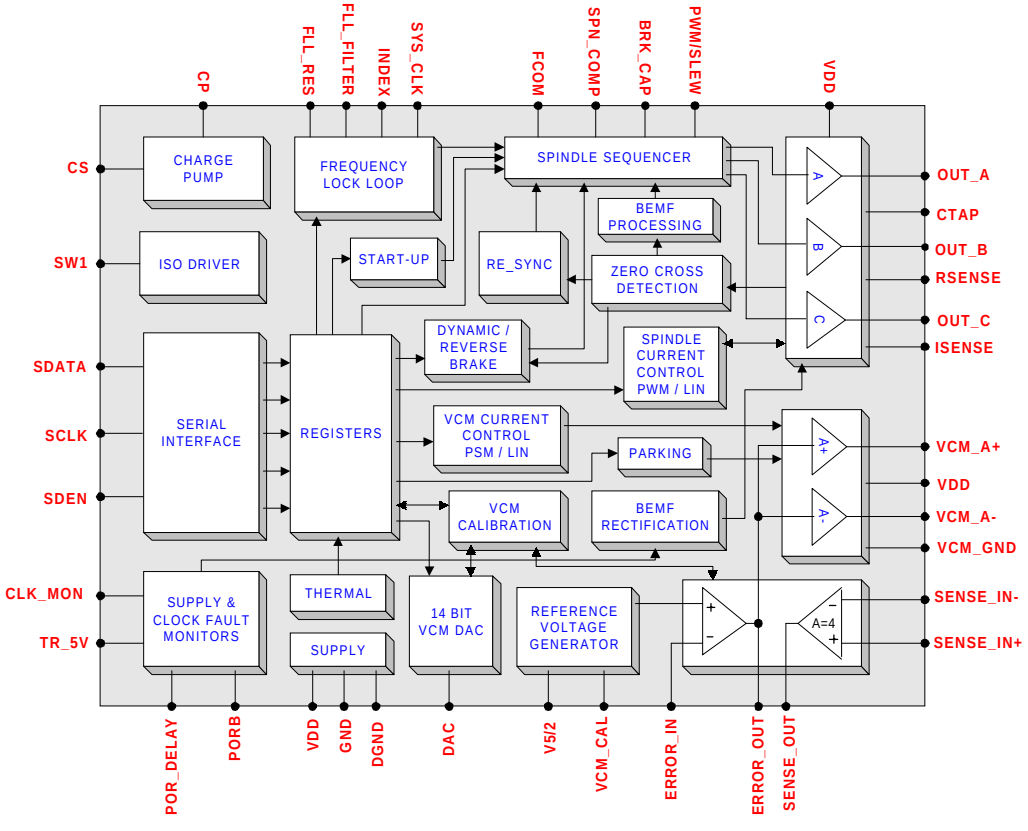
AN1138
APPLICATION NOTE

L6275
5V DISK DRIVE POWER COMBO IC

by Carlo Vertemara

The L6275 integrate into a single chip both SPINDLE and VCM controllers as well as power stages. The device is designed for 5V disk drive application requiring up to 2.0A of Spindle and 1.5 of Vcm peak current. A serial port with up to 25MHz capability provides easy interface to the microprocessor. A register controlled Frequency Locked Loop (FLL) allows flexibility in setting the Spindle Speed. Integrated BEMF processing, digital masking, digital commutation delay, and sequencing minimize the number of external components required. Power On Reset (POR) circuitry is included. Upon detection of a low voltage condition, POR is asserted, the internal registers are reset, and Spindle power circuitry is tri-stated. The BEMF is rectified providing power for the actuator retraction followed by dynamic Spindle braking. The device is built in BICMOS technology allowing dense digital / analog circuitry to be combined with a high power DMOS output stage.The serial interface lines SDEN, SCLK and SDATA are 3.3V compatible.



INDEX OF CHAPTERS

INDEX OF FIGURES

INDEX OF TABLES & EQUATIONS

1.0 FEATURES

- GENERAL

- 5V (+/- 10%) OPERATION.
- REGISTER BASED ARCHITECTURE
- MINIMUM EXTERNAL COMPONENTS
- BICMOS + VERTICAL DMOS TECHNOLOGY (1.5µm)

- VCM DRIVER

- 1.5A DRIVE CAPABILITY
- 0.9Ω TOTAL BRIDGE IMPEDANCE AT 25°C
- LINEAR MODE
- PHASE SHIFT MODULATION (PSM MODE)
- INSTANTANEOUS, (GLITCH FREE) SWITCH BETWEEN THE TWO MODES.
- CLASS AB OUTPUT DRIVERS
- ZERO CROSSOVER DISTORTION
- 14 BIT DAC DEFINE OUTPUT CURRENT
- SELECTABLE TRANSCONDUCTANCE
- 4 PROGRAMMABLE RETRACT VOLTAGE
- DYNAMIC BRAKE

- SPINDLE DRIVER

- 2.0A DRIVE CAPABILITY
- 0.8Ω TOTAL BRIDGE IMPEDANCE AT 25°C
- BEMF, INTERNAL / EXTERNAL, PROCESSING, SENSOR-LESS MOTOR COMMUTATION
- PROGRAMMABLE COMMUTATION DELAY
- LINEAR MODE AND CONSTANT TOFF PWM OPERATION MODE
- INTERNAL FREQUENCY LOCKED LOOP SPEED CONTROL (FLL)
- BEMF RECTIFICATION DURING RETRACT
- RESYNCHRONIZATION
- BUILT-IN ALIGN & GO START UP
- INDUCTIVE SENSE START UP OPTION
- DYNAMIC & REVERSE BRAKE
- PROGRAMMABLE OUTPUT SLEW RATE

- OTHER FUNCTIONS

- 5V MONITORING WITH EXTERNAL SET TRIP POINTS AND HYSTERESIS
- POWER UP/DOWN SEQUENCING
- LOW VOLTAGE SENSE
- 3.3V INPUT LOGIC COMPATIBILITY
- THERMAL SHUTDOWN AND PRE-THERMAL WARNING
- SYSTEM CLOCK WATCHDOG

2.0 AUXILIARY CIRCUITS

2.1 SUPPLY MONITORS AND POWER-ON RESET

The power supply monitor pin, TR_5V (#35), monitors the main supply VDD by means of external resistor dividers. The threshold of the comparators is set internally by the band-gap reference (2V) and hysteresis is 100mV. The ratio of the external resistors set the rising trip point. At Power-Up, PORB output pin (#34) remain low until the POR delay capacitor at POR_DELAY pin (#36) is charged. It goes low whenever either monitor input goes below the threshold . If power returns to normal during a retract sequence, the retract will finish before the POR delay can begin, which means that the PORB (pin #34) output will remain low until the end of the retract plus the PORB delay time. For POR_delay, the charge current is 2µA and the threshold is 3V. POR delay capacitor is calculated as follow:

- 1. $\frac{Time}{1.5\times10^{6}}$ POR Delay Capacitor (nF)

Example with Por Delay Time=150milli seconds : $Capacitor = \frac{150\times10^{-3}}{1.5\times10^{6}} = 100nF$.

The bits required for VCM parking are not reset by PORB and they must be always written (Reg #5.3, Reg #8.2 and Reg #9.0.1.3.6). All the other bits are reset whenever PORB is low. As long as PORB is low the Spindle outputs are in brake by means of turning on the Low Side Dmos. In case of Power Down with the Spindle previously spinning, at the falling edge of PORB, the Spindle outputs tristate for BEMF rectification and the Voice Coil initiate the retract phase according to the time and voltage set. After the retract time, the Spindle outputs brake. The complete Power Off Sequence is depicted in Figure #16 in chapter 5 (Voice Coil Circuit), section 5.3 (Retract)

2.2 THERMAL WARNING / SHUTDOWN

The Thermal Warning/Shutdown circuit senses the temperature of the die. When the temperature reaches the thermal warning threshold, the “THERMAL_WARN” bit will be reset (=0). If the temperature rises to 25°C higher than the thermal warning threshold, then “THERMAL” bit will be reset (=0). At this time, the Spindle outputs are tristated and Voice Coil will retract. After retract, also Voice Coil outputs will be tristated. The registers are not reset by Thermal Shutdown. The Spindle and VCM will be tristated as long as the internal Thermal Shutdown signal is active. To release this signal, the die temperature has to decrease and RUN bit (Reg #2.3) or VCM_EN bit (Reg #9.5) has to be toggled.

The “THERMAL” has approximately 60°C of hysteresis, so that the “THERMAL” bit will not return to normal until the chip has cool to 60°C below the rising “THERMAL” trip point. (“THERMAL” and “THERMAL_WARN” are in Reg #7.0.1).

2.3 CHARGE-PUMP

Two external pins (CP pin 23, CS pin 24) are used. The base of a PNP transistor is connected to Cs pin, the collector of the transistor is connected through a diode to CP pin. A storage capacitor is connected from CP to ground. An Inductor whit a resistor in series is connected between VDD and the collector of the transistor. The voltage at the CP pin is called Vboost. An internal oscillator is running at 200KHz. This can be checked at CS pin. The Vboost Voltage is typical VDD+5.7V. It is possible to stop the oscillator by setting the VB/DIS bit (Reg#10.7). However these are used mainly for testing purpose. The internal oscillator is not stopped by POR condition and it is keeping running for all the retract time as long as the VDD is greater than 2V. The current that CP pin requires is about 3mA with both Spindle and VCM in Linear mode operation and it is about 5mA with Spindle and VCM in PWM mode operation.

2.4 SYSTEM CLOCK WATCHDOG

The device monitors the SYSTEM CLOCK frequency throughout pin #40 (CLK_MON). A 15pF capacitor is connected between SYS_CLK pin (#15) and CLK_MON pin. A 10KΩ resistor goes from CLK_MON pin to ground. In the operating condition, a maximum time of 10µS +/- 30% is allowed between SYS_CLK edges (rise and fall). If no edge is detected within this time the system will cause a POR.
To disable the System Clock Watchdog function, simply do not connectect any components to CLK_MON pin (#40).

3.0 SERIAL INTERFACE CIRCUITS

3.1 SERIAL PORT

The The serial port interface is used to program the internal registers in addition to interfacing with the status and ID registers. The serial port is enabled for data transfer when the Serial Data Enable pin (SDEN pin #16) is high (=1). SDEN must be asserted high prior to any transmission and it should remain high until the completion of the transfer. At the end of each transfer SDEN should be brought low (=0).
When SDEN is high, the data presented to the Serial Data pin (SDATA pin #17) will be latched on each rising edge of the Serial Clock pin (SCLK pin #18). Rising edges of SCLK should only occur when the desired bit of address or data is being presented on the serial data line. The data is latched into the internal register after the 16th rising edge of SCLK. Each transmission consists of a write control bit followed by a 7 bit address word and 8 bit data word. The address bits select the internal register to be written. The address and data fields are input LSB first, MSB last, where LSB is defined as bit zero. When the write control bit is “1”, the contents of the read register that is being addressed will be shifted out on the SDATA pin LSB first (positive SCLK edge). The SDATA pin will switch from input (high Z) to driving output after the falling edge of the 8th SCLK pulse.
All write bits in the registers are reset whenever PORB is low except for Retract time Reg#9.3.6 and Retract voltage Reg#9.0.1 which must be set or reset by the external controller soon after power-up to insure that the correct value will be programmed in case of a power down situation. Even though the Reg#8.2 bit is not used, it must be also reset (=0) by the external controller. The serial interface is 3.3V compatible.

Figure 1. Serial Port Write Timing

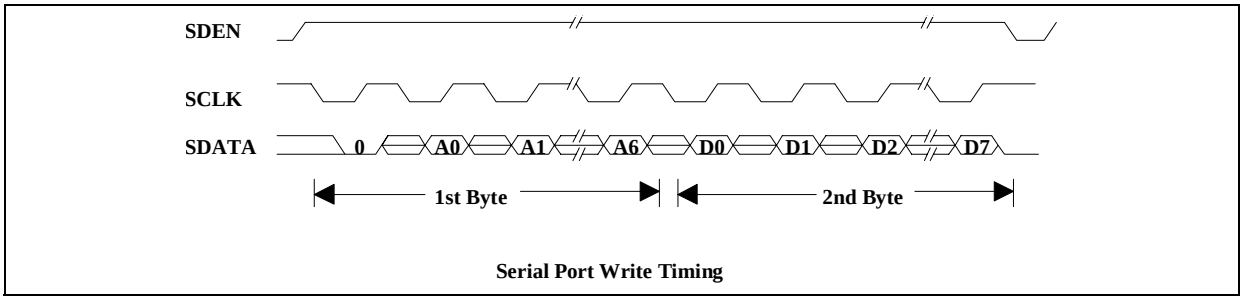
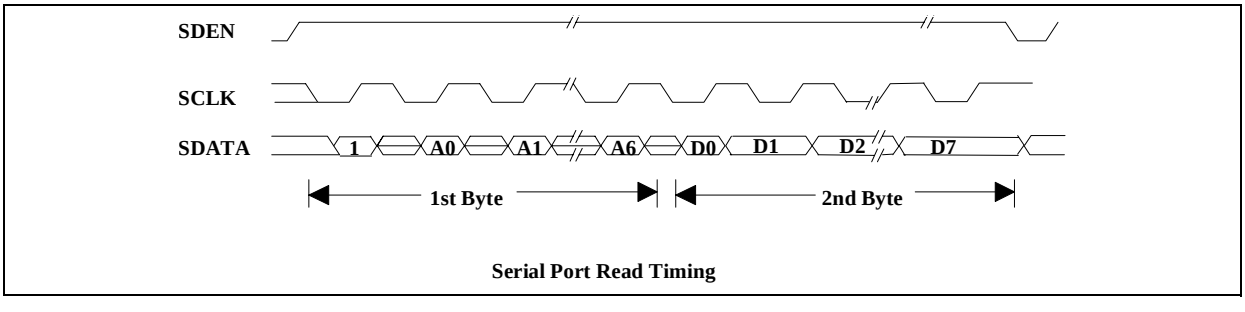


Figure 2. Serial Port Read Timing



3.2 REGISTERS MAP

Table 1. Register Map

Reg.	Description	Bit 7	Bit 6	Bit 5	Bit 4	Bit3	Bit 2	Bit 1	Bit 0	Address
0	VCM DAC HIGH BYTE	VCM _CAL	PSM/ LIN	VDAC BIT13	VDAC BIT12	VDAC BIT11	VDAC BIT10	VDAC BIT9	VDAC BIT8	0Eh 00001110
1	VCM DAC LOW BYTE	VDAC BIT7	VDAC BIT6	VDAC BIT5	VDAC BIT4	VDAC BIT3	VDAC BIT2	VDAC BIT1	VDAC BIT0	1Eh 00011110
2	SPINDLE CONTROL	EXT/ INT	PWM/ LIN	MECH/ ELEC	SPIN_E N	RUN	R_SE Q	START _UP	INCRE _SEQ	2Eh 00101110
3	SPINDLE DELAY	SD0	SD1	SD2	SD3	8_12_ POLE	MIN1	MIN2	MASK _TIME	3Eh 00111110
4	FLL COARSE COUNTER	C11	C10	C9	C8	C7	C6	C5	C4	4Eh 01001110
5	COARSE/FINE COUNTER	C3	C2	C1	C0	SET 0	F10	F9	F8	5Eh 01011110
6	FLL FINE COUNTER	F7	F6	F5	F4	F3	F2	F1	F0	6Eh 01101110
7	SPINDLE STATUS	GO	ALIGN	ERROR _LOCK	MASK _TIME	FAULT	ROTOR_ STUCK	THERMAL _WARN	THERMAL	7Fh 01111111
8	SPINDLE FLL	CPH	CPL	IL0	IL1	ISNS	SET 0	ICP	SSLEW	8Eh 10001110
9	SYSTEM CONTROL	RETRACT	RT1	VCM_EN	DOUBLE	RT0	VR	PKV_2	PKV_1	9Eh 10011110
10	TEST CONTROL	VB/DIS	SET 0	REV_ BRAKE	FLL_OUT	SET 0	SET 0	SET 0	SET 0	AEh 10101110
11	VCM CONTROL	SET 0	BRAKE	TRISTATE	SET 0	COM SLEW	SLEEP	VCMH	VCMS	BEh 10111110
12	CHIP ID	ID_ REV_7	ID_ REV_6	ID_ REV_5	ID_ REV_4	ID_ REV_3	ID_ REV_2	ID_ REV_1	ID_ REV_0	FFh 11111111

Reg#0: Bit0~Bit5, D8~D13 of VCM DAC
Bit6, “1” VCM PSM mode, “0” VCM Linear mode
Bit7, “1” VCM calibration

Reg#1: Bit0~Bit7, D0~D7 of VCM DAC

Reg#2: Bit0, “0” to “1” transition increments Spindle sequencer
Bit1, “1” Spindle Internal Align & Go Start-Up, “0” Spindle External Start-Up
Bit2, “1” Resets Spindle sequencer to phase 1 (A -B)
Bit3, “1” Starts Spindle Align & Go Start-Up, “0” Resets control logic and brakes the Spindle
Bit4, “1” Enables Spindle outputs, “0” disables Spindle outputs
Bit5, “1” Electrical cycle for Spindle FLL control, “0” Mechanical cycle for FLL control
Bit6, “1“ Spindle PWM mode, “0” Spindle Linear mode
Bit7, “1” External Spindle loop feedback (via Index pin), “0” Internal Spindle Loop feedback

Reg#3: Bit0, “1” 7.5° Spindle Mask time, “0” 15° Spindle Mask time

Bit1~Bit2, Spindle PWM mode Minimum ON time
Bit3, "1" 8 pole Spindle motor, "0" 12 pole Spindle motor
Bit4~Bit7, Spindle Commutation Delay control SD0....SD3

Reg#4: Bit0~Bit7, Spindle FLL coarse counter C4...C11

Reg#5: Bit0~Bit2, Spindle FLL fine counter F8...F10
Bit3, "1" Spindle Two phase brake ((*)only L6254), "0" Spindle Three phase brake
Bit4~Bit7, Spindle FLL coarse counter C0...C3

Reg#6: Bit0~Bit7, Spindle FLL fine counter F0...F7

Reg#7: Read Only Register

Bit0, "0" Thermal Shutdown (>= 160°C)
Bit1, "0" Thermal Warning (>= 125°C)
Bit2, "0" Spindle Bemf not detected
Bit3, "1" Rapid deceleration of the Spindle motor or High frequency on FCOM signal.
Bit4, Spindle Mask time signal, "0" Spindle bemf is masked
Bit5, "0" Spindle speed lock error (>16µs/sample)
Bit6, "0" Spindle is in the Internal Start-Up ALIGN Phase
Bit7, "0" Spindle is in the Internal Start-Up GO Phase

Reg#8: Bit0, Spindle PWM (chopping) slew rate,"1" 20V/µs, "0" 10V/µs
Bit1, Charge pump current in Spindle FLL loop, "1" 25µA, "0" 100µA
Bit3, "1" Presets Spindle inductive sense start-up circuits
Bit4~Bit5, Spindle start-up current limitation
Bit6, "1" Forces Spindle FLL charge pump low
Bit7, "1" Forces Spindle FLL charge pump high

Reg#9: Bit0~Bit1, VCM retract voltage
Bit2, "1" Connects the voltage reference (2V) for VCM calibration
Bit3~Bit6, VCM retract time
Bit4, "1" Doubles the times for the internal Spindle Align and Go Start-Up
Bit5, "1" Enables VCM
Bit7, "1" Initiates the VCM retract

Reg#10: Bit4, "1" Spindle Mechanical or Electrical output, "0" Spindle Zero cross output
Bit5, "1" Spindle reverse brake
Bit7, "1" Disables Vboost

Reg#11: Bit0, VCM PSM (chopping) Slew Rate,"1" 20V/µs, "0" 10V/µs
Bit1, "1" Forces VCM outputs to be high in PSM mode
Bit2, "1" Unused (Future Power saving mode)
Bit3, "1" Spindle PWM (phase commutation) slew rate "1" 2V/µs, "0" 30V/µs

Reg#12: Read Only Register

Bit0~Bit7, for chip ID

3.3 STATUS AND IDENTIFICATION REGISTERS

The Status register (Reg#7) and the Identification register (Reg#12) are READ only registers, while all the others are only WRITE registers. The **Status register** provides informations of the Spindle motor (Bits 2...7) and the Thermal behaviour of the device (Bits 1,2).

- **THERMAL** - Bit #0. Normally is set to 1, when the Junction temperature reaches about 160°C, the bit will be set to 0. This will automatically tristate the Spindle outputs and the Voice Coil will initiate the

retract.The bit will return to normal when the temperature will drop about 60°C. All the functions of the device but the reading in the serial port, will be in stand-by as long as this bit is set to 0.

- **THERMAL_WARN** - Bit #1. Normally is set to 1, when the Junction temperature reaches about 125°C, the bit will be set to 0. The device will take no action upon this bit set to 0. Corrections are demanded to the µP. This bit will return automatically to normal (=1) when the temperature will drop below the set point (about 125°C).
- **ROTOR_STUCK** - Bit #2. Normally is set to 1, if the Spindle BEMF is not detected, the bit will be set to 0. This can happen for example during Start-Up because the Spindle motor may be Stucked. The device will tristate the Spindle and VCM outputs upon this bit is set to 0.
- **FAULT** - Bit #3. Normally is set to 0, if the Spindle motor decelerate rapidly or the FCOM signal change to high frequency, the bit will be set to 1. The device will take no action upon this bit set to 0. Corrections are demanded to the µP.
- **MASK_TIME** - Bit #4. By reading continuosly this bit, the internal BEMF mask time signal can be monitored. When the bit is 0, the BEMF is masked.
- **ERROR_LOCK** - Bit #5. When the Spindle speed is at target speed +/- 16µS sample, the bit will be set to 1, otherwise will be set to 0.
- **ALIGN** - Bit #6. Normally is set to 1, when the Spindle motor is in the ALIGN phase of the internal Align&GO start-up, the bit will be set to 0.
- **GO** - Bit #7. Normally is set to 1, when the Spindle motor is in the GO phase of the internal Align&GO start-up, the bit will be set to 0.

By reading and decoding the **Identification register**, the revision of the Device can be retrieved.

4.0 SPINDLE CIRCUITS

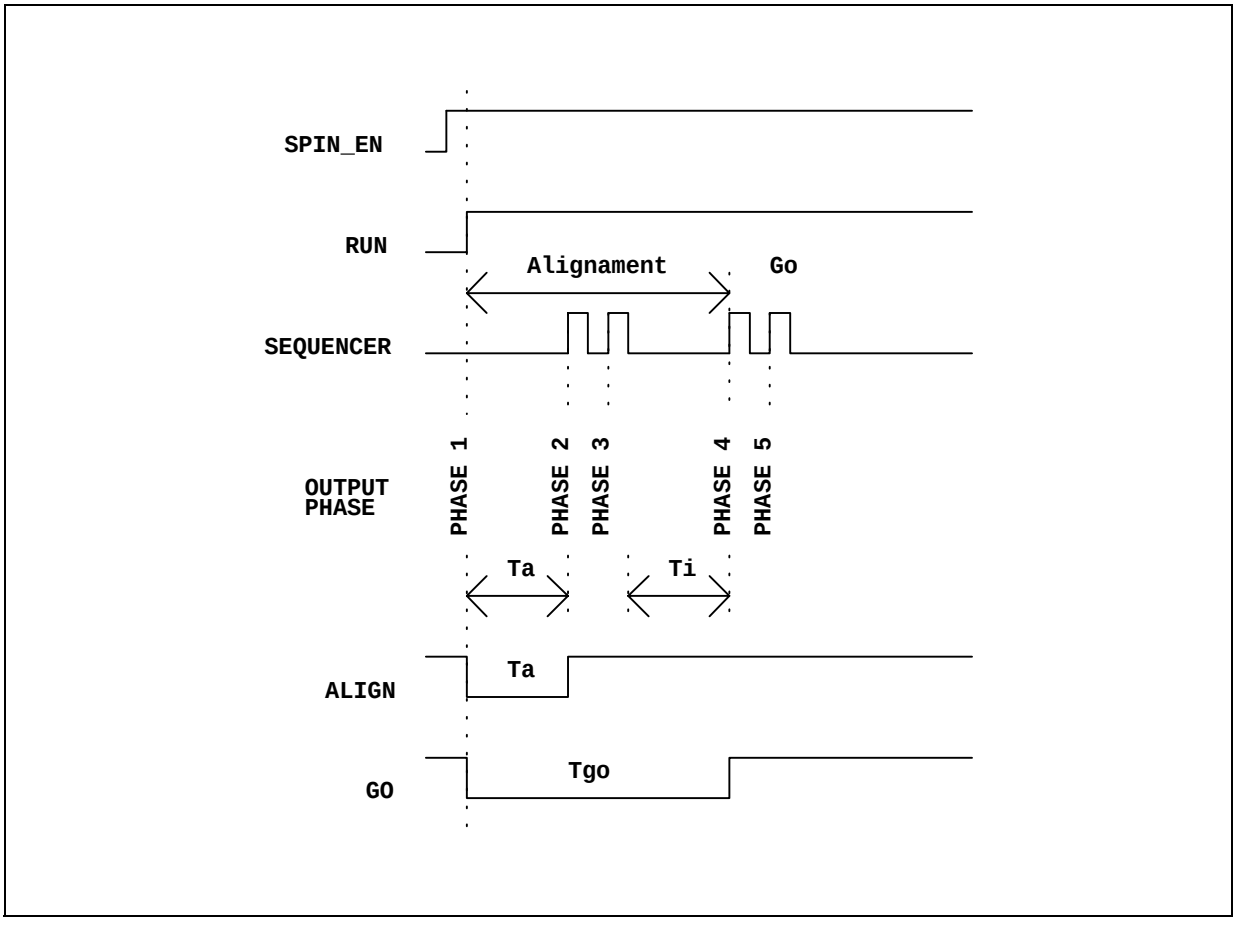
4.1 START-UP DESCRIPTION

The spindle driver has the ability to stand alone start-up the motor using an internal “Align & Go” algorithm. Although the internal Start-Up will spin-up the motor consistently, the possibility exist where certain applications might require complete microprocessor control. In this case, the chip provides all the function and signals to externally build a custom Start-Up.

