1** describes the default jumper group settings.

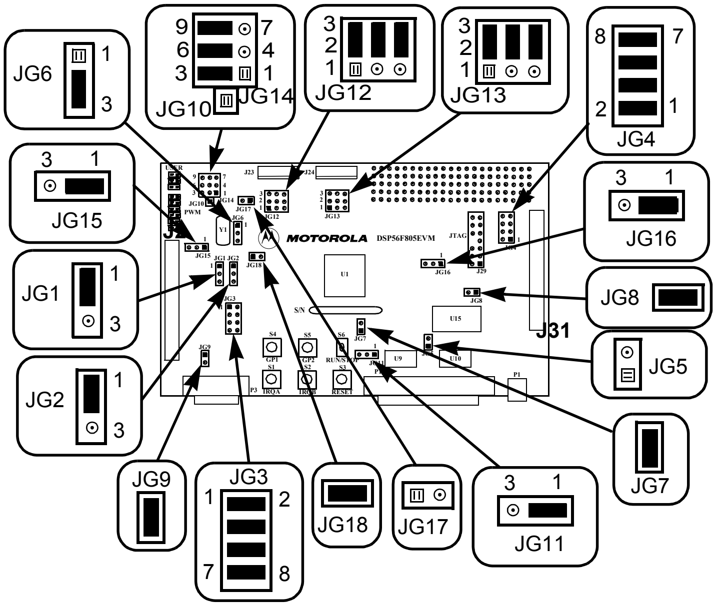


Figure 4-3. DSP56F805EVM Jumper Reference

Table 4-1. DSP56F805EVM Default Jumper Options

Jumper Group	Comment	Jumpers Connections
JG1	PD0 input selected as a high	1–2
JG2	PD1 input selected as a high	1–2
JG3	Primary UNI-3 serial selected	1–2, 3–4, 5–6 & 7–8
JG4	Secondary UNI-3 serial selected	1–2, 3–4, 5–6 & 7–8
JG5	Enable on-board Parallel JTAG Host Target Interface	NC
JG6	Use on-board crystal for DSP oscillator input	2–3
JG7	Selects DSP’s Mode 0 operation upon exit from reset	1-2

Table 4-1. DSP56F805EVM Default Jumper Options (Continued)

Jumper Group	Comment	Jumpers Connections
JG8	Enable on-board SRAM	1–2
JG9	Enable RS-232 output	1–2
JG10	Secondary UNI-3 Analog Temperature Input unused	1–2
JG11	Use Host power for Host Target Interface	1–2
JG12	Primary Encoder Input Selected	2–3, 5–6 & 8–9
JG13	Secondary Encoder Input Selected	2–3, 5–6 & 8–9
JG14	Primary UNI-3 3-Phase Current Sense Selected as Analog Inputs	2–3, 5–6 & 8–9
JG15	Primary UNI-3 Phase A Over-Current Selected for FAULTA1	1–2
JG16	Secondary UNI-3 Phase B Over-Current Selected for FAULTB1	1–2
JG17	CAN termination unselected	NC
JG18	Use on-board crystal for DSP oscillator input	1–2

An interconnection diagram is shown in **Figure 4-4** for connecting the PC and the external 12V DC power supply to the DSP56F805EVM board.

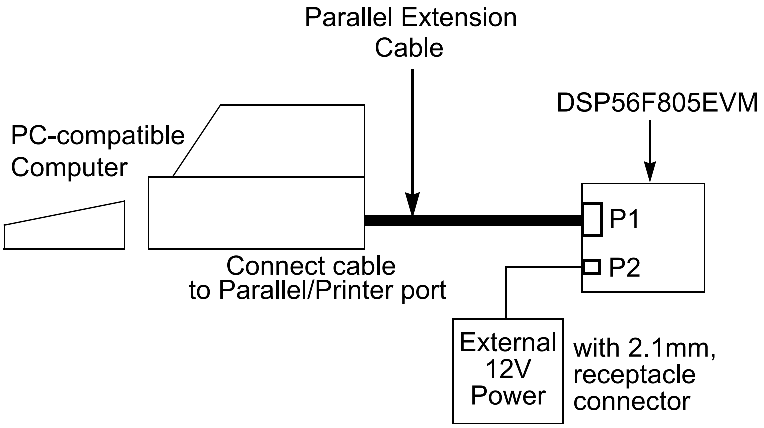


Figure 4-4. Connecting the DSP56F805EVM Cables

Perform the following steps to connect the DSP56F805EVM cables:

4. Connect the parallel extension cable to the Parallel port of the host

computer.

5. Connect the other end of the parallel extension cable to P1, shown in **Figure 4-4**, on the DSP56F805EVM board. This provides the connection which allows the host computer to control the board.
6. Make sure that the external 12V DC, 4.0A power supply is not plugged into a 120V AC power source.
7. Connect the 2.1mm output power plug from the external power supply into P2, shown in **Figure 4-4**, on the DSP56F805EVM board.
8. Apply power to the external power supply. The green Power-On LED, LED10, will illuminate when power is correctly applied.

4.4 3-Phase Switched Reluctance High-Voltage Power Stage

Motorola's embedded motion control series high-voltage (HV) switched reluctance (SR) power stage is a 180 watt (1/4 horsepower), 3-phase power stage that will operate off of dc input voltages from 140 volts to 230 volts and ac line voltages from 100 volts to 240 volts. In combination with one of Motorola's Embedded Motion Control Series control boards and an optoisolation board, it provides a software development platform that allows algorithms to be written and tested, without the need to design and build a power stage. It supports a wide variety of algorithms for controlling switched reluctance motors.

Input connections are made via 40-pin ribbon cable connector J14. Power connections to the motor are made on output connector J13. Phase A, phase B, and phase C are labeled Ph. A, Ph. B, Ph. C on the board. Power requirements are met with a single external 140-volt to 230-vo lt dc power supply or an ac line voltage. Either input is supplied through connector J11. Current measuring circuitry is set up for 2.93 amps full scale. Both bus and phase leg currents are measured. A cycle-by-cycle overcurrent trip point is set at 2.69 amps.

The HV SR power stage has both a printed circuit board and a power substrate.

The printed circuit board contains IGBT gate drive circuits, analog signal conditioning, low-voltage power supplies, power factor control circuitry, and some of the large passive power components. This board also has a MC68HC705JJ7 microcontroller used for board configuration and identification. All of the power electronics that need to dissipate heat are mounted on the power substrate. This substrate includes the power IGBTs, brake resistors, current-sensing resistors, a power factor correction MOSFET, and temperature sensing diodes. **Figure 2-1** shows a block diagram.

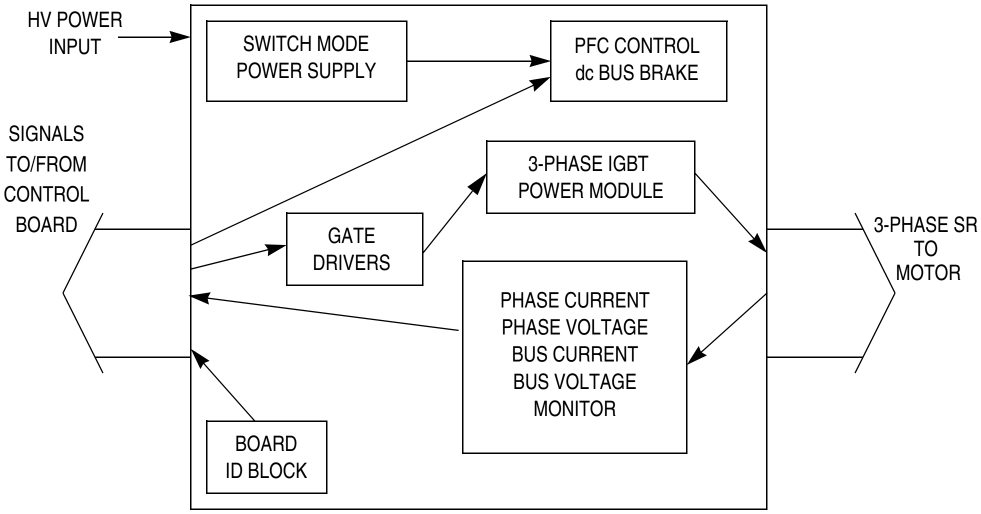


Figure 4-5. Block Diagram

The electrical characteristics in **Table 2-1** apply to operation at 25 C with a 160-Vdc supply voltage.

Table 4-2. Electrical Characteristics

Characteristic	Symbol	Min	Typ	Max	Units
dc input voltage	Vdc	140	160	230	V
ac input voltage	Vac	100	208	240	V
Quiescent current	I_{CC}	—	70	—	mA
Min logic 1 input voltage	V_{IH}	2.0	—	—	V
Max logic 0 input voltage	V_{IL}	—	—	0.8	V
Input resistance	R_{In}	—	10 kΩ	—	
Analog output range	V_{Out}	0	—	3.3	V
Bus current sense voltage	I_{Sense}	—	563	—	mV/A
Bus voltage sense voltage	V_{Bus}	—	8.09	—	mV/V
Peak output current	I_{PK}	—	—	2.8	A
Brake resistor dissipation (continuous)	P_{BK}	—	—	50	W
Brake resistor dissipation (15 sec pk)	$P_{BK(Pk)}$	—	—	100	W
Total power dissipation	P_{diss}	—	—	85	W

4.5 Optoisolation Board

Motorola's embedded motion control series optoisolation board links signals from a controller to a high-voltage power stage. The board isolates the controller, and peripherals that may be attached to the controller, from dangerous voltages that are present on the power stage. The optoisolation board's galvanic isolation barrier also isolates control signals from high noise in the power stage and provides a noise-robust systems architecture.

Signal translation is virtually one-for-one. Gate drive signals are passed from controller to power stage via high-speed, high dv/dt, digital optocouplers. Analog feedback signals are passed back through HCNR201 high-linearity analog optocouplers. Delay times are typically

250 ns for digital signals, and 2 µs for analog signals. Grounds are separated by the optocouplers' galvanic isolation barrier.

Both input and output connections are made via 40-pin ribbon cable connectors. The pin assignments for both connectors are the same. For example, signal PWM_AT appears on pin 1 of the input connector and also on pin 1 of the output connector. In addition to the usual motor control signals, an MC68HC705JJ7CDW serves as a serial link, which allows controller software to identify the power board.

Power requirements for controller side circuitry are met with a single external 12-Vdc power supply. Power for power stage side circuitry is supplied from the power stage through the 40-pin output connector.

The electrical characteristics in **Table 4-3** apply to operation at 25°C, and a 12-Vdc power supply voltage.

Table 4-3. Electrical Characteristics

Characteristic	Symbol	Min	Typ	Max	Units	Notes
Power Supply Voltage	Vdc	10	12	30	V	
Quiescent Current	I_{CC}	70(1)	200(2)	500(3)	mA	dc/dc converter
Min Logic 1 Input Voltage	V_{IH}	2.0	—	—	V	HCT logic
Max Logic 0 Input Voltage	V_{IL}	—	—	0.8	V	HCT logic
Analog Input Range	V_{In}	0	—	3.3	V	
Input Resistance	R_{In}	—	10	—	kΩ	
Analog Output Range	V_{Out}	0	—	3.3	V	
Digital Delay Time	t_{DDLY}	—	0.25	—	µs	
Analog Delay Time	t_{ADLY}	—	2	—	µs	

1. Power supply powers optoisolation board only.
2. Current consumption of optoisolation board plus DSP EMV board (powered from this power supply)
3. Maximum current handled by dc/dc converters

4.6 Motor-Brake Specifications

The SR Motor Brake set incorporates a 3-Phase SR Motor and attached BLDC motor brake. The detailed specifications are listed in **Table 4-4**.

