Attitude Sensor and Actuator Interfacing for micro-satellites

By

Zamokwakhe P. Dlamini

Thesis presented in partial fulfillment of
the requirements for the degree of
Master of Science in Engineering Science
at the
University of Stellenbosch

Supervisor: Prof. J.J. du Plessis

Department of Electrical and Electronic Engineering
University of Stellenbosch

December 2002

Declaration

I the undersigned hereby declare that the work contained in this thesis is my own original work and has not previously in its entirety or in part been submitted at any university for a degree.

Signature: Date:

Summary

The purpose of this thesis was to design and test the interfaces of sensors and actuators. In the complete Attitude Determination and Control System (ADCS) the Interface Control Processor (ICP) must be interfaced to various sensors and torquers.

The testing of interfaces involves the simulation of sensors and actuators and also two of the Attitude Control Processor's functions, i.e. sending actuator commands and receiving sensor data. The tested interfaces are for two actuators and three sensors. Both analog and digital sensors and actuators are interfaced i.e. reaction wheels, magnetorquers, magnetometer and horizon /fine sun sensor. The simulated sensors and actuators are to be employed on three axis controlled, low earth orbiting micro-satellites.

To test the interfaces, a test circuit was developed and the design is presented. Finally the software to facilitate the testing was developed and is also presented with the analysis based on protocol implementation and data rates.

Opsomming

Die doel van hierdie tesis was om sensor en aktueerder koppelvlakke te ontwerp en te toets. In die volledige Attitude Determination and Control System (ADCS) moet die Interface Control Processor (ICP) aan verskeie sensore en torquers gekoppel word.

Die toets van koppelvlakke behels die simulasie van sensore en aktueerders en ook twee van die Attitude Control Processor (ACP) se funksies, naamlik die stuur van aktueerder bevele en die ontvang van sensor data. Die koppelvlakke vir twee aktueerders en drie sensore is getoets. Beide analoog en digitale sensore en aktueerders is gekoppel, naamlik die reaksiewiele, magneetspoele, magnetometer en horison / fyn son sensor. Die gesimuleerde sensore en aktueerders sal gebruik word op drie-as beheerde, lae aardbaan mikrosatelliete.

Om die koppelvlakke te toets is 'n stroombaan ontwikkel waarvan die ontwerp hier aangebied word. Sagteware om die toetse te fasiliteer is ontwikkel en word ook hier aangebied tesame met analises gebaseer op protokol implementering en data tempo's.

Acknowledgements

I would like to express my sincere gratitude to the following people. Without their help and assistance this thesis would not have been possible.

- Mr. Japie Engelbrecht who as my mentor, rendered valuable guidance and assistance.
- Prof. J. J. du Plessis my supervisor, for his guidance in documenting the thesis.
- Mr. Andre Bester who, as my first mentor, started working with me on this thesis.
- Mr. Jock De Bliquiy for his assistance in designing and laying out the test circuit board.
- Mr. X. C. Farr for his assistance in providing me with the documents of SUNSAT's ADCS.
- My family, especially my father and my mother whose encouragement has been of great value.
- My Almighty God for being the guiding light through out.
- Lastly, I would like to thank the Department of Communications for giving me the opportunity to do this study.

List of Figures

Acronyms

ACP	Attitude Control Processor
ADC	Analog to Digital Converter
ADCS	Attitude Determination and Control System
CCD	Charge Coupled Device
CPU	Central Processing Unit
DAC	Digital to Analog Converter
FSA	Finite State Automation
ICP	Interface Control Processor
IGRF	International Geomagnetic Reference Field
ISR	Interrupt Service Routine
LED	Light Emitting Diode
LEO	Low Earth Orbit
MCU	Micro-controller Unit
OBC	On-board Computer
Op-Amp	Operational Amplifier
OUST	University of Surrey Satellite
PWM	Pulse Width Modulation
RW	Reaction Wheels
SCP	Star Camera Processor
SFR	Special Function Register
SUNSAT	Stellenbosch University Satellite
TTL	Transistor -Transistor Logic
UART	Universal Asynchronous Receiver Transmitter

Contents

Chapter 1

Introduction

The ADCS hardware can be divided into three categories namely: the ADCS onboard processors (ICP and the ACP), actuators and sensors. The ADCS components described here are applicable to small, low earth orbit (LEO) satellites. The ADCS stabilizes and orients the satellite in the desired direction during the mission despite the external disturbance torques acting on the satellite. This requires the satellite to determine its attitude using the sensors. The attitude control is achieved by using the actuators.

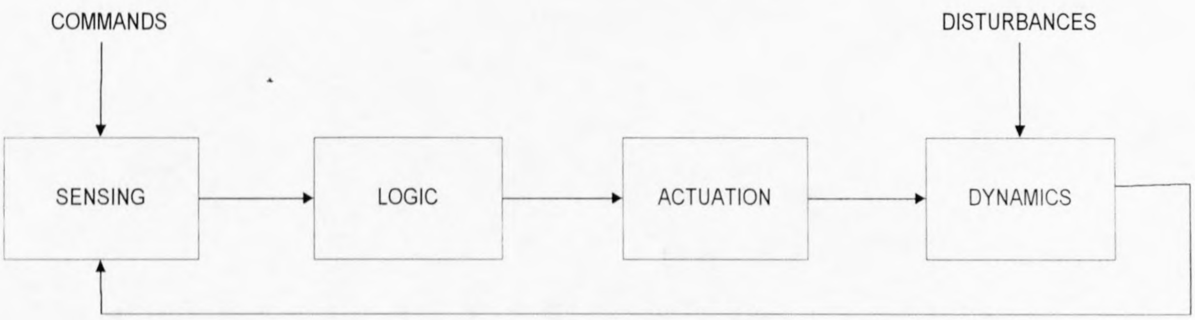


Figure 1.1 General Satellite Attitude Control System [1]

In general, a spacecraft's attitude control system consists of the following four major functional sections: sensing, logic, actuation, and vehicle dynamics. The sensing function determines the satellite's attitude. The logic programs the electronic signals in a correct sequence to the torque producing elements, which in turn rotate the spacecraft about the center of mass of the spacecraft. The resulting motion (dynamics) is then monitored by the vehicle sensors, which thus close the loop of the spacecraft attitude control system (see Figure 1.1).

The ADCS consist of two main processors, namely the ICP and the ACP. The function of the ICP is to manage the sensors and the actuators on the ADCS. The ICP is an 80C31-based micro-controller. On SUNSAT the ICP has four main communication routes, namely three UART-routes to OBC1, OBC2 (Onboard Computers) and to the ACP respectively, and a general bus (8031BUS) connected to a multi-plexer and to the sensors and actuators [7]. The ACP, a T800 transputer implements all the control system software.

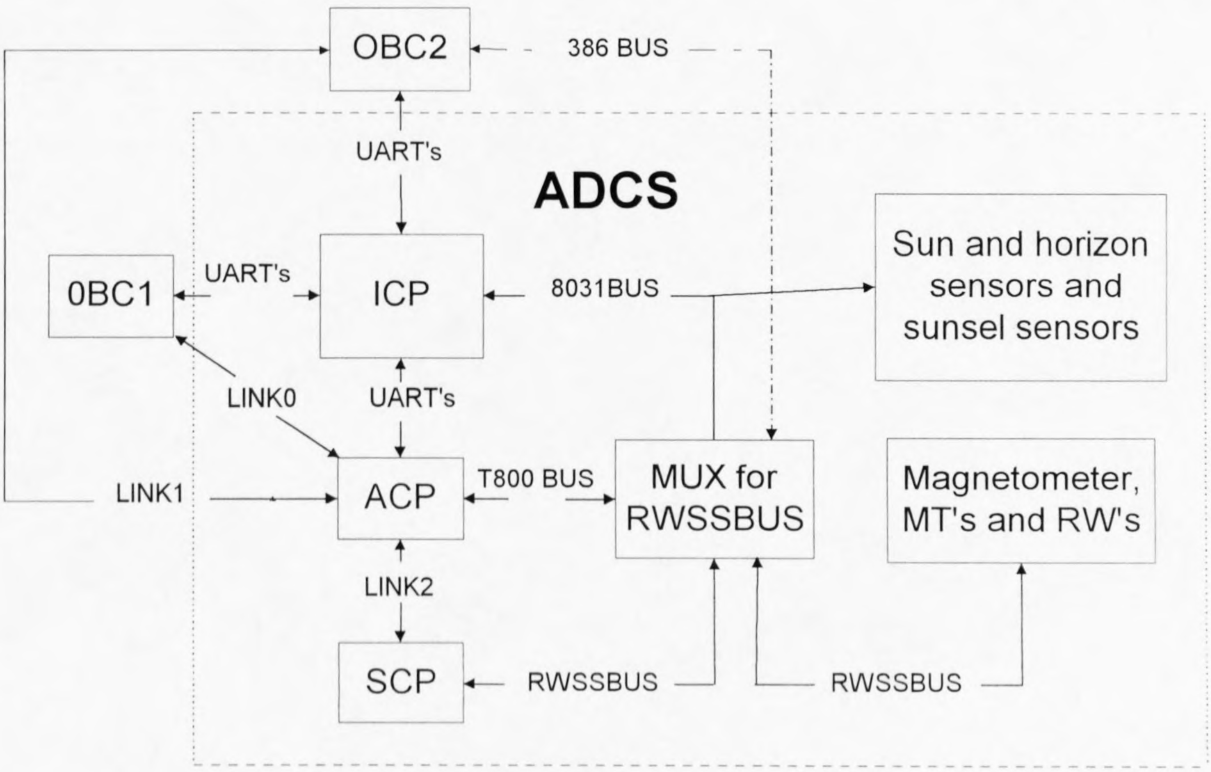


Figure 1.2: The Basic Structure of the ADCS Communication on SUNSAT [7]

Figure 1.2 shows the basic communication routes on the ADCS of SUNSAT. Since the ADCS is a time critical system the codes running in both the ACP and the ICP must also be time dependent processes.