4.1.1 Internal Start-Up

The Internal Start-Up is asserted by setting (=1) the START_UP bit (Reg#2.1). Assuming that the motor is stationary, when RUN and SPIN_EN bits (Reg#2.3.4) are set (=1), after a resynchronization time, the motor will be in Align mode with Phase 1 active (output A high and output B low. Please note that this output configuration is true at Power on condition or whenever reset sequencer is asserted, by toggling the R_SEQ bit (Reg#2.2), otherwise can be different). After a Ta time (align time) the sequencer double increments the outputs to Phase 3 (output B high and output C low). After a Ti time (increment time) the sequencer double increments again the outputs and the controller enters the GO mode, with the sequencer automatically incrementing the output phase upon detection of the Motor’s Bemf. The times labeled Ta and Ti are two delays that are 25% and 75% respectively of the total delay time and they are determined by the frequency of the system clock signal present on pin#15, (SYS_CLK). By setting the DOUBLE bit (Reg#9.4), Ta and Ti can be doubled. Ta and Ti are calculated as formula #2 and #3. The Figure#3 depicts the internal Align & GO auto Start-Up profile.

Figure 3. Internal Align & Go Auto Start-up Profile



- $2.\left(\frac{2.56\times10^{6}}{Sys\_Clk}\right)\cdot(1+DOUBLE)$ Internal Start-Up Align Time - Ta (mS)

- $3.\left(\frac{7.68\times10^{6}}{Sys\_Clk}\right)\cdot(1+DOUBLE)$ Internal Start-Up Increment Time - Ti (mS)

Example with 20MHz Sys_Clk and DOUBLE=0:

$$Ta=\left(\frac{2.56\times10^{6}}{20\times10^{6}}\right)\cdot(1+0)=128ms \quad - \quad Ti=\left(\frac{7.68\times10^{6}}{20\times10^{6}}\right)\cdot(1+0)=384ms$$

4.1.2 External Start-Up

When the START_UP bit (Reg#2.1) is reset (=0) the internal Auto-StartUp algorithm is disabled and the sequencer is accessible allowing the microprocessor to control the phase commutation of the motor. The chip offers this capability via the INCRE_SEQ bit (Reg#2.0). The transition 0 to 1 of this bit increments the sequencer to the next position. A feedback from the motor (zero cross) is available on pin #1, FCOM. FCOM toggles at each successive zero crossing and it can be monitored to assure that motion occurred. With this approach the microprocessor or DSP can access all the function of the chip and different Start-Up algorithm can be applied to spin-up the motor. Following are three Start-Up algorithm examples: **Align & GO**, **Stepping** and **Inductive Sense**.

4.1.2.1 Align & Go

The external Align & Go Start-Up is structured as the Internal one. The times **Ta** and **Ti** along with the commutation must be provided externally by the µP allowing further flexibility to the system. Assuming that the motor is stationary, when RUN and SPIN_EN bits (Reg#2.3.4) are set (=1), the motor is in Align mode with Phase 1 active (output A high and output B low, please note that this output configuration is true at Power ON condition or whenever reset sequencer is asserted, by toggling the R_SEQ bit (Reg#2.2), otherwise can be different). After a **Ta** time (align time handled by external µP) the sequencer can be double incremented to Phase 3 (output B high and output C low) by toggling the INCRE_SEQ bit (Reg#2.0). After a **Ti** time (increment time) the sequencer can be double incremented again which should produce torque in the desired direction. A little waiting time greater than the length of the Recirculation Spike produced by the motor coil during commutation (typical 3µS) must be applied prior the end-over operation. At this time, by setting the START_UP bit (Reg#2.1), the sequencer is controlled by the BEMF zero crossing and the motor should ramp up to speed.

4.1.2.2 Stepping

This approach is driven in a similar fashion to a stepper motor. The START_UP bit (Reg#2.1) must be reset (=0), RUN and SPIN_EN bits (Reg#2.3.4) must be set=1. By toggling the INCRE_SEQ bit (Reg#2.0), the commutation rate is continually increased until the BEMF voltage is large enough to reliably use the zero-crossing for commutation timing. Once this point has been reached, a little waiting time greater than the length of the Recirculation Spike produced by the motor's coil during commutation (typical 3µs) must be applied. By setting the START_UP bit (Reg#2.1), the BEMF zero crossing will automatically control the sequencer and the Motor Control circuit will take over to bring the motor into frequency lock. The Stepping approach takes longer than other Start-Up algorithm because the initial commutation frequency and subsequent ramp rate must be low enough so that the motor can follow without slipping. This implies that to have a reliable algorithm, the initial frequency and ramp must be carefully calculated according to the mechanical and electrical motor characteristics and under worst case conditions.

4.1.2.3 Inductive Sense

Since hard drive heads are not designed for back rotation, it is important to know which Spindle phase should be energized first, in order to avoid back-rotation.

During operation, the electrical phase of the motor can be detected by sensing the BEMF induced in the unenergized winding by the rotor's magnet in motion, however, when the motor is stopped, detection of the rotor position is more difficult. For this purpose, the Inductive Sense algorithm has been developed. To get the position of a stopped three phase permanent magnet rotor, the inductance of each of the six motor commutation phases is sensed by applying a voltage across each combination of two motor winding and measuring the time required for the current through the windings to reach a certain arbitrary selected value. The shortest rise time should correspond to the lowest inductance and this commutation phase should be the one closest to the mechanical position of the rotor.

Altought no Inductive Sense algorithm is built inside the device, it provides all the hardware functions that interacting with the µP can externally perform the Inductive Sense Start-Up. The ISNS bit (Reg#8.3) must first be set (=1) in order to preset the internal circuit to sense the current in the coils. This allow the FCOM pin (#1) to become the information signal of threshold reached.

There are four different threshold values which adjust the maximum voltage on sense resistor. The threshold can be selected by changing the **IL0** and **IL1** bit in the Reg#8.4.5. The Table #2 shows the available values.

Table 2. Spindle Inductive Sense Current Limit

"IL0"	"IL1"	"ISNS"	V_SENSE_LIMIT (+/- 15%)
0	0	1	.15V
1	0	1	.20V
0	1	1	.25V
1	1	1	.30V

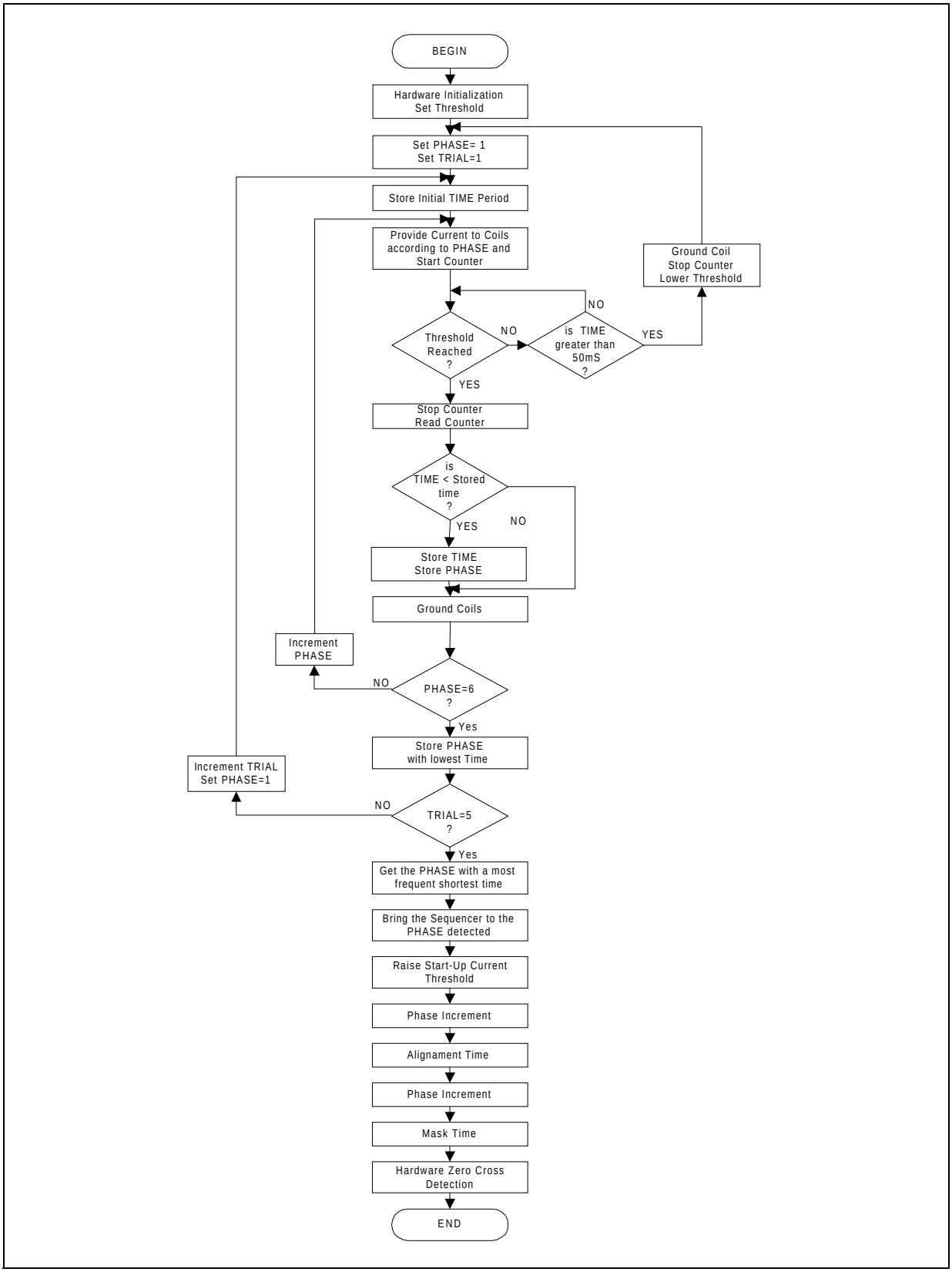
The current that flow into the coil is also a function of the Sense resistor connected to the Rsense pin and it is calculated as follow:

- 4. $\frac{V\_sense\_limit}{Rsense}$ Spindle Inductive Sense Maximum Coil Current (Amp)

Example with Rsense=0.3Ω and IL0/IL1/ISNS=1 : Max Coil Current = (0.3 / 0.3) = 1 A.

Figure#4 shows the flow chart of the Inductive sense Start-Up routine. A step by step procedure to build the Inductive sense Start-Up, can be found in chapter 8 (Appendix), section 8.3 (Inductive sense start-up step by step).

Figure 4. Inductive Sense Flow Chart



4.1.3 Resynchronization

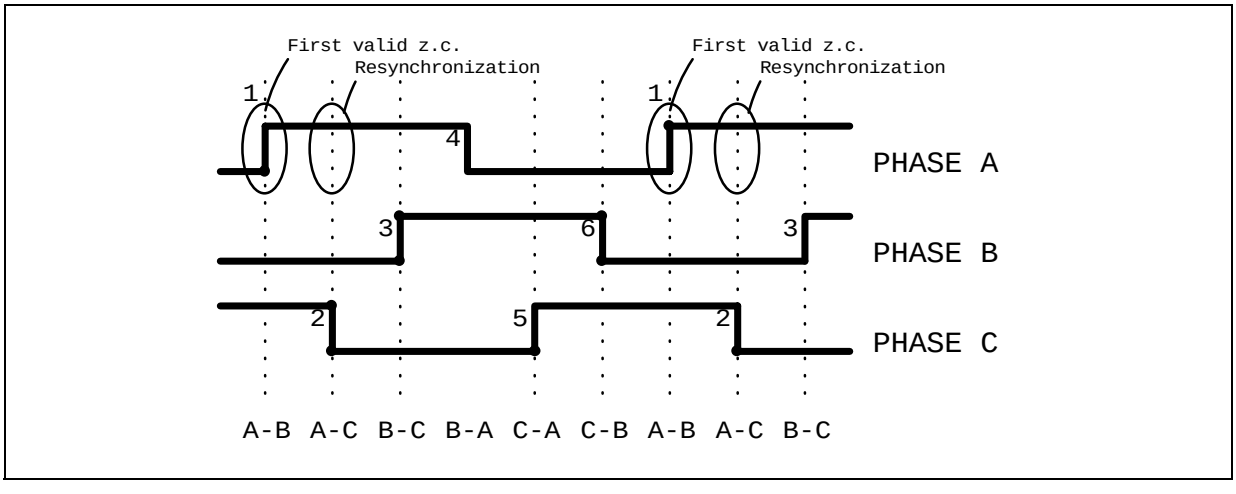
The Resynchronization is the ability to spin-up the motor without going through the Start-Up routine when the Power is momentarily lost with the Motor still moving and BEMF is detected. Supposing that the Spindle is running at a certain speed, when a POR (Power On Reset) occurs, the spindle outputs are tristated and the motor will coast. If the Power come back, PORB high, the internal circuit will first check for the BEMF and if some consecutive zero cross are detected, the system will be able to resynchronize the sequencer and get the motor at speed again. Please note that when POR condition occurs, most of the internal registers are reset, for this reason, it is important to restore the desired bits value as fast as possible upon POR gets back. Also note that the PORB stays at low level until Por Delay time expire. The Spindle resynchronization circuit stand-by for the BEMF (zero cross transition) for a **Tsync** time, which it is calculated as follow:

- 5. $\frac{8.4\times10^6}{SYS\_CLK}$ Spindle Resyncronization Time - Tsync (mS)

Example with 20MHz Sys_Clk : $Tsync = \left(\frac{8.4\times10^6}{20\times10^6}\right) = 420ms$.

The number of consecutive Zero Cross that the chip needs to detect before resynchronizing depends from the instant that the system is activated (PORB High and REG#2.1.3.4 =1), after for example a glitch in the power supply. The first valid zero cross, as it is shown in Figure #5, is set in point #1 (Phase A-B, this is true right afer POR goes on or whenever R_SEQ bit (Reg#2.2) is toggled). The points (Zero Crosses) between the system activation and the Resynchronization point (point #2) are the number of Zero Cross that the system has to detect, therefore the time before the resynchronization can initiate. If the motor is coasting and the chip is activated between point 1 and 2, the number of zero cross necessary will be 7. If the chip will be activated between point 6 and 1 the chip will resynchronize after two zero crosses. If the Spindle outputs are disabled by resetting (=0) the SPIN_EN bit (Reg#2.4), while the motor was spinning, the system will keep track of the zero crosses and it will resynchronize at the next zero cross upon the SPIN_EN bit is re-asserted.

Figure 5. Resynchronization Timing



After START_UP, RUN and SPIN_EN bits are set (REG#2.1.3.4=1), the system is halted waiting for a zero cross to occur within a Tsync time as explained before. If not enough zero crosses are detected, after Tsync time, the internal Align&GO algorithm is initied. If another Start-Up is required (i.e. Inductive sense), it is mandatory to reset the SPIN_EN bit before the end of the resynchronization time in order to avoid the internal Start-Up to be initiated. In other word, the external µP, in parallel with the chip, needs to keep track the time and

the BEMF (looking at FCOM pin (#1)) and take action whether internal or external Start-Up is to be initiated. If the motor is spinning and enough zero cross are detected, no actions are required, the chip sequencer will automatically lock on to the proper phase and bring the motor speed up to frequency lock.

4.1.4 Motor Stuck

The purpose of this feature is to protect the motor and the circuits against an over-heating due to high current in the same phase when the motor is not moving. During Start-Up, when a phase is energized according to the sequencer in the GO phase, the current starts to flow into the motor winding producing a torque which should result a mechanical movement of the rotor to the next position. If for some reason the rotor does not move (not enough torque or the motor is stuck) and the BEMF is not detected within **Tstuck** time, the system will shut off the current in the Spindle tristating the outputs and the controller will enter the 'Stuck Rotor Hold' state. To resume from this condition, RUN bit (Reg#2,3) must be toggled. Following is the formula to calculate the Tstuck time.

- 6. $\frac{8.4\times10^{6}}{\text{Sys\_Clk}}$ Spindle Motor Stuck Time - Tstuck (ms)

Example with 20MHz Sys_Clk: $Tstuck = \left(\frac{8.4\times10^{6}}{20\times10^{6}}\right) = 420ms$

The above description is valid always and not only at Start-Up. This means that the Stuck rotor counter is active all the time, also when the motor is running at steady speed. If for some reason, which it is, of coarse unlikely to happens (unless the speed gets extremely low such as 'Tzero_cross > Tstuck'), the BEMF is not detected within Tstuck time, the Stuck rotor circuit will react as explained before. The Stuck Rotor information can be read as a '0' in the Reg#7.2.

4.2 BRAKE

The Brake is the ability to completely stop a Spindle Motor in a reasonable time. There are two different way to stop the spindle motor, one is Commanded through the serial port and the other one is when the Power supply is turned off causing a POR condition. The device offers three different mode to brake, **Three phase brake** and **Reverse brake**. In a power-down condition only Three phase brake are available.

4.2.1 Three Phase Brake

The brake function is asserted by resetting (=0) the RUN bit (Reg#2.3) in power on condition and after retract in case of power loss. The low side output DMOS of phase A, B and C at this time are turned on.

In a power-down condition, after retract, when spindle brake initiate, the voltage on VDD drops very rapidly because the low side DMOS are turned on shorting the motor BEMF. In order to keep the spindle sink transistors ON in this mode, an external capacitor is used on the BRK_CAP pin (#12) to hold the gates of the sink devices ON. This capacitor is charged up whenever VDD is high to about 5V, and it is isolated from all supplies by blocking diodes. Following is the formula to calculate Brake storage Capacitor where **Time** is the duration in seconds of the brake time required.

- 7. $25\times10^{-8}\cdot Time$ Spindle Brake Capacitor (µF)

Example with Brake Time=8 seconds : $\text{Brake Capcitor} = (25\times10^{-8}\cdot 8) = 2\mu F$.

4.2.2 Reverse Brake

It is available only when power is present (PORB=1). It is asserted by setting the REV_BRAKE bit (Reg #10.5).

At this time the current in the winding of the motor is reversed respect its previous flowing, resulting a rapid deceleration of the motor. The system will end over to the normal three/two phase brake (according to the Reg#5.3 bit value) as soon as the time between two zero cross is reaching a predefined value.

Three different values are available by toggling the REV_BRAKE bit (Reg#10.5) according to the following table (#3).

Table 3. Zero Crossing Time after reverse brake

Reg#10.5	Zero Cross Time
1 Toggle	10ms
2 Toggles	20ms
3 Toggles	40ms

Since the saturation of the Powers are required to succeed the Reverse Brake operation, the above statement is valid if PWM Current Control mode is operating. In case of Linear Control Loop, a little trick needs to be used. Right before setting the REV_BRAKE bit , the CPH bit (Reg#8.7) must be set (=1). This will disconnect the FLL loop from the Current control loop bringing quickly the low side DMOS to saturation.

4.3 CURRENT CONTROL

The current in the spindle motor is sensed by a small resistor (typ. 0.3Ω) connected to the bottom of the H bridge and ground. The voltage drop across the resistor is amplified by a sense amplifier with a gain of four. The output voltage of the sense amplifier is sent to the Error Amplifier Stage where it is compared to the Current command coming from the FLL circuit. The Error Amplifier circuit will drive the power stage according to the driving mode selected. The device provides two different Current Control Loop, Linear and Pulse Width Modulation (PWM). By changing the Reg#2.6 bit, the selection of the two modes can be done any time regardless the motor is running or not.

4.3.1 Current Limitation

During initial Start-Up, the error signal from the output of the FLL will be at compliance in order to quickly bring the motor up to correct speed.

The maximum current flowing into the coil at the beginning (when the motor is stopped and no BEMF is present) is given by the following equation where RdsON is the ON resistance of the power DMOS.

Icoil = Vsupply / (2*RdsON + Rmotor + Rsense)

The motor current during this condition can be safely limited to a predetermined value by setting the appropriate bits IL0 and IL1 (Reg#8.4.5) according to the table #4.

Table 4. Spindle Rsense Voltage Limit

"IL0"	"IL1"	V_SENSE_LIMIT (+/- 15%)
0	0	.45V
1	0	.50V
0	1	.55V
1	1	.75V

The current flowing into the coil will be calculated as follow:

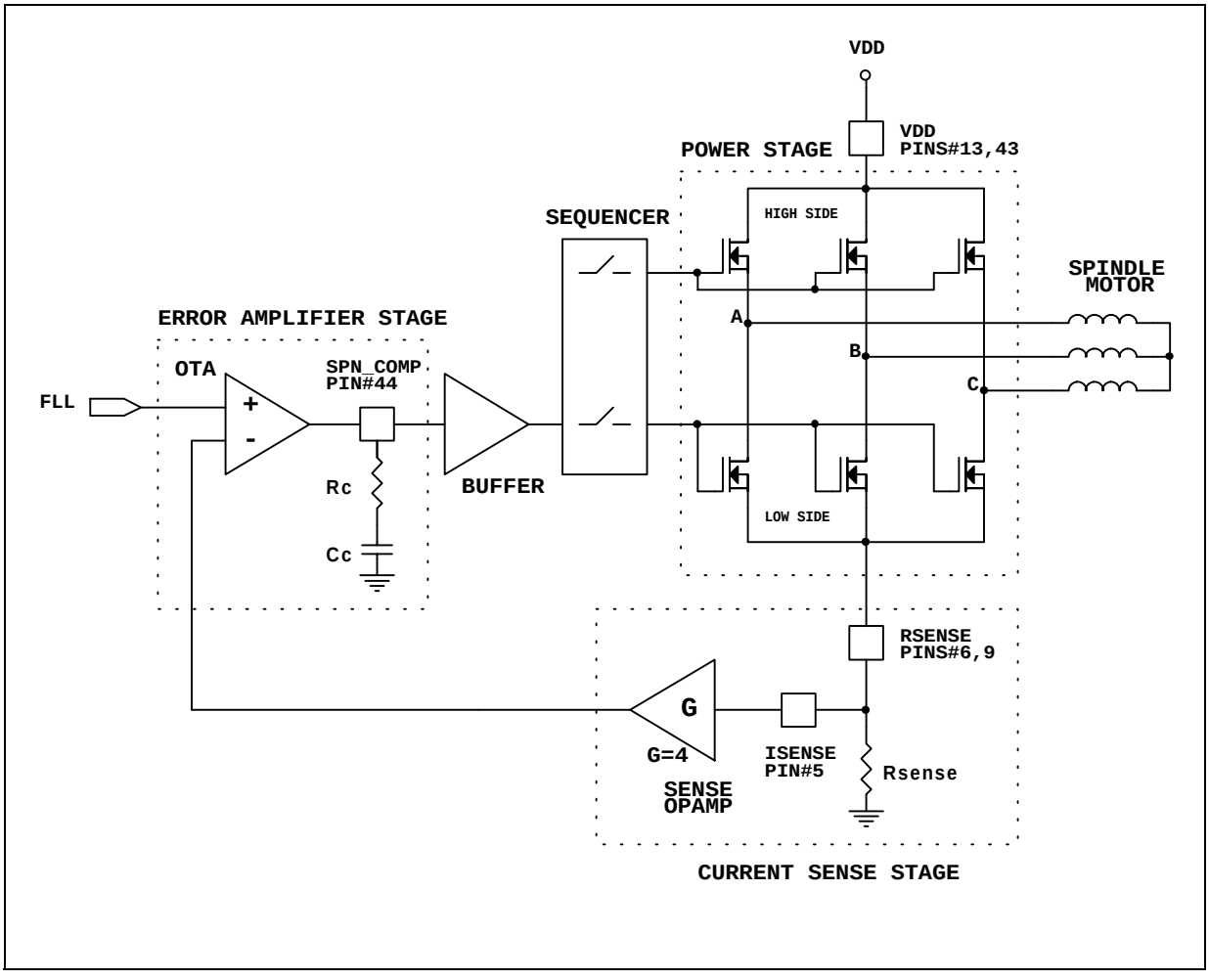
- 8. $\frac{V\_sense\_limit}{Rsense}$ Spindle Maximum Coil Current (Amp)

Example with Rsense=0.3Ω and IL0/IL1=0 : Max Coil Current = (0.45 / 0.3) = 1.5 A.

4.3.2 Linear Current Control

The Linear mode of operation for the Spindle driver is activated by a " 0" on bit 6 in the Spindle Control register Reg #2.6.

Figure 6. Spindle Linear Current Control Loop

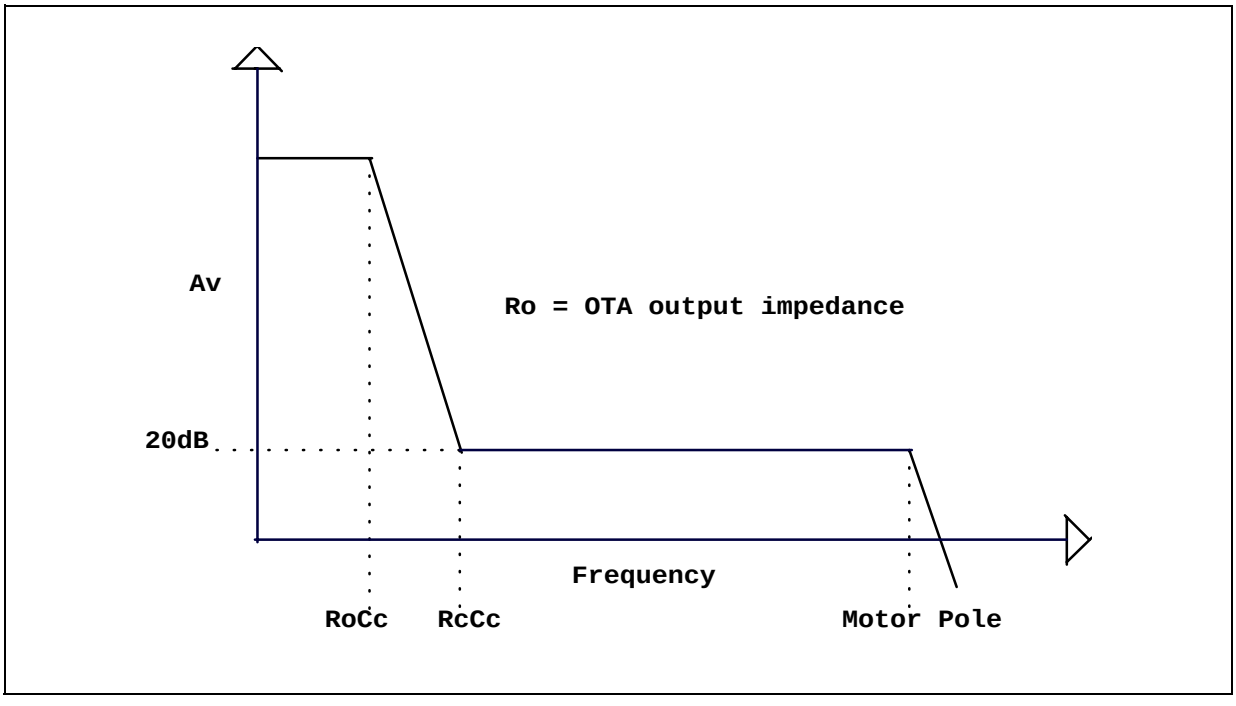


The output current is controlled in a Linear fashion via a transconductance loop. Referring to Figure #6, the sourcing FET (High Side Dmos) of one phase is forced into full conduction, while the sinking FET (Low Side Dmos) of an appropriate phase operate as a transconductance element. During a run condition, the current in the sinking FET, is monitored by a RSENSE resistor. The resulting voltage that appears across the resistor is amplified by a factor of four by the sense amplifier opamp and is sent to the Error Amplifier Stage where it is compared to the Current command coming from the FLL circuit. The Error Amplifier circuit along with the Buffer, provides sufficient drive to the sinking FET in order to maintain the motor speed at the proper level as commanded by the Speed Controller (FLL).