The SR motor has six stator poles and four rotor poles. This combination yields 12 strokes (or pulses) per single mechanical revolution. The SR motor is characterized by a dedicated inductance profile. The motor inductance profile as a function of mechanical position is shown in **Figure 4-6**. The mechanical angle 90^{o}_{mech} corresponds to one electrical period of the stroke. The presented profile was used for the determination of the advanced commutation angle.

On the motor brake shaft, a position encoder and position Hall sensor are attached. They allow position sensing if it is required by the control algorithm. The introduced drive uses the Encoder for the position determination

Table 4-4. Motor - Brake Specifications

Set Manufacturer	EM Brno, Czech Republic	
Motor Specification:	**eMotor Type:**	SR40V (3-Phase SR Motor)
	Stator / Rotor Poles:	6/4
	Speed Range:	< 5000 rpm
	Nominal Voltage:	3 x 300V
	Nominal Current:	1.2A
Brake Specification:	**Brake Type**	SG40N 3-Phase BLDC Motor
	Nominal Voltage:	3 x 27V
	Nominal Current:	2.6 A
Position Encoder	**Type**	Baumer Electric BHK 16.05A 1024-12-5
	Pulses per Revolution	1024

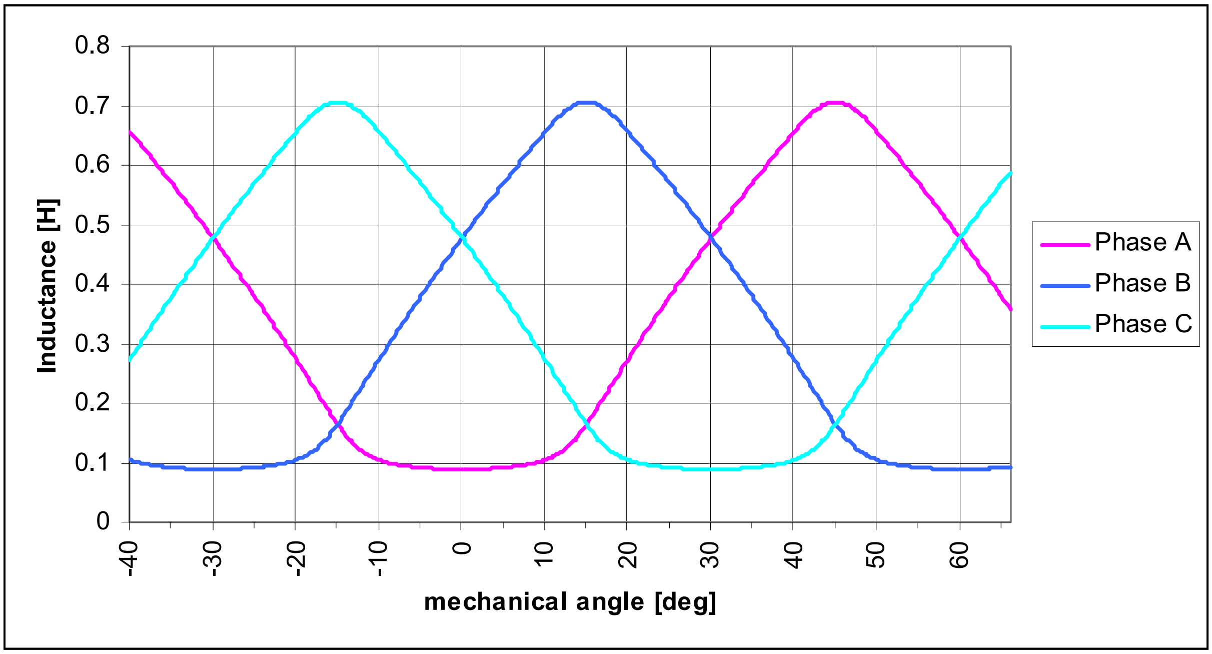


Figure 4-6. Inductance Characteristic

4.7 Hardware Documentation

All the system parts are supplied and documented according to the following references:

- U1 - Controller Board for DSP56F805:
  - supplied as: DSP56805EVM
  - described in: DSP56F803EVMUM/D DSP Evaluation Module Hardware User's Manual
- U2 - 3-Phase SR High-Voltage Power Stage
  - supplied as a kit with an Optoisolation Board as: ECOPTHIVSR
  - described in: MEMC3PSRHVPSUM/D Motorola Embedded Motion Control 3-Phase SR High-Voltage Power Stage User's Manual
- U2 - 3-Phase SR High-Voltage Power Stage

  - supplied as a kit with an Optoisolation Board as: ECOPTHIVSR
  - described in: MEMC3PSRHVPSUM/D Motorola Embedded Motion Control 3-Phase SR High-Voltage Power Stage User's Manual
- U3 - Optoisolation Board
  - supplied with 3 ph AC/BLDC High Voltage Power Stage as: ECOPTHIVACBLDC
  - or supplied alone as: ECOPT - optoisolation board
  - described in: *Motorola Embedded Motion Optoisolation Board User's Manual* MEMCOBUM/D
- MB1 Motor-Brake AM40V + SG40N
  - supplied as: ECMTRHIVAC

Detailed descriptions of individual boards can be found in comprehensive User's Manuals belonging to each board. The manuals are available on the Motorola web. The User's Manual incorporates the schematic of the board, description of individual function blocks and a bill of materials. An individual board can be ordered from Motorola as a standard product.

WARNING: ***It is strongly recommended to use an opto-isolation (optocouplers and optoisolation amplifiers) during development time to avoid electric shock and any damage to the development equipment.***

Section 5. Software Design

5.1 Contents

5.2 Introduction

This section describes the design of the software blocks of the drive. The software will be described in terms of:

- Control algorithm data flow
- State diagram
- Software implementation

5.2.1 Data Flow

The control algorithm of a closed loop SR drive is described in **Figure 5-1** and **Figure 5-2**. It is based on the system description.

The individual processes are described in detail in the following sections.