1.1 Objectives of the Thesis

The objective of the thesis was to develop the interfaces for testing various sensors and actuators. The development of these interfaces includes the development of the test hardware and software. The test hardware must be able to support the simulation of the sensors and actuators, both digital and analog. The sensors and actuator interfaces must be compatible with the general ADCS communication protocols.

1.2 Methodology

The interface test hardware was constructed as shown below in figure 1.3. For the purpose of the thesis, only the partial system inside bold dashed lines including the ACP was developed.

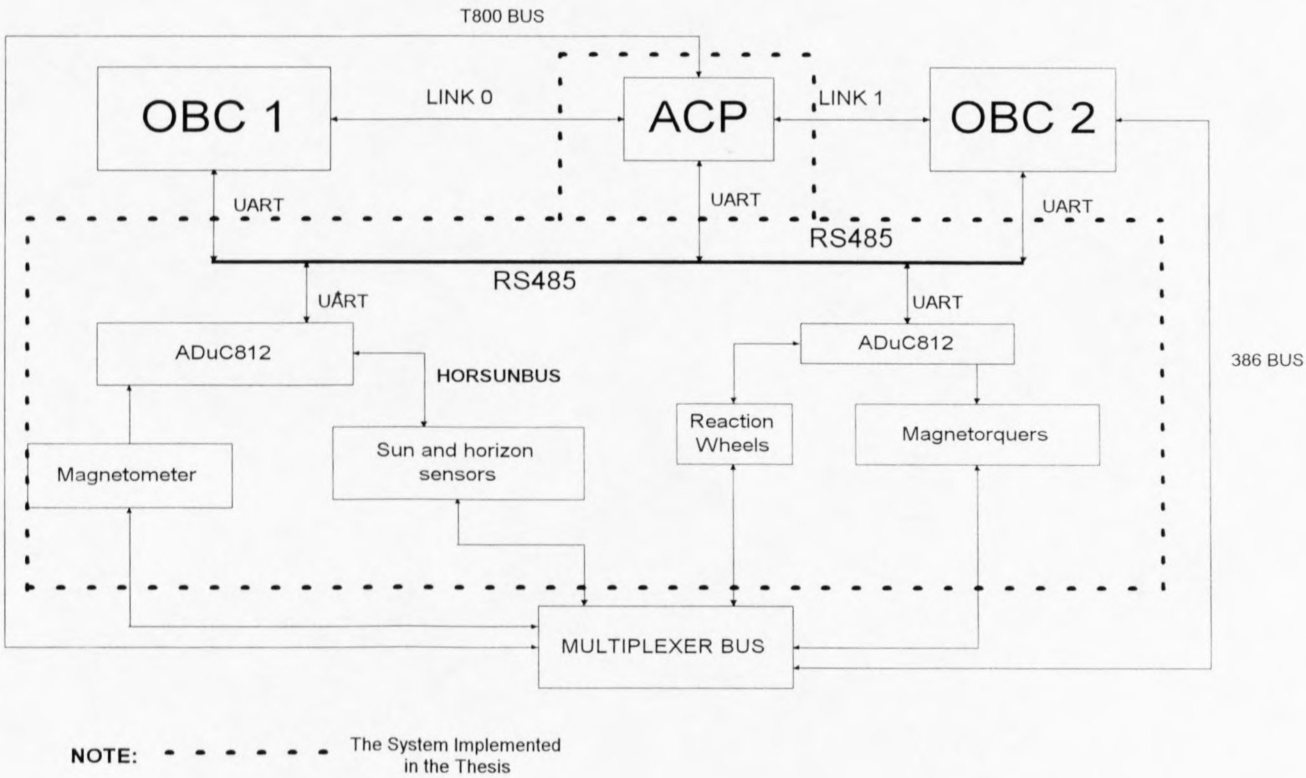


Figure 1.3 Interface of the Test Circuit with the ADCS

To implement the RS-485 protocol, two ADuC812 based slaves were interfaced with the simulated sensors and actuators. The ACP was simulated using a PC. The only functions of the ACP implemented by the PC are to send actuator commands and receive sensor data.

The scope of this thesis document covers the following topics:

- Chapter 2 introduces various types of attitude determination and control components for three-axis stabilized micro-satellites.
- Chapter 3 presents the interfacing of the sensors and actuators to the ADCS of SUNSAT. The interfacing of the simulated actuators and sensors with the test hardware is discussed.
- Chapter 4 presents the discussion on the software implemented in the hardware circuit. The code implemented in slave1, slave2, horizon / fine sun sensor data and the master (PC) are discussed in the form of flow diagrams respectively.
- Chapter 5 looks into the implementation of the software and hardware to simulate the sensors and actuators. The results of the measurements done on the hardware and the analysis of the hardware and software are presented in this chapter.
- Chapter 6 summarizes and reflects on the results presented in the thesis. Some recommendations on the possible interfaces are also outlined.

Chapter 2

Background

This chapter discusses the different types of attitude determination and control components. The sensors discussed here measure the orientation of the satellite axes with respect to external references such as the earth, sun or the stars.

ATTITUDE DETERMINATION

2.1 Magnetometer

The magnetometer measures the strength of the geomagnetic field vectors in three axes. The satellite attitude (e.g. pitch, roll and yaw angles and angular rates) is determined by comparing magnetometer readings with the IGRF (International Geomagnetic Reference Field) model of the earth's magnetic field.

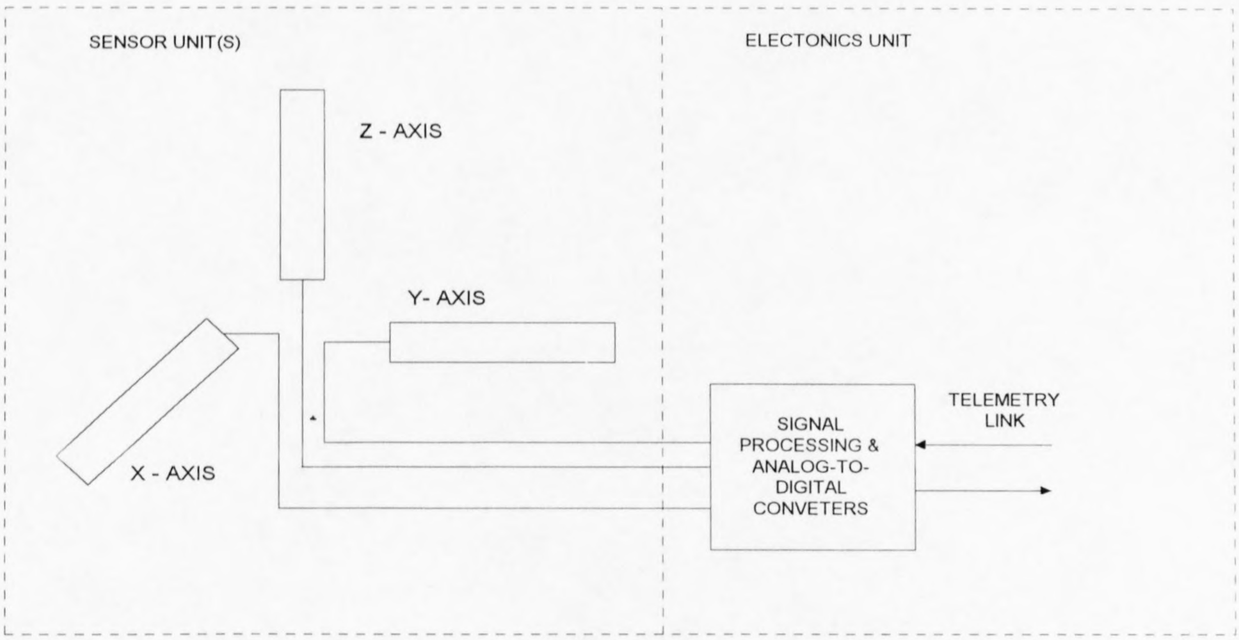


Figure 2.1: Generalized Block Diagram of a Magnetometer Unit [2]

The Earth's magnetic field strength decreases with distance from the Earth as 1/r^3. For altitudes above 1000km, the residual spacecraft magnetic biases dominate the magnetic field measurement [9]. The magnetometers are suitable for LEO satellites since their orbits are less than 1000km.

Magnetometers consist of two main units, an electronic unit and a magnetic sensor (see Figure 2.1). The electronics unit converts the sensor measurements into a suitable digital format. To improve accuracy, the magnetometer data can be combined with data from the sun or horizon sensor [9].