4.3.2.1 Transconductance Loop Stability

The RC network connected to the Compensation pin (SPIN_COMP pin#44 - Figure #6) provides a single pole/ zero compensation scheme. The pole/zero locations are adjusted such that a few dB (typ. 20dB) remains in the transconductance loop at frequencies higher than the zero. The inductive characteristic of the load provides the pole necessary for loop stability. Thus the loop bandwidth is actually limited by the motor itself.

Figure 7. Spindle Linear Current Loop Bode Diagram



The Bode plot in Figure #7, depicts the normal way to achieve stability in the loop. The pole and the zero are used to set a gain of 20dB at a higher frequency and the pole of the motor cuts the gain to achieve stability.

Loop instability may be caused by two factor:

1) The motor pole is too close to zero. The zero is not able to decrement the shift of the phase, and when the effect of the motor pole is present, the phase shift may reach 180° and the loop will oscillate. To rectify this situation the pole/zero must be shifted at lower frequencies by increasing the compensation capacitor (Cc).
2) The motor capacitance itself can interfere with the loop, creating double poles. If the gain at higher frequencies is sufficiently high, the double pole slope of 40dB/decade can cause the phase shift to reach 180°, resulting in oscillation. By leaving the pole unchanged and increasing the zero, the stable response can be achieved. The Model and a Mathcad analysis for the Spindle Current Control Loop can be found in chapter 8 (Appendix), section 8.1.1 (Spindle Current Control loop).

4.3.2.2 Slew Rate Control

A 3-phase motor appears as an inductive load to the power supply. The power supply sees a disturbance when one motor phase turns OFF and another turns ON because the DMOS turn-off time is much shorter than the L/R rise time. Furthermore, the turn-off of the DMOS can even cause current recirculation back into the supply. However, the need for a snubber circuit can be eliminated by controlling the turn-OFF time of the DMOS. The rate at which the upper and lower drivers turn ON and OFF is programmable via an external resistor, Rslew, connected to the PWM/SLEW pin (#3). This resistor define an internal current source that is utilized to limit the voltage slew rate at the outputs during transition, thus minimizing the load change that the power supply sees.

To insure proper operation, the range of the resistor value indicated should not be exceeded.

If a relatively low value of Rslew is selected, the resultant fast Slew Rate will result in increasing commutation cross-over current, higher EMI and large amount of commutation current.

Higher values of Rslew result of course in slow Slew Rate at the outputs which is, under most conditions, the desired case since the problems associated with fast rates are reduced. The additional advantage is lower acoustical noise. Problems can occur though if the Slew Rate for a given application is too slow. The problem manifest itself as the motor begins to spin-up. At lower RPMs, the Bemf of the motor is relatively small resulting in higher amounts of commutation current. The excessively long Slew Rate may exceed the mask period and the commutation spike can be detected as a zero cross.

With a high value of Rslew, also the Power Dissipation needs to be considered. Following is the formula to calculate the intended Slew Rate Resistor in Linear mode.

- 9. $\frac{(3 \cdot 10 \times 10^4)}{\text{Slew Rate}}$ Spindle Output Slew Rate Resistror - Rslew (Ω)

Example for Slew Rate = 6V/µS : $Rslew = \frac{(3 \cdot 10 \times 10^4)}{6} = 50K\Omega$

4.3.3 Pwm Current Control

The PWM mode of operation for the Spindle driver is activated by a ”1” on the Spindle Control register Reg #2.6 bit.

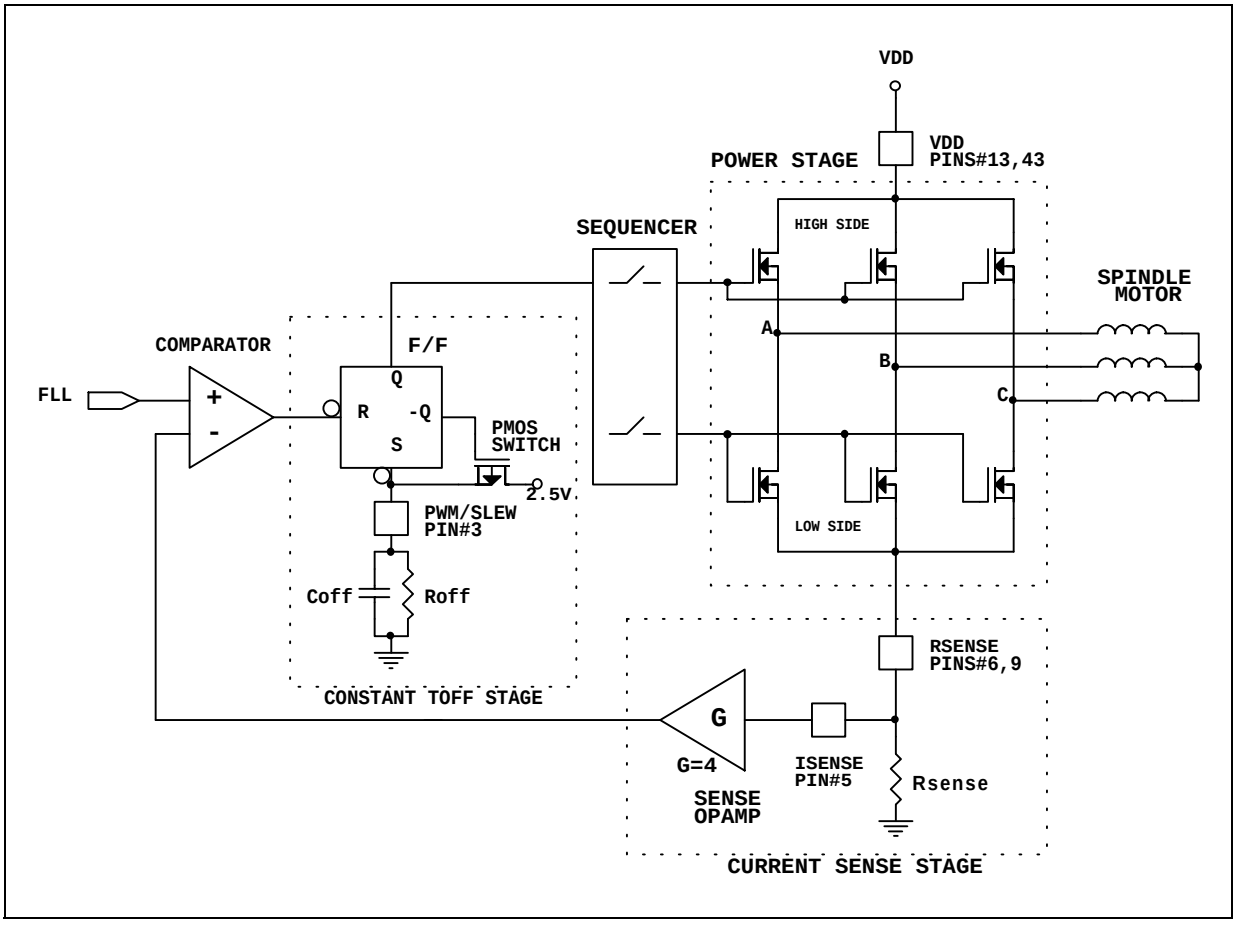
In the switch mode current control operation, only the high side drivers are pulse width modulated to control the current in the motor. The low side drivers are fully switched ON or OFF according to their commutation sequence timing. The PWM control of the high side drivers is achieved by a constant TOFF single shot circuit. Constant TOFF control operates on the principle of monitoring the motor current and comparing it to a reference or control level (FLL). When the motor current reaches this commanded level, the output drivers turn OFF and remain OFF for a constant time. After this time the drivers turn back ON to repeat the cycle. Referring to Figure #8, the current in the motor windings is monitored via the voltage dropped in the sensing resistor, Rsense. This voltage is multiplied by a factor of four in the Current Sense Amplifier and sent to the negative input of a comparator. The control voltage (FLL), is applied to the positive input of the comparator. When the output of the Current Sense Amplifier reaches a level that is equal to the commanded level, the output of the comparator switches low resetting the Q output of the F/F.

This causes the upper driver to turn OFF and through -Q (which will go high) turn off the PMOS switch allowing the capacitor (Coff) to discharge through Roff, initiating then the Constant TOFF time.

When the voltage on Coff reaches 1.2V, the F/F will be set again causing the upper drive to turn ON and the Coff capacitor itself to be quickly recharged.

The Figure #8 shows the Current Control Loop in PWM mode.

Figure 8. Spindle Pwm Current Control Loop



4.3.3.1 Constant OFF Time

The timing of the single shot circuit can be programmed by an external RC network at PWM/SLEW pin (#3). The constant OFF time can be calculated as follow:

- 10. $0.69 \cdot Roff \cdot Coff$ Spindle PWM Constant OFF Time - Toff (µS)

Example for Toff = 10µS and Roff = 100KΩ (Typ.): $Coff = \frac{10\times10^{-6}}{0.69} \cdot 100\times10^{3} = 150pF$.

4.3.3.2 Minimum ON Time

In a Spindle PWM current control system, the chopping in a commanded phase causes a swinging also in the floating phase. For this reason the BEMF zero cross signal is masked most of the time. The BEMF detection is possible by opening a small window in which the floating phase is monitored.

The window will be opened 6.4µs (based on 20MHz SYS_CLK) after the turn ON of the driver and it will be closed when the zero cross occurs or at the turn OFF of the driver. In the case of minimum ON time less than 6.4µs, the window will be opened only at the point where the output is about to be turned OFF and then it will be closed right away.

Since the noise caused by the turning ON of the high side driver can be detected as a false zero cross, the

minimum ON time has to be chosen large enough to cover those spike. In the other hand has not to be too large otherwise the motor can not slow down fast when required or it may run at higher undesired speed. For this reason four different ON times value are available through the serial port, MIN1 and MIN2 bits (Reg#3.1.2), Those timing are not dependent from the external SYS_CLK signal but are internally generated. The table #5 shows the timing available.

Table 5. Spindle PWM Minimum ON Time

"MIN1"	"MIN2"	Minimum ON Time
0	0	5.9µS
0	1	1.4µS
1	0	12.0µS
1	1	5.21µS

4.3.3.3 Slew Rate Control

In the PWM current control mode, two different slew rate control are present, the first one during commutation and the second one during chopping. Both are addressable through the serial port. The COMSLEW bit (Reg#11.3) will performs slew rate at commutation with '0' for 30V/µS and '1' for 2V/µS. The SSLEW bit (Reg#8.0) will perform slew rate during chopping with '0' for 10V/µS and '1' for 20V/µS.

4.3.3.4 PWM Design Methodology

The bandwidth of the PWM loop was optimized to reject unwanted switching noise while providing sufficient response, commensurate with the switching speed of the output drivers. At higher frequencies the switching losses inherent in the drivers start to negate any of the power dissipation savings gained with PWM operation. At lower frequencies, less that 20KHz, the switching of the coil may result in the undesirable acoustic range. A good compromised PWM frequency can be assumed around:

(1) Chopping Frequency: **30-40KHz.**

The time that the current flowing into an inductor needs to reach a certain value is equal to a time that it needs to get from that value to zero. This time is depending from the Inductor (Lm) and the resistor (Rm) values. In the case of a running motor the BEMF generated by the motor itself at that speed needs to be taken in consideration. Due to the fact that BEMF is opposing to the power supply, the voltage across the motor will be less (Vmotor = Vpower-Bemf). For this reason the charging and discharging of the inductor will result with different voltage and so different duration. The ratio to have a good regulation margin will be about:

(2) Ton **/** Toff Ratio: **TON = 70% - TOFF = 30%**

The device architecture has been designed to work in constant OFF time mode, this means that the system is using the frequency (changing only the ON time) to adjust the speed. The ratio TON / TOFF provide the motor with the required amount of current to spin at desired speed.

Assuming that 40KHz chopping frequency is used, at steady speed, the period will be equal to 25µS. For the previous assumption (2), the TOFF needs to be 30% of 25µS so will be about 7.5µS. In this case the TON will be equal to 17.5µS (TON=Period -TOFF). If this Ratio can not be achieved, most probably the motor is not been designed to run at that speed (BEMF provided is too high).

4.4 SPEED CONTROL (FLL)

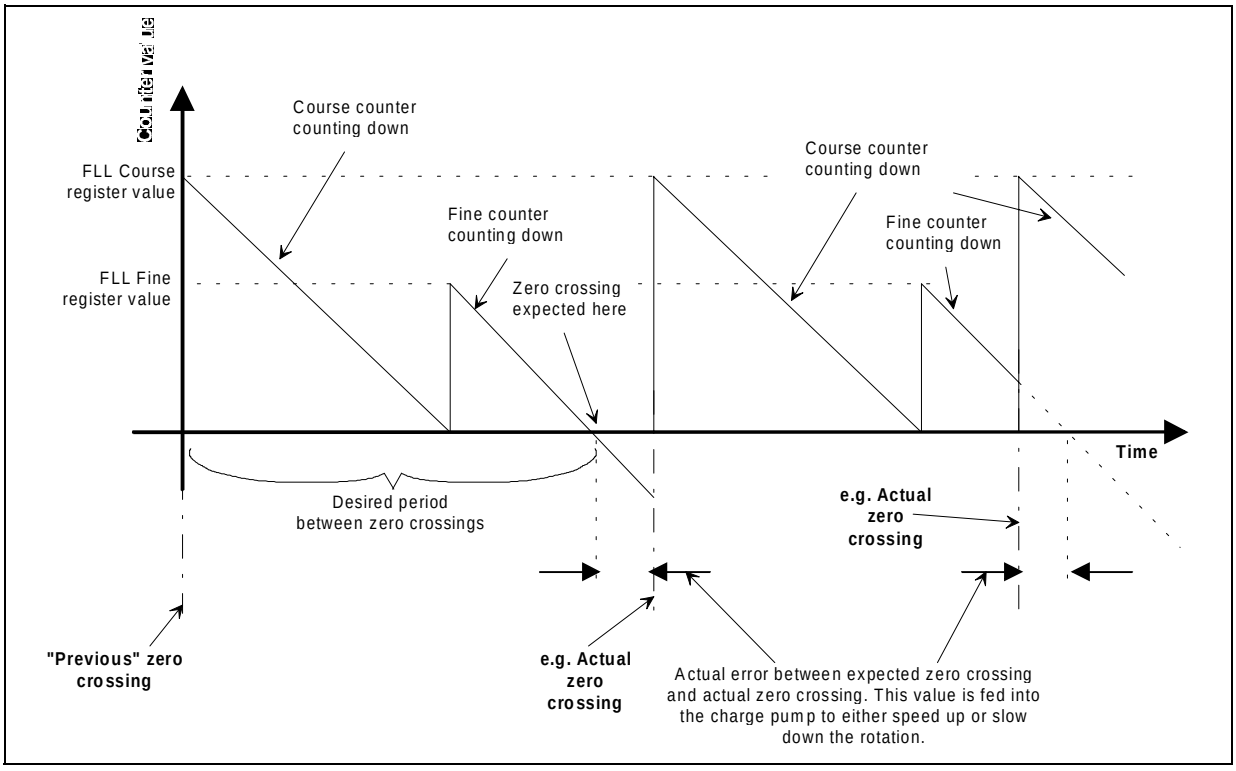
The rotational position of the motor is inferred from the BEMF waveform generated by the floating coil. The chip uses the instant of a particular zero-crossing and the period between successive zero crossings to dictate the

commutation timings. The complete control loop is on chip and the speed is controlled by a reference clock at pin SYS_CLK (pin#15).

The speed control loop uses a frequency locked loop (FLL) which in conjunction with an external compensation network brings the frequency of the tachometer signal to be equal to the internally generated reference frequency. The tachometer signal can either be the BEMF signal divided down to a once per mechanical or electrical revolution signal or an externally generated tachometer signal, sector burst. The output of the speed control is a current demand signal that goes to the Spindle Driver.

There are a "Fine" and a "Coarse" counters that define the speed of the motor. The register #4 and half of the #5 are the FLL Coarse Counter registers (12bits total, only count down to zero). The register #6 and the other half of the #5 are the FLL Fine Counter registers (11bits, count down to 2's complement of the 11 bit value). The figure #9 shows the counters.

Figure 9. FLL Counters



In more detail, those three registers are used in conjunction with two down counters which form a frequency detector that in turn creates feedback through to a charge pump to maintain the motor's speed regulation. The frequency clock applied to SYS_CLK pin is internally divided by a factor of 5.

The coarse counter is 12 bits and it is clocked at 1/64th the rate of the internal frequency clock. The fine counter is 11 bits and it is clocked at 1/4th the rate of the internal frequency clock. The on-chip Frequency Locked Loop uses the electrical or mechanical cycle pulses, according to the Reg#2.5 setting to adjust the speed of the motor. Upon the first pulse, the coarse register's contents (loaded via the serial port) is loaded into the internal coarse counter which immediately starts to count down. When this coarse counter reaches zero, the fine counter is then loaded from its corresponding register. The fine counter then also immediately starts to count down. The fine counter can count down through zero and continue counting down to the 2's complement of the original fine counter value.

The period between the start of the coarse counter and the zero crossing during the fine counter operation is

the programmed period. Any differences between the desired period and the pulse is the error in the transconductance loop and corrective action is taken by the charge pump.

The pulse measurement is initiated by an edge of the Feedback Frequency (Figure #10). If the feedback frequency is higher than the reference frequency, then a DOWN pulse will be generated whose width is the difference in pulse width of the two inputs. If the feedback frequency is lower than the reference frequency than an UP pulse will be generated whose width is the difference on pulse width of the inputs. The maximum pulse width of the frequency comparator is set by the value programmed into the fine counter. UP and DOWN pulses control the output of a current source or sink respectively which are connected to the FLL_FILTER pin (#21). The current source and sink may also be turned on by the CPH and CPL bits in the serial port (Reg#8.6.7), however these are mainly for testing purpose. The value of the source and sink currents is set by the ICP bit (Reg#8.1) in the serial port. The result of this is a current pulse, whose width is proportional to the speed error, this is presented to the FLL_FILTER pin which is used with an external RC network to construct the Loop Filter for the motor speed control. The FLL_FILTER pin is also internally connected to the input of a unity gain buffer. The output of this buffer is clamped to a voltage set by ILO, IL1 and ISNS bits (Reg#8.3.4.5.) in the serial port. The value for the clamp voltage are outlined in tables #2 and #4.

The coarse and fine counter arrangement is guaranteed to work in all possible circumstances. For example if the zero crossing is within or outside the fine window or even if the zero crossing is in the coarse register range. This system will even work if the zero crossing occurs across multiple coarse/fine cycles. The FLL has a prescaler (defined by the System Control Register bits MEC/ELEC and 8_12_POLE (Reg#2.5 & Reg#3.3) that changes the cycle counting mechanism between electrical or mechanical (8 pole or 12 pole) i.e. dividing the electrical period clock by 1, 4 or 6.

The equation for setting the Coarse and Fine Counters are calculated as shown in formulas #11 and #12 where T0 is the period of the updating frequency that can be either electrical or mechanical and it is calculate as **T0 = 60 / Rpm * Cycle**. **Cycle** is the cycle counting mechanism and as explained before it can be 1,4 or 6 according to desired cycle. Use 1 if Mechanical cycle is selected, use 4 or 6 if electrical cycle is chosen according to the motor pole. Conventionally, **90%** of T0 goes into the Coarse Counter, **10%** goes into the Fine Counter. **Rpm** is the desired speed.

The **CoarseCounterPeriod** is the time of one period of the Coarse Counter and it is equal to **1/Frequency(SYS_CLK) * 5 * 64.** The **FineCounterPeriod** is the time of one period of Fine Counter and it is equal to **1/ Frequency(SYS_CLK) *5 *4**. The **CoarseCounterError** is the error from the Coarse Counter calculation and needs to be added to the FINE register calculation.

- 11. $COARSE\_REG = \frac{(0.9 \cdot T0)}{CoarseCounterPeriod}$ FLL Coarse Counter

- 12. $COARSE\_FREQ = \frac{(0.1 \cdot T0) + CoarseCounterError}{FineCounterPeriod}$ FLL Fine Counter]

Example whit a speed of 5400Rpm, Cycle=Mechanical and SYS_CLK=20MHz.

1. Calculate T0 reference period.

T0 = 60 / (Speed*Cycle) = 60 / (5400 *1) = **11 mS**

2. Calculate Coarse Counter Period.

CoarseCounterPeriod = (1 / Frequency(SYS_CLK)) *5 * 64 = (1 / 20e6) * 5 * 64 = **16μS**

3. Calculate Fine Counter Period.

FineCounterPeriod = (1 / Frequency(SYS_CLK)) *5 * 4 = (1 / 20e6) * 5 * 4 = **1μS**

4. Calculate Coarse Register value.

COARSE_REG = (0.9 * T0) / CoarseCounterPeriod = (0.9 * 11e-3) / 16e-6 = **618.75**
The number 618 is the value that has to be programmed into the Coarse Counter Register. The decimal number 0.75 is the error that needs to be added to the computation of the Fine Register.

5. Calculate Coarse Counter Error.

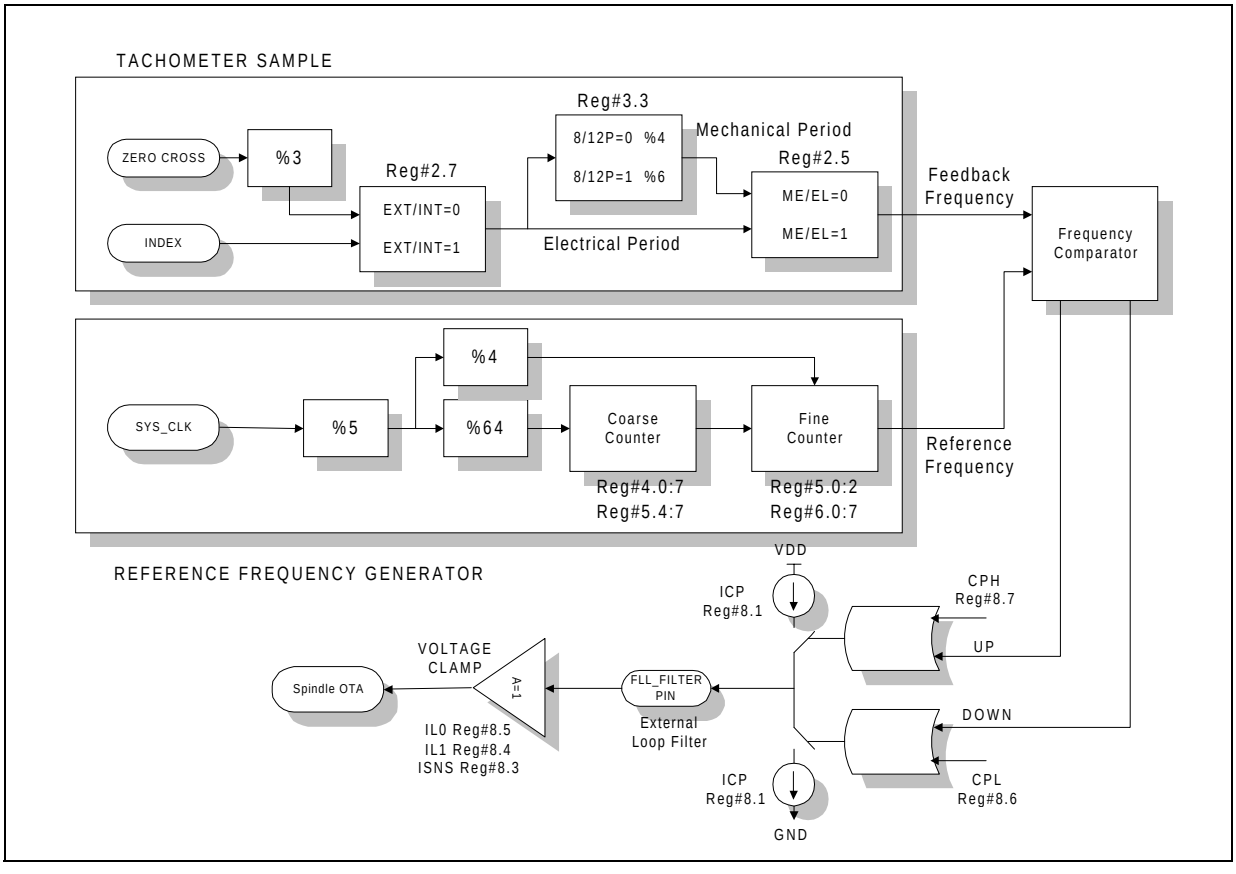
CoarseCounterError = 0.5 * CoarseCounterPeriod = 0.75 * 16e-6 = **12µS**

6. Calculate Fine Register Value.

FINE_REG = (0.1*T0+CoarseCounterError) / CoarseCounterPeriod = (0.1*11e-3+12e-6) / 1e-6 = **1112**

The number 1112 is the value that has to be programmed into the Fine Counter Register.

Figure 10. FLL LOOP



Since the Coarse Counter is a 12 bit counter, the maximum count available is 4096. Therefore, from the above example, the full Coarse count set limit will be 65,536mS (4096*CoarseCounterPeriod). The Fine Counter is only a 11 bit counter for a total of 2048 counts. The full Fine count set limit will be 2,048mS (2048*FineCounter-Period). If a greater number will result from the calculation, the percentage of T0 used in the Fine Counter equation needs to be decreased and added to the Coarse Counter equations (for example , 91% of T0 goes into the Coarse Counter and 9% of T0 goes into the fine Counter). This procedure needs to be repeated until the number will fit the Fine Counter Register. Figure #10 depicts the complete FLL loop block diagram.