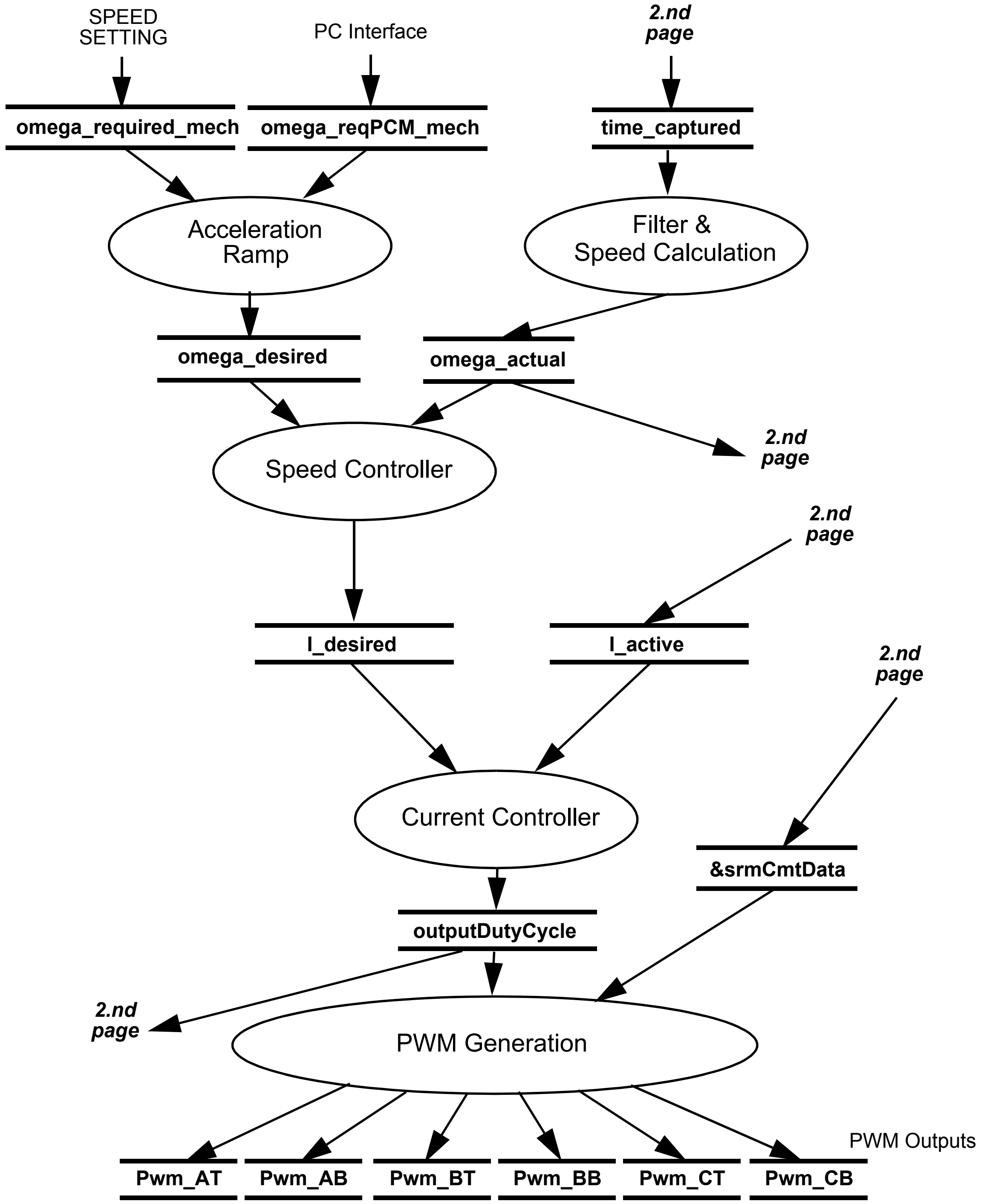


Figure 5-1. System Data Flow I - Speed & Current Control

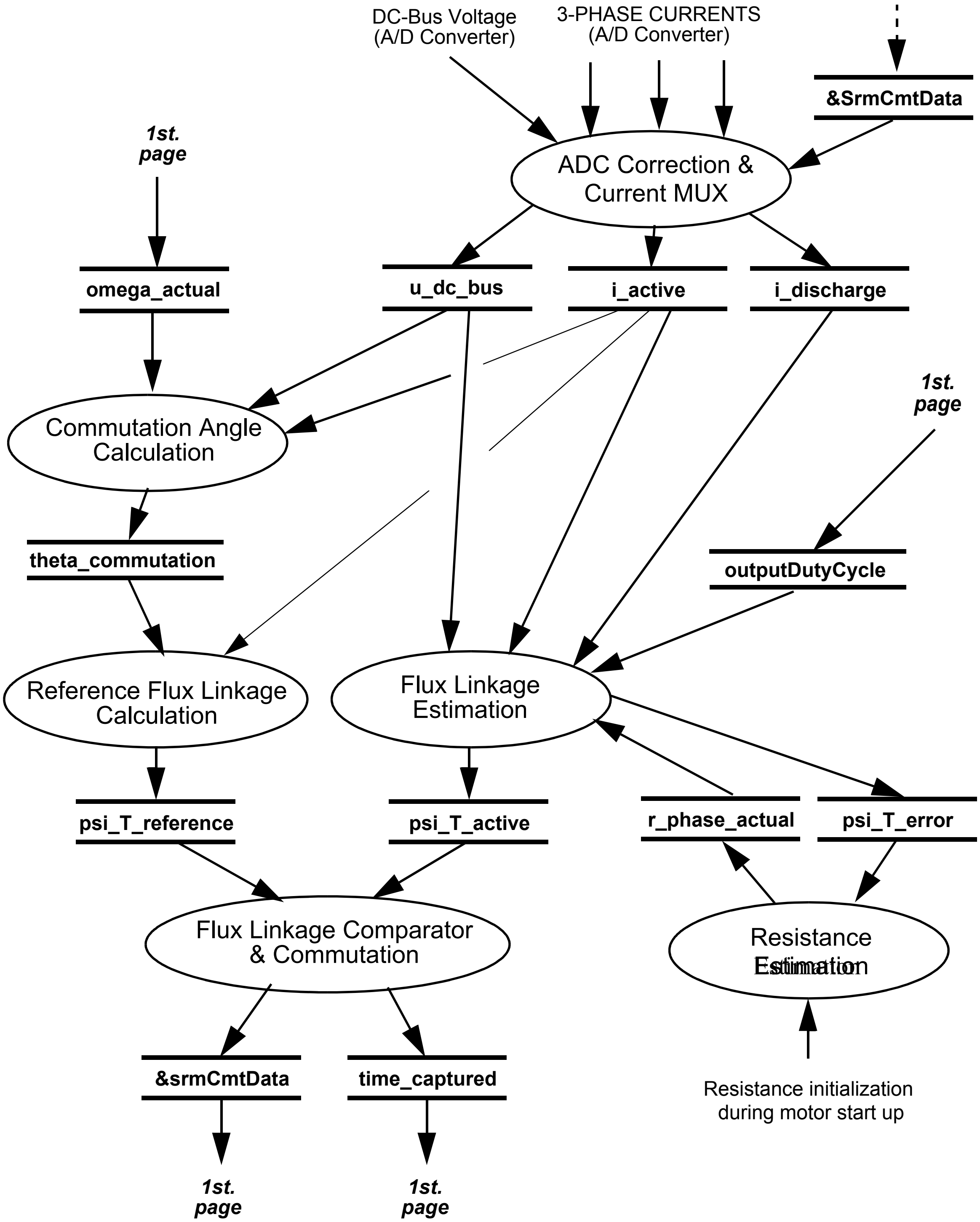


Figure 5-2. System Data Flow II - Commutation

5.2.1.1 Acceleration Ramp

This process calculates the desired speed based on the required speed according to the acceleration / deceleration ramp. The required speed is set either manually, using the push buttons (when in manual operational mode), or by PC master software (when in PC master software operational mode).

5.2.1.2 Filter and Speed Calculation

The process calculates the actual speed of the motor. The calculation is based on the evaluation of the time between the commutation instances.

Each time the commutation is performed, the actual time is captured. The process reads the time between the sequential commutation events and calculates the actual motor speed accordingly.

A software moving average filter applied to the speed measurement is incorporated into the process for greater noise immunity. The actual motor speed is calculated as the average value of the last four measurements.

5.2.1.3 Speed Controller

This process calculates the desired phase current according to the speed error. Speed error is the difference between the actual speed and desired speed. A Proportional-Integrational (PI) type of controller is implemented. The constants of the speed controller are tuned experimentally according to the load profile and the speed limits.

5.2.1.4 Current Controller

This process calculates the duty cycle of the PWM based on phase current error. Phase current error is the difference between the actual phase current and desired phase current. A PI type of controller is implemented. The current controller constants are tuned experimentally according to the type of motor used.

5.2.1.5 PWM Generation

This process sets the on-chip PWM module for generation of the control pulses for the 3-Phase SR motor power stage. Generation of these pulses is based on the software control register that is formulated by the process of Commutation Calculation and is based on the required duty cycle generated by the Speed Controller process. The calculated software control word is loaded into the proper PWM register and the PWM duty cycle is updated according to the required duty cycle. The PWM Generation process is accessed regularly at a rate given by the PWM frequency. It is frequent enough to ensure the precise generation of commutation pulses.

5.2.1.6 ADC Correction and Current MUX

This process takes care of the Analog-to-Digital Converter. The sampling of the ADC is synchronized to the PWM pulses. The process selects the proper ADC channels to be converted and reads and processes the results of the ADC conversion.

The active and discharge phase currents are selected and corrected using the measured reference noise signal. The DC-Bus voltage and temperature are filtered using a moving average filter. See **Section 3.3.4 Current and Voltage Measurement** for a detailed description.

5.2.1.7 Flux Linkage Estimation

This process calculates the actual flux linkage. The calculation of the active flux linkage is started with each commutation of phases. Flux linkage error is captured at the end of the current pulse and is further used for phase resistance estimation (see **Section 3.3.3** and **Section 2.3.2**).

5.2.1.8 Commutation Angle Calculation

This process calculates the commutation angle according to the actual speed, the DC-Bus voltage, and desired current (see **Section 3.3.3**).

5.2.1.9 Reference Flux Linkage Calculation

This process calculates the reference flux linkage according to the stored characteristic $\Psi(i_{phase})$ of the aligned position. The process requires the commutation angle and the actual phase current for determination of the reference flux linkage (see **Section 3.3.3**).

5.2.1.10 Flux Linkage Comparator & Commutation

This process compares the reference flux linkage and the active flux linkage to determine commutation events. When the actual flux linkage exceeds the reference, a commutation is performed (see **Section 3.3.3**). Also, the actual time is captured to be used for actual speed calculation.

The DSP on-chip PWM module is used in a mode for generation of independent output signals that can be controlled either by software or by the PWM module.

The commutation technique distinguishes the three following cases:

- When the PWM output needs to be modulated, the PWM generator controls the channel directly
- When the PWM output needs to be switched to an inactive state (0), the software output control of the corresponding PWM channel is handed over and the channel is turned off manually
- When the PWM output needs to be switched to the active state (1), the software output control of the corresponding PWM channel is handed over and the channel is turned on manually

The on-chip PWM module enables control of the outputs from the PWM module either by the PWM generator, or by using the software. Setting the output control enable bit, OUTCTLx, enables software to drive the PWM outputs instead of the PWM generator. In independent mode, with OUTCTLx = 1, the output bit OUTx controls the PWMx channel. Setting or clearing the OUTx bit activates or deactivates the PWMx output. The OUTCTLx and OUTx bits are in the PWM output control register.

This control technique requires the preparation of the output control register. For the calculation of the OUTCTLx and OUTx bits in the PWM output control register, a dedicated commutation algorithm, **3-Phase SR**

Motor Commutation Handler for H/W Configuration 2-Switches-per-Phase, `srmcmt3ph2spp,` was developed. The algorithm generates an output control word according to the desired action and the desired direction of rotation. For example, when Phase A needs to be turned off, the algorithm sets the corresponding OUTCTLx bits to enable the output control of the required PWMs and clears the OUTx bits to turn off the PWMs. The other output control register bits are not affected.

5.2.1.11 Resistance Estimation

This process evaluates the flux linkage estimation error at the end of the phase current stroke and estimates the actual phase resistance (see **Section 3.3.3**).

5.2.2 State Diagram

The processes described above are implemented in a single state machine, as illustrated in **Figure 5-3** The state machine provides a transition among the application states INIT, STOP, RUN, FAULT. The following variables are used to invoke the transition between the individual states:

- switchState (Stop, Run): state of the Start/Stop switch
- appFault (NO_FAULT, any fault): fault occurrence
- appOpMode (change from Manual to PC and vice versa): change operational mode

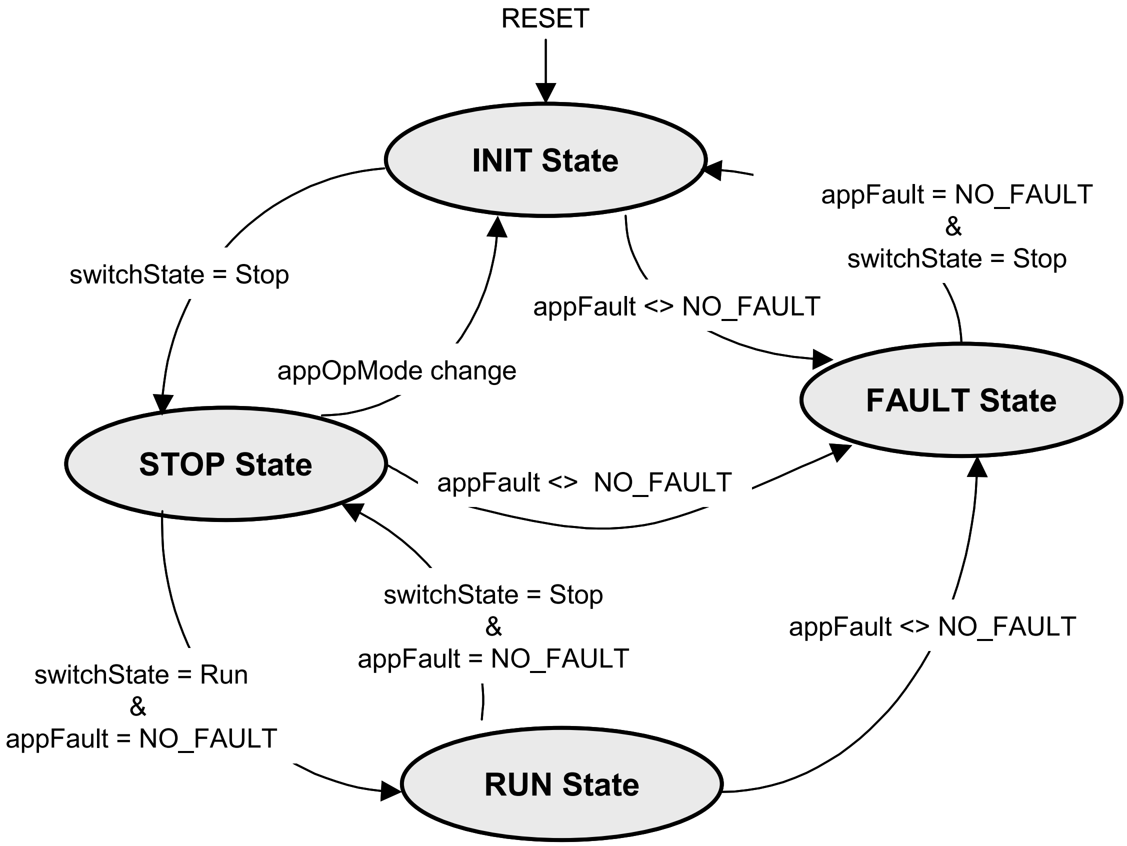


Figure 5-3. Application State Diagram

5.2.2.1 Application State - INIT

After RESET the application enters the INIT state. In this state, the drive is disabled and the motor cannot be started.

If any fault is detected, the application transits to the FAULT state (protection against faults). If no fault is present, and the Start/Stop switch is detected in the STOP position, the application transits to the STOP state (protection against start after reset if the Start/Stop switch is accidentally in START position).

5.2.2.2 Application State - STOP

The STOP state can be entered either from the INIT state or from the RUN state. In the STOP state, the drive is enabled and the application waits for the START command.

When the application is in the STOP state, the operational mode can be changed - either from MANUAL mode to PC master software mode or vice versa. When the operational mode is changed, the application always transits to the INIT state.

If any fault in the STOP state is detected, the application enters the FAULT state (fault protection). If no fault is present and the start command is accepted, the application transits to the RUN state and the motor is started.

5.2.2.3 Application State - RUN

The RUN state can be entered from the STOP state. In the RUN state the drive is enabled and the motor is running.

If any fault in the RUN state is detected, the application enters the FAULT state (fault protection). If no fault is present and the stop command is accepted the application transits to the STOP state and the motor is stopped.

5.2.2.4 Application State - FAULT

The STOP state can be entered from any state. In the FAULT state, the drive is disabled and the application waits for the faults to be cleared.