2.2 Horizon and Sun Sensors

2.2.1 Horizon Sensors

Horizon sensors are infrared devices that detect the contrast between the cold of deep space and the heat of the Earth's atmosphere. On SUNSAT (see figure 2.2), two orthogonal linear CCD and lens assemblies look 27.3° below local level to obtain orthogonal measurements of the sunlit earth horizon [6]. A ± 15° view on each 2048 element CCD is used to obtain pitch and roll attitude angles to an accuracy of 0.5 mrad if the satellite yaw angle is known [6].

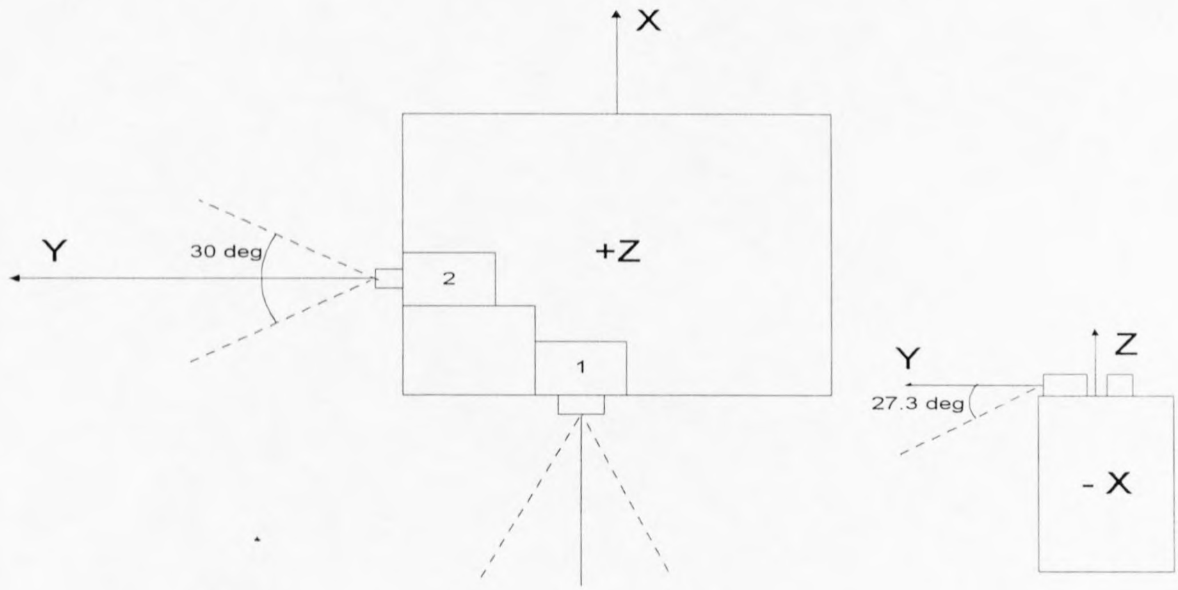


Figure 2.2: Horizon Sensors Position on SUNSAT

Horizon sensors are used when both CCDs detect a valid sunlit horizon, and accurate attitude information is needed. The rest of the time the horizon sensor will be inactive [6].

2.2.2 Fine Sun Sensor

Similar linear CCD technology as used in horizon sensors is applied to obtain a sun azimuth measurement within a 60° view with an angular resolution of 1mrad (see figure 2.3), [6]. The sensor head consists of a slit aperture perpendicular to the CCD array [6].

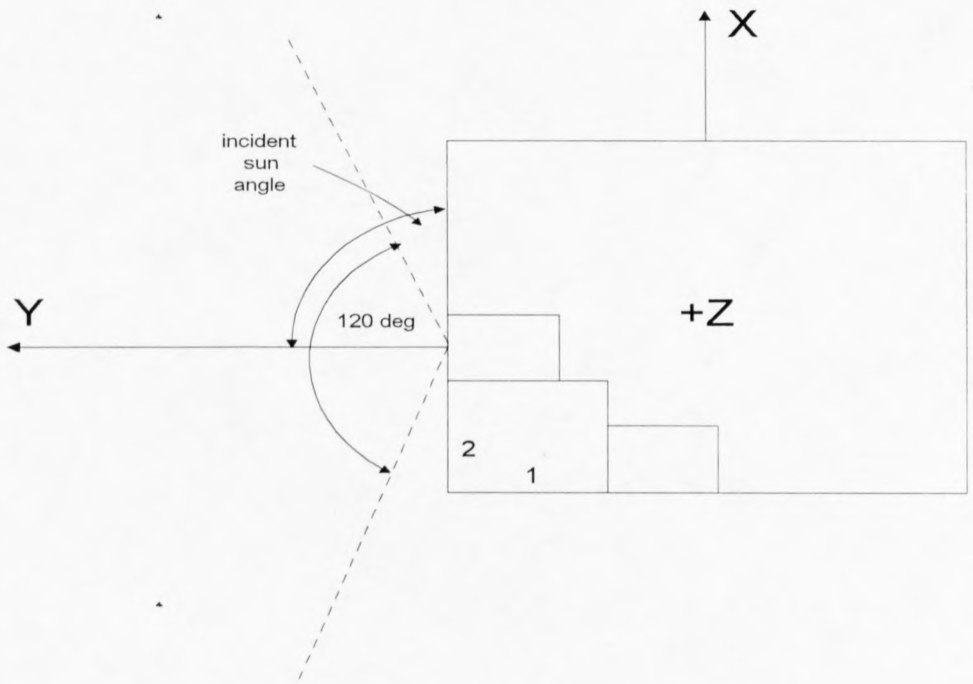


Figure 2.3: Sun Sensor Positions on SUNSAT

To obtain accurate yaw measurements, the fine sun sensor is used. The fine sun sensor is used on SUNSAT during imaging when it is important to have accurate yaw attitude information. For two-axis output, two sensors can be mounted perpendicular to one another (see figure 2.3). The fine sun sensor is switched off when the satellite is in the earth's shadow to reduce power consumption.

2.2.3 Coarse Sun Sensors

The coarse sensors or cosine detectors are sensors based on the sinusoidal variation of the output current of a solar cell with the sun angle (see figure 2.4).

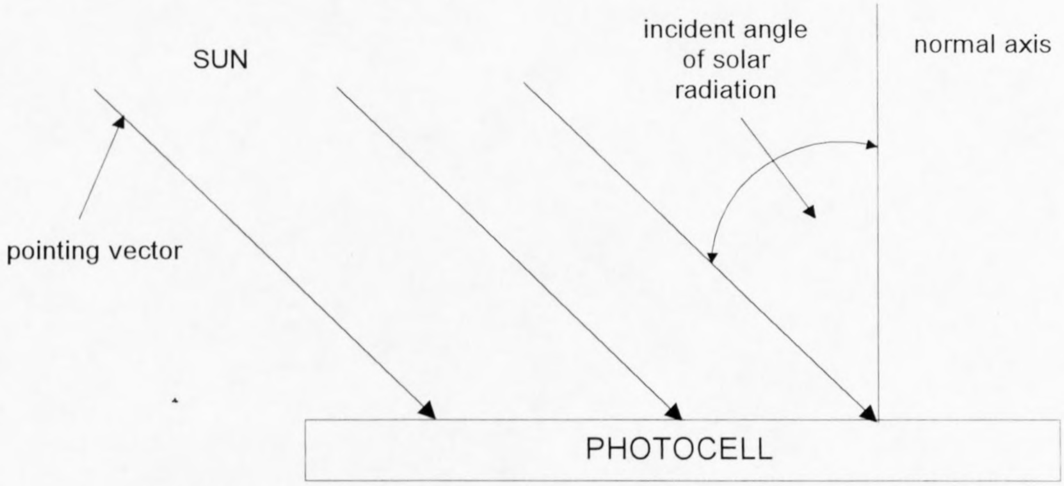


Figure 2.4: Cosine Detector Sun Sensor

The energy deposited in a photocell and consequently the output current, I, is proportional to the cosine of the angle of incidence of the solar radiation [2]:

$$I(\theta) = I(0)*\cos(\theta) \qquad 2.1$$

The effective photocell area and Fresnel reflections cause small transmission losses that are omitted from equation 2.1. The currents from each cell are compared to obtain the sun vector direction with respect to the satellite body [2].

2.3 Star Sensors

Star sensors represent the most common sensor for high accuracy missions [9]. Star sensors are categorized into scanners and trackers. Scanners consist of multiple slits that allow stars to pass through in a scanner's view. After several star crossing, the vehicle's attitude can be derived. Scanners are usually employed on spinning spacecraft.

On 3-axis stabilized spacecraft, trackers are used. Trackers are employed to track one or more stars to derive 2-or 3-axis attitude information.

On SUNSAT at least 2 separated stars are detectable within the sensor's field of view and enable an algorithm using a star catalogue to calculate the pitch, roll and yaw angles. The pitch resolution depends on the stars' separation distance. During earth imaging, the star sensor is always pointing towards the orbit anti-normal, so only a small part of a full star catalogue has to be present on board the satellite [6].

Star sensors are susceptible to being blinded by the Sun and the Moon. The star sensors are usually employed together with gyros in high accuracy missions.