4.4.1 Internal Frequency Feedback

The internal frequency for the Speed Loop Feedback is taken internally from the FCOM signal divided down by 3 and it is selectable by resetting (=0) the EXT/INT bit (Reg#2.7). The Figure#10 shows the configuration.

4.4.2 External Frequency Feedback

An INDEX pin (#11) is provided for external Speed Loop Feedback frequency. This pin will allow the user to control the motor speed with a signal taken from the data read off of the disk itself. This mode is selectable by setting (=1) the EXT/INT bit (Reg#2.7). The Figure#10 shows the configuration.

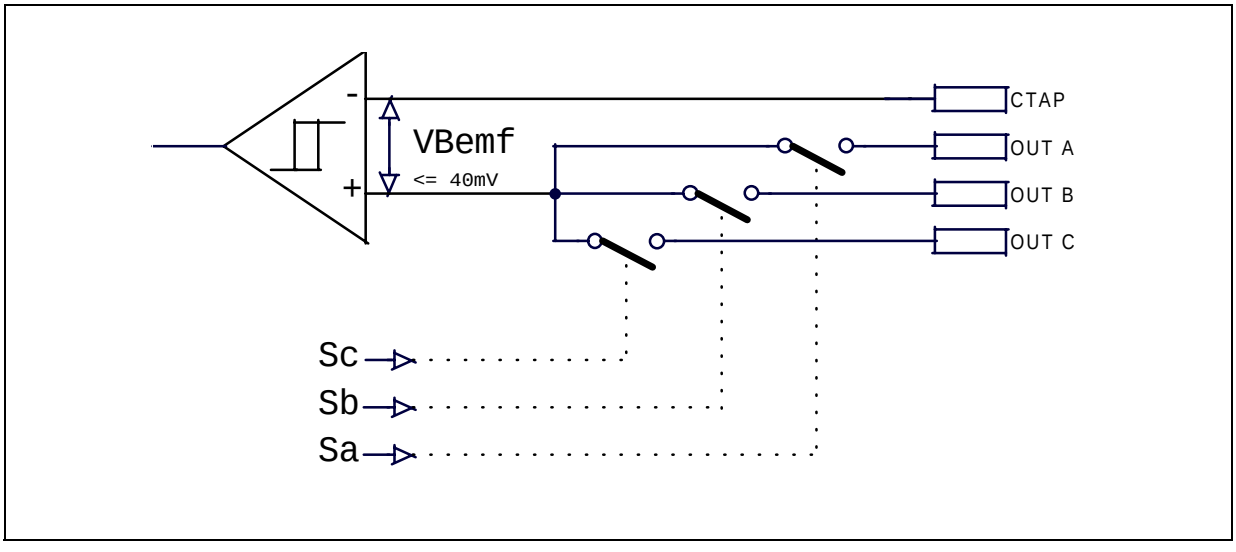
4.4.3 Speed Loop Compensation

The Frequency Comparator and Charge Pump generate and error current based on the frequency difference between a programmed reference frequency. This error current is converted to an error voltage by the loop filter, which programs the current delivered to the motor through the OTA loop. The Model and a Mathcad analysis for the Spindle Speed Control Loop can be found in chapter 8 (Appendix), section 8.1.2 (Spindle Speed Control loop).

4.5 BEMF DETECTION

Since no Hall Effect Sensors are required, the commutation information is derived from the BEMF voltage zero-crossing of the undriven phase with respect to the center tap. The Bemf comparator and associated signal levels are depicted in Figure #11. For reliable operation, the BEMF signal amplitude should be a minimum of +/- 40mV to be properly detected. In order to provide for noise immunity, internal hysteresis (typically 15mV) is incorporated in the detection circuitry to prevent false zero crossing detection.

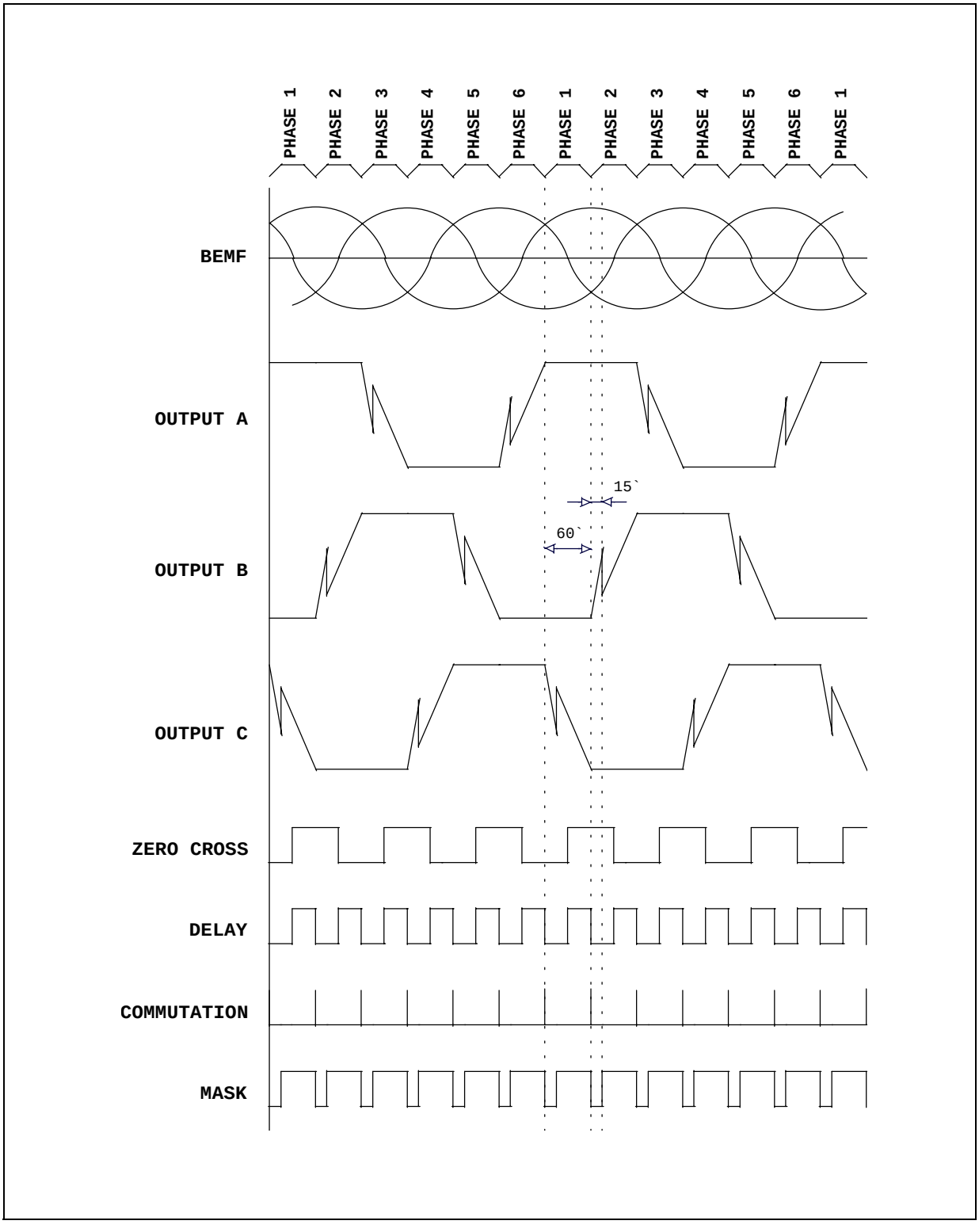
Figure 11. BEMF Amplifier



4.6 DELAY, COMMUTATION & MASK

The Figure #12, represent the internal signal timing associated with the Motor Bemf and Output stage.

Figure 12. Internal Signals Timing



The BEMF waveform is shown as a reference along with a dashed line to indicate the occurrence of a zero crossing. A typical sequence starts when the outputs switch states. Referring to Figure #12, during Phase 1,

output A goes high, while output B is low. During this phase, output C is floating and the Bemf is monitored. The outputs remain in this state for 60 electrical degrees as indicated by the first set of dashed lines. After this period the output switches to phase 2 with output A high and C low with the Bemf amplifier monitoring output B.

In order to prevent commutation current noise being detected as a false zero crossing, a **Masking** circuit automatically blanks out all incoming signals as soon as a zero crossing is detected. When the next commutation occurs an internal counter starts counting down to set the time that the masking pulse remains. This time is equal to 7.5° or 15° of the electrical cycle and it can be programmed through the serial port by setting or resetting the Reg#3.0. Thus the actual masking period is the total of the time from the detected zero crossing to the commutation plus 12.5% (7.5°) or 25% (15°) of the previous period. The selection of the Mask has to be done in order to cover (as shown in Figure #12, between dashed lines) the spike produced by the commutation. This can be sperimentally achieved by looking with an oscilloscope the phase commutation Voltage during start-up (worse case condition due to high current). The time of the Mask signal has to be large enough to cover the duration of the commutation spike.

After the masking period, the Bemf voltage at output B is monitored for a zero crossing. Upon detection of the zero crossing the output is commutated after a **Delay** (in case of Figure #12, 30 electrical degrees) insuring maximum torque. The Delay can be programmed through the serial port by setting Reg#3.4:7 bits. The adjustment range is from 1.875 through 30 electrical degrees in 1.875 degree increments. The Delay has to be chosen in order to optimize the commutation point to insure maximum torque. This can be sperimentally achieved by looking with an oscilloscope the Voltage and the Current in a phase and adjusting the delay in order to have those waveforms as much symmetric as possible.

4.7 FCOM SIGNAL

The FCOM signal available on pin#1 has two different functions. Normally when the motor is running it will provide feedback information of the rotor's position (Zero Crossing, Electrical or Mechanical). During Spindle Inductive sense Start-Up, it will provide informations whether the current threshold is reached or not. In this last case, if Spindle Linear current control is used the FCOM pin will start low and it will go high when the current in the sense resistor will reach the threshold set by IL0,IL1 and ISNS bits (Reg#8.3.4.5) . The FCOM pin will stay high until the Spindle outputs are disabled then it will go low again. If Spindle PWM current control is used, the FCOM will behave like in Linear mode but instead to stay all the time high, it will start chopping with 50% duty cycle and the half period will be equal to the TOFF time set by the external components attached on PWM/SLEW pin (#3). The FCOM pin will continue to chop until the Spindle outputs are disabled then it will go low. The table #6 shows the options of the FCOM output signal available by the combination of the ISNS (Reg#8.3), FLL_OUT(Reg#10.4) and MECH/ELEC(Reg#2.5) bits.

Table 6. FCOM Output Signal

ISNS	FLL_OUT	MECH/ELEC	FCOM SIGNAL
0	0	0	Toggle every Zero Crossing
0	1	0	Toggle every Mechanical turn
0	1	1	Toggle every Electrical turn
1	X	X	Inductive Sense

4.8 BEMF RECTIFICATION

In case of a power down, at the falling edge of the POR signal, the spindle outputs will be tristated and the retract sequence will initiate. The energy for the retract function in power down condition comes from the rectified BEMF of the motor acting as a generator on VDD. Because of the bottom spindle transistors are commutated in sync with the BEMF, the voltage on VDD is only one diode drop less than the BEMF voltage (the body diode of the top side spindle drivers).

4.9 POWER STAGE

The Spindle Power driver consist of a 3 Phase H bridge power transistor. Both the low and the high side drivers are NMOS transistor. The drivers are built by a special low Rdson DMOS structure. The 12 volts power supply is connected to the H bridge via an external power NMOS device. The H bridge returns to ground through the sink pins and via an external resistor. An on chip Boost voltage generator is being used to switch the high side drivers and the external isolation NMOS as explained in chapter 2 (Auxiliary circuits), section 2.3 (Charge Pump).

4.10 EXTERNAL ISOFET

An external NMOS device is used to connect the Power Supply to the Spindle and VCM high side Dmos. The meaning of this device is to isolate the Spindle and VCM power stage from the power supply during retract at Power Off. This will insure the Bemf voltage available from the Spindle motor to stay inside the chip and thus providing the necessary voltage to retract the Voice Coil. Without this device, the Bemf voltage will be discharged very fast through the power supply. The external Isofet is turned on upon the POR signal goes HIGH and is turned off when the POR signal goes low. The high voltage (~17V) to apply to the gate necessary to turn it on, it is provided by the charge pump circuit and it is available on SW1 pin (33). If no particular care is needed on the voltage drop of the external ISOFET, it can be substituted with a normal Schottky diode.

4.11 EXTERNAL BEMF PROCESSING OPTION

The external Bemf processing is available by re-setting (=0) the Reg#2.1 bit. At this time the FCOM signal has to be monitored by the external µP as a zero crossing reference. Upon the transition of the FCOM signal, a sequencer commutation needs to be programmed after a delay calculated by the µP itself. The Delay has to be calculated as a percentage of the previous period (typical 30 electrical degree equal to half of the previous period). The Mask time like in internal Bemf processing is also calculated as a percentage of the previous period, but it is also performed by the high state of the commutation signal INCRE_SEQ (Reg#2.0 bit). When the INCRE_SEQ bit is set (=1) the bemf is masked. The actual mask time signal is a combination of the two. The Figure #19 in the chapter 6 (Application and Tools), section 6.1 (application configurations) depicts this feature.

4.12 EXTERNAL SPEED CONTROL OPTION

The external Speed Control is available through the FLL_FILTER pin (21). By removing the components attached to this pin, the control of the input of the spindle OTA can be reached by an external analog voltage signal (see Figure #10). The feedback frequency for the external FLL/PLL can be taken from the FCOM pin or from the data read off of the disk itself. During Start-up, the OTA input is internally forced high until the Speed reach the value set by the internal FLL counters, then the control is ended over to the charge pump leaving the FLL_FILTER pin at high impedance. For this reason if the external speed control is chosen, the FLL registers has to be set to a lower speed than the target one. Another way to perform the external speed control is through the serial port. This feature can be accomplished by turning ON and OFF the CPH and CPL bits (Reg#8.6.7, see Figure #10). In this case the external filter components are to be connected. The Figure #20 in the chapter 6 (Application & Tools), section 6.1 (application configurations) depicts this feature.

5.0 VOICE COIL CIRCUITS

5.1 DESCRIPTION

A complete Voice Coil control circuit plus 14 bit DAC and DMOS full bridge drivers are integrated in the chip. The class AB power stage which do not require snubber networks for load compensation, yet maintaining high accuracy on gain and offset parameter, allows a well controlled quiescent current, therefore maximum precision in the positioning of the head actuator. The reference voltage for the system is VDD/2 to allow symmetrical use of the available power supply. The DAC command input to the actuator is a voltage centered around VDD/2 that swings ±1V.

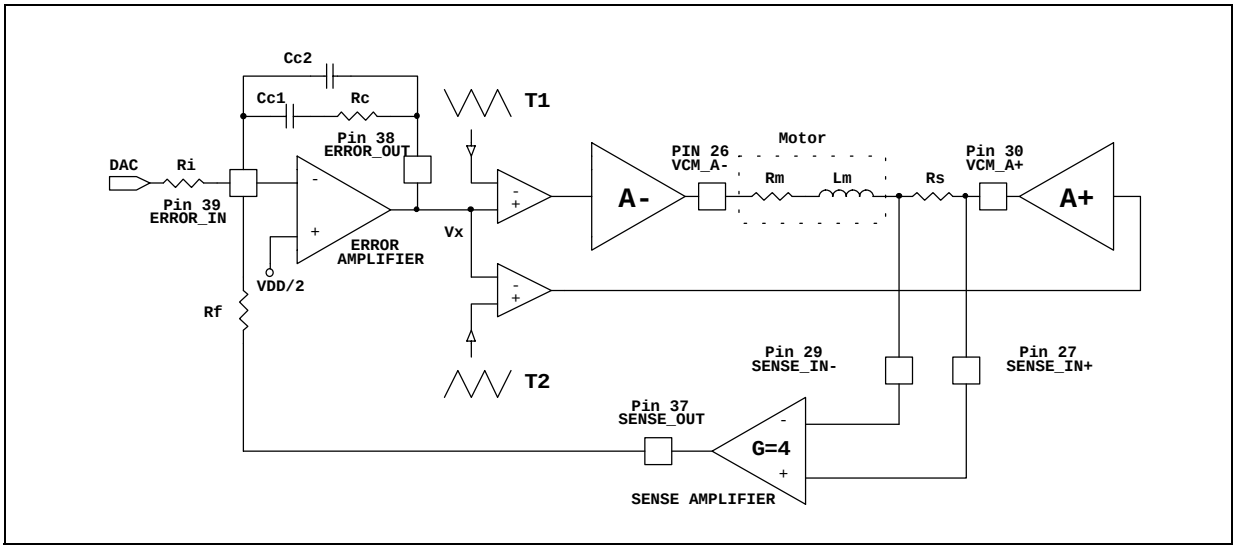
5.2 CURRENT CONTROL

The current in the actuator is sensed by a small resistor (typ. 250mΩ) in series to the coil. The voltage drop across the resistor is amplified by a differential amplifier with a gain of four. The output voltage of this amplifier is proportional to the current in the coil. This voltage and the DAC voltage summed at the input of the error amplifier, represent the difference between the desired current and the actual motor current. The output of the error amplifier is the signal that will drive the power stage according to the driving mode selected. Two different modes are available, Linear mode and Phase Shift Modulation
Mode (PSM). By changing the Reg#0.6 bit, the selection of the two modes can be done any time regardless the Voice Coil motor is moving or not.
The Model and a Mathcad analysis for the VCM Current Control Loop can be found in chapter 8 (Appendix), section 8.1.3 (Voice Coil Current Control loop).

5.2.1 Linear Control Loop

The Linear control loop is selected by a '0' in the Reg#0.6 bit. The complete system is described in Figure #13.

Figure 13. Voice Coil Linear Control Loop



The loop consists of a 14 bit DAC, error amplifier referenced to VDD/2, fixed gain (A=4) differential input current sense amplifier and positive and negative power output amplifiers (referenced to VDD/2). The output of the current sense amplifier (which is referenced to VDD/2) and the output of the DAC are summed with external resistors at the input of the error amplifier. The error amplifier allows the bandwidth of the VCM to be compensated with an external RC filter and drives the positive and negative power amplifiers. The transconductance of the

loop formed by the error amplifier, power amplifiers and current sense amplifier is set by 4 times the value of the external current sense resistor (in series with the VCM), and the ratio of the two summing resistors Ri and Rf. The open loop bandwidth of the Error

amplifier is 10MHz with a gain of 100dB. The open loop bandwidth of the Power Stage is 1MHz for A+ and 1MHz for A- with a gain of +8 for A+ and -8 for A-. Therefore the total gain of the power stage is 16. The open loop bandwidth of the Sense amplifier is 10MHz and the gain is set to 4.

If the Capacitor Cc2 (Figure #13) is choosen to be 10 times smaller than Cc1, the pole created by Cc2 is further reason the Cc2 capacitor will not be included in the Transfer Function and the equation can be simplified as follow.

- 13. $$\frac{I_o}{V_i} = \frac{R_f}{GRiRs} \cdot \frac{1}{s\left(\frac{RfCc1(Rs+Rm)}{2AGRs}+1\right)}$$ VCM Loop Transfer Function

The single Pole response is independent from Rc (if set $Cc1Rc = \frac{Lm}{Rs+Rm}$).

The Bandwidth is set with Cc1:

- 14. $$Wo = \frac{2AGRs}{Rf(Rs+Rm)Cc1}$$ VCM loop Bandwidth

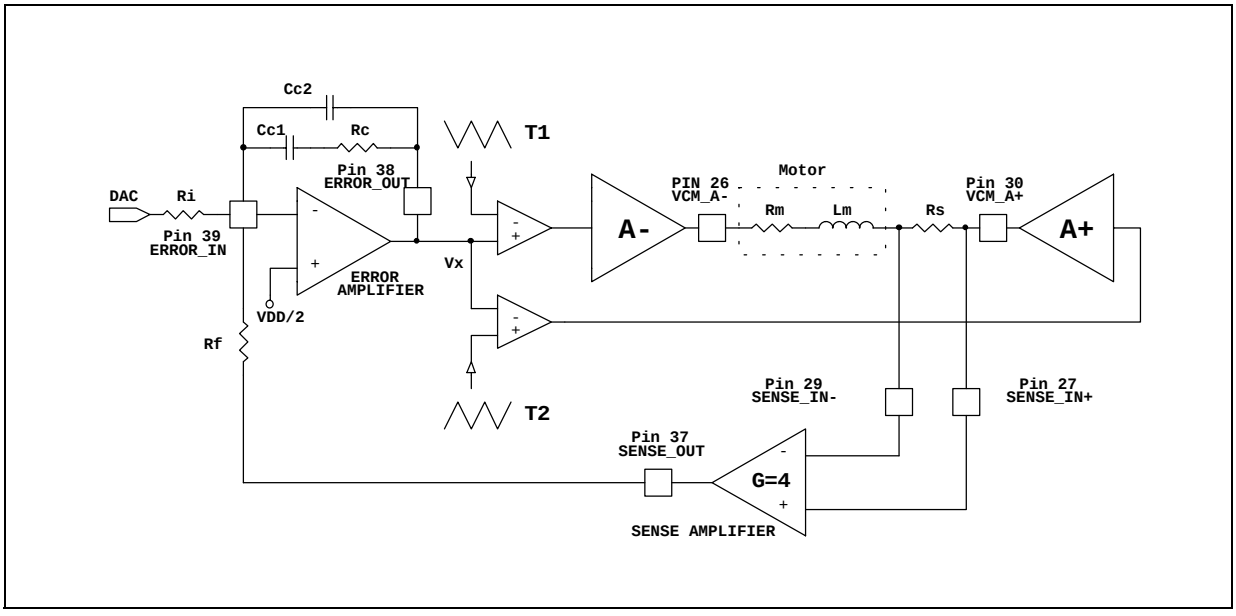
- 15. $$\frac{Io}{Vi} = \frac{Rf}{RiGRs}$$ VCM loop low frequency gain

5.2.2 PSM Control Loop

The PSM control loop is selected by a '1' in the Reg#0.6 bit. The complete system is described in Figure #14.

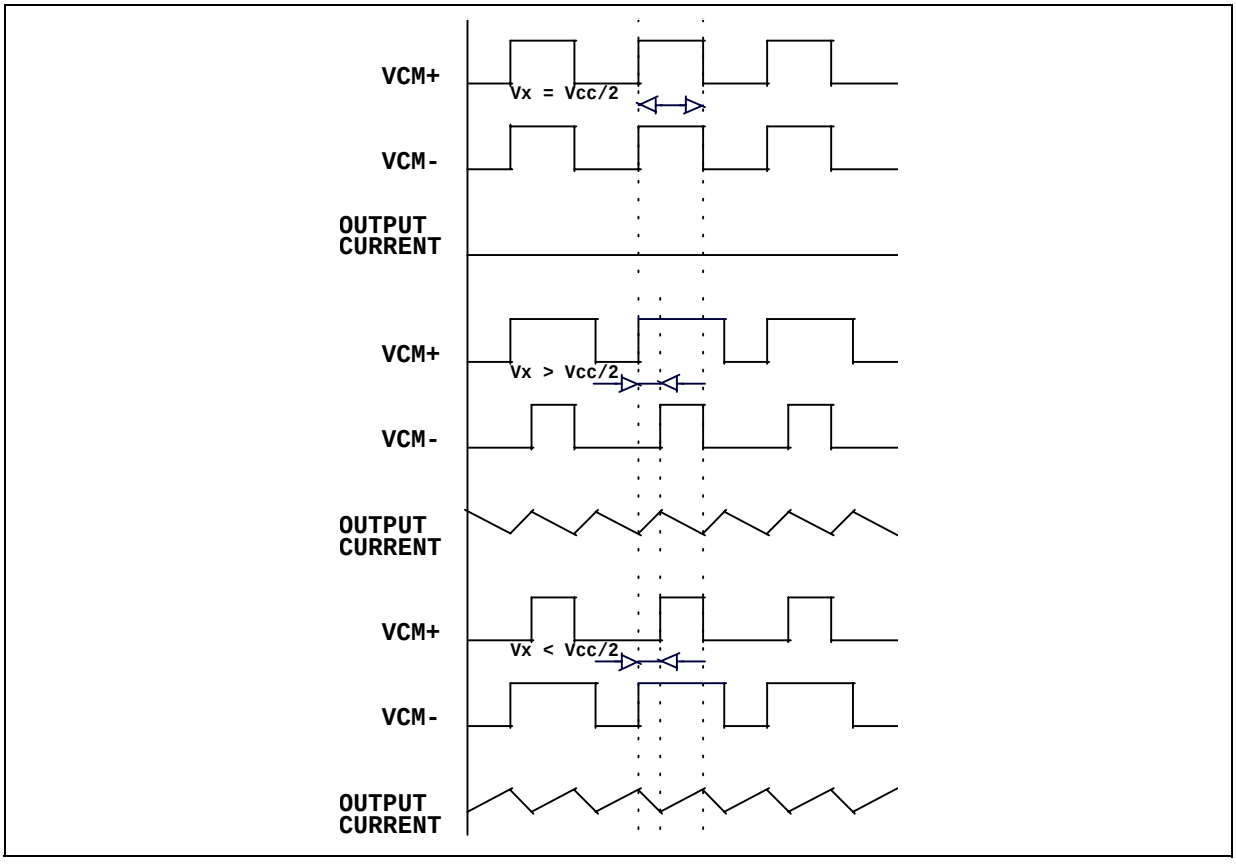
The architecture of the current sense, error amplifier and control voltage (from DAC) are the same as for the linear mode of operation. However the output of the error amplifier (Vx) is input to two comparators which switch the VCM+ and VCM- outputs ON and OFF.

Figure 14. Voice Coil PSM Control Loop



The second input of the two comparators is one of the two anti-phase triangle waves whose 50% voltage is VDD/2. As shown in Figure #15, when the Vx signal is equal to VDD/2, the outputs voltage VCM+ and VCM- are in phase (both high or both low) resulting zero current in the coil. When the Vx signal is grater or less than VDD/2, the duty cycle of VCM+ and VCM- changes (if one increase the other decrease) resulting a positive or negative current into the coil.