When it is detected that the fault has been eliminated, and the fault clear command is accepted (the Start/Stop switch is moved to the stop position), then the application transits to the INIT state.

5.2.3 Software Design

The general software diagram incorporates: (1) the Main routine entered from Reset, and (2) the Interrupt Service Routines (ISR). The diagram is illustrated in **Figure 5-4**.

After Reset, the Main routine provides board identification, initialization of the DSP, initialization of the application, and then it enters an infinite background loop. The background loop contains Fault Detection, Application State Machine, and a scheduler routine.

The scheduler routine provides the timing sequence for two tasks called Timeout 1 and Timeout 2. The Timeout 1 and Timeout 2 flags are periodically set to predetermined intervals by the ADC Conversion Completed ISR. The scheduler utilizes these flags and calls the required routines:

- the routine in Timeout 1 provides a user interface, calculates the required speed, the start-up routines, and the speed ramp (acceleration/deceleration)
- the routine in Timeout 2 calculates the Speed Controller and Resistance Estimator

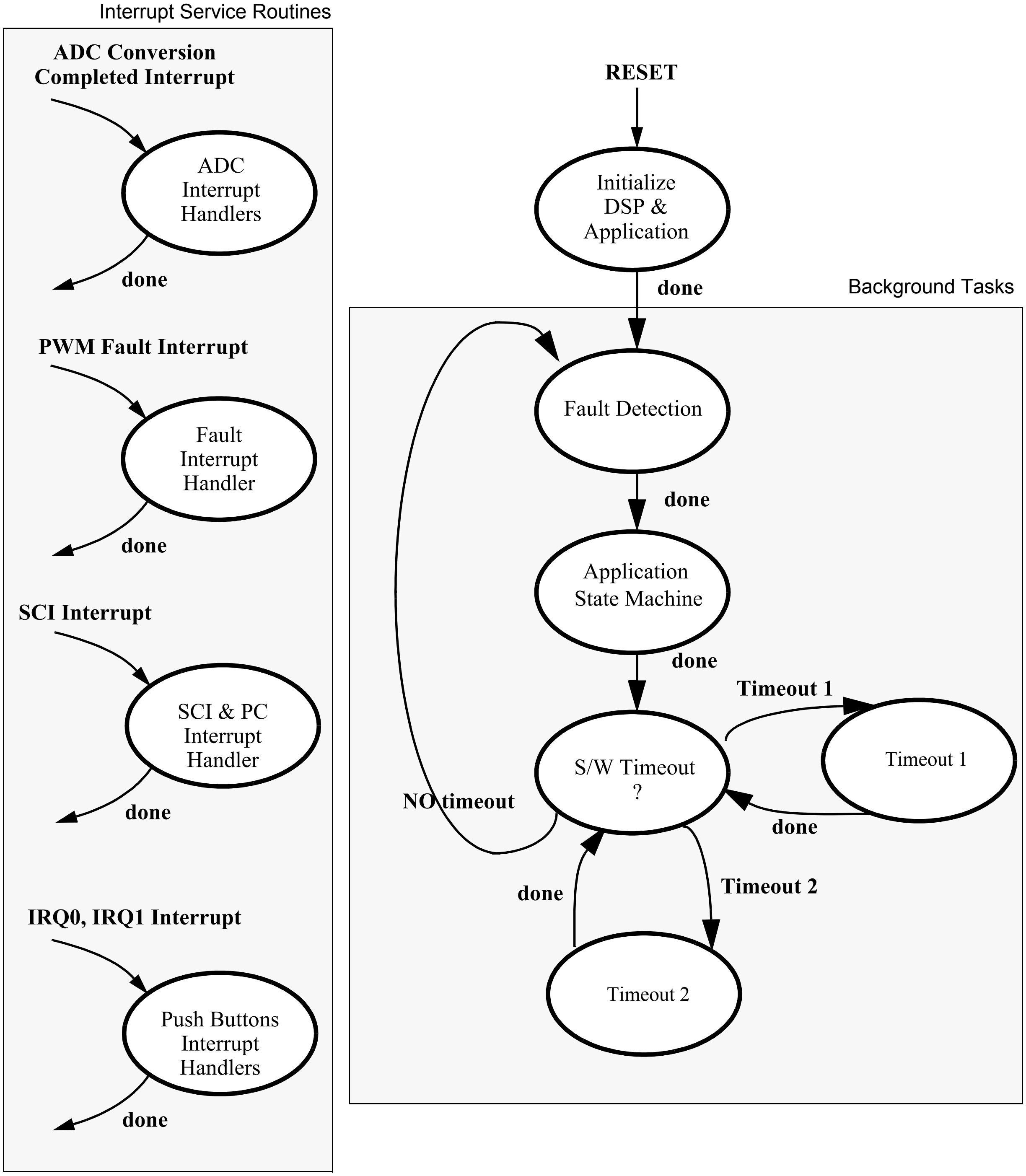


Figure 5-4. Software Design - General Overview

The Timeout 1 and Timeout 2 tasks are performed in the RUN state, instead of interrupt routines, in order to reduce time and avoid software bottlenecks.

The following interrupt service routines are utilized:

- ADC Conversion Completed ISR - services ADC and provides all the control tasks linked to the event; the ADC is synchronized with the PWM pulses.
- Fault ISR - services faults invoked by external hardware fault
- SCI ISR - services PC master software communication
- Push Button Up ISR - services Up Push Button
- Push Button Down ISR - services Down Push Button

5.2.3.1 Initialization

After Reset, the initialization of the DSP is performed. At the beginning of the initialization, interrupts are disabled; at the end of initialization they are enabled.

DSP initialization:

- Disable interrupts
- Identify power stage board
  - identify SR High-Voltage H/W set
- Initialize PWM on-chip module:
  - center-aligned independent PWM mode, positive polarity
  - set PWM modulus for PWM frequency 16kHz
  - set PWM interrupt reload each PWM pulse
  - set FAULT2 (DC-Bus over-current fault) in manual mode, interrupt enabled
  - set FAULT1 (DC-Bus over-voltage fault) in manual mode, interrupt enabled
  - associate interrupt with PWM Fault events

- Initialize ADC on-chip module
  - ADC triggered simultaneously
  - associate interrupt with ADC conversion completed event
  - 1st sample of ADC_A: Current Phase A
  - 2nd sample of ADC_A: DC-Bus Voltage
  - 3rd sample of ADC_A: Temperature
  - 1st sample of ADC_B: Current Phase B
  - 2nd sample of ADC_B: Current Phase C
  - 3rd sample of ADC_B: void
- Initialize Quad Timer A0 on-chip module (speed measurement)
  - count up
  - Prescaler 128
- Initialize Quad Timer A1 on-chip module (position reference for visualization using PC master software)
  - count Quadrature Decoder input
  - count repeatedly up to 255
- Initialize Quadrature Decoder on-chip module (position reference for PC master software)
  - set digital filter for input signals
  - connect Quadrature Decoder signals to the Quad TimerA1
- Initialize brake driver
- Initialize LED driver
- Initialize push buttons
  - push buttons on interrupts IRQ0, IRQ1
- Initialize switch driver
  - switch driver used for DSP56F805EVM

Application initialization:

- Set individual application parameters to their initial values

- Start ADC conversion
- Measure offset of individual current sensors
- Measure DC-Bus voltage and temperature
- Calculate application parameters according to DC-Bus voltage
- Initialize Quad Timer C2 driver (ADC-PWM Synchronization)
    - set ADC synchronization delay to 0
    - enable Quad Timer C2 to be started on first SYNC
- ADC driver initialization
    - set ADC synchronization to ON
    - enable 8-sample conversion
- Initialize all variables for motor start-up
- Set ADC according to start-up phase
- Enable interrupts

5.2.3.2 Fault Detection

The Fault Detection routinely checks for application faults. If a fault occurs, it disables the PWM outputs and sets the application FAULT status. Note that in the case of over-current and over-voltage faults, PWM outputs are disabled directly via internal PWM module fault protection (see **Section 5.2.3.7**).

5.2.3.3 Application State Machine

The Application State Machine provides transition between the individual states of the application: INIT, STOP, RUN, and FAULT. For reference, see **Section 5.2.2**.

5.2.3.4 Scheduler Timeout 1

This routine is accessed from the main scheduler at a period of Timeout 1 (10 msec). The following tasks are then performed:

- Push button filter - debounces push button switching noise

- Start/Stop switch filter - debounces Start/Stop switch noise
- According to operational mode, desired speed is calculated
  - in manual mode according to the push buttons
  - in PC master software control mode, according to the PC master command
- Start-up routine is performed if required and start-up switching pattern is generated. For a detailed description refer to **Section 3.3.2**.
- Speed command is calculated using the acceleration / deceleration ramp using the desired speed setup
- LED is controlled according to the state of the drive. It can indicate a STOP state, RUN state or FAULT state.

5.2.3.5 Scheduler Timeout 2

This state is accessed from the main scheduler in period of Timeout 2 (2.5 msec). The following tasks are then performed:

- Speed controller calculates the desired phase current according to the actual and the desired speed. The speed controller constants are determined experimentally and set during the initialization of the chip.
- Resistance estimator estimates the phase resistance according to the flux linkage estimation error

5.2.3.6 ADC Conversion Completed ISR

The ADC Conversion Completed ISR is the most critical and the routine most demanding of the processor's time. Most of the application control processes need to be linked with this ISR.

The Analog-to-Digital converter is initiated synchronously with a PWM reload pulse (center of the PWM pulse). It scans all three phase currents, the DC Bus voltage and the temperature at once. When the conversion is finalized, the ADC Completed ISR is called.

The routine provides the following services and calculations:

- Reads the time for speed calculation reference
- Reads the ADC conversion results (phase currents, DC-Bus voltage, temperature)
- Calculates the ADC offsets for the phase currents
- Calculates the reference and the actual flux linkage and determines commutation
- Current controller calculates the output duty cycle according to the desired and the actual phase currents
- Provides commutation when required
- Provides speed measurement
- Records selected recorder variables (PC master software)
- Loads PWM registers
- Calculates the references for software timers Timer1 and Timer2
- Enables the next ADC synchronization trigger

5.2.3.7 Fault ISR

The PWM Fault ISR is the highest priority interrupt implemented in the software. In the case of a DC-Bus over-current or a DC-Bus over-voltage fault detection, the external hardware circuit generates a fault signal, that is detected on the Fault input pin of the DSP. The signal disables the motor control PWM outputs in order to protect the power stage and generates a Fault interrupt, where the fault condition is handled. The routine records the corresponding fault source to the fault status register.

5.2.3.8 SCI ISR

This interrupt handler provides SCI communication and PC master software service routines. These routines are fully independent of the motor control tasks.

5.2.3.9 Push Button UP/Down ISR

The Push Button Interrupt Handlers take care of the push button service. The Up Button Interrupt Handler sets the Up Button flag, the Down Button Interrupt Handler sets the Down Button flag. The desired speed is incremented/decremented according to the debounced Up/Down button flag.

5.3 Implementation Notes

5.3.1 Scaling of Quantities

The SR motor control application uses a fractional representation for all real quantities except time. The N-bit signed fractional format is represented using 1.[N-1] format (1 sign bit, N-1 fractional bits). Signed fractional numbers (SF) lie in the following range:

$$-1.0 \le \mathrm{SF} \le +1.0 \text{ -}2^{-[N-1]} \qquad \textbf{(5-1)}$$

For words and long-word signed fractions, the most negative number that can be represented is -1.0, whose internal representation is $8000 and $80000000, respectively. The most positive word is $7FFF or 1.0 - 2^{-15}, and the most positive long-word is $7FFFFFFF or 1.0 - 2^{-31}.