ATTITUDE CONTROL

Attitude control is the process of achieving an orientation of the spacecraft in a specified and predetermined direction. Attitude control is separated into two forms, namely:

1) Attitude stabilization which is the process of maintaining the present orientation.
2) Attitude maneuver is the process of reorienting the spacecraft from one attitude to another.

The motion of the satêllite can be separated into motion of the satellite's center of mass in the earth-centered coordinate system and motion of the body of the satellite about the center of mass. The yaw axis points in the direction of the center of the earth. The roll axis is in the plane of the orbit, perpendicular to the yaw axis and oriented in the direction of the velocity.

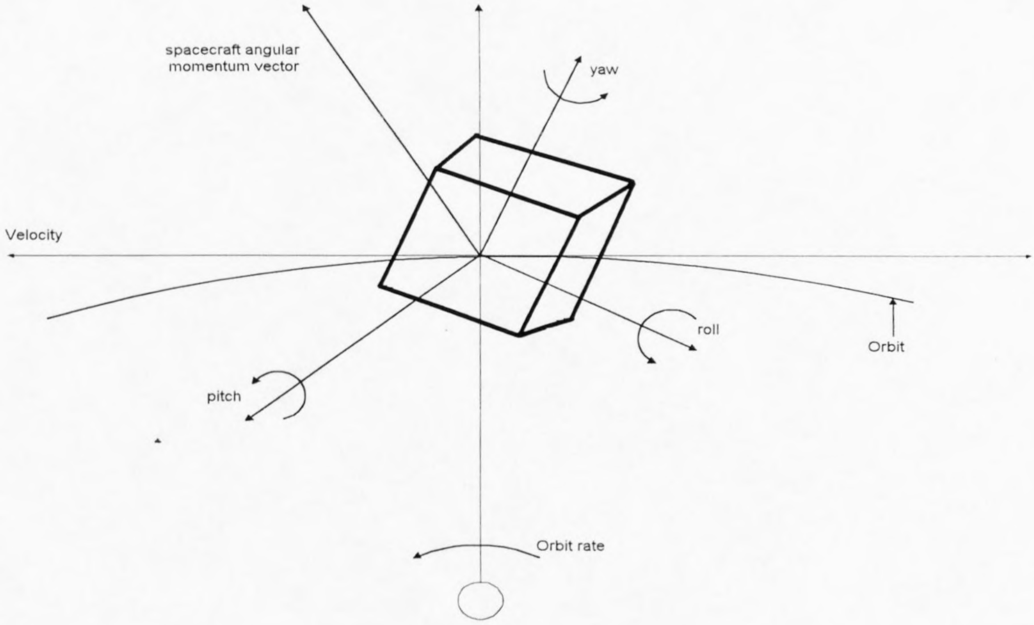


Figure 2.5 Attitude Control Coordinate Axes for an Orbiting Satellite

The pitch axis is perpendicular to the roll and yaw axes (see Figure 2.5).

2.4 Passive (Gravity Gradient) Control

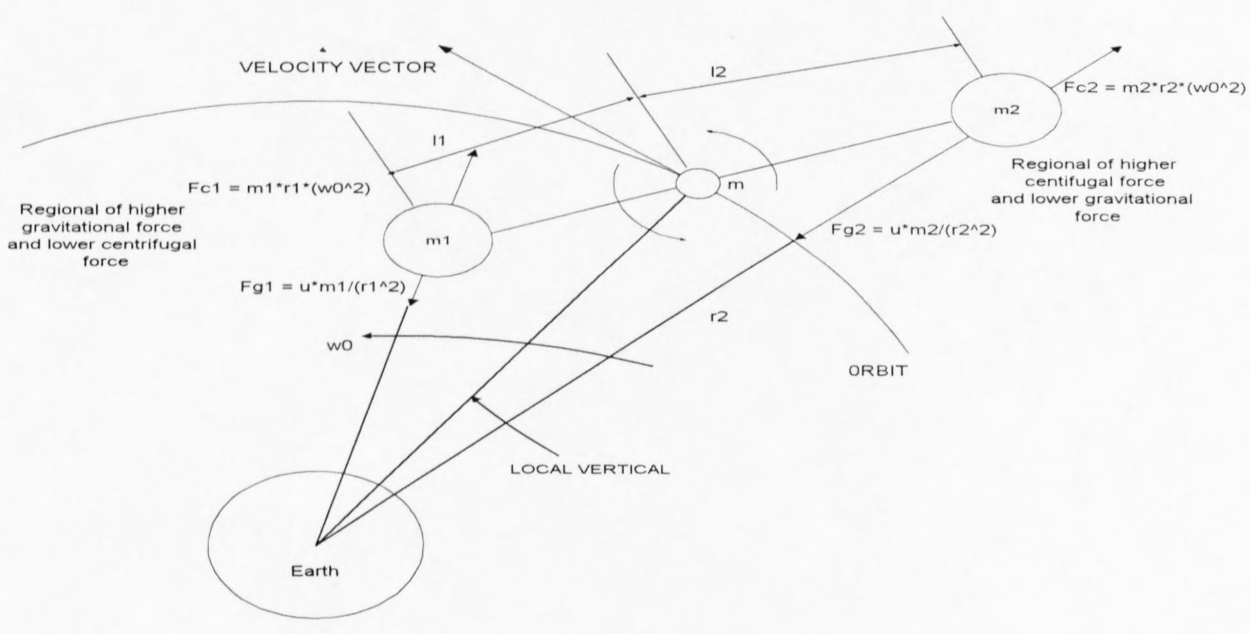


Figure 2.6: Gravity Gradient Restoring Torque [1]

This system (passive control) uses a deployable gravity gradient boom with a tip mass. The satellite is subjected to the earth's gravitational field that determines the movement of the center of mass of the satellite.

Active control element components such as reaction wheels are not employed for passive gravity gradient control. Figure 2.6 shows a gravity gradient restoring torque acting on the dumbbell shaped satellite. The torque is the result of the inverse relationship between the gravity gradient and the distance to the center of the earth.

The satellite will react in such a way that the forces acting on the mass center are equalized. The passive control is used for pointing, with no accuracy involved.

2.5 Active Control System

2.5.1 Reaction Wheels

Reaction wheels are actuators of momentum exchange type. The angular momentum absorbed by the reaction wheels is transferred to the spacecraft or satellite. Accelerating and decelerating an inertial wheel changes the magnitude of angular momentum vector.

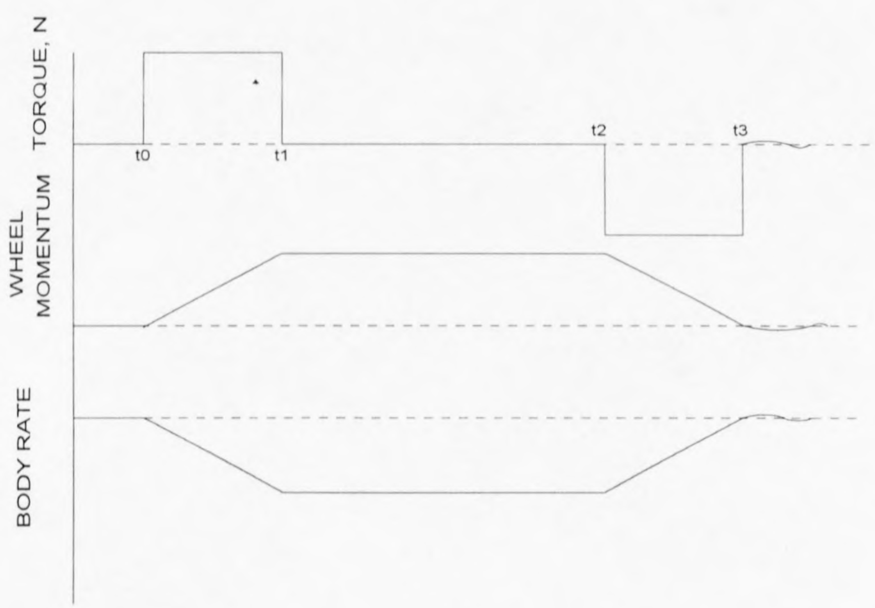


Figure 2.7: Single-Axis Slew Maneuver [2]

In three-axis reaction wheel system, usually three reaction wheels are used to control a satellite with the wheel axes aligned with the body principal axes; a redundant fourth wheel is also employed. On SUNSAT the fourth wheel is aligned to the Z-axis.

Figure 2.7 shows torque, wheel momentum and body rate of the vehicle. The torque motor rotates at maximum speed when a new angular position is commanded at t_0. The momentum of the wheel increases linearly until the maximum limit of the wheel momentum or body rate is reached at time t_1. The body rate $d\theta/dt$ then decreases linearly during the time interval, t_0-t_1, and causes the position angle, θ, to change with time. At time t_2 a braking torque is applied to slow the body rate. This braking causes the target position angle to be reached with a small angular velocity. At t_3, the target is reached but with small residual errors. These residual errors are then eliminated by the stabilization control system.

2.5.2 3-Axis Magnetorquer Coils

Magnetic coils or electromagnets are used to generate magnetic dipole moments for attitude control. The magnetic coils ensure a high reliability (lack of moving parts), and are digitally switched on/off with dual polarity. Magnetic coils with n turns of area, S, create a magnetic moment, $\mathbf{M} = n\mathbf{I}S$ (Am^2), when fed with a current **I** [2].