Figure 15. Voice Coil PSM Output Voltages & Current Signals



The amplitude of the anti-phase triangle waves (T1,T2) has been chosen to keep the effective gain from Vx to the output the same as in linear mode. This minimize transient DC shift in the control loop voltages when switching between PSM and Linear modes. The frequency of the triangle waves is generated by an internal oscillator and its value is about 50KHz +/- 30%. As can be noticed in Figure #15, the effective chopping frequency is twice the frequency of the oscillator.

5.2.3 D/A converter

A 14 bit DAC is available through the serial port in the Reg#0.0:5 (Vcm Dac High) and Reg#1.0:7 (Vcm DAC Low). It is required to write on register #1 to make effective changes on register #0. This means that if a change is done to a bit in a low VCM_DAC register (Bit 0 through Bit 7), only the Register #1 needs to be written. If a change is done to a bit in a high VCM_DAC register (Bit 8 through Bit 13), both register #0 (first) and register #1 need to be written.

The ladder of the DAC and the DAC circuit itself, are connected internally to a floating power supply in which a feedback system is keeping the supply centered to VDD/2. This is done mainly to compensate the variation of the power supply.

The output of the DAC swings from VDD/2 + 1V (DAC set as Hex.4000 or Dec.16384) and VDD/2 -1V (DAC set

as Hex.0 or Dec.0). The zero current is set as VDD/2 (DAC set as Hex.2000 or Dec.8192). The resolution voltage (per LSB) is given by the full scale voltage (2V) divided by 14Bit (16384) and it is equal to 122µV.

5.3 RETRACT

The retract profile is set by the “PKV_1”, ”PKV_2“ bits (Reg #9.0.1), for Parking Voltage and ” RT0”, “RT1” bits for Parking Time (Reg #9.3.6) in the serial port. Because these bits are not reset by PORB, they must be configured through the serial port after power-up. When programming Retract Time, the Reg#8.2 bit must be always set to 0. Tables #7 and #8 show respectively the Retract Voltage and Retract Time options.

Table 7. VCM Retract Voltage

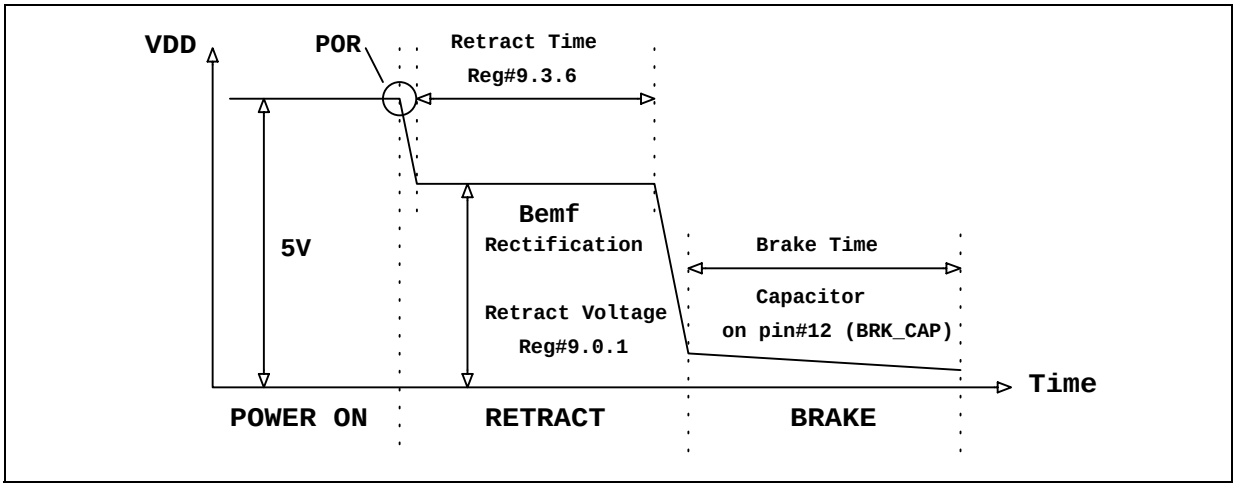
PKV_1	PKV_2	Retract Voltage
0	0	0.850V
0	1	0.650V
1	0	1.600V
1	1	1.150V

Table 8. VCM Retract Time

RT0	RT1	Retract Time
0	0	160ms
0	1	320ms
1	0	80ms
1	1	160ms

The retract sequence is initiated by a falling edge of PORB or by a “0” to ”1” transition on the ”RETRACT” bit (Reg #9.7). Figure #16 shows the Vcm Retract and Spindle Brake sequence at power off.

Figure 16. Power Off Sequence



The device offers the possibility to Tristate or Brake the Voice Coil outputs for half of the Retract time selected. This allows more flexibility to the Retract function. The TRISTATE and BRAKE bits are available respectively inside Reg#11.5 and Reg#11.6.

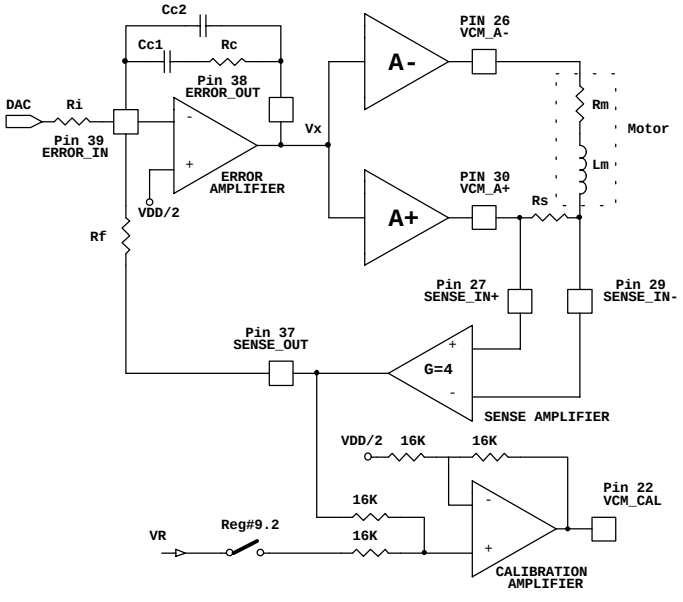
5.4 OFFSET CALIBRATION

A calibration mode can be enabled through the serial port by setting VCM_CAL bit (Reg#0.7) which allows the user to determine the offset of the Sense Amplifier. This is accomplished by enabling all of the VCM control circuits except for the output drivers. This means that offset of the Isense amplifier can be read directly on the VCM_CAL pin (#22). The output offsets is calculated by programming zero current in the normal operation mode (VCM_CAL bit=0) and measuring the output of the VCM_CAL. By adjusting the DAC bits, monitoring in the mean time the VCM_CAL pin until the voltage on that pin reach the same value previously measured (when the VCM_CAL bit was set (=1)), the DAC value obtained will represent the real zero current reference. The VR bit (Reg #9.2) needs to be set prior to start the offset calibration. This allow to reference the calibration amplifier to a very precise voltage (2V). The voltage on VCM_CAL pin during calibration is 2V +/- offset. Figure #17 shows the Vcm Calibration circuit.

Following is the step by step procedure to calibrate the Voice Coil offset.

1) Enable Voice Coil Circuits by setting VCM_EN bit (=1) (Reg#9.5).
2) Apply a reference voltage to the Calibration Amplifier by setting VR bit (=1) (Reg#9.2).
3) Set VCM_CAL bit (=1) (Reg#0.7). Control circuits are ON but power drivers are OFF.
4) Use an external ADC to measure the Voltage on pin VCM_CAL as **Voffset**.
5) Reset VCM_CAL bit (=0) (Reg#0.7).
6) Adjust DAC output through VCM DAC register (Reg#0 and Reg#1) until **V(VCM_CAL) =Voffset**.
7) Save the DAC value as a zero current reference.

Figure 17. Voice Coil Offset Calibration



5.5 POWER STAGE

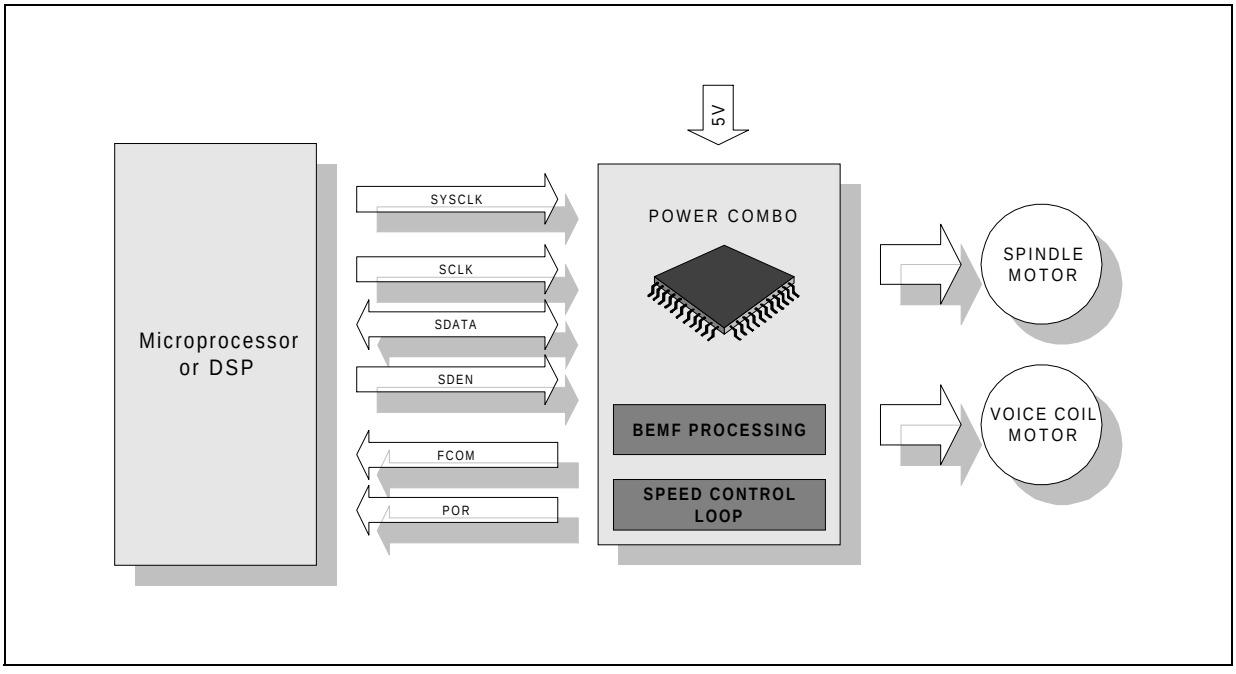
The Voice Coil is a class AB, H bridge type power driver with all power devices internal to the device. The H bridge consist of four NMOS power transistor built by a special low Rdson DMOS power structure. The 12 volts power supply is connected to the H bridge via an external power NMOS device. The H bridge returns to ground via VCM_GND pin (28).

6.0 APPLICATION & TOOLS

6.1 APPLICATION CONFIGURATIONS

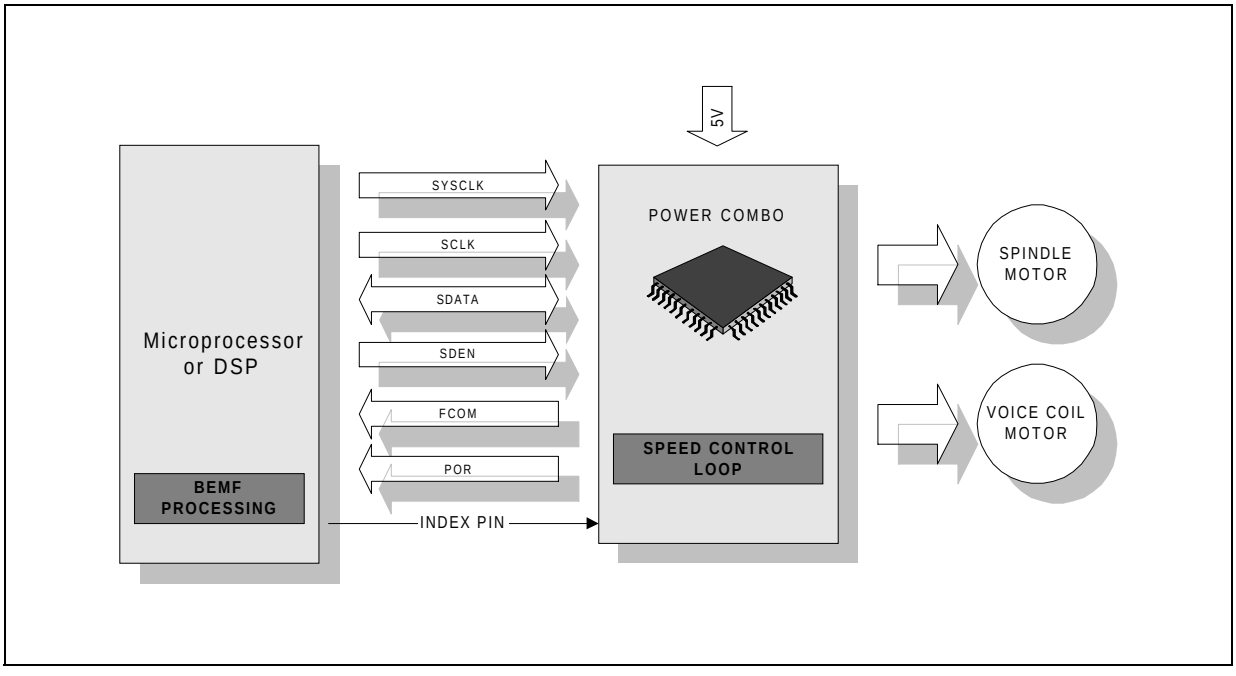
Altought the Device offers the complete BEMF processing and Speed Control Loop inside, the possibility may exists where certain applications might require complete microprocessor control of those functions. The flexible architecture of this device give the µP the option to externally provide those features. The figure #18 depicts the Stand Alone configuration where the Bemf processing and Speed Control Loop are demanded to the Power Combo. In this case the µP has to provide the SYSCLK frequency and POR signal has to be monitored for under voltage detection. FCOM and the status register may also be monitored for fault condition. The registers have to be programmed once according to the desired bits configuration.

Figure 18. Stand Alone Application Configuration



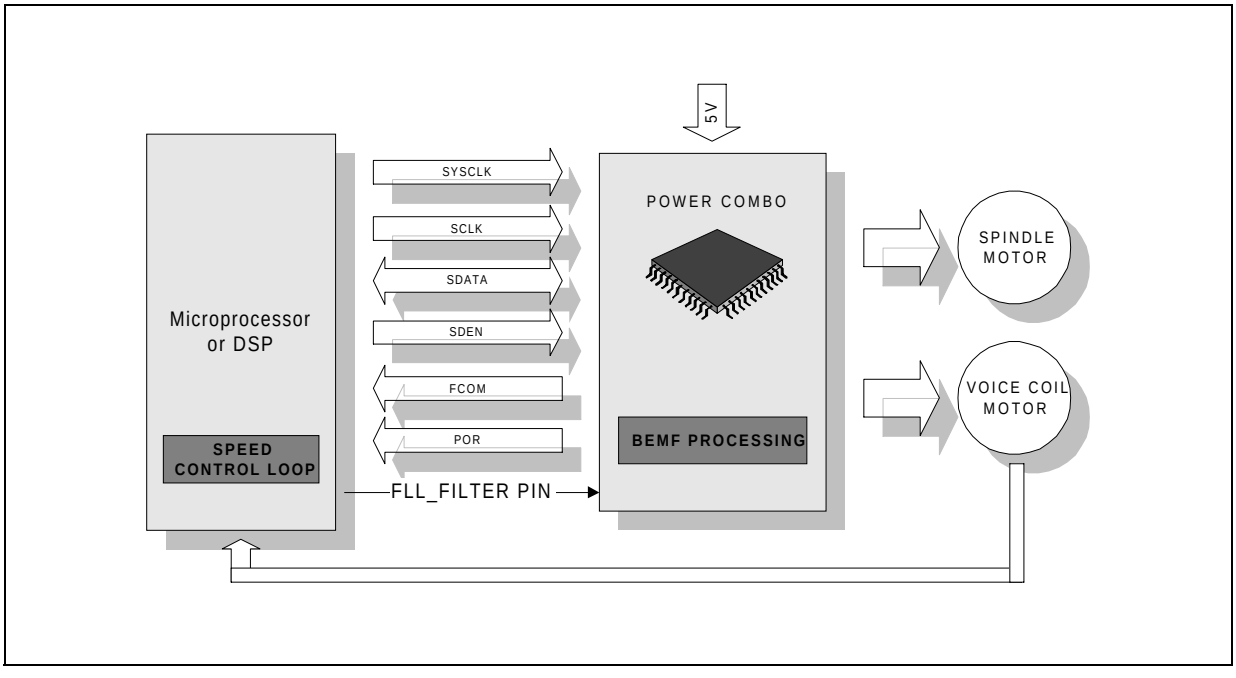
The Figure #19, shows the External Bemf processing configuration. In this case the FCOM pin has to be continuously monitored and the commutation signal has to be provided to INDEX pin (#11). An explanation of this configuration setup can be found in chapter 4 (Spindle Circuits), section 4.11 (External BEMF Processing option).

Figure 19. External Bemf Processing Configuration



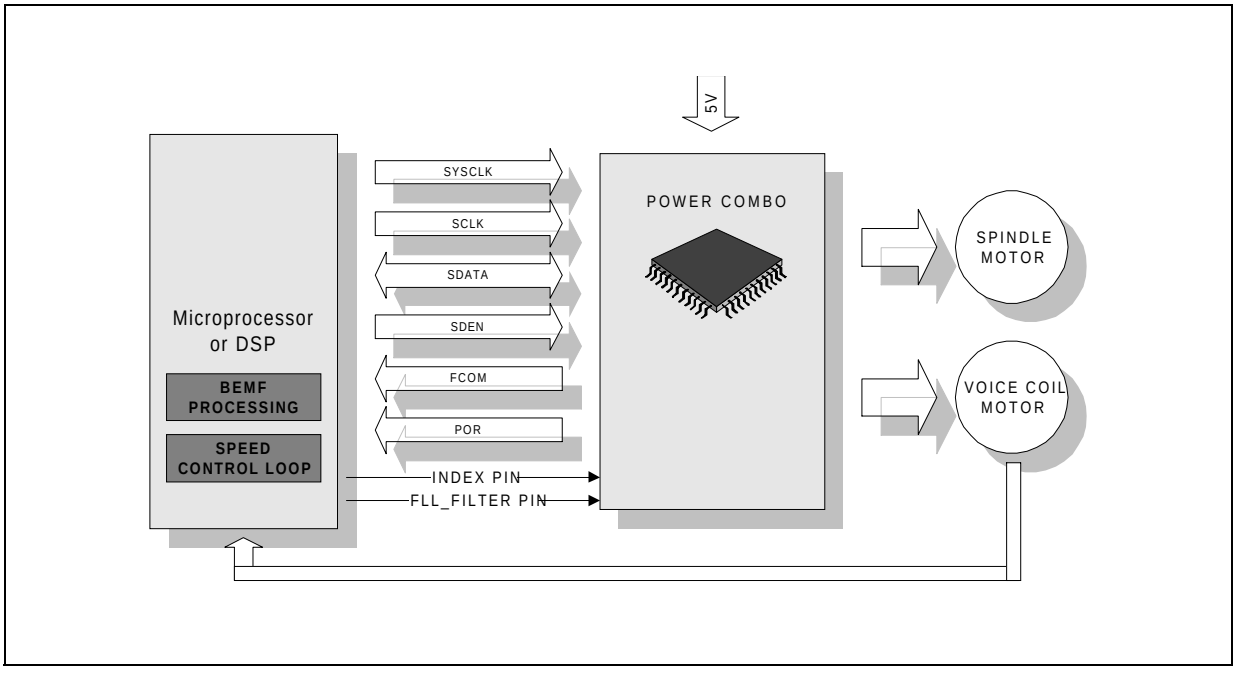
The Figure #20, shows the External Speed control configuration. The Feedback frequency is taken from the disk itself and the control voltage is entered on FLL_FILTER pin (#21). An explanation of this configuration setup can be found in chapter 4 (Spindle Circuits), section 4.12 (External Speed Control option).

Figure 20. External Speed Control Configuration



The Figure #21, shows both the External BEMF processing and the Speed control together in the same configurations. This option is the combination of the two features and it is illustrated in the explanation of the previous Figures #19 and Figure #20.

Figure 21. External BEMF Processing & Speed Control Configuration



6.2 TYPICAL APPLICATION

In Figure #22 is depicted a typical application circuit. The external components are selected to work with a 5400Rpm Spindle motor with inductance Lm=1.2mH phase to phase and resistor Rm=5.3Ω phase to phase. The Voice Coil is assumed with an inductance Lm=1.5mH and a resistor Rm=13.3Ω.

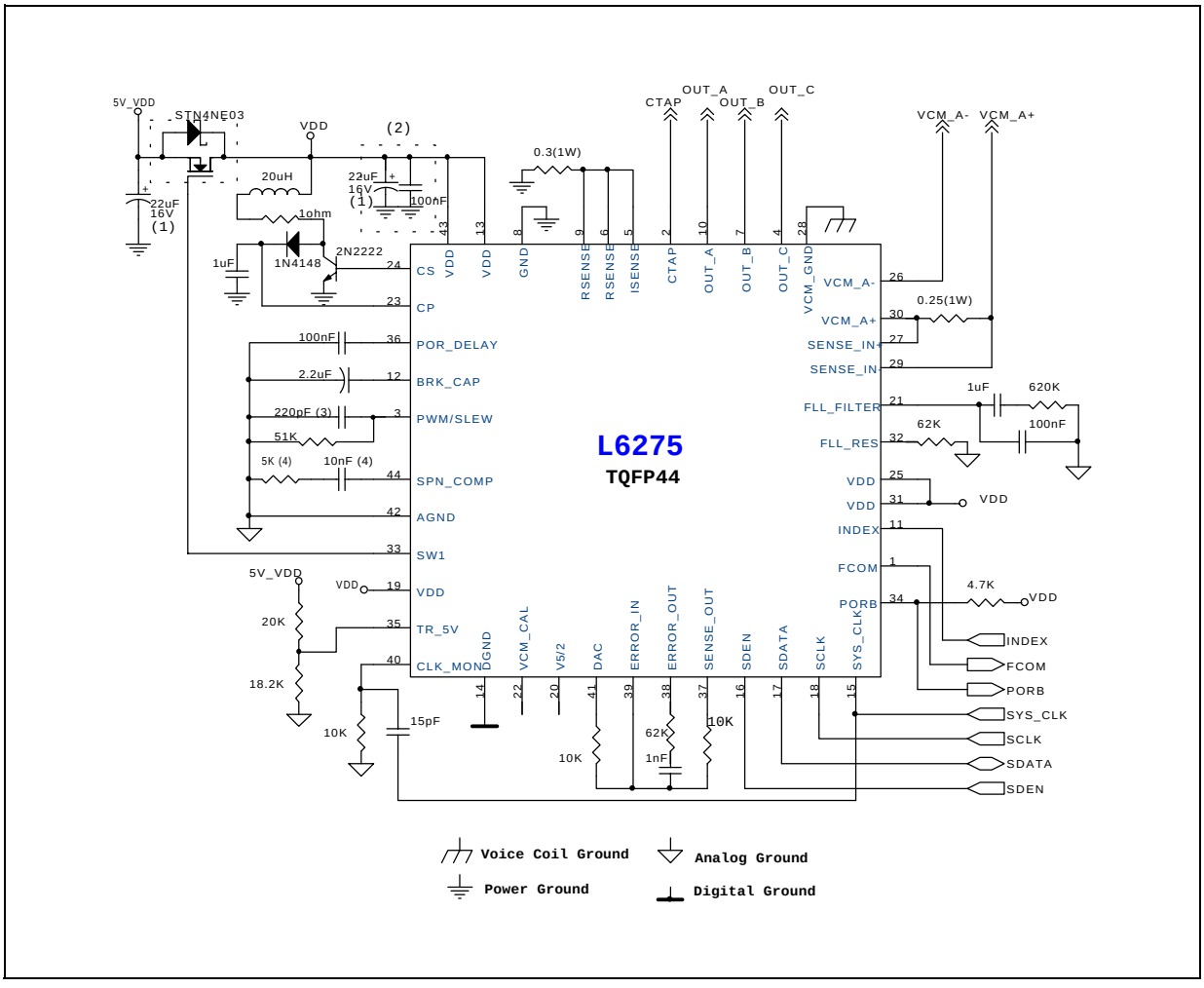
The maximum Spindle Start-Up current is set as VDD / (Rm+2rdson+Rsense) and it is equal to 1.85A with Reg#8.4.5 bits set to 1. The VDD is equal to 5V. The maximum Voice Coil current is set to be 1A.

Since high current is demanded through the Spindle and Voice Coil sense resistor, 1W power dissipation capacity is recommended.

If Spindle Linear current control loop is selected, the capacitor (3), attached to PWM/SLEW pin (#3) can be avoided. If Spindle PWM current control loop is selected, the components, one resistor and one capacitor (4), attached to SPIN_COMP pin (#44) can be avoided.

The layers that goes from SENSE_IN+ and SENSE_IN- pins to the Voice Coil sense resistor must be at equal distance. A capacitor (1) connected to VDD and Ground is required in order to clamp possible spike due to the current recirculation in the Spindle windings. Since the absolute maximum rating of the VDD (pin #13) line is 6V, precautions need to be taken in order not to allow the voltage on this pin to go above the limits. A recommended capacitor on VDD pin for a 5400Rpm, two platters application is 22µF Tantalum type. This capacitor needs to be placed as closest as possible to the VDD pin (#13). Also an under voltage on VDD can be very. This can happen for example during plug and unplug test of the drive power supply connector. It is indeed recommended to use a protection circuit in order to better protect the Supply lines. A good layout by separating the grounds will improve the noise immunity of the system.