The following equation shows the relationship between the real and the fractional representations:

$$\text{Fractional Value} = \frac{\text{Real Value}}{\text{Real quantity range}} \qquad \textbf{(5-2)}$$

where:

Fractional Value is the fractional representation of the real value [Frac16]

Real Value is the real value of the quantity [V, A, RPM, etc.]

Real quantity range is the maximal range of the quantity, defined in the application [V, A, RPM, etc.]

5.3.2 Voltage Scaling

The application voltages are scaled to the maximal measured voltage. For DC-Bus voltage the scaling equation is the following:

$$u\_dc\_bus = \frac{V_{DC\_BUS}}{V_{MAX}} \tag{5-3}$$

Where:

u_dc_bus is the scaled variable of the DC-Bus voltage [Frac16]

V_{DC_BUS} is the measured DC-Bus voltage [V]

V_{MAX} is the maximal measurable DC-Bus voltage [V]

In the application, V_{MAX} = 407V for the high voltage platform.

The other application voltage variables are scaled in the same way (active phase voltage `u_active`, discharge phase voltage `u_discharge`, DC-Bus under-voltage limit, start-up voltage).

5.3.3 Phase Resistance Scaling

There is no general way for scaling the resistance. In order to decrease the calculation requirements, in the application the phase resistance was scaled according to the scaling of the measured voltage and the phase current. For the actual phase resistance, the scaling equation is the following:

$$r\_phase\_actual = \frac{R_{phase\_actual}}{\frac{u_{MAX}}{i_{phase\_max}}} \tag{5-4}$$

Where:

r_phase_actual is the scaled variable of the actual phase resistance [Frac16]

R_{phase_actual} is the measured actual phase resistance [Ω]

u_{MAX} is the maximal measurable DC-Bus voltage [V]

$i_{phase-max}$ is the maximal measurable phase current [A]

In the application, $u_{MAX}/i_{phase\_max} = 407V/5.86A = 69.4\Omega$

The other application resistance variables are scaled the same way (resistance sample, `r_phase_sample`).

5.3.4 Phase Inductance Scaling

There is no general way for scaling the inductance. In order to decrease the calculation requirements, in the application the phase inductance was scaled according to the scaling of the measured voltage and the phase current. For unaligned phase inductance, the scaling equation is the following:

$$L\_unaligned = \frac{L_{unaligned}}{\frac{u_{MAX}}{i_{phase\_max}}} \tag{5-5}$$

Where:

$L\_unaligned$ is the scaled variable of the unaligned phase inductance [Frac16]

$L_{unaligned}$ is the unaligned phase inductance [H]

u_{MAX} is the max. measurable DC-Bus voltage [V]

$i_{phase-max}$ is the max. measurable phase current [A]

In the application, $u_{MAX} / i_{phase\_max} = 407V/5.86A = 69.4V/A$.

5.3.5 Flux Linkage Scaling

The application phase linkage is calculated as a flux linkage divided by a sampling period T **(2-6)**. The 16-bit phase flux increments (u_k-r_k*i_k) are summed to the 32-bit flux linkage sum variable (Ψ_N/T). The integration output can overflow if more than 65,536 samples are calculated. The sampling period T is defined by the PWM frequency of 16kHz. In the application: $T = 1/16000 = 62.5 \cdot 10^{-6}$sec.

The 32-bit flux linkage, `psi_T_active_sum`, is further scaled to the 16-bit variable `psi_T_active`.

$$psi\_T\_active = psi\_T\_active\_sum \cdot 256 \tag{5-6}$$

Where:

psi_T_active is the scaled variable of the active flux linkage [Frac16]

psi_T_active_sum is the scaled variable of the active flux linkage sum [Frac32].

The other application 16-bit flux linkage variables are scaled in the same way (flux linkage error, `psi_T_error`, reference flux linkage, `psi_T_reference`, delta flux linkage, `psi_T_delta`)

5.3.6 Electrical Angle Scaling

The application electrical angle is scaled to the electrical angle in the aligned position (see **Figure 5-5**). For the electrical commutation angle the scaling equation is the following:

$$theta\_commutation\_el = \frac{\vartheta_{commutation\_el}}{180^o} \tag{5-7}$$

Where:

theta_commutation_el is the scaled variable of the electrical commutation angle [Frac16]

$\vartheta_{commutation\_el}$ is the desired commutation angle [$^o{}_{el}$]

In the application, $\vartheta_{aligned\_el} = 180^o{}_{el}$

The other application electrical angle variables are scaled in the same way (delta theta required for phase current to reach the desired current, t `theta_delta_el`, theta where stator and rotor poles start to overlap, `theta_start_to_overlap_el`).

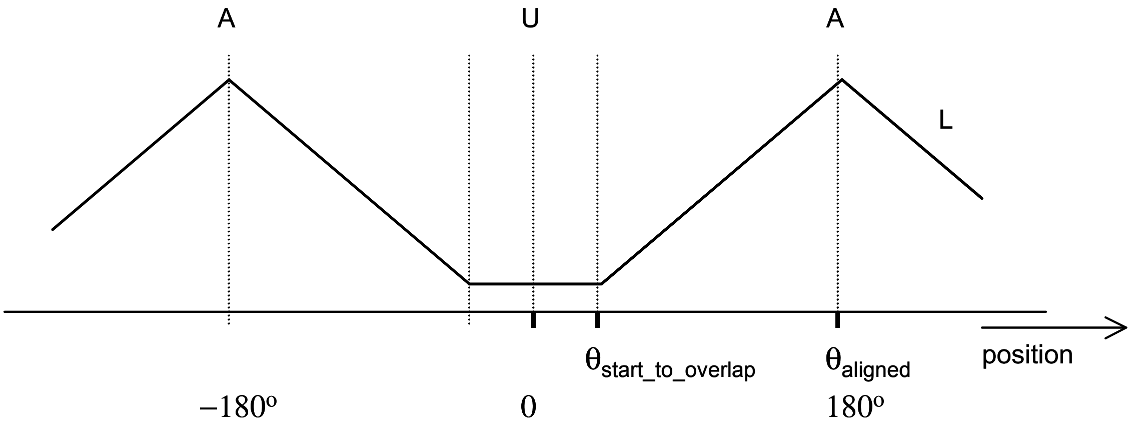


Figure 5-5. Electrical Angle Definition

5.3.7 Speed Scaling

Speed is scaled to the maximal speed of the drive. For the desired start-up speed, the scaling equation is the following:

$$omega\_desired\_startup = \frac{\omega_{start\_up}}{\omega_{MAX}} \quad \textbf{(5-8)}$$

Where:

omega_desired_startup is the scaled variable of the desired start-up speed [Frac16]

$\omega_{start\text{-}up}$ is the desired start-up speed [RPM]

ω_{MAX} = maximal speed of the drive [RPM]

In the application, ω_{MAX} = 3000 RPM.

The other application speed variables are scaled in the same way (actual speed, `omega_actual_mech`, speed limits, `omega_reqMAX_mech & omega_reqMIN_mech`, push button speed increment, `omega_increment_pb`).

5.3.8 Duty Cycle Scaling

The duty cycle is scaled to the maximal duty cycle of the drive. For the output duty cycle, the scaling equation is the following:

$$output\_duty\_cycle = \frac{duty\_cycle_{output}}{duty\_cycle_{MAX}} \tag{5-9}$$

Where:

output_duty_cycle is the scaled variable of output duty cycle [Frac16]

$duty\_cucle_{output}$ is the desired output duty cycle [%]

$duty\_cycle_{MAX}$ is the max. applicable duty cycle [%]

In the application, $duty\_cycle_{MAX} = 100\%$

The other application duty cycles are scaled in the same way (high and low duty cycle limits for speed controller, start up output duty cycle `outputDutyCycleStartup`).

5.3.9 Velocity Calculation

The actual speed of the motor is calculated from the time, *TimeCaptured,* captured by the on-chip Quad Timer between the two following edges of the position Hall sensors. The actual speed, *OmegaActual* is calculated according to the following equation:

$$OmegaActual = \frac{SpeedCalcConst}{TimeCaptured} \tag{5-10}$$

where:

OmegaActual is the actual speed [RPM]

TimeCaptured is the time, in terms of number of timer pulses, captured between two edges of the position sensor [-]

SpeedCalcConst is a constant defining the relationship between the actual speed and number of captured pulses between the two edges of the position sensor

The constant *SpeedCalcConst* is calculated as:

$$SpeedCalcConst = 2^{15} \times \frac{SpeedMin}{SpeedMax} \tag{5-11}$$

where:

SpeedMin is the minimal measured speed [RPM]

SpeedMax is the maximal measured speed [RPM]

Minimal measured speed, *SpeedMin,* is given by the configuration of the sensors and parameters of the DSP on-chip timer used for speed measurement. It is calculated as:

$$SpeedMin = \frac{\frac{1}{NoPulsesPerRev} \times 60}{\frac{2^{15}}{BusClockFreq} \times Presc} \tag{5-12}$$

where:

NoPulsesPerRev is the number of sensed pulses of the position sensor per single revolution [-]

Presc is the prescaler of the Quad Timer used for speed measurements

BusClockFreq is the DSP Bus Clock Frequency [Hz]

Maximal measured speed, *SpeedMax,* is selected as:

$$SpeedMax = k \times SpeedMin \tag{5-13}$$

where:

k is an integer constant greater than 1

Then the speed calculation constant is determined as:

$$SpeedCalcConst = BusClockFreq \times \frac{60}{NoPulsesPerRev \times Presc \times SpeedMax} \tag{5-14}$$

In the application:

NoPulsesPerRev = 12 Hall sensor pulses per 1 revolution of the motor

Presc = 128

$BusClockFreq = 36*10^6$ Hz

$SpeedMax = 3000$ RPM

Then, $SpeedCalcConst = 468$ [rev^{-1}]

Section 6. Application Setup

6.1 Contents

6.2 Application Description

The 3-Phase SR Sensorless Motor Control Application demonstrates the sensorless Switched Reluctance Motor Control application using the flux linkage position estimation on the DSP56F805 processor. An estimation of the phase resistance for low speed range is included.

Control Process

After RESET, the drive enters the INIT state in manual mode.When the RUN/STOP switch is detected in the STOP position (using the RUN/STOP Switch or the PC master software command) and there are no faults pending, the STOP application state is entered. When the start command is detected (using the RUN/STOP switch or the PC master software Start button), the drive enters the RUN application state; the motor is started. The following start-up sequence with the rotor alignment is provided:

- MOTOR_STOPPED Motor stopped

- ALIGNMENT_COMMAND Alignment command accepted
- ALIGNMENT_STAGE_ONE Alignment in progress; phases B and C switched on
- ALIGNMENT_STAGE_TWO Alignment in progress; phase B switched on
- START_UP_COMMAND Alignment finalized; start the motor
- START_UP_FINISHEDMotor running; start-up finalized

The rotor position is evaluated using the sensorless flux linkage estimation algorithm. The actual flux linkage is calculated using the PWM frequency rate and compared with the reference flux linkage for the given commutation angle. The commutation angle is calculated according to the desired speed, current and actual DC Bus voltage. When the actual flux linkage exceeds the reference flux linkage, the commutation of the phases in the desired rotation direction is done. The flux linkage error is used for the phase resistance estimation in low speeds (US Patent Pending). The commutation instances are used for actual motor speed calculation. According to the control signals (RUN/STOP switch, UP/DOWN push buttons) and PC master software commands (during PC master software control), the reference speed command is calculated using an acceleration/deceleration ramp. The comparison between the actual speed command and the measured speed generates a speed error. Based on the error, the speed controller generates the desired phase current. When the phase is commuted, it is turned on with duty cycle 100 percent (or Output_duty_cycle_startup during motor start-up). Then during each PWM cycle, the actual phase current is compared with the desired current. As soon as the actual current exceeds the command current, the current controller is turned on. The procedure is repeated for each commutation cycle of the motor. The current controller generates the desired duty cycle. Finally, the 3-phase PWM SR Motor Control signals are generated.