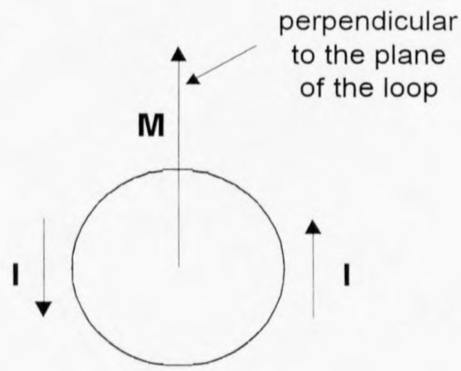


Figure 2.8: Magnetic Moment due to a Current Loop

This magnetic moment can generate a torque **T** by interaction with the terrestrial magnetic field **B** [2]:

$$\mathbf{T} = \mathbf{M} \times \mathbf{B} \quad \text{(Nm)} \qquad (2.2)$$

For a single wire loop enclosing an area S through which a current **I** is flowing the magnetic moment **M** is given by

$$\mathbf{M} = \mathrm{IS}\mathbf{n} \qquad (2.3)$$

where **n** is a unit vector normal to the plane of the loop (see figure 2.8)[2].

For a coil with N turns, the principle of superposition gives $\mathbf{m} = \mathrm{NIS}\mathbf{n}$. The direction of the magnetic moment is the direction of the thumb of the right hand when the fingers of the right hand are cupped in the direction of the current in the loop. The magnetic dipole moment depends on the material enclosed by the current carrying coil and is given by [2]:

$$\mathbf{d} = \mu\mathbf{m} \qquad (2.4)$$

and $\mathbf{T} = \mathbf{d} \times \mathbf{H}$, where μ, is the magnetic permeability and **H** is the magnetic field strength or magnetic intensity. The magnetic torquers on SUNSAT use a PWM technique to control the amplitude of the generated magnetic dipole moment. Weight, power consumption and bulk of the magnetic coils influence the selection of the coils.

In the following chapter the interfacing of the sensors and actuators to the ADCS on SUNSAT is discussed together with the interfacing of the simulated sensors and actuators to the interface test circuit.

Chapter 3

Interface Simulation Test Circuit Design

The first section of this chapter discusses the features on the micro-converter. The section also gives reasons why the ADuC812 is the interface micro-controller of choice. The remaining sections discuss the design and interfacing of different parts of the test circuit, with the simulated senŝors and actuators.

Further more the simulation of magnetometer, manetorquers, and horizon/sun sensor data is discussed in this chapter. Finally the interfacing of the sensors and actuators to the ADCS main board is also discussed in this chapter.

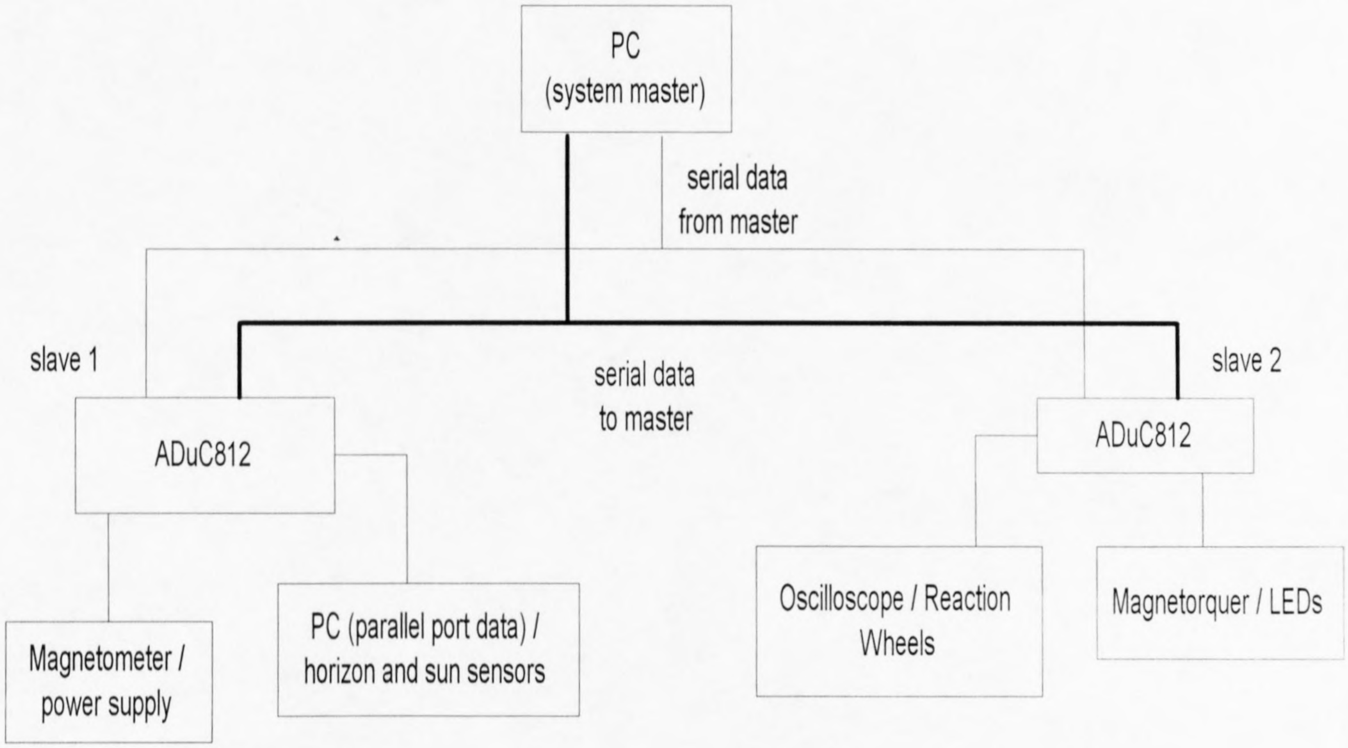


Figure 3.1: Master / Slave Serial Network

3.1 Sensors and Actuators Controller

The controller is used to read the magnetometer voltage outputs and horizon sensor data and transmit this data serially to the simulated ACP via the UART. The micro-controller is also used to give pulses to the reaction wheels and control bits to the MOSFET driving the magnetorquers. The sensors and actuators were interfaced as shown in figure 3.1.

The controller of choice is the ADuC812 micro-converter manufactured by AD (Analog Devices). The ADuC812 is a fully integrated 12-bit data acquisition system incorporating a high performance self-calibrating multi-channel ADC, two 12-bit ADCs and programmable 8-bit (8051-compatible) MCU on a single chip. 8K bytes Flash/EE program memory, 640 bytes Flash/EE data memory and 256 bytes of on-chip data RAM support the programmable 8051-compatible core. The additional MCU support functions include Power Supply Monitor, 32 Programmable I/O lines and Standard full duplex UART Serial Port I/O are provided for multiprocessor interfaces and I/O expansion. When clocked at 12MHz the current needed when the processor is running from 5V power supply is 26mA, and 15mA in idle mode and 50μA when in power down mode.

The ADuC812 has three 16-bit Timer/Counters, namely: Timer 0, Timer 1 and Timer 2. The on-chip ADC has been designed to run at a maximum speed of one sample every 5μs (i.e., 200kHz sampling rate).

The controller has an integrated ADC and DAC circuits, space is saved on the circuit board, since fewer components are needed to interface with the 8051-compatible core. The code can be downloaded to the ADuC812 via the standard UART serial port of the PC while the chip is in a target circuit.

3.2 Magnetometer Interfacing

3.2.1 Interfacing the Magnetometer to the ADCS

The magnetometer is interfaced to the ADCS through the magnetometer piggyback plug shown in Appendix B.5. The output pins MAGM1, MAGM2 and MAGM3 carry analogue voltage output proportional to the magnetic flux measured along the X, Y and Z axes of the magnetometer respectively. Pin MAGM4 carries an analogue voltage proportional to the temperature of the magnetometer housing.

The process of converting from analogue signals to digital signals is done by the analogue to digital converters.

3.2.2 Interfacing the Differential Power Supply to the Micro-Controller

To simulate the magnetometer bipolar output voltage and also to power the Op-Amps, dual tracking dc power supplies were used. The Op-Amps and the Analog to Digital Converters (ADCs) provide the interface between the power supply and the micro-controller. The ADCs are the integral part of the micro-converter (ADuC812). The Op-Amps, are employed to convert the differential output voltage from the power supply into single ended dc voltage. The ADC converts the single ended voltage into a 12-bit digital format.

The input differential voltage to the Op-Amps was varied to simulate the changing magnetic field strength. The software program (Java) was written to read the digital data (converted analog data) from the micro-controller). This Java program reads the data from the serial port of the PC (Personal Computer) and displays the data on the screen. The functions of the Java program are explained further in chapters 4 and 5.

3.3 Horizon and Fine Sun Sensor Interfacing

3.3.1 Interfacing the Horizon and Fine Sun Sensors to the ADCS

The two horizon sensors and the fine sun sensor contain linear CCDs and identical circuits which detect a transition from light to darkness [4]. The two horizon sensors detect the image of the earth's horizon using the CCDs. The fine sun sensor uses the narrow slit, which allows a strip of sunlight to fall on its CCD. Each CCD contains 2048 pixels. The transition from light to dark can be represented by 11-bit values, one value for each sensor.