Figure 22. Typical Application Circuit



6.3 SMART POWER DEVELOPMENT SYSTEM (SPDS)

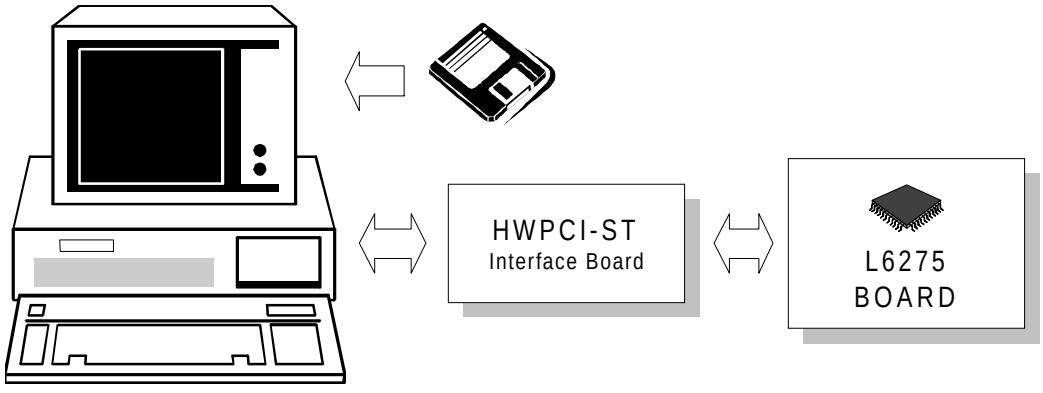
The STMicroelectronics Smart Power development System is a family of hardware and software tools that are used with a personal computer to develop applications of smart power ICs. Using this tool engineers can quickly evaluate smart power ICs and optimize their application designs obtaining the best possible performance and reliability in the shortest time.

The SPDS is based on a IBM PC AT compatible, as shown in Figure #23. A PC interface board, suitable for any data transfer operation from or to the PC, is inserted into one PC ISA expansion slot and interface the computer with the external devices.

In addition, an evaluation board specific for one or more STMicroelectronics products, controlled by the SPDS software installed on the PC, provide the direct user application management.

Designed for one or more specific STMicroelectronics Smart Power products, the SPDS software provides a set of tools dedicated to the design development and optimization of applications which make use of the STMicroelectronics Smart Power ICs. SPDS software permits the evaluation and optimization of external components. SPDS software support the SPDS hardware and permits the driving of the application to test, modify and optimize in real time the application parameters.

Figure 23. Smart Power Development System Setup



The SPDS tool for the devices described in this application note is available through the STMicroelectronics representative.

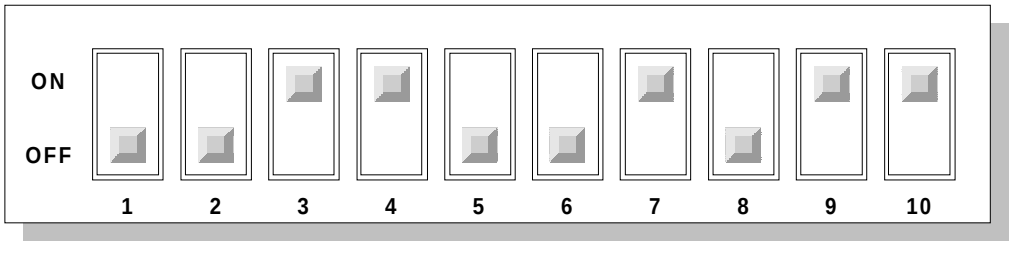
6.3.1 SPDS Hardware

The SPDS hardware consist of two boards, PC interface board (HWPCI-ST) and application board. The interface board provides a programmable interface between the PC and the external devices. Inserted into one PC ISA expansion slot , the board is memory-mapped: the PC reads/writes the expansion board as a common memory location.
The interface board contains 4 eight-bit digital ports which can be programmed as outputs and/or inputs; a timing section consisting of four general purpose counting/timing channels and one programmable rate generator channel is controlled by one 8MHz on-board crystal clock.
The interface board is connected to the application board (external to the PC) with two flat cables. The application board is designed specifically for the devices depicted in this manual and provides all the external components necessary to the device itself. The Figure #25 and Figure #26 show the connection between interface and application boards.

6.3.1.1 Interface Board

The interface card has a switch that defines the PC memory location in which it is allocated. The switch has to be set at HexCD00 address as explained in the Figure #24.

Figure 24. SPDS Interface Board Memory Address, Dip Switch Setting



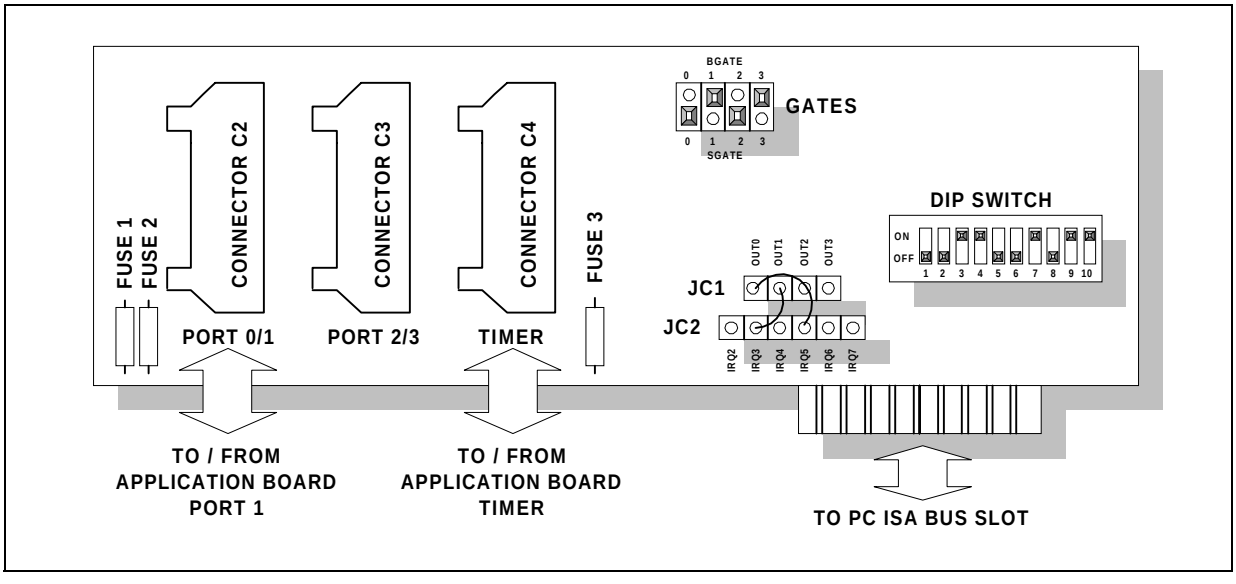
The OUT0 pin of the JC1 jumper must be connected to IRQ5 pin of the JC2 jumper connector. The OUT1 pin of the JC1 jumper must be connected to IRQ3 pin of the JC2 jumper connector. This will allow the outputs of the Timer0 and Timer1 to generate an Interrupt request respectively on IRQ5 and IRQ3 lines of the PC interrupt controller. The TGATE3 pin must be connected to BGATE3 pin of the GATE jumper connector. The TGATE1 pin must be connected to BGATE1 pin of the GATE jumper connector. This setting will allow the application board to control the counting of the Timer1 and Timer3. The Timer 0 and 2 are enabled by software. The Figure #25 shows the connectors and jumpers location. Three connectors along with three flat cables interface the Interface Board with the external Application Board. Connector C2 carries 16 bits input/output data through two 8 bits port, Port 0 and Port 1. Connector C3 also carries 16 bits input/output data through Port 2 and Port 3. Each port is configurable by software as input or output. The connector C3 is not utilized in the application of the devices depicted in this manual.

Connector C4 is the Timer connector and carries the GATE, CLOCK and OUT signal for each of the four Timer available on board. The Rate Generator output, which is programmable, is also available on Connector C4. Three 5V supply lines capable of 1A of current each isolated with fuses (Fuse 1, Fuse 2 and Fuse 3) are available on pin #34 of each cables.

A detailed explanation of the Interface card is available in a separate manual inside the SPDS tool box.

Important!!! The SPDS makes use of the IRQ3 and IRQ5 interrupt lines. If another board inside the PC is using the same lines, it is necessary to remove it to avoid the Interrupt Hardware conflit .

Figure 25. SPDS Interface Board, Connectors And Jumpers Location



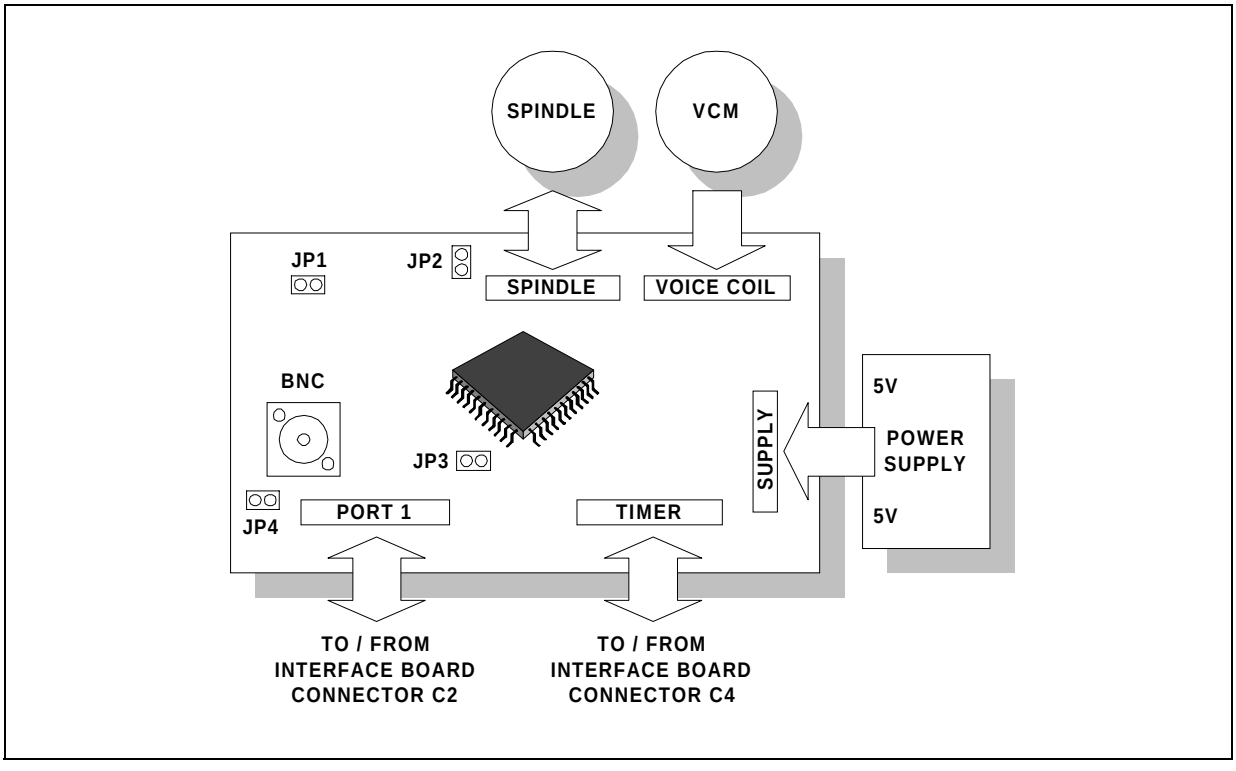
6.3.1.2 Application Board

The application board is connected to the interface board with two flat cables. The Port 0/1 cable, which provides all the digital input output signals, is connected between Port 1 connector of the application board and connector C2 of the interface board. The Timer cable, which provides the timings, is connected between Timer Connector of the application board and connector C4 of the interface board.

Two 5V supply lines available on pin #34 of Port 1 and Timer cables supply some of the Application board ICs. For this reason, please make sure that the Fuses mounted on the Interface Board are working.

An oscillator running at 20MHz required by the Combo IC is available on board. A different frequency can be input at the BNC connector. The selection between on board and external frequencies is done by software.

Figure 26. SPDS Application Board, Connectors And Jumpers Location



The Supply to the board is provided through the Supply connector. 5V is required. The Spindle and the VCM are connected respectively to the Spindle and Voice Coil connectors. A print on the PCB depicts the phases and polarity connection. Four jumpers are available on board for evaluation purpose.

Jumper JP1 inserts a Schottky diode between the VDD and the ICs in case the ISOFET is not required. JP2 connects the Input of the Sense Amplifier (Isense) to the sense resistor (must be inserted). JP3 connects the VDD power supply to the Voice Coil section (must be inserted). JP4 connects the PC ground to the Application board ground. A separate manual with a detailed schematic and explanation of the Application Board is available inside the SPDS tool box.

6.3.2 SPDS Software

The Smart Power Development System Software is available on a 1.4M floppy disk. The diskette content is a number of files (7) that must be loaded and run from the fixed disk. It is suggested to set up a subdirectory for example “SPDS” and copy all the files into it. The software runs under DOS environment and the following modification needs to be done to the CONFIG.SYS file.

C:\...\EMM386.EXE NOEMS X=CD00-CFC0

FILES=40 (or higher)

BUFFERS=40 (or higher)

A file called README.TXT is include in the diskette, to read it use a common ASCII editor. Following is the directory of the files present on the floppy disk.

ST.BAT	Starts the SPDS program lunching the LOGO.EXE file
LOGO.EXE	Displays the STMicroelectronics logo
MOUSE.EXE	Installs the MOUSE driver

MAINM.EXE SPDS Application program

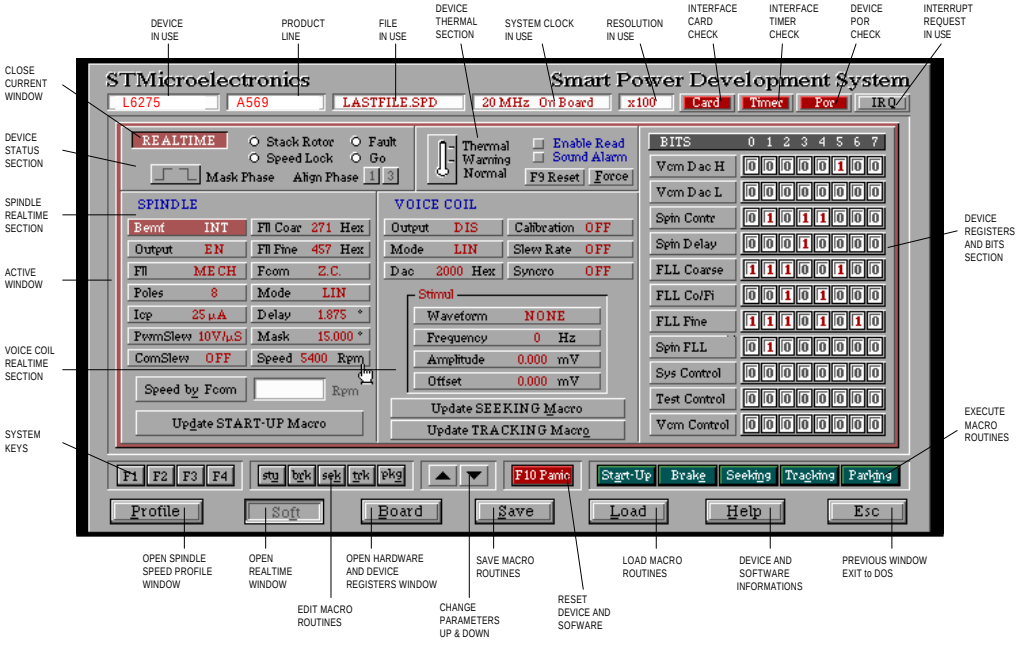
TMSRB.FON SPDS fonts

LASTFILE.SPD Contains SPDS program macro routine

README.TXT Contains SPDS setup information

The source files of the SPDS program are also available upon request, please ask a STMicroelectronics representative. To start the program simply type **ST** and press **Enter**. After the ST logo is displayed, type any key to enter the SPDS environment. Following in Figure #27 is the Real-time frame of the SPDS environment.

Figure 27. SPDS Software Realtime Frame



The SPDS software is completely accessible through a mouse and/or keyboard. Using the mouse, move the pointer to the desired location on the screen. The pointer shape will change from an arrow to a little hand, then press the left button. Using the keyboard, an under lined letter determines the short key to enter the function. Where the under lined letter is not available, use the TAB key to move between the parameters. Use Arrow UP and DOWN keys to change the parameters value. A separate dedicated manual with a detailed explanation of the SPDS software is available inside the SPDS tool box.

7.0 PACKAGE & THERMAL

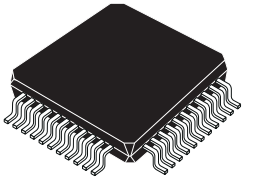
7.1 TQFP44 PACKAGE

The device is mounted in a Thin Quad Flat 44 pin Package (TQFP44). The dimensions of the body are 10x10x1.40mm. The mounting information are outlined in the following Figure #28.

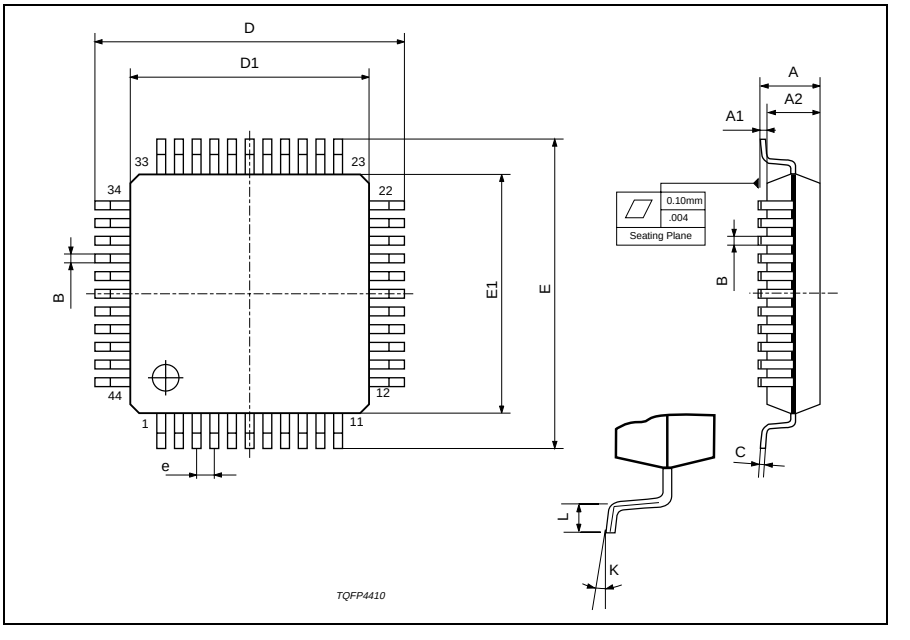
Figure 28. TQFP44 Dimensions

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			1.60			0.063
A1	0.05		0.15	0.002		0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
B	0.30	0.37	0.45	0.012	0.014	0.018
C	0.09		0.20	0.004		0.008
D		12.00			0.472	
D1		10.00			0.394	
D3		8.00			0.315	
e		0.80			0.031	
E		12.00			0.472	
E1		10.00			0.394	
E3		8.00			0.315	
L	0.45	0.60	0.75	0.018	0.024	0.030
L1		1.00			0.039	
K	0°(min.), 3.5°(typ.), 7°(max.)					

OUTLINE AND MECHANICAL DATA



TQFP44 (10 x 10)



7.2 THERMAL CHARACTERISTICS

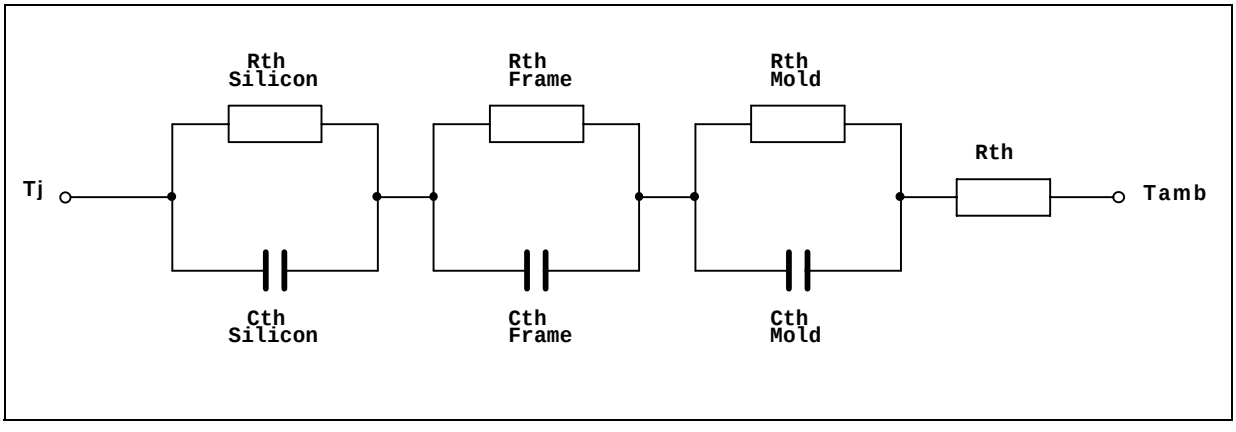
The thermal characteristics are influenced by many factors. These can not be described by a single thermal resistance, in fact a set of experimental curves gives the best presentation. Two thermal resistance are more important, the Junction to Ambient Thermal Resistance, **Rthj-a,** and the Junction to case Thermal Resistance, **Rthj-c.**

Rthj-a, represents the thermal resistance of the system and comprises the silicon die, the package, and any thermal mass in contact with the package to dissipate heat to the ambient. At a given dissipation level Pd, the increase in junction temperature ΔTj over ambient temperature Ta is given by:

ΔTj = Rthj-a * Pd.

Rthj-a is made up for many elements both within the device and external to it. If the device is considered alone, Rthj-a is given by the dissipation path from the silicon die to the lead frame, to the molding compound. The Rthj-a for the TQFP44 package is 40 to 45°C/W. The Figure #29 shows the equivalent thermal circuit simplified.

Figure 29. Equivalent Thermal Circuit Simplified Package



Rthj-c, is the thermal resistance from the junction to a give area of the package's external surface. The junction to top case thermal resistance for this package is 11°C/W. However this value is obtained in an extreme environment where 100% of the heat is dissipated through package top case. In the application environment, only less than 10% of the heat is dissipated through this path. Therefore, Tj should be calculated in the following way:

- 16. $T_j = T_c + 10\% \cdot (P_d \cdot R_{thj-c})$ Junction Temperature

This gives Tj ~ Tc + Pd. Hence the junction temperature is only slightly higher than the case temperature.

Example with:

- Multilayer printed circuit Board: 4 Layers.
- Board size: 10x10cm 'L' shaped.
- Die size: $20mm^2$
- Dissipating area: $20mm^2$
- Spindle Spin-Up: 2W for 3sec.
- Average Power Dissipation. Spindle + Vcm: 0.5W

According to this condition, the Thermal Resistance Junction to Ambient should be around 42-45×°C/W (without any forced air flow). If the board is mounted in some metal enclosure, this can help to remove the heat reducing the Rthj-a to 35-40×°C/W (provided that there is enough space around the package to have an almost

free air convention flow).

In steady state condition (stable running dissipation Spindle + Vcm) the Junction Temperature increase is in the range of 20-25×°C maximum.

For the transient behaviour, up to 3 seconds (spindle Spin-Up) the board dimension are not so important as, within this time, almost all the heat is still inside the package. In this case the key parameters to identify the transient thermal impedance are the die size, the dissipating area, the package geometry. Under the condition of this example, the thermal impedance @3sec is about 18×°C/W. This means tha at 3 seconds of Spin -Up, the temperature increase is between 35 to 40×°C.

In conclusion, the Spin-Up phase induces a temperature increase for the junction of about 35-40×°C; after that when the steady state running will take place the Junction Temperature will go down and will stabilize around 20-25×°C. To have the absolute Junction Temperature, simply add the ambient temperature.

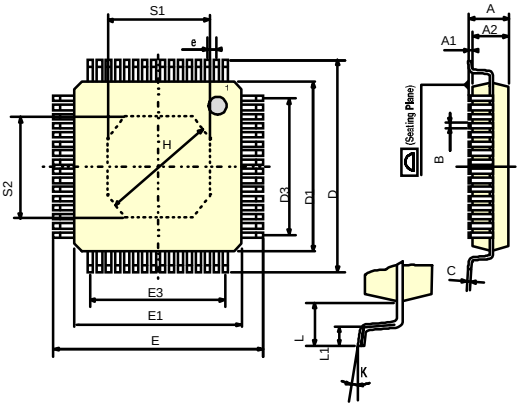
7.3 SLUG

The device provides a piece of metal (SLUG) mounted on the back of the package. This metal, melted in the plastic of the package is designed to handle higher power dissipation by lowering the Junction to ambient thermal resistance. When the slug is soldered to the PCB, the Rthj-a is 30 to 35×°C/W.

Following in Figure #30 is outlined the Slug dimension.

Figure 30. TQFP with SLUG, Package Outline

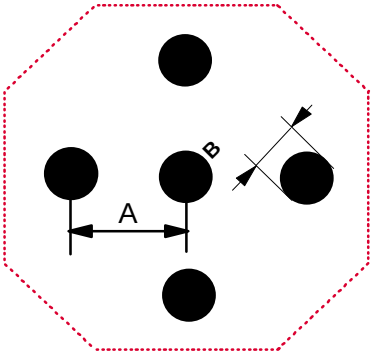
Dimensions	Tolerances (mm)	Lead Count	
		44	64
A	Max.	1.60	
A1	+/-0.05	0.10	
A2	+/-0.05	1.40	
D	+/-0.20	12.00	
D1	+/-0.10	10.00	
D3	Typ.	8.00	7.50
E	+/-0.20	12.00	
E1	+/-0.10	10.00	
E3	Typ.	8.00	7.50
L	+/-0.15	0.60	
L1	Typ.	1.00	
e	Typ.	0.80	0.50
B	+/-0.05	0.35	0.23
C	Max.	0.20	
S1	Typ.	6.00	
S2	Typ.	6.00	
H	Typ.	5.90	
K(deg.)		0°-7°	



The Figure #31 shows rhe suggested pattern to sold the device slug to the drive board. Figure #32 depicts the design and dimension of the Slug mounted on the back of the TQFP44 package.

Figure #33 shows the device temperature rise with and without Slug.