Drive Protection

The DC Bus voltage, DC Bus current and power stage temperature are measured during the control process. They protect the drive from Overvoltage, Undervoltage, Overcurrent and Overheating. The Undervoltage and Overheating protection is performed by software, while the Overcurrent and Overvoltage fault signal utilizes a fault input of the DSP. The power stage is identified using board identification. If the correct power stage is not identified, the fault "Wrong Power Stage" disables the drive operation. Line voltage is measured during application initialization and the application automatically adjusts itself to run at either 115 V AC or 230 V AC, depending on the measured value. If the line voltage is detected to be out of the -15% to +10% of nominal voltage, the fault "Out of the Mains Limit" disables the drive operation.

If any of the above-mentioned faults occur, the motor control PWM outputs are disabled in order to protect the drive and the application enters the FAULT state. The FAULT state can be left only when the fault conditions disappear and the RUN/STOP switch is moved to the STOP position

The application can run on:

- External RAM or Flash
- 3-Phase SR High-Voltage Power Stage powered by 115V AC or 230V AC
- Manual or PC Master Operating Mode

The correct power stage and voltage level is identified automatically and the appropriate constants are set.

The 3-phase SR motor control application can operate in two modes:

1. Manual Operating Mode
   The drive is controlled by the RUN/STOP switch. The motor speed is set by the UP and DOWN push buttons (see **Figure 6-1**). The actual state of the application is indicated by the user LEDs (see **Figure 6-2**). If the application runs and motor spinning is disabled (i.e., the system is ready), the GREEN user LED will flash at a frequency of 2Hz. When motor spinning is enabled, the GREEN user LED will be *On*. If a fault occurs on the power stage, the GREEN user LED will flash at a frequency of 8Hz. The actual state of the PWM outputs are indicated by PWM output LEDs.

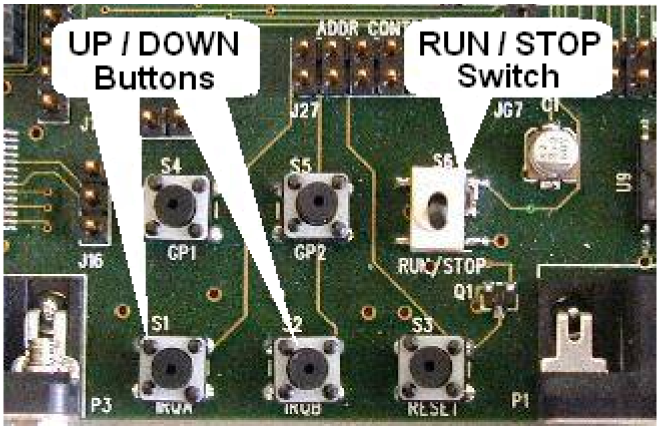


Figure 6-1. RUN/STOP Switch and UP/DOWN Buttons

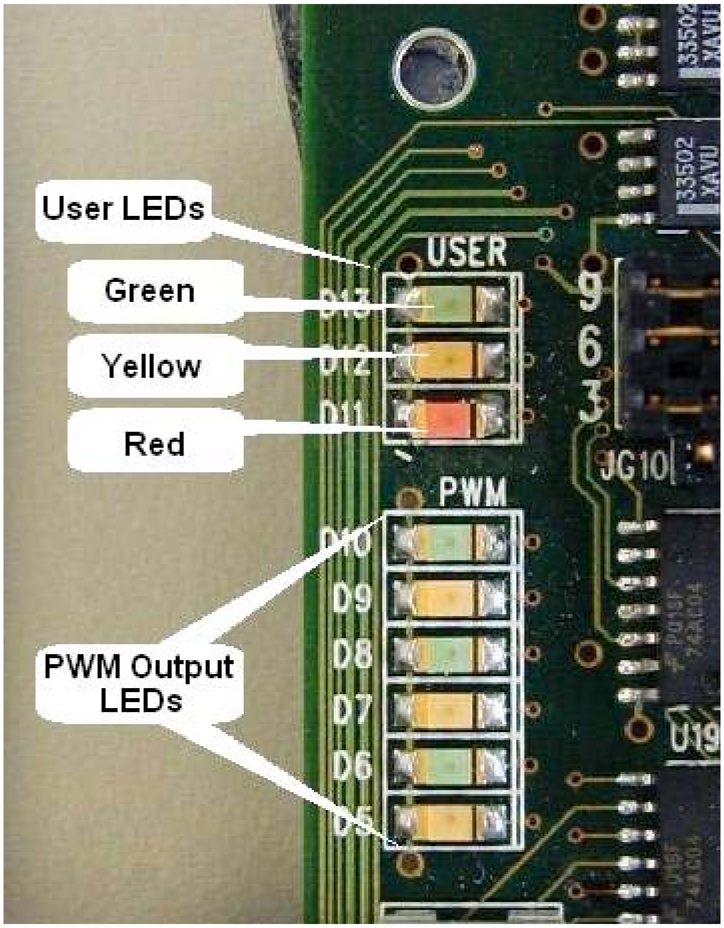


Figure 6-2. USER and PWM LEDs at DSP56F805EVM

Table 6-1. Motor Application States

Application State	Motor State	Green LED State
Stopped	Stopped	Blinking at a frequency of 2Hz
Running	Spinning	On
Fault	Stopped	Blinking at a frequency of 8Hz

2. PC master software (Remote) Operating Mode
   The drive is controlled remotely from a PC through the SCI communication channel of the DSP device via an RS-232 physical interface. The drive is enabled by the RUN/STOP switch, which can be used to safely stop the application at any time. PC master software enables to set the required speed of the motor.

The following PC Master control actions are supported:

- Set PC Master Mode of the motor control system
- Set Manual Mode of the motor control system
- Start the motor
- Stop the motor
- Set the Required Speed of the motor

PC Master displays the following information:

- Required Speed of the motor
- Actual Speed of the motor
- Application status - Init/Stop/Run/Fault
- DC Bus voltage level
- Identified line voltage
- Fault Status - No_Fault/Overvoltage/Overcurrent/Undervoltage/Overheating
- Identified Power Stage

Speed Scope monitors:

- Required Speed
- Actual Speed
- Desired Phase Current

Start-up Recorder captures:

- Desired Phase Current
- Active Phase Current
- Reference Flux Linkage
- Active Flux Linkage
- Output Duty Cycle
- Encoder Position Reference

Start-up Recorder is initiated with motor start only.

Flux Linkage Recorder captures:

- Active Phase Current
- Discharge Phase Current
- Active Flux Linkage
- Discharge Flux Linkage
- Reference Flux Linkage
- Encoder Position Reference

Flux Linkage Recorder may be initiated any time during the motor run.

Current Controller Recorder captures:

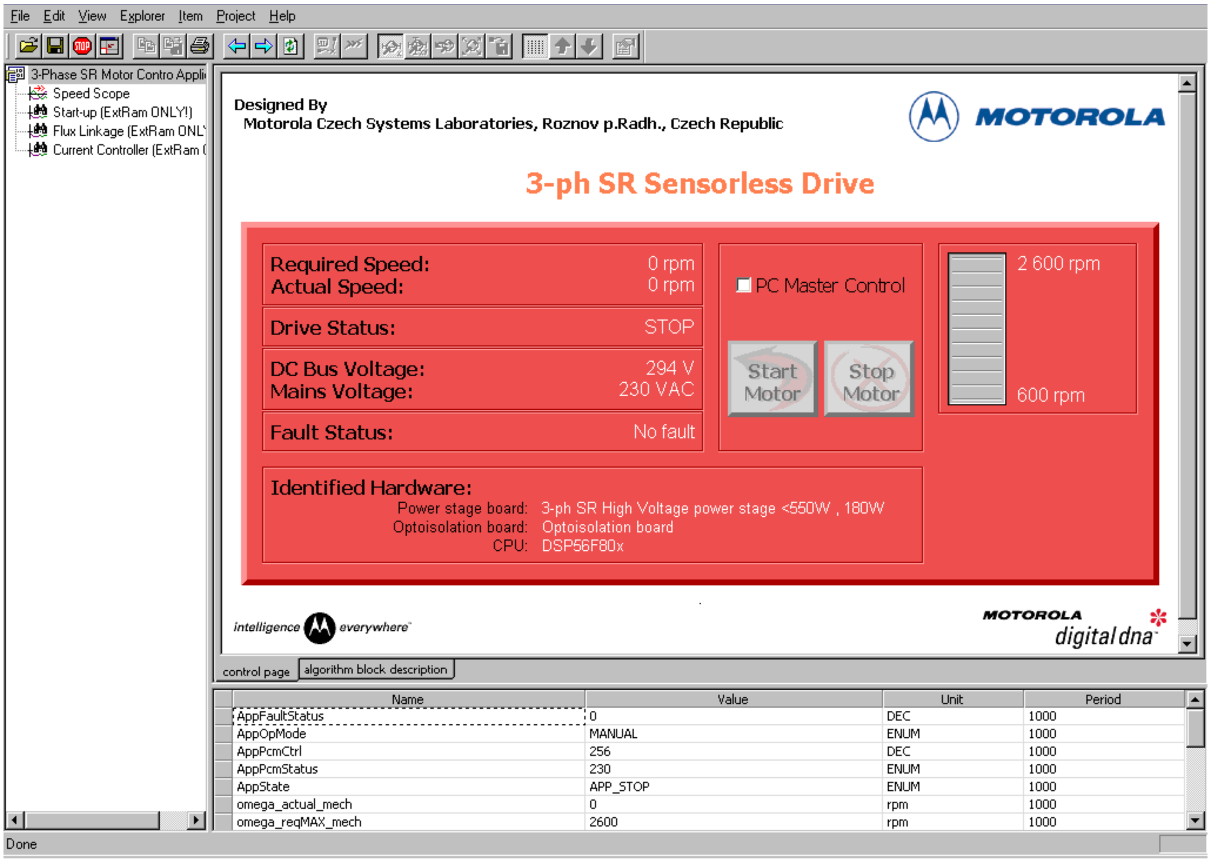
- Desired Phase Current
- Active Phase Current
- Output Duty Cycle
- Encoder Position Reference

Current Controller Recorder may be initiated any time during the motor run.

The recorder can be used **only** when the application is running from **External RAM** due to limited on-chip memory. The length of the recorded window may be set in “Recorder Properties” => bookmark “Main” => “Recorded Samples”. It is limited by the dedicated memory space in the *appconfig.h* file. The recorder samples are taken every 64.5 μsec.

Start the PC master software window’s application, *3ph_srm_sensorless_sa.pmp*. **Figure 6-3** illustrates the PC master software control window after this project has been launched.