The horizon and fine sun sensors interface with the ADCS tray through the horizon-sun sensor piggyback plug shown in Appendix B.4. This piggyback plug is the same type used on SUNSAT.

D_REG_0

A7	A6	A5	A4	A3	A2	A1	A0

D_REG_1

BB	B10	B9	B8	AA	A10	A9	A8

D_REG_2

B7	B6	B5	B4	B3	B2	B1	B0

D_REG_3

S7	S6	S5	S4	S3	S2	S1	S0

D_REG_4

-	-	-	-	SS	S10	S9	S8

Figure 3.2 Organization of Registers D_REG_0 to D_REG_4

Symbol	Name and Significance
A10..0	X-horizon sensor data: 11-bit value representing X horizon transition
B10..0	Y-horizon sensor data: 11-bit value representing Y horizon transition
S10..0	fine sun sensor data: 11-bit value representing sun transition
AA	X-horizon sensor saturation bit
BB	Y-horizon sensor saturation bit
SS	fine sun sensor saturation bit

The horizon / sun sensors are digital outputting devices with the hardware containing five 8-bit registers. The three horizon sensor registers are accessed through pins D_REG_0, D_REG_1 and D_REG_2. The 8-bit data from the registers is read by the micro-controller via port 8031_D (Appendix B.4). Pins D_REG_0 to D_REG_4 may only be driven low one at a time, other wise bus contention will occur. The organization of the horizon and sun sensor data registers on SUNSAT is as shown in figure 3.2. The same organization of the registers is proposed for testing the interface circuit.

The two fine sun sensor registers are accessed through pins D_REG_3 and D_REG_4. Pins AA, BB and SS receive bits to indicate if the CCD is saturated. These bits AA, BB and SS are stored in registers D_REG_1 and D_REG_4 (see Figure 3.2).

3.3.2 Horizon and Fine Sun Sensor Data Simulation

The two horizon sensors and the single fine sun sensor contain linear CCDs. Each CCD contains 2048 pixels. The pixels are averaged and a threshold value is then used to determine a valid horizon condition. The threshold is represented by three 11-bit values, one for each sensor.

To test the ADuC812 software (code in slave 1 in figure 3.1), the micro-converter was interfaced to the parallel port of the PC via port 0 and port 2. Port 0 lines are used to read in the digital parallel data from the PC. The Java code (horizon / fine sun sensor data) running in the PC is used to send 8-bits of data to the micro-controller. The 8-bit data simulates the data coming from the five horizon and sun sensor registers. Port 2 pins of the micro-controller are used to select the register (379H of the PCs parallel port) lines. The five bits (3 to 7) of the printer status register (379H) can be made low via the external lines from the micro-controller. When one of the lines is made low, data is output from the PC parallel port via the printer data register (378H).

As soon as one of the lines of port address 379H goes low and is detected, the Java code writes out the parallel data from the PC to the micro-controller. Making one of these lines (port address 379H lines) low corresponds to selecting one of the horizon and sun sensors registers. Only one line at a time is made low to prevent bus contention.

Using port 0, the micro-controller reads in this data and sends it serially to another PC (network master). Each parallel data read subroutine execution in the micro-controller reads in five bytes.

3.4 Magnetorquer Interfacing

3.4.1 Interfacing the Magnetorquer to the ADCS

The magnetorquers can be switched on and off using the pulse width modulation technique. The magnetorquers are magnetic torquer coils. Appendix B.2 shows the schematic of the MOSFET drivers for the X1 (x-axis) magnetic torquer coil. The MOSFET driver circuits are identical for all the coils. The MOSFET drivers directly interface to the ADCS via the 80C51. The 80C51 provides the polarity status bits to the MOSFET drivers. The voltage at pins SX1A and SX1B (see Appendix B.2) control the current in the coil. Pins SX1B and SX1A can have two discrete voltages. The voltage can be 14V and 0V (Ground), similar to SUNSAT. The material and resistance in the coil can influence the size of the discrete voltages.

The differential voltage between SX1A and SX1B creates the magnetic dipole on coil X1. If there is no differential voltage then current flow is zero, and the magnetorquer is off. If SX1A is at 14V and SX1B is at ground then the current will flow in the direction of SX1B.

The magnetic torquer coils interface with the ADCS via the magnetorquer piggyback plug, shown in Appendix B.6. Coil X2, which is along the x-axis, is connected to SX2A and SX2B.

Coil Y1, which is along the y-axis, is connected to SY1A and SY1B. Coil Y2, which is also along the y-axis, is connected to SY2A and SY2B. Coil Z1, which is along the z-axis, is connected to SZ1A and SZ1B, and Z2 is connected to SZ2A and SZ2B.

3.4.2 Magnetorquer Status Test Circuit

Two status bits are used to control each coil, i.e. X1A and X1B.The bits come directly from the micro-controller. To test the status of the coils the circuit shown in Appendix B.3 was constructed. The circuit is made out of LEDs. When the status of the bit corresponding to an LED is high, the LED is off. When the status of the bit changes to low the LED is switched on. The LEDs were interfaced to port2 and higher nibble of port3 of the ADuC812 (slave 2). The LEDs' circuit connected to the higher nibble of port3 is the same as the one in Appendix B.3 except that it consists of only four LEDs.

3.5 Reaction Wheel Interfacing

The reaction wheel is a pre-manufactured unit and the interface is designed with the configuration of the reaction wheel in mind. The reaction wheels are servo motor components with brushes or without brushes. For the servomotor without a brush the digital control system can be configured as shown in figure 3.3 below.

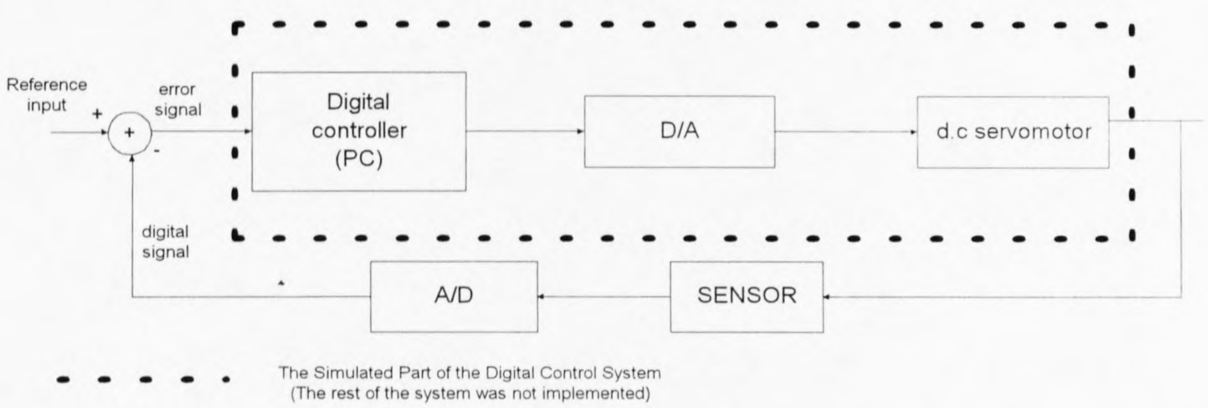


Figure 3.3 Digital Control System

The D/A converter is employed to convert the digital control data from the PC, into a suitable analogue signal (i.e. voltage, current) for the motor. It should be noted that the actual reaction wheels were never interfaced to the interface test circuit. In testing the control voltage for the reaction wheels the voltage from the D/A converter was measured and compared to the calculated D/A voltage. The output of the D/A converter after amplification can drive the motor to the required velocity or position [10].

Chapter 4

Software Description

In this chapter, the execution flow of the test programs is discussed. The test programs discussed include the horizon/ fine sun sensor data simulation program, the ACP (network master) simulation using Java, the simulated sensor read program and the simulated actuator control program.

4.1 High Level Language

When writing in C, a compiler will get the code from high-level language to machine codes. If assembly language is used, an assembler will get the code from mnemonics to the equivalent numeric code. Using either language of the two mentioned above the initialization of the SFRs will require the knowledge of the structure of the SFRs. The amount of reusable code in assembly language is usually small and therefore the division of code into functions, as is done in C, results in better structure of the software. The functions can be brought from one project to another.

4.2 The Main Functions of Micro-Controller Software

The codes in the ADuC812 micro-converters read data from the simulated attitude sensors that include the horizon/ fine sensors and the magnetometer control the simulated actuators that include the 3-axis magnetorquers, and transmit sensor data to the simulated ACP. The ADuC812 performs these functions after receiving the correct commands from the PC.

4.3 Polling and Interrupt Driven Data Transfers

The polling process is preferred when the CPU spends most of the time doing no work. Polling is a method of using the resources for simplifying both the hardware and software associated with an application. The polling method uses resources even though service is not wanted at the particular instance.

In the partial system for the thesis, polling is utilized in the code that simulates the horizon and sun sensor data. The Java code polls the parallel port of the PC to detect if any of the lines corresponding to port address 379H is low. If any of the lines is low, then the PC will output the corresponding byte to the micro-controller through the parallel port. For most of the time this Java program does nothing other than polling the parallel port.