Figure 31. Raccomended Pattern for SLUG SOLDERING TO PCB



Package Sizes	A	B
10x10x1.4mm	2.00 mm	1.00 mm

Figure 32. TQFP SLUG design

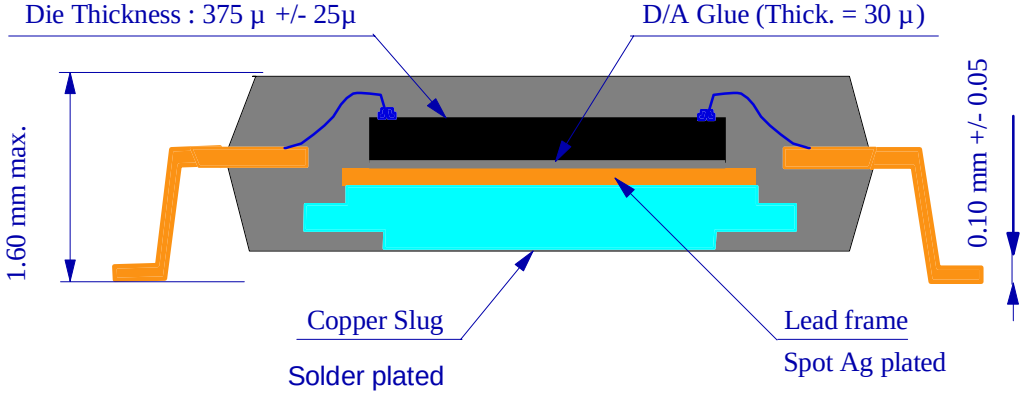
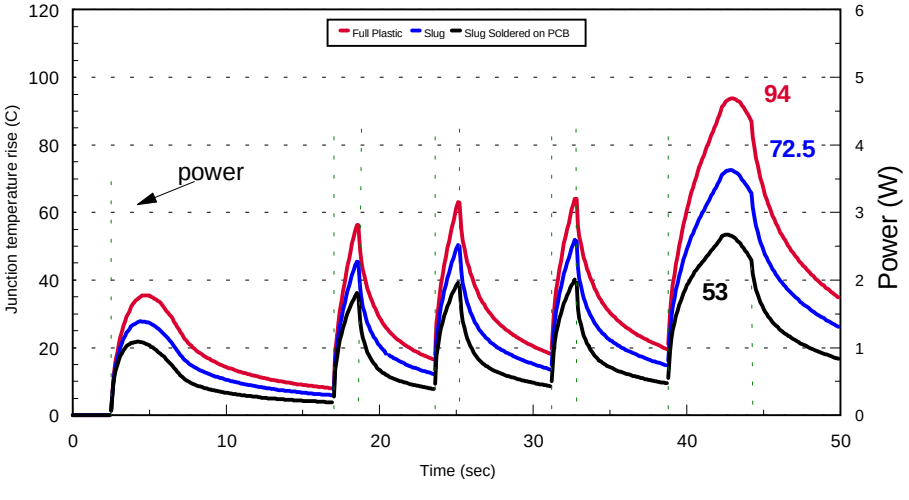


Figure 33. IC Temperature Crise



8.0 APPENDIX

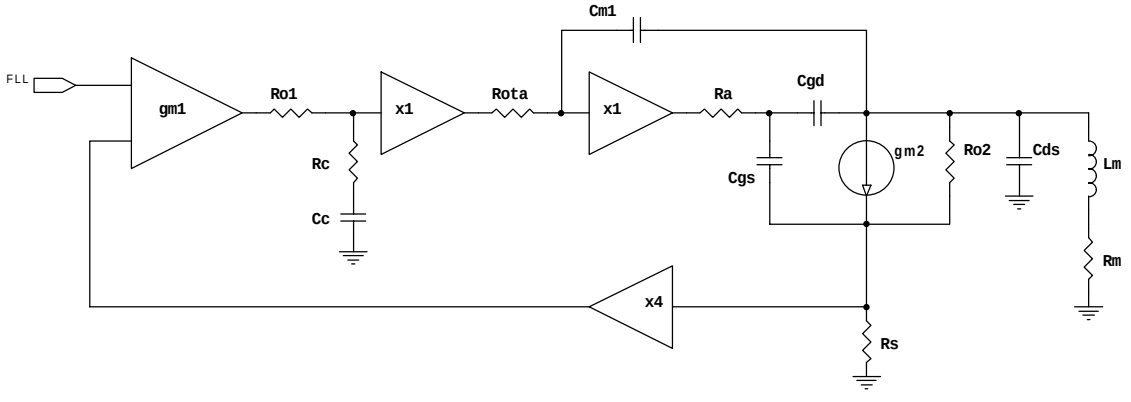
8.1 MODELS & MATHCAD ANALISYS

Following are the models and Mathcad analisys for the Spindle Current control loop, Spindle Speed control loop and Voice Coil Current control Loop. The external component used are referred to the typical application depicted in Figure #22. The 5.0 release of Mathcad has been used and a floppy disk with the three analisys is available through the STMicroelectronics representative.

8.1.1 Spindle Current Control Loop

Figure #34 shows the Spindle Current Control Loop model used for the analisys. Figures 35a, 35b, 35c show the Mathcad analisys of the loop.

Figure 34. Spindle Current Control Loop Model



(User specified parameters)

$Rc := 5 \cdot 10^{3}$ **(External compensation resistance)**

$Cc := 10 \cdot 10^{-9}$ **(External compensation capacitance)**

$Rm := 5.3$ **(Motor's winding resistance - Phase to Phase)**

$Lm := 1.2 \cdot 10^{-3}$ **(Motor's winding inductance - Phase to Phase)**

$Rs := 0.3$ **(External sense resistor)**

(Device parameters)

$gm1 := 0.6 \cdot 10^{-3}$ **(Transconductance of OTA)**

$Ro1 := 5.3 \cdot 10^{6}$ **(Output resistance of OTA)**

$Rota := 10 \cdot 10^{3}$ **(Output resistance of OTA buffer)**

$Cml := 20 \cdot 10^{-12}$ **(Miller capacitor value)**

$Ra := 130$ **(Output resistance of predriver)**

$Id := 110 \cdot 10^{-3}$ **(Coil current)**

$gm2 := 2.7 \cdot \sqrt{Id}$ **(Transconductance of DMOS)** $gm2 = 0.895$

$Ro2 := \frac{5.2}{Id}$ **(Output resistance of DMOS)** $Ro2 = 47.273$

$Cds := 30 \cdot 10^{-12}$ **(Drain-source capacitance of DMOS)**

$Cgd := 12 \cdot 10^{-12}$ **(Gate-drain capacitance of DMOS)**

$Cgs := 30 \cdot 10^{-12}$ **(Gate-source capacitance of DMOS)**

$A := 4$ **(Gain of sense amplifier)**

Figure 35a. Mathcad Analisys of the Spindle Current Control Loop

1ST STAGE (OTA) TRANSFER FUNCTION :

$A1 := gm1 \cdot Ro1$ $\quad A1 = 3.18 \cdot 10^3$

$wz := \frac{1}{Rc \cdot Cc}$ $\quad Fz := \frac{wz}{2 \cdot \pi}$ $\quad Fz = 3.183 \cdot 10^3$

$wp := \frac{1}{Cc \cdot (Rc + Ro1)}$ $\quad Fp := \frac{wp}{2 \cdot \pi}$ $\quad Fp = 3$

$G(s) := A1 \cdot \frac{\left(1 + \frac{s}{wz}\right)}{\left(1 + \frac{s}{wp}\right)}$ **(Transfer function of OTA after compensation)**

2ND STAGE TRANSFER FUNCTION :

$i := \sqrt{-1}$

$f(n) := 10^n$

$S(n) := 2 \cdot i \cdot \pi \cdot 10^n$

$n := 1, 1.01 .. 6$

$ZL(n) := Lm \cdot S(n) + Rm$ **(motor)**

$H1(n) := A1 \cdot \frac{\left(1 + \frac{S(n)}{wz}\right)}{\left(1 + \frac{S(n)}{wp}\right)}$ **(OTA)**

$Ci := Cgs + Cgd$

$Co := Cgd + Cds$

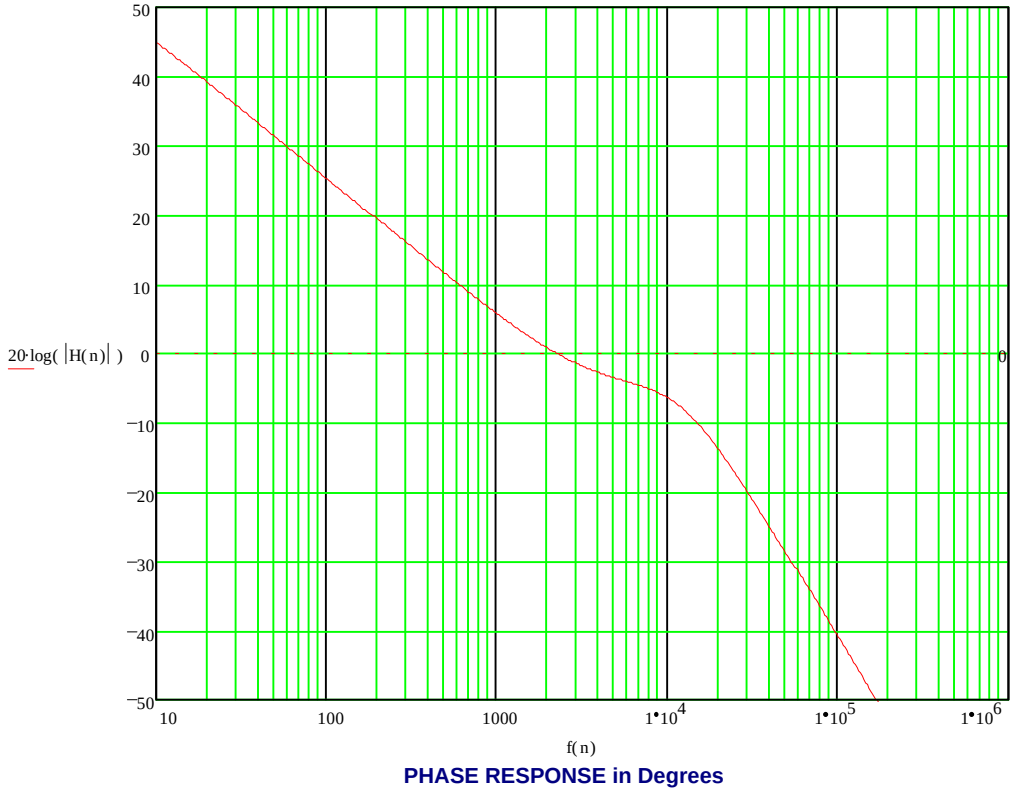
$$H2(n) := \frac{\left\| \begin{bmatrix} \frac{1}{Ra} + Ci \cdot S(n) & -Cgd \cdot S(n) & 0 & -\frac{1}{Ra} \\ -(Cgd \cdot S(n) - gm2) & (Cml + Co) \cdot S(n) + \frac{1}{Ro2} + \frac{1}{ZL(n)} & 0 & -Cml \cdot S(n) \\ -(Cgs \cdot S(n) + gm2) & -\left(\frac{1}{Ro2} + Cds \cdot S(n)\right) & 0 & 0 \\ 0 & -Cml \cdot S(n) & \frac{1}{Rota} & \frac{1}{Rota} + Cml \cdot S(n) \end{bmatrix} \right\|}{\left\| \begin{bmatrix} \frac{1}{Ra} + Ci \cdot S(n) & -Cgd \cdot S(n) & -Cgs \cdot S(n) & -\frac{1}{Ra} \\ -(Cgd \cdot S(n) - gm2) & (Cml + Co) \cdot S(n) + \frac{1}{Ro2} + \frac{1}{ZL(n)} & -\left(gm2 + \frac{1}{Ro2} + Cds \cdot S(n)\right) & -Cml \cdot S(n) \\ -(Cgs \cdot S(n) + gm2) & -\left(\frac{1}{Ro2} + Cds \cdot S(n)\right) & (Cgs + Cds) \cdot S(n) + gm2 + \frac{1}{Ro2} + \frac{1}{Rs} & 0 \\ 0 & -Cml \cdot S(n) & 0 & \frac{1}{Rota} + Cml \cdot S(n) \end{bmatrix} \right\|}$$

Figure 35b. Mathcad Analisys of the Spindle Current Control Loop

OPEN LOOP RESPONSE

$H(n) := H1(n) \cdot H2(n)$ $\quad |H(0)| = 584.28$

MAGNITUDE RESPONSE in DB



PHASE RESPONSE in Degrees

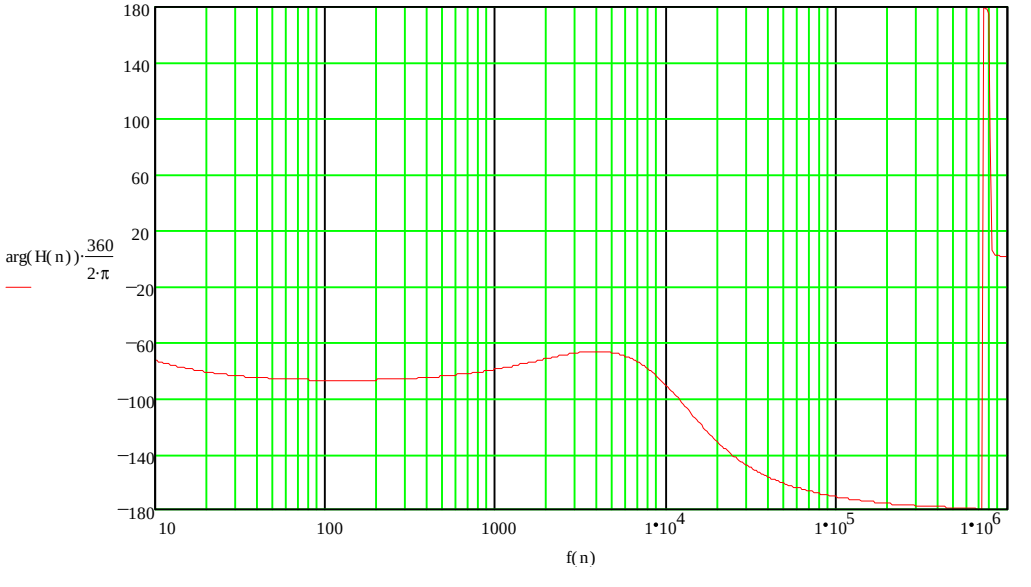
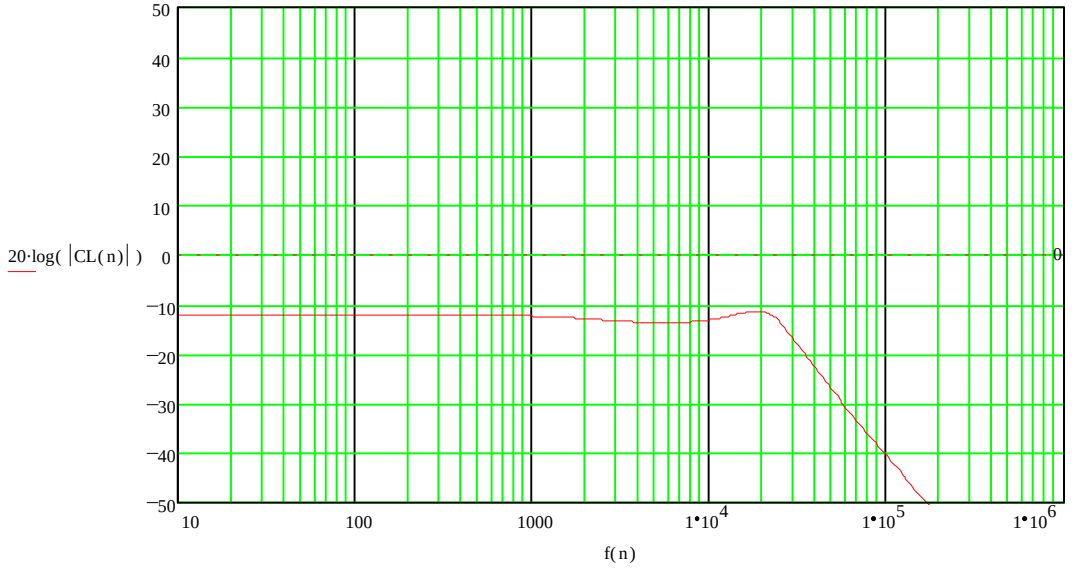


Figure 35c. Mathcad Analisys of the Spindle Current Control Loop

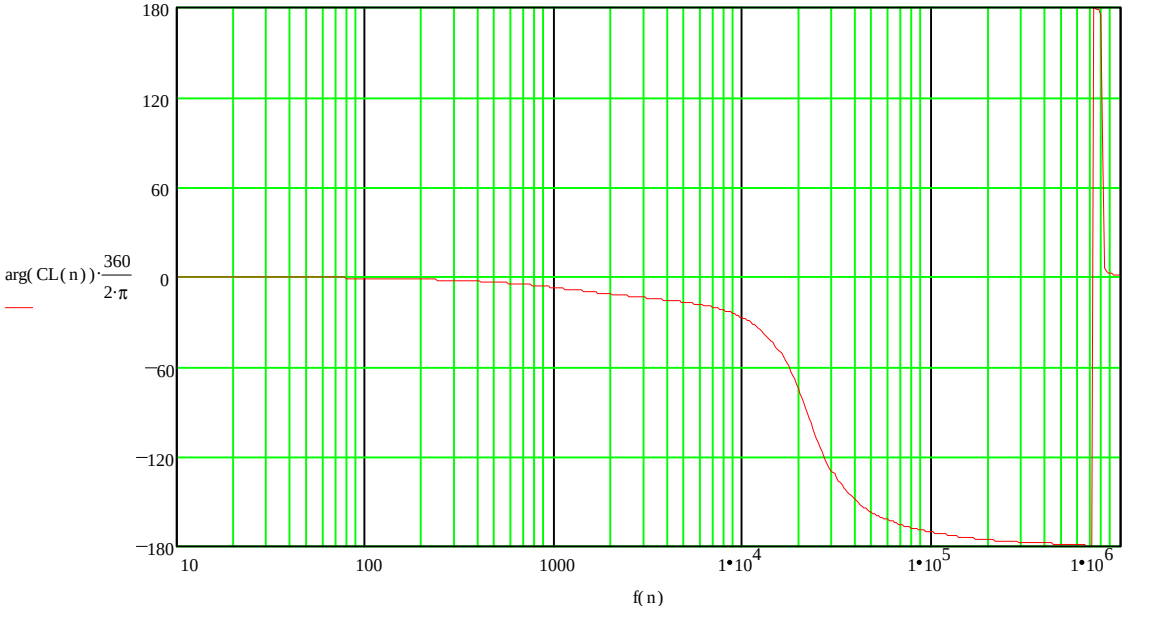
CLOSE LOOP RESPONSE

$$CL(n) := \frac{H(n)}{1 + A \cdot H(n)}$$

MAGNITUDE RESPONSE in DB



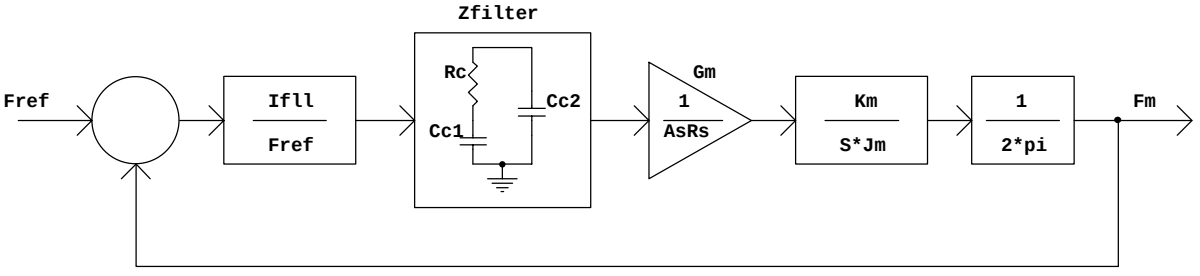
PHASE RESPONSE in Degrees



8.1.2 Spindle Speed Control Loop

Figure #36 shows the Spindle Speed Control Loop model used for the analisys. Figures 37a, 37b, 37c show the Mathcad analisys of the loop.

Figure 36. Spindle Speed Control Loop



$Fref := 90$ **Reference Clock Frequency(Hz) Speed at 5400Rpm**

$Kfll := 1$ ***Charge Pump Current Gain(1or4)**

$Ifll := 25 \cdot 10^{-6} \cdot Kfll$ **Charge Pump Current(A)** $Ifll = 2.5 \cdot 10^{-5}$

$As := 4$ ***Sense Amp Gain**

$Rs := 3 \cdot 10^{-1}$ ***Sense Resistor(0.3ohms)**

$Gm := \frac{1}{As \cdot Rs}$ **Current Loop Transconductance(A/V)** $Gm = 0.833$

$Km := 125$ ***Motor Torfque Constant(gm-cm/A)**

$Jm := 0.2$ ***Motor Inertia(gm-cm-sec-sec)**

$Rc := 0.43 \cdot 10^{6}$ ***FLL Compensation Resistor**

$Cc1 := 1 \cdot 10^{-6}$ ***FLL Compensation Capacitor**

$Cc2 := 0.1 \cdot 10^{-6}$ ***FLL Compensation Capacitor**

Figure 37a. Mathcad Analisys of the Spindle Speed Control Loop

VELOCITY LOOP TRANSFER FUNCTION :

$$i := \sqrt{-1}$$

$$f(n) := 10^{n}$$

$$S(n) := 2 \cdot i \cdot \pi \cdot 10^{n}$$

$$n := -2, -1.99 .. 3$$

$$Sz := \frac{1}{Rc \cdot Cc1}$$ **Filter zero**

$$fz := \frac{Sz}{2 \cdot \pi} \qquad fz = 0.37$$

$$Sp := \frac{1}{Rc \cdot \frac{Cc1 \cdot Cc2}{Cc1 + Cc2}}$$ **Filter pole**

$$fp := \frac{Sp}{2 \cdot \pi} \qquad fp = 4.071$$

$$Kc := \frac{1}{Cc1 + Cc2}$$

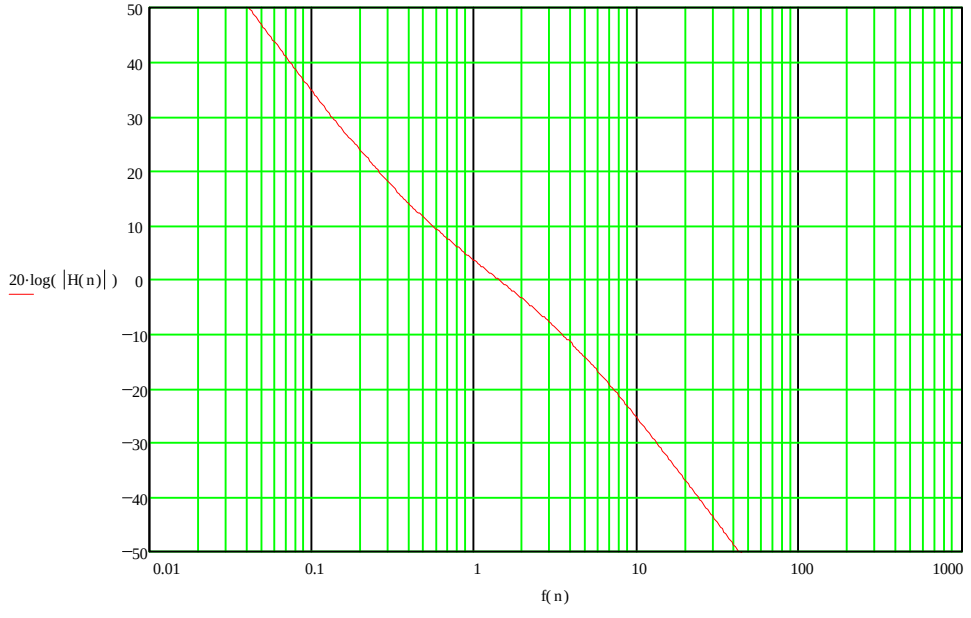
$$Zfilter(n) := Kc \cdot \frac{\frac{S(n)}{Sz} + 1}{S(n) \cdot \left(\frac{S(n)}{Sp} + 1 \right)}$$

Figure 37b. Mathcad Analisys of the Spindle Speed Control Loop

Open Loop Gain

$$H(n) := \frac{Ifll}{Fref} \cdot Zfilter(n) \cdot Gm \cdot \frac{Km}{Jm \cdot S(n)} \cdot \frac{1}{2 \cdot \pi} \qquad \text{let} \quad K := \frac{Ifll}{Fref} \cdot \frac{Gm}{Jm} \cdot \frac{Km}{2 \cdot \pi} \cdot Kc \qquad K = 20.933$$

OPEN LOOP MAGNITUDE RESPONSE in DB



OPEN LOOP PHASE RESPONSE in degree

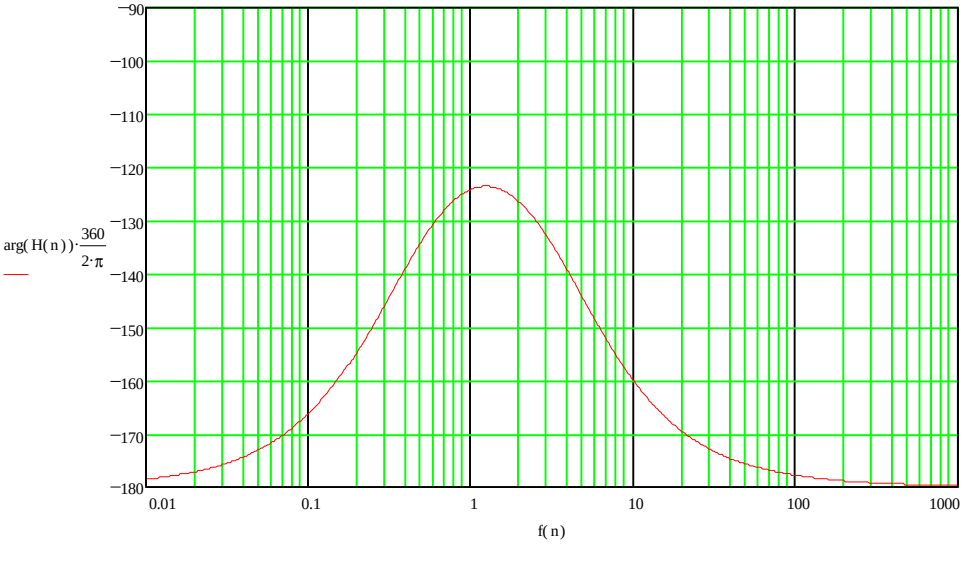
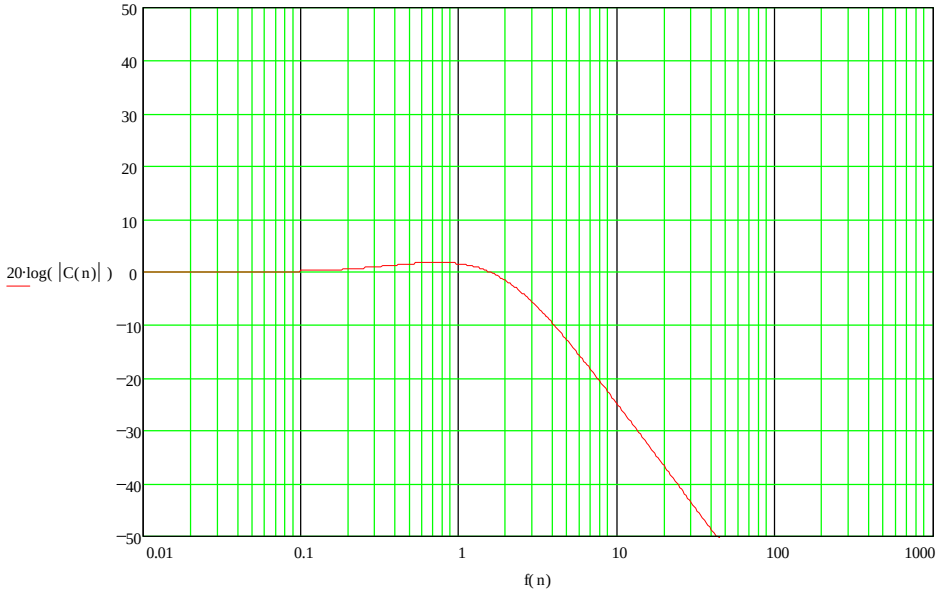