NOTE: *If the PC master software project (.pmp file) is unable to control the application, it is possible that the wrong load map (.elf file) has been selected. PC master software uses the load map to determine addresses for global variables being monitored. Once the PC master software project has been launched, this option may be selected in the PC master software window under Project/Select Other Map FileReload.*



Designed By
Motorola Czech Systems Laboratories, Roznov p.Radh., Czech Republic

3-ph SR Sensorless Drive

Required Speed: 0 rpm
Actual Speed: 0 rpm
Drive Status: STOP
DC Bus Voltage: 294 V
Mains Voltage: 230 VAC
Fault Status: No fault

☐ PC Master Control

2 600 rpm
600 rpm

Identified Hardware:
Power stage board: 3-ph SR High Voltage power stage <550W , 180W
Optoisolation board: Optoisolation board
CPU: DSP56F80x

Name	Value	Unit	Period
AppFaultStatus	0	DEC	1000
AppOpMode	MANUAL	ENUM	1000
AppPcmCtrl	256	DEC	1000
AppPcmStatus	230	ENUM	1000
AppState	APP_STOP	ENUM	1000
omega_actual_mech	0	rpm	1000
omega_reqMAX_mech	2600	rpm	1000

Figure 6-3. PC Master Software Control Window

6.3 Application Set-Up

Figure 6-4 illustrates the hardware set-ups for the 3-phase SR Motor Control applications. The motor’s Encoder connector attached to connector J23 on the EVM Board is not required for the motor operation. It serves only for PC Master position reference.

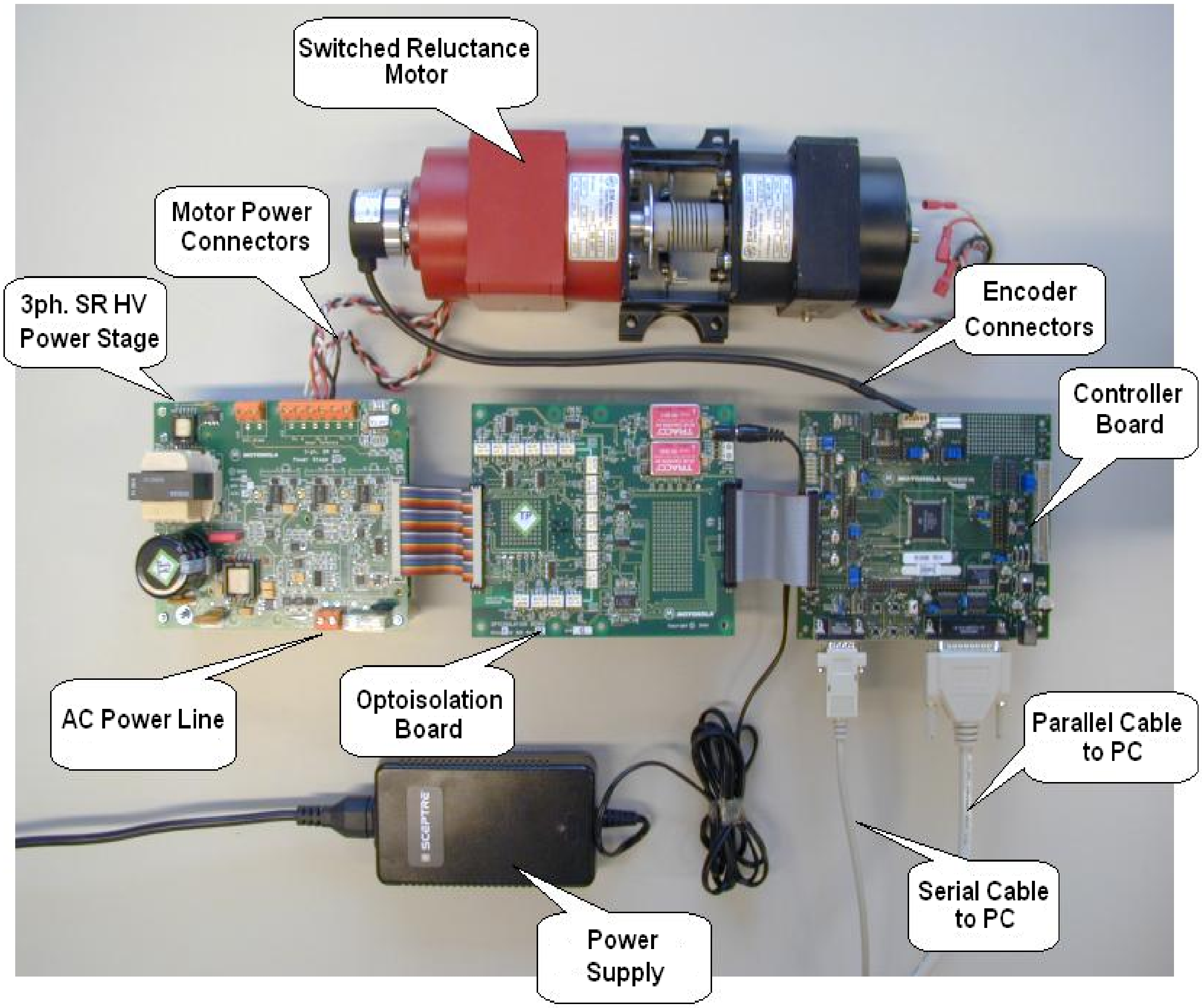


Figure 6-4. Set-up of the 3-Phase SR Motor Control Application

The system consists of the following components:

- Switched reluctance motor Type 40 V, EM Brno s.r.o., Czech Republic
- Load Type SG 40N, EM Brno s.r.o., Czech Republic
- Encoder BHK 16.05A1024-12-5, Baumer Electric, Switzerland
- 3-ph. SR HV Power Stage 180 W:
  - supplied as ECINLHIVSR

- Optoisolation Board
  - ECOPT
- DSP56F805 Evaluation Module, supplied as DSP56F805EVM
- The serial cable - needed for the PC master software debugging tool only.
- The parallel cable - needed for the Metrowerks Code Warrior debugging and s/w loading.

The correct order of phases (phase A, phase B, phase C) for the SR motor is:

- phase A = white wire
- phase B = red wire
- phase C = black wire

When facing a motor shaft, the motor shaft should rotate clockwise (i.e., positive direction, positive speed).

For detailed information, refer to the dedicated application note (see References).

6.3.1 DSP56F805EVM Set-Up

To execute the 3-Phase SR Sensorless Motor Control, the DSP56F805EVM board requires the strap settings shown in **Figure 6-5** and **Table 6-2**.

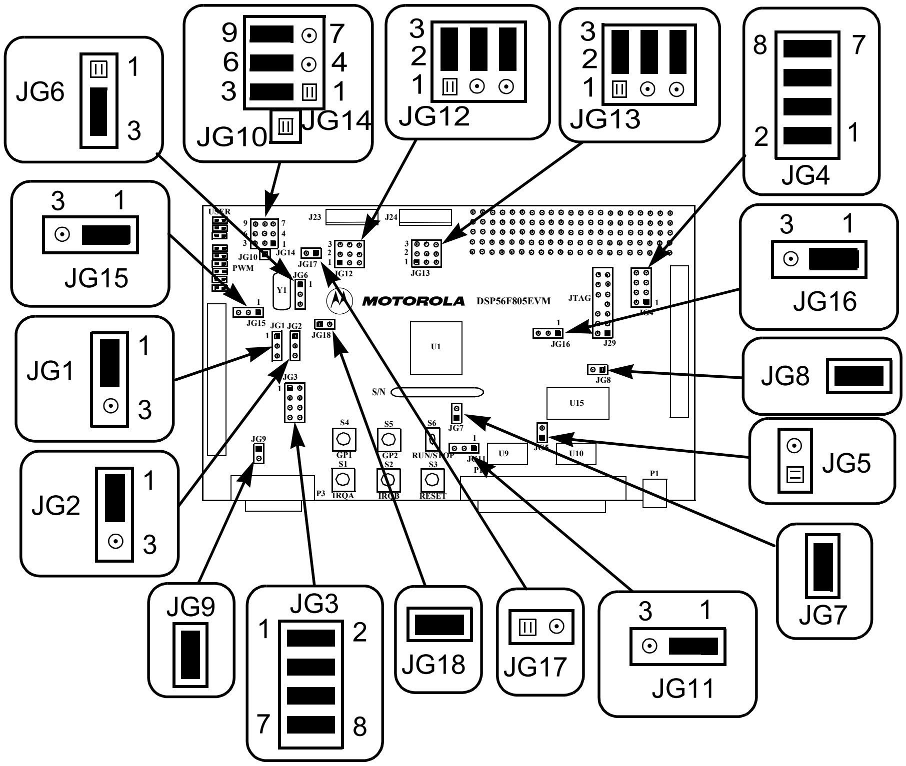


Figure 6-5. DSP56F805EVM Jumper Reference

Table 6-2. DSP56F805EVM Jumper Settings

Jumper Group	Comment	Connections
JG1	PD0 input selected as a high	1-2
JG2	PD1 input selected as a high	1-2
JG3	Primary UNI-3 serial selected	1-2, 3-4, 5-6, 7-8
JG4	Secondary UNI-3 serial selected	1-2, 3-4, 5-6, 7-8
JG5	Enable on-board parallel JTAG Command Converter Interface	NC
JG6	Use on-board crystal for DSP oscillator input	2-3
JG7	Select DSP’s Mode 0 operation upon exit from reset	1-2
JG8	Enable on-board SRAM	1-2
JG9	Enable RS-232 output	1-2

Table 6-2. DSP56F805EVM Jumper Settings

Jumper Group	Comment	Connections
JG10	Secondary UNI-3 Analog temperature input unused	NC
JG11	Use Host power for Host target interface	1-2
JG12	Primary Encoder input selected for quadrature encoder signals	2-3, 5-6, 8-9
JG13	Secondary Encoder input selected	2-3, 5-6, 8-9
JG14	Primary UNI-3 3-Phase Current Sense selected as Analog Inputs	2-3, 5-6, 8-9
JG15	Secondary UNI-3 Phase A Overcurrent selected for FAULTA1	1-2
JG16	Secondary UNI-3 Phase B Overcurrent selected for FAULTB1	1-2
JG17	CAN termination unselected	NC
JG18	Use on-board crystal for DSP oscillator input	1-2

NOTE: *When running the EVM target system in a stand-alone mode from Flash, the JG5 jumper must be set in the 1-2 configuration to disable the command converter parallel port interface.*

6.4 Projects Files

The 3-Phase SR Sensorless Motor Control application is composed of the following files:

- **...\3ph_srm_sensorless_sa\3ph_srm_sensorless.c,** main program
- **...\3ph_srm_sensorless_sa\3ph_srm_sensorless_sa.mcp,** application project file
- **...\3ph_srm_sensorless_sa\ApplicationConfig\appconfig.h,** application configuration file
- **...\3ph_srm_sensorless_sa\SystemConfig\ExtRam\linker_ram.cmd,** linker command file for external RAM
- **...\3ph_srm_sensorless_sa\SystemConfig\Flash\linker_flash.cmd,** linker command file for Flash

- **...\3ph_srm_sensorless_sa\SystemConfig\Flash\flash.cfg,** configuration file for Flash
- **...\3ph_srm_sensorless_sa\PCMaster\3ph_srm_sensorless.pmp**, PC master software file

These files are located in the application folder.

Motor Control algorithms used in the application:

- **...\controller.c, .h**: source and header files for PI controller
- **...\ramp.c, .h**: source and header files for ramp generation
- **...\SrmCmt3Ph2spp.c, .h**: source and header files for SR Motor commutation algorithm
- **...\srmcac.c, .h**: source and header files for the mechanical and the electrical quantities calculation algorithms

Other functions used in the application:

- **...\boardId.c, .h**: source and header files for the board identification function

This application runs stand-alone, i.e. all the needed files are concentrated in one project folder. Quick_Start libraries are:

- **...\3ph_srm_sensorless_sa\src\include**, folder for general C-header files
- **...\3ph_srm_sensorless_sa\src\dsp56805**, folder for the device specific source files, e.g. drivers
- **...\3ph_srm_sensorless_sa\src\pc_master_support**, folder for PC master software source files
- **...\3ph_srm_sensorless_sa\src\algorithms**, folder for algorithms
- **...\3ph_srm_sensorless_sa\src\bsp**, folder for the board identification function source file

6.5 Application Build & Execute

When building the 3-Phase SR Sensorless Motor Control Application, the user can create an application that runs from internal *Flash* or *External RAM*. To select the type of application to build, open the *3srm_hall_sa.mcp* project and select the target build type, as shown in **Figure 6-6**. A definition of the projects associated with these target build types may be viewed under the *Targets* tab of the project window.