When the timing or CPU resources become problematic, interrupt driven data transfers become an alternative. The wait for poll time associated with polling can be too long and this can sometimes lead to dry polls. There are no dry polls associated with interrupt-driven data transfers and interrupt service can begin immediately after the request is detected.

The interrupt-driven data transfer is implemented by the codes running in the ADuC812 micro-converters. The interrupts used include the serial port, Timer0, Timer1 and the ADC interrupt.

4.4 An FSA (Finite State Automation)

The FSA is a software machine that moves from one state to another based on the input received. This is called state transition. The FSA is employed to parse the incoming byte stream into a valid message. The FSA has an initial state in which it is looking for a synch/start byte. After the initial state, several intermediate states that correspond to the next expected part of the message follow. The final state reads and verifies the checksum. The FSA will return to the initial state if at any stage a byte is received that does not agree with the format of a valid message.

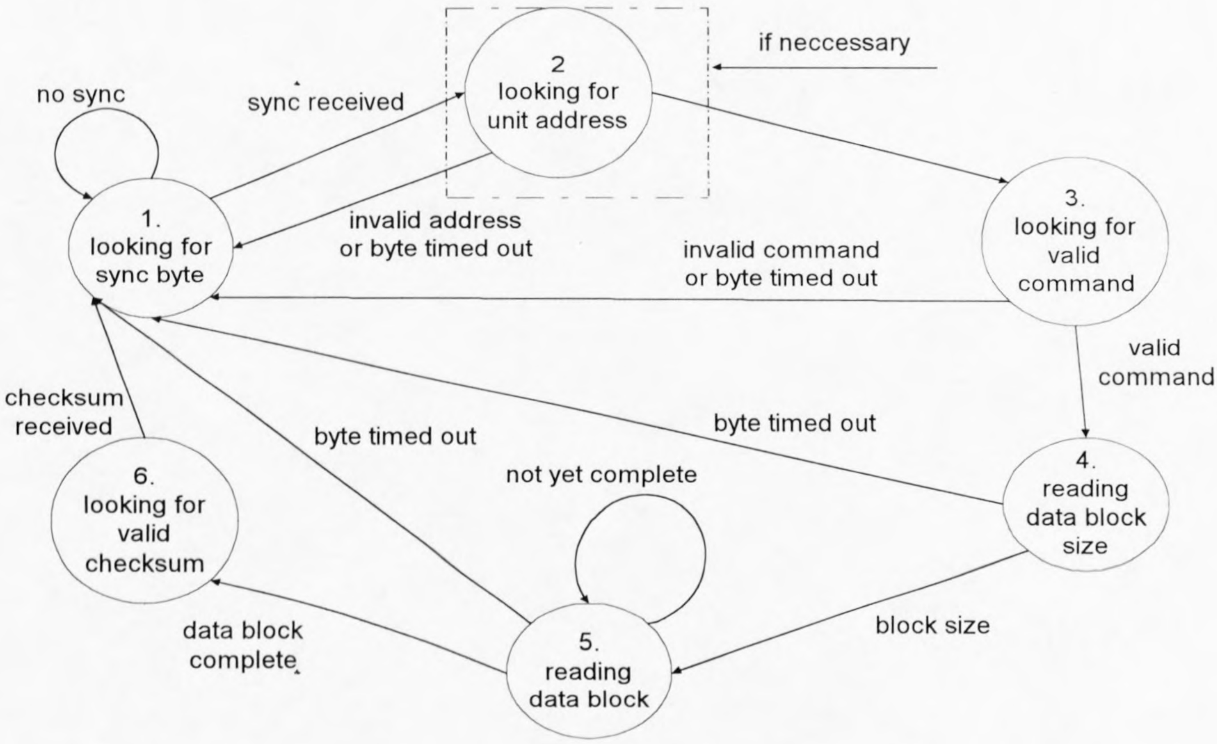


Figure 4.1: General Receive FSA

When a valid message is received it is parsed into the queue and processed by the main loop. The FSA system allows the micro-controller to spend more of the time working on tasks other than serial communications. The model in figure 4.1 represents the code that runs the FSA.

The code that runs in both slaves implements the FSA method. Slave2 receives messages containing the control data for the reaction wheel and magnetorquers. Slave2 also sends an acknowledgement to the PC if the message is received correctly. If the message is invalid slave2 will send the NACK packet. The simulated sensors are only read after the correct message is received. If incorrect message is received, the PC resends the message.

4.5 Horizon / Fine Sun Sensor Data Simulation Program

This section presents the execution flow of the horizon /fine sun sensor data simulation program represented by a flow diagram in figure 4.2. The program is written in Java and runs in the PC.

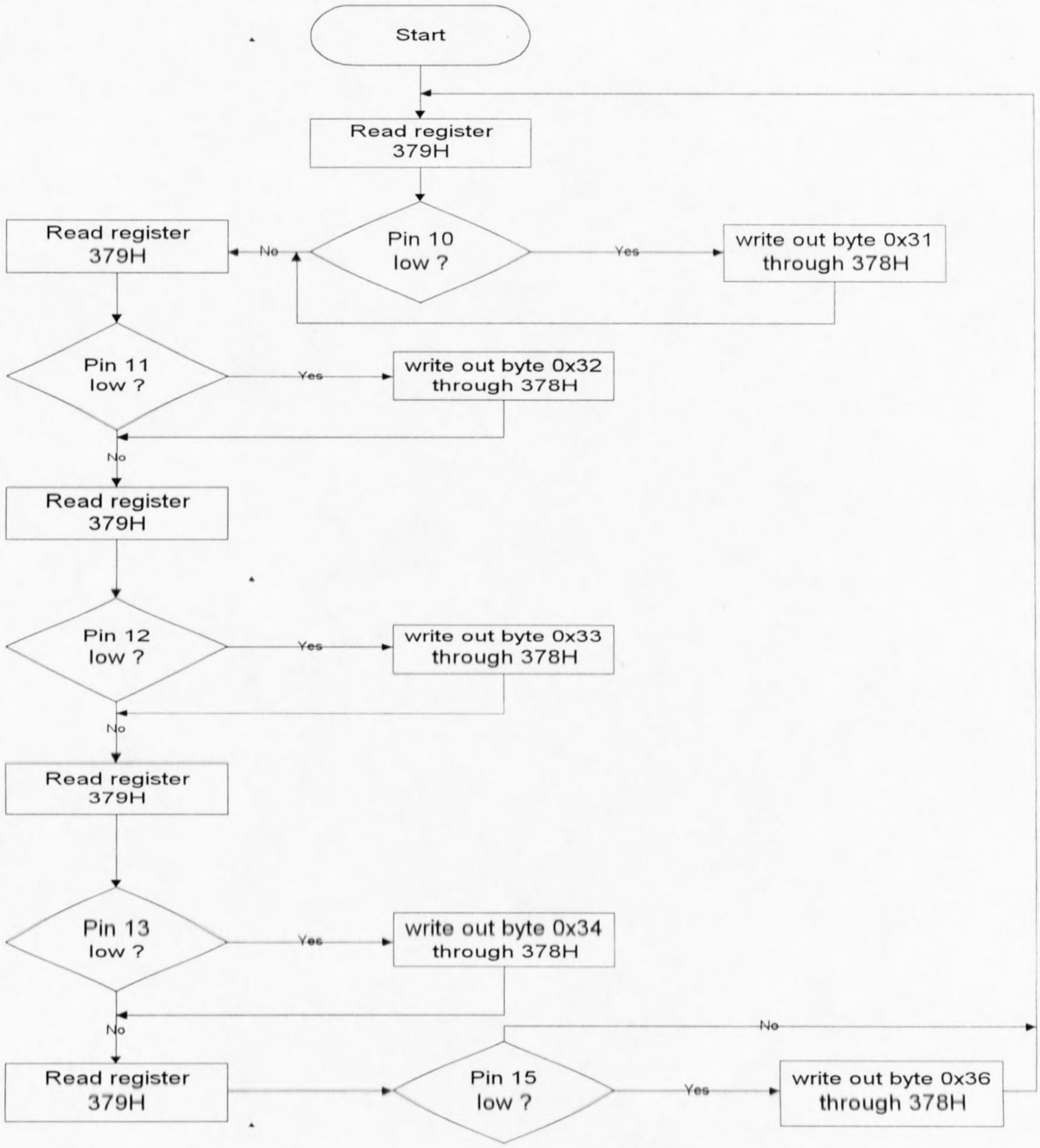


Figure 4.2: Horizon / Fine Sun Sensor Data Simulation Program Flowchart

The program polls the parallel port of the PC to detect if any of the lines corresponding to port address 379H is low. If any of these lines is low, the PC outputs the corresponding byte to the ADuC812 through the parallel port (register address 378H). For most of the time this Java program does nothing other than polling the port.

Only one line (port address 379H lines) at a time is set low by the micro-controller to prevent the PC from sending more than one different byte before the data is read by the micro-controller. The data on the parallel lines changes after a new byte has been output.

4.6 Sensors Reading Program

This section presents the execution flow of the A/D and digital parallel data reading program represented by a flow diagram in figure 4.3. This program runs in slave1 micro-controller.