Figure 37c. Mathcad Analisys of the Spindle Speed Control Loop

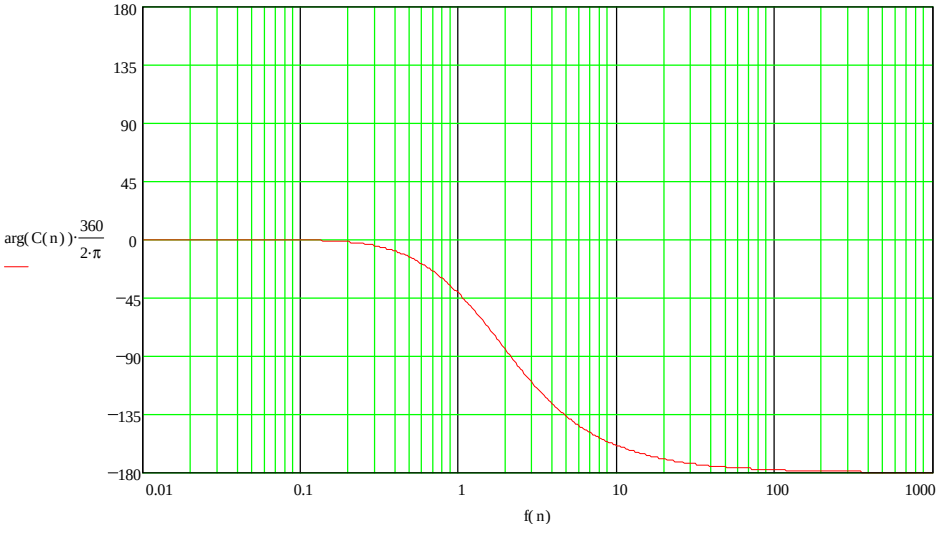
Close Loop Gain

$$C(n) := \frac{H(n)}{1 + H(n)}$$

CLOSE LOOP MAGNITUDE RESPONSE in DB



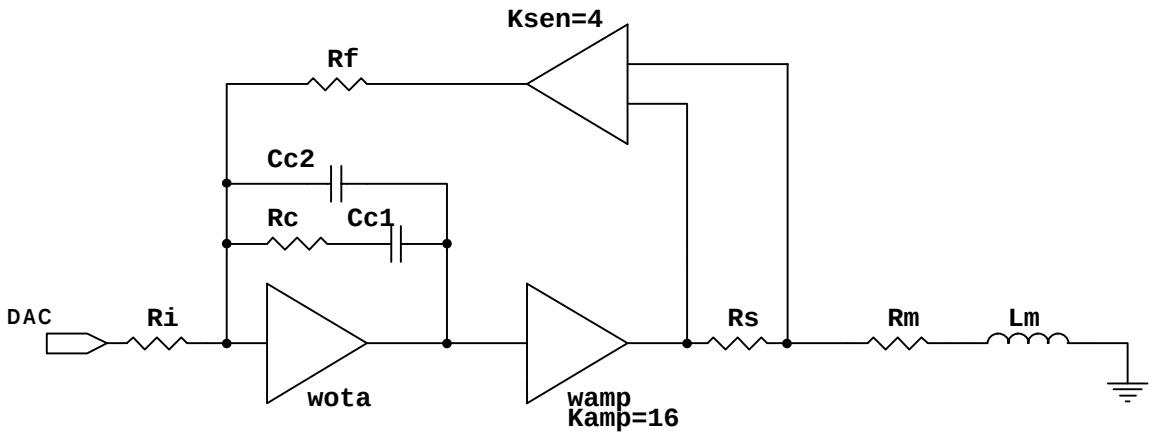
CLOSE LOOP PHASE RESPONSE in degrees



8.1.3 Voice Coil Current Control Loop

Figure #38 shows the Voice Control Loop model used for the analisys. Figures 39a, 39b, 39c show the Mathcad analisys of the loop.

Figure 38. Voice Coil Current Control Loop Model



Device parameters

Kamp := 16 — **gain of power stage**

wamp := $600 \cdot 10^{3}$ — **pole of power stage**

Ksen := 4 — **gain of sense amplifier**

wota := $60 \cdot 10^{6}$ — **pole of OTA**

User specified value

Lm := $1.5 \cdot 10^{-3}$ — **coil inductance**

Rm := 13.3 — **coil resistance**

Rs := 0.25 — **sense resistance**

Ri := $1 \cdot 10^{4}$ — **input resistor**

Kgm := 1 — **target DC loop transconductance(A/V)**

Rf := $1 \cdot 10^{4}$ — **feedback resistor** *suggested Rf value* Rfs := Ksen·Rs·Ri·Kgm Rfs = $1 \cdot 10^{4}$

BW := $1 \cdot 10^{4}$ — **target bandwidth(Hz)**

Cc1 := $1.8 \cdot 10^{-9}$ — **compensation capacitance** *suggested Cc1 value* $Cc1s := \frac{Ksen \cdot Rs \cdot Kamp}{Rf \cdot (Rs + Rm) \cdot 2 \cdot \pi \cdot BW}$ $Cc1s = 1.879 \cdot 10^{-9}$

Cc2 := $1.8 \cdot 10^{-10}$ — **compensation capacitance** *suggested Cc2 value* $Cc2s := \frac{Cc1}{10}$ $Cc2s = 1.8 \cdot 10^{-10}$

Rc := $6.2 \cdot 10^{4}$ — **compensation resistance** *suggested Rc value* $Rcs := \frac{Lm}{Cc1 \cdot (Rs + Rm)}$ $Rcs = 6.15 \cdot 10^{4}$

Figure 39a. Mathcad Analisys of the Voice Coil Current Loop

$i := \sqrt{-1}$

$f(n) := 10^n$

$S(n) := 2 \cdot i \cdot \pi \cdot 10^n$

$n := 1, 1.01 .. 6$

$Kc := \frac{1}{(Cc1 + Cc2) \cdot Ri}$ $\quad Kc = 5.051 \cdot 10^4$

$wz := \frac{1}{Rc \cdot Cc1}$ $\quad Fz := \frac{wz}{2 \cdot \pi}$ $\quad Fz = 1.426 \cdot 10^3$

$wp := \frac{1}{Rc \cdot \left(\frac{Cc1 \cdot Cc2}{Cc1 + Cc2} \right)}$ $\quad Fp := \frac{wp}{2 \cdot \pi}$ $\quad Fp = 1.569 \cdot 10^4$

$H1(n) := Kc \cdot \frac{\left(1 + \frac{S(n)}{wz}\right)}{\left(1 + \frac{S(n)}{wota}\right) \cdot \left(1 + \frac{S(n)}{wp}\right) \cdot S(n)}$ **compensator**

$H2(n) := \frac{Kamp}{1 + \frac{S(n)}{wamp}}$ **power stage**

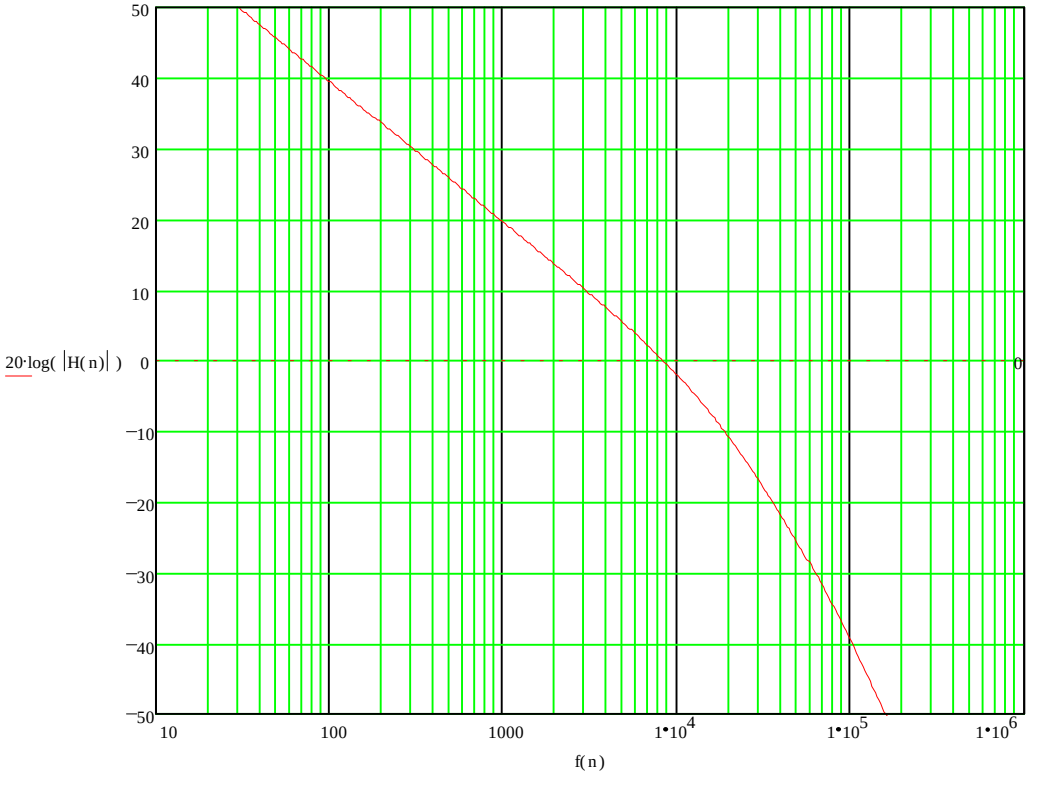
$H3(n) := \frac{1}{Lm \cdot S(n) + Rm + Rs}$ $\quad fl := \frac{Rm + Rs}{2 \cdot \pi \cdot Lm}$ $\quad fl = 1.438 \cdot 10^3$

Figure 39b. Mathcad Analisys of the Voice Coil Current Loop

OPEN LOOP RESPONSE

$H(n) := H1(n) \cdot H2(n) \cdot H3(n)$ $\quad |H(1)| = 949.152$

MAGNITUDE RESPONSE in DB



PHASE RESPONSE in Degrees

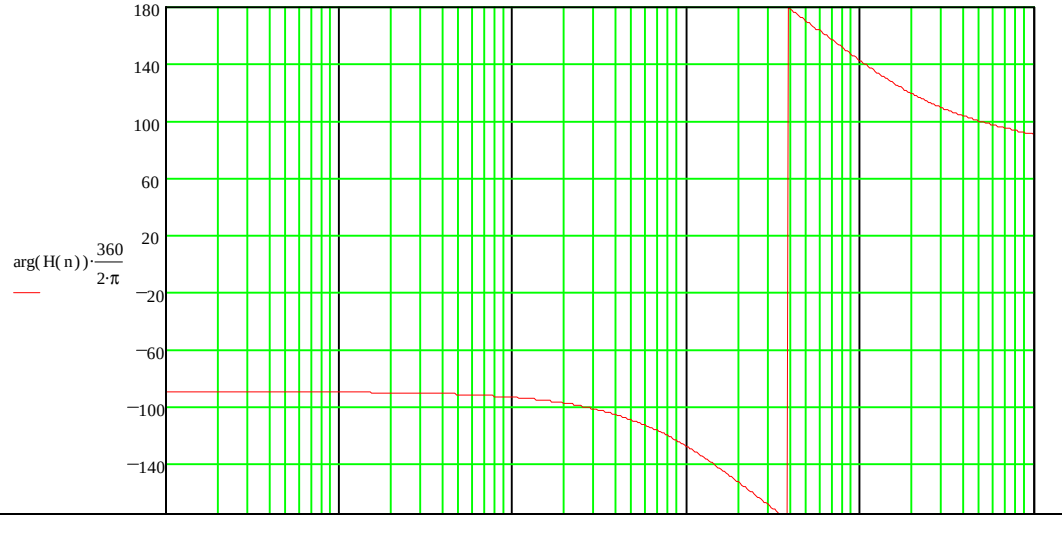
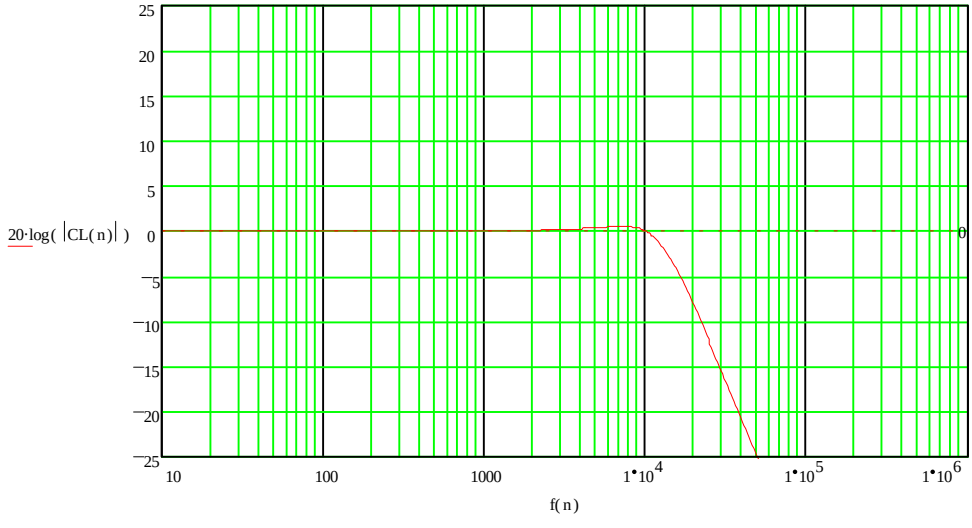


Figure 39c. Mathcad Analisys of the Voice Coil Current Loop

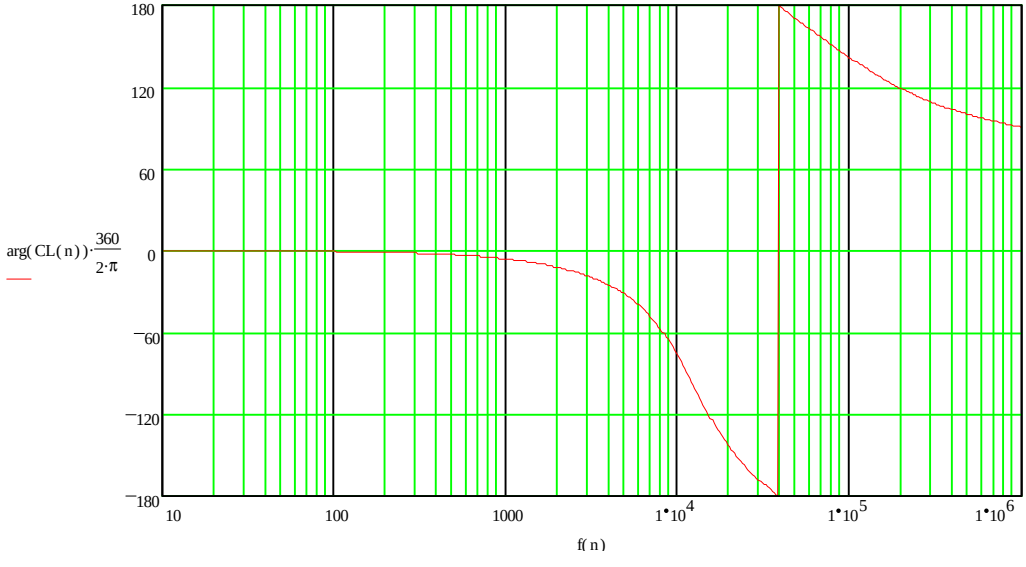
CLOSE LOOP RESPONSE

$A := Ksen \cdot Rs \cdot \frac{Ri}{Rf}$ $A = 1$ $CL(n) := \frac{H(n)}{1 + A \cdot H(n)}$ $20 \cdot \log(|CL(1)|) = 1.464 \cdot 10^{-6}$

MAGNITUDE RESPONSE in DB



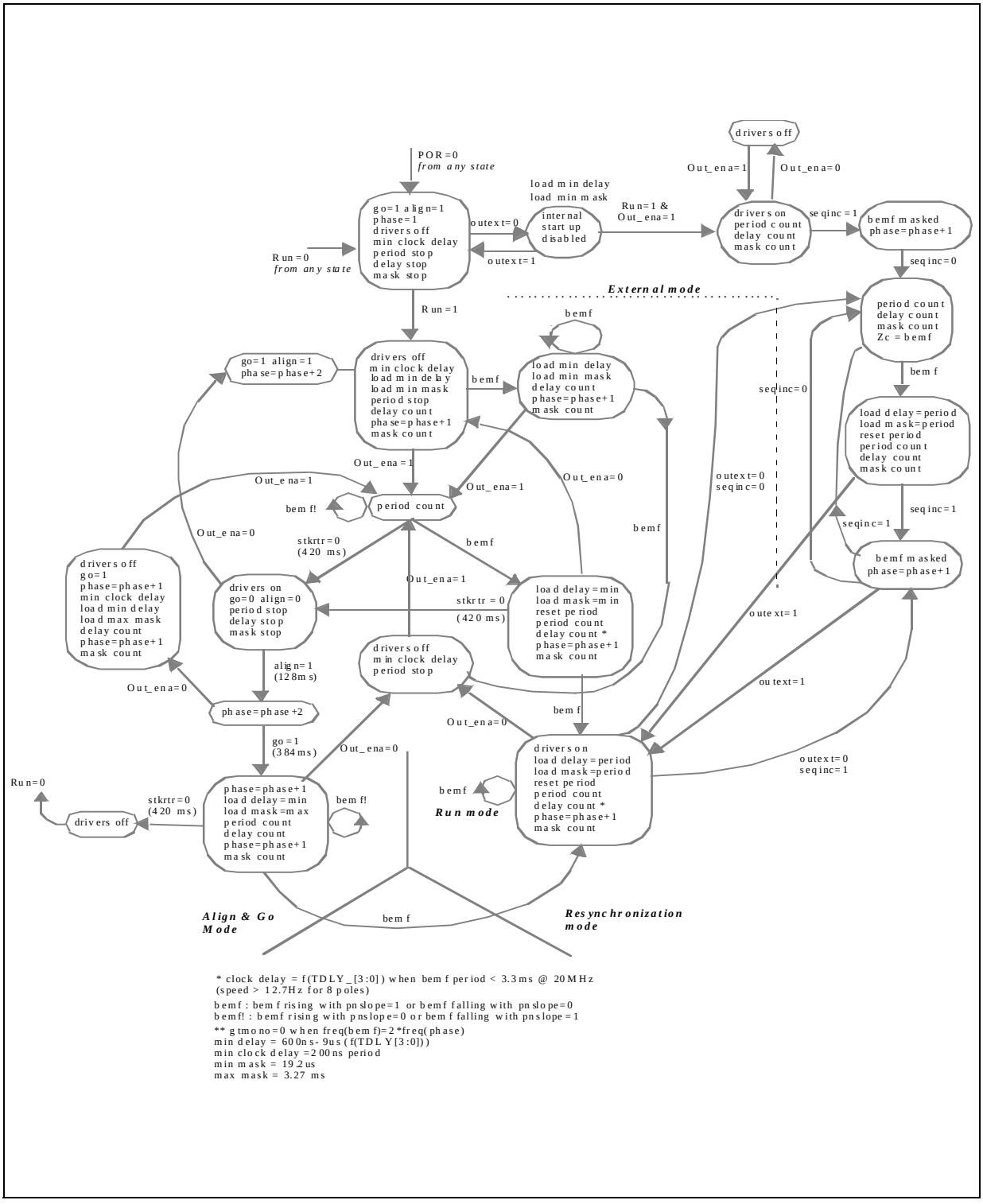
PHASE RESPONSE in Degrees



8.2 STATE DIAGRAM

The Figure #40 depicts the complete Spindle Motor Controller State diagram. This include, Internal Align & Go Start-Up, External Start-Up and Resynchronization.

Figure 40. Spindle Motor Controller State Diagram



8.3 INDUCTIVE SENSE START-UP STEP BY STEP

Following is the Step by Step approch to perform the Inductive Sense Start-Up Option. The device has to be programmed exactely in the order described. A flow Chart of this Start-Up is available in Figure #4 inside chaper 4 (Spindle Circuits) in section 4.1.2.3 (Inductive Sense).

GENERAL SETUP

1. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 0**.
2. **START_UP** - Reg#2.1 [1 = internal start-up, 0 = external]. **Set to 0**.
3. **R_SEQ** - Reg#2.2 [1 = reset sequencer to phase 1]. **Set to 1**.
4. **R_SEQ** - Reg#2.2 [1 = reset sequencer to phase 1]. **Set to 0**.
5. **RUN** - Reg#2.3 [1 = start internal Align&Go start-up, 0 = reset logic]. **Set to 0**.
6. **SPIN_EN** - Reg#2.4 [1 = enable spindle outputs, 0 = disable]. **Set to 0**.
7. **EXT/INT** - Reg#2.7 [1 = external spindle feedback ,0 = internal]. **Set to 1**.
8. **MEC/ELEC** - Reg#2.5 [1 = electrical cycle for FLL, 0 = mechanical]. **Set as required**.
9. **8_12_POLE** - Reg#3.3 [1 = 8 pole, 0 = 12 pole]. **Set as required**.
10. **ICP** - Reg#8.1 [1 = 25µA, 0 = 100µA**]. Set as required**.
11. **PWM/LIN** - Reg#2.6 [1 = pwm current control, 0 = linear]. **Set to 1**.
12. **FLL**, Fine and Coarse Counters - Regs# 4,5,6. **Set as required**.
13. **ISNS** - Reg#8.3 [1 = presets spindle inductive start-up sense circuits]. **Set to 1**.
15. **RUN** - Reg#2.3 [1 = start internal Align&Go, 0 = reset logic]. Set to 1.

PHASE DETECTION

16. Set **TRIAL** counter to 1.
17. Set **PHASE** counter to 1.
18. Preset a **COUNTER** with at least 500nS resolution.
19. Store Initilal **Time** Period.
20. **SPIN_EN** - Reg#2.4 [1 = enable spindle output, 0 = disable]. Set to 1.
21. Start **COUNTER**.
22. Monitor at the same time, the **FCOM** pin #1 output for transition from low level (0 volt) to high level (5 volt) and the **COUNTER** for the end of count (>50mS).
23. If **COUNTER** is greater than 50mS, set to 0 SPIN_EN Reg#2.4 and lower the threshold according to step #14. Then restart from step # 16.
24. If the Threshold is reached (**FCOM** is at high level), stop **COUNTER** and reset to 0 **SPIN_EN** - Reg#2.4 bit.
25. If **COUNTER** time is shortest than the previous one, store **Time** and **PHASE**.
26. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 1**.
27. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 0**.
28. Increment **PHASE** and reapet from step #20 until **PHASE**=6.
29. Store **PHASE** with lowest time.
30. Increment **TRIAL** and repeat from step # 19 until **TRIAL**=5.

31. Get the **PHASE** with a most frequent lowest time. (**PHASE** is a position where the rotor is suppose to be).

STARTING

32. **ISNS** - Reg# 8.3 [1 = preset spindle inductive start-up sense circuits]. **Set to 0**.
33. **PWM/LIN** - Reg#2.6 [1 = pwm current control, 0 = linear]. **Set as required**.
34. **IL1/IL0** - Reg#8.4.5 [Phase Detection Current Threshold]. **Set as required**.
35. **R_SEQ** - Reg#2.2 [1 = reset sequencer to phase 1]. **Set to 1**.
36. **R_SEQ** - Reg#2.2 [1 = reset sequencer to phase 1]. **Set to 0**.
37. If **PHASE**=1 go to Step # 41.
38. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 1**.
39. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 0**.
40. Repeat from step # 38 for **PHASE** minus 1 times. (Example: if **PHASE**=3 repeat 2 times. This is to bring the Sequencer to the same phase of the Rotor).
41. **SPIN_EN** - Reg#2.4 [1 = enable spindle outputs, 0 = disable]. **Set to 1**.
42. **START_UP** - Reg#2.1 [1 = internal start-up, 0 = external]. **Set to 1**.
43. Wait **10mS** (this time may vary).
44. **START_UP** - Reg#2.1 [1 = internal start-up, 0 = external]. **Set to 0**. (Steps # 42, #43, #44 are to charge the external Compensation Capacitor if **LINEAR** mode is used. If **PWM** mode is used, they may be skipped).
45. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 1**.
46. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 0**.
47. Wait **20mS** (this time may vary).
48. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 1**.
49. **INCRE_SEQ** - Reg#2.0 [transition 0 to 1 increments spindle sequencer]. **Set to 0**.
50. Wait **3mS** (this time may vary).
51. **START_UP** - Reg#2.1 [1 = internal start-up, 0 = external]. **Set to 1**.
52. Check for desired **Speed**.
53. **SD0...SD3** - Reg#3.7.6.5.4. [commutation delay]. **Set as required**

littlediode
COMPONENTS FOR ALL

Littlediode supplies new, hard to find or obsolete electronic components and semiconductors all over the world.

With over two million different components listed you are sure to find the part you need.

Feel free to visit us today at our online store:

LittleDiode.com

Looking forward to providing you with the best possible service.