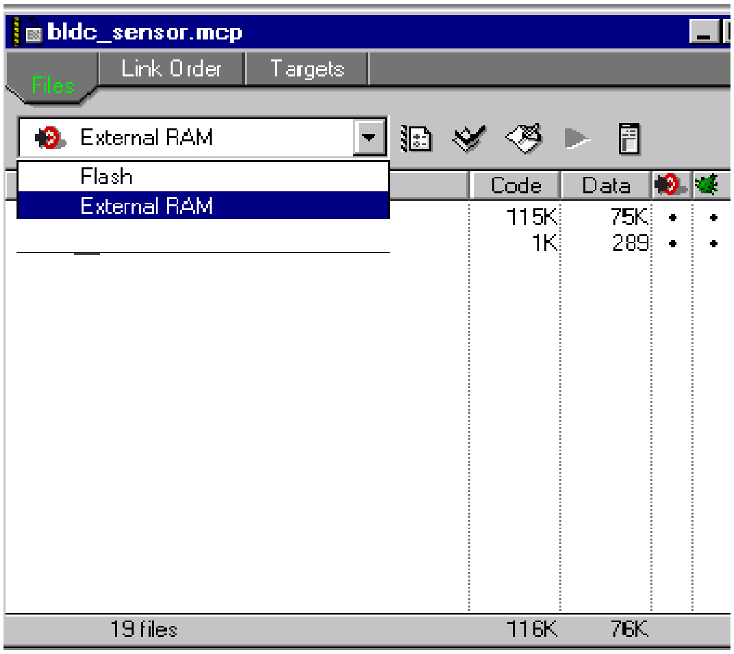


Figure 6-6. Target Build Selection

The project may now be built by executing the *Make* command, as shown in **Figure 6-7**. This will build and link the 3-Phase SR Sensorless Motor Control Application and all needed Metrowerks and Quick_Start libraries.

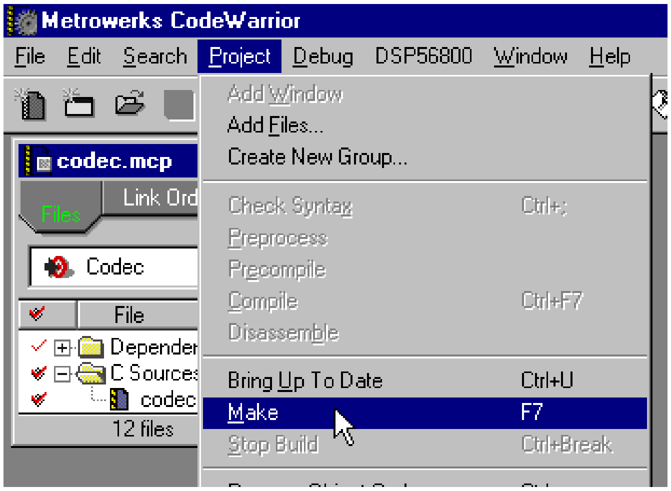


Figure 6-7. Execute *Make* Command

To execute the 3-Phase SR Sensorless Motor Control application, select *Project\Debug* in the CodeWarrior IDE, followed by the *Run* command. For more help with these commands, refer to the CodeWarrior tutorial documentation in the following file located in the CodeWarrior installation folder:

<...>\CodeWarrior Documentation\PDF\Targeting_DSP56800.pdf

If the Flash target is selected, CodeWarrior will automatically program the internal Flash of the DSP with the executable generated during *Build*. If the External RAM target is selected, the executable will be loaded to off-chip RAM.

Once Flash has been programmed with the executable, the EVM target system may be run in a stand-alone mode from Flash. To do this, set the JG5 jumper in the 1-2 configuration to disable the parallel port, and press the RESET button.

Once the application is running, move the RUN/STOP switch to the RUN position and set the required speed using the UP/DOWN push buttons. Pressing the UP/DOWN buttons should incrementally increase the motor speed until it reaches maximum speed. If successful, the SR motor will be spinning.

NOTE: *If the RUN/STOP switch is set to the RUN position when the application starts, toggle the RUN/STOP switch between the STOP and RUN positions to enable motor spinning. This is a protection feature that prevents the motor from starting when the application is executed from CodeWarrior.*

You should also see a lighted green LED, which indicates that the application is running. If the application is stopped, the green LED will blink at a 2Hz frequency. If an Undervoltage fault occurs, the green LED will blink at a frequency of 8Hz.

Appendix A. References

1. Chalupa, L., *Pohon se spinanym reluktancnim motorem*, Master's Thesis, FEI-VUT BRNO, UPVE, 1994
2. Chalupa, L., Visinka, R., On-Fly Phase Resistance Estimation of Switched Reluctance Motor for Sensorless based Control Techniques, *Conference Power Conversion and Intelligent Motion*, Nurnberg, PCIM, 2000
3. Motorola, *Apparatus and Method for Estimating the Coil Resistance in an Electric Motor*, Chalupa, L., Visinka, R., *US Patent, 6,366,865*, 2002-04-02
4. Gallegos-Lopez, G., A New Sensorless Low-Cost Method for Switched Reluctance Motor Drives, *University of Glasgow - SPEED Laboratory*, 1997
5. Miller, T.J.E., *Switched Reluctance Motors and Their Control*, Magna Physics Publishing and Clarendon Press, ISBN 0-19-859387-2, 1993
6. Lyons, J.P., MacMinn, S.R., Preston, Flux/Current Methods for SRM Rotor Position Estimation, *Proc. IEEE-IAS'91*, 1991
7. 3-Phase SR Motor Control with Hall Sensors using DSP56F80x, *AN1912/D*, Motorola, 2000
8. CodeWarrior for Motorola DSP56800 Embedded Systems, CWDSP56800, Metrowerks, 2001
9. DSP56F800 16-bit Digital Signal Processor, Family Manual, DSP56F800FM/D, Motorola, 2001
10. DSP56F80x 16-bit Digital Signal Processor, User's Manual, DSP56F801-7UM/D, Motorola, 2001
11. DSP56F805 Evaluation Module Hardware User's Manual, DSP56F805EVMUM/D, Motorola, 2001

12. Embedded Software Development Kit for 56800/56800E, MSW3SDK000AA, available on Motorola SPS web page, Motorola, 2001

13. Motorola Embedded Motion Optoisolation Board User's Manual, MEMCOBUM/D, Motorola, 2000

14. Motorola Embedded Motion Control 3-Phase Switched Reluctance High-Voltage Power Stage User's Manual, MEMC3PSRHVPSUM/D, Motorola, 2000

15. Motorola Embedded Motion Control 3-Phase Switched Reluctance Low-Voltage Power Stage User's Manual, MEMC3PSRLVPSUM/D, Motorola, 2000

16. **CodeWarrior for Motorola DSP56800 Embedded Systems**, CWDSP56800, Metrowerks 2001

17. **DSP56F805 Evaluation Module Hardware User's Manual**, DSP56F805EVMUM/D, Motorola 2001

18. **Motorola Embedded Motion Optoisolation Board User's Manual**, MEMCOBUM/D, Motorola 2000

19. **User Manual** for PC master software, Motorola 2001

20. **Motorola SPS web page**: *http://www.motorola.com/*

21. **DSP56800_Quick_Start User's Manual**, MCSL 2002

22. **Motor Control Algorithms Description**, MCSL 2002

Appendix B. Glossary

AC — Alternating Current.

ADC — See “analogue-to-digital converter”.

brush — A component transfering elektrical power from non-rotational terminals, mounted on the stator, to the rotor

BLDC — Brushless dc motor.

commutation — A process providing the creation of a rotation field by switching of power transistor (electronic replacement of brush and commutator)

commutator — A mechanical device alternating DC current in DC commutator motor and providing rotation of DC commutator motor

COP — Computer Operating Properly timer

DC — Direct Current.

DSP — Digital Signal Prosessor.

DSP56F80x — A Motorola family of 16-bit DSPs dedicated for motor control.

DT — see “Dead Time (DT)”

Dead Time (DT) — short time that must be inserted between the turning off of one transistor in the inverter half bridge and turning on of the complementary transistor due to the limited switching speed of the transistors.

duty cycle — A ratio of the amount of time the signal is on versus the time it is off. Duty cycle is usually represented by a percentage.

GPIO — General Purpose Input/Output.

Hall Sensors - A position sensor giving six defined events (each 60 electrical degrees) per electrical revolution (for 3-phase motor)

HV — High Voltage (115 V AC or 230 V AC)

interrupt — A temporary break in the sequential execution of a program to respond to signals from peripheral devices by executing a subroutine.

input/output (I/O) — Input/output interfaces between a computer system and the external world. A CPU reads an input to sense the level of an external signal and writes to an output to change the level on an external signal.

JTAG — Interface allowing On-Chip Emulation and Programming.

LED — Light Emitting Diode

logic 1 — A voltage level approximately equal to the input power voltage (V_{DD}).

logic 0 — A voltage level approximately equal to the ground voltage (V_{SS}).

LV — Low Voltage (12 V DC)

PI controller — Proportional-Integral controller.

phase-locked loop (PLL) — A clock generator circuit in which a voltage controlled oscillator produces an oscillation which is synchronized to a reference signal.

PM — Permanent Magnet

PMSM - Permanent Magnet Synchronous Motor.

PWM — Pulse Width Modulation.

Quadrature Decoder — A module providing decoding of position from a quadrature encoder mounted on a motor shaft.

Quad Timer — A module with four 16-bit timers.

reset — To force a device to a known condition.

RPM — Revolutions per minute.

SCI — See "serial communication interface module (SCI)."

serial communications interface module (SCI) — A module that supports asynchronous communication.

serial peripheral interface module (SPI) — A module that supports synchronous communication.

software — Instructions and data that control the operation of a microcontroller.

software interrupt (SWI) — An instruction that causes an interrupt and its associated vector fetch.

SPI — See "serial peripheral interface module (SPI)."

timer — A module used to relate events in a system to a point in time.

HOW TO REACH US:

USA/EUROPE/LOCATIONS NOT LISTED:

Motorola Literature Distribution;
P.O. Box 5405, Denver, Colorado 80217
1-303-675-2140 or 1-800-441-2447

JAPAN:

Motorola Japan Ltd.; SPS, Technical Information Center,
3-20-1, Minami-Azabu Minato-ku, Tokyo 106-8573 Japan
81-3-3440-3569

ASIA/PACIFIC:

Motorola Semiconductors H.K. Ltd.;
Silicon Harbour Centre, 2 Dai King Street,
Tai Po Industrial Estate, Tai Po, N.T., Hong Kong
852-26668334

TECHNICAL INFORMATION CENTER:

1-800-521-6274

HOME PAGE:

http://motorola.com/semiconductors

Information in this document is provided solely to enable system and software implementers to use Motorola products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part.

MOTOROLA

Motorola and the Stylized M Logo are registered in the U.S. Patent and Trademark Office. digital dna is a trademark of Motorola, Inc. All other product or service names are the property of their respective owners. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

© Motorola, Inc. 2003

Freescale Semiconductor, Inc.