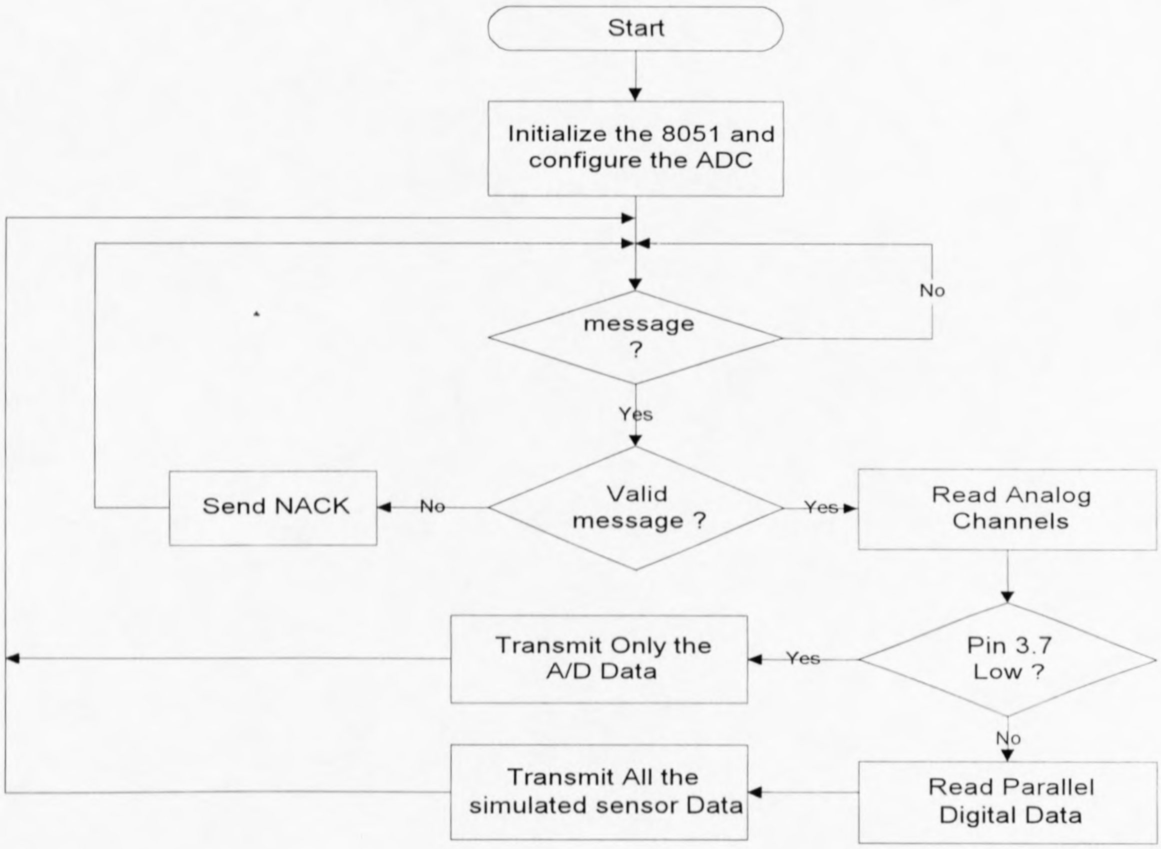


Figure 4.3: Process Running in Slave1

After initialing the ADuC812 the program stays in the infinite for-loop waiting for the messages. When the message is received its validity is checked. The micro-controller responds with the NACK to the invalid message.

The valid message for slave1 carries the read sensor command. The analog-to-digital channels are read first. If pin 3.7 is low the parallel digital data (horizon / fine sun sensor) will not be read. The external line controls the status of pin 3.7. The status of pin 3.7 simulates the Tele-command signal indicating if the horizon /fine sun sensors are on or off.

4.7 Actuator Control Program

The actuator code that runs in slave2 implements the FSA method described in section 4.4. Messages containing the control data for the simulated reaction wheels and magnetorquers are received in slave2. Slave2 also sends an acknowledgement to the PC if the message is received correctly. If the message is invalid slave2 will send the NACK packet and the PC will have to re-transmit the message.

The interrupts implemented in slave2 code are Serial Port and Timer 0 interrupt. Every second the PC sends the reaction wheel control data, and if the message is not received correctly by slave-2, the NACK message will be sent to the PC and on receiving the NACK message the PC sends the message again.

The PC sends the magnetorquer control data to slave2 every 10 seconds. The slave program switches on the magnetorquers for 8 seconds over the 10 seconds period. To have the magnetorquers switch on for periods less than 8 seconds, slave2 code should have a function that calculates the period according to the message received from the master (PC). When the 10ms interrupt occurs, the 10ms counter is incremented. When the 10ms counter reaches 100 it is re-initialized to zero and the Second-counter is incremented. When the Second-counter is one, the magnetorquers are switched on, and when the Second-counter reaches 9, timer0 is stopped and the magnetorquers are switched off. It should be noted that pairs of LEDs that are both on or off simulate the off magnetorquers.

The flow diagram shown in figure 4.4 represents the process that runs in slave2.

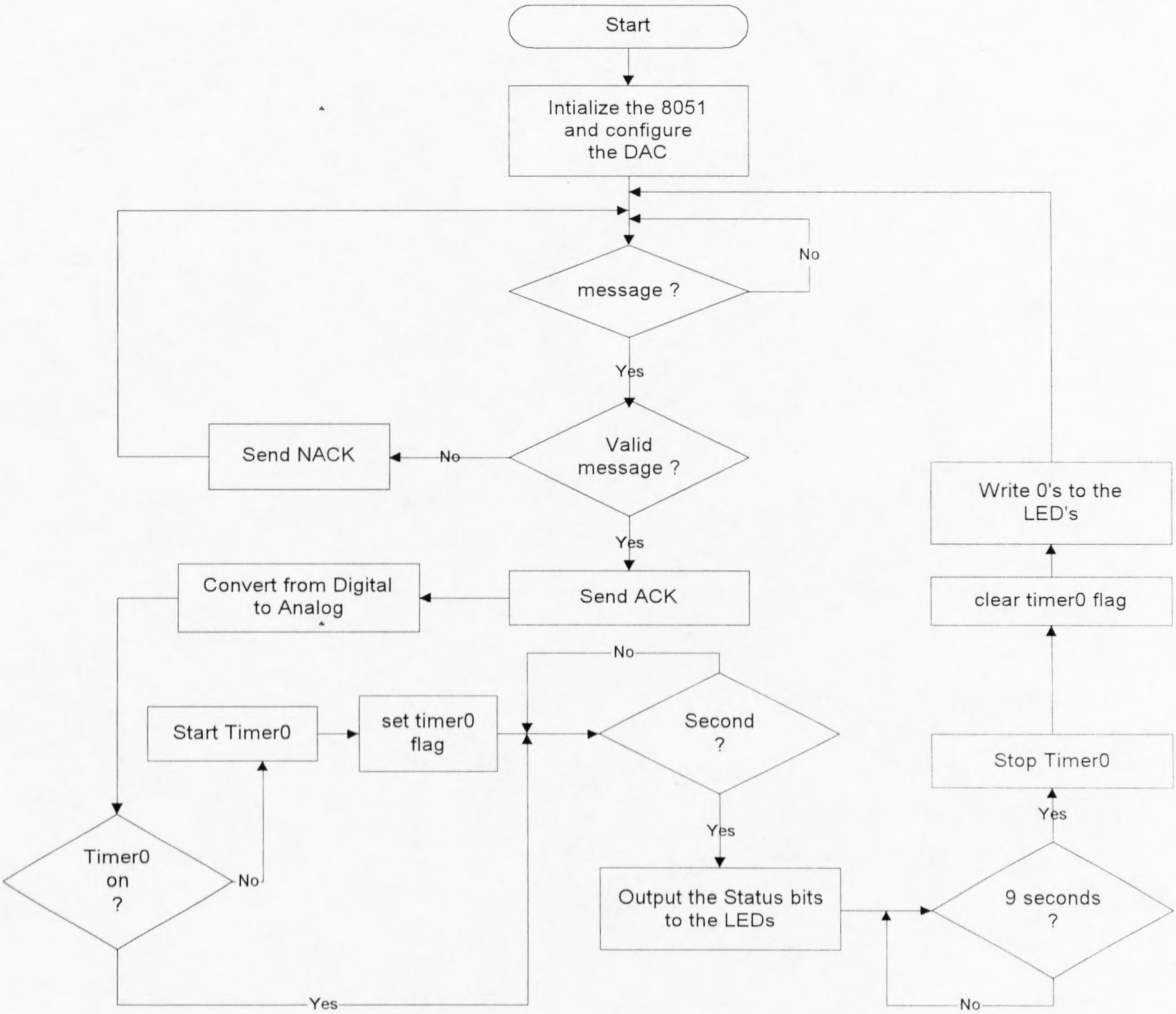


Figure 4.4: Process Running in Slave2

The handling of the 10ms counter and the Second-counter is done inside the Timer0 interrupt service routine. The interfaced LEDs will be switched on with the status held in the receiving buffer.

4.8 ACP Simulation Program

The ACP simulation program is written in Java and runs in the PC. The PC communicates with the slaves on the RS-485 network. Receiving the sensor data and sending of actuator control messages are the only functions of the ACP simulated by this program. The communication protocol between the slaves and the PC (master) is discussed in chapter 5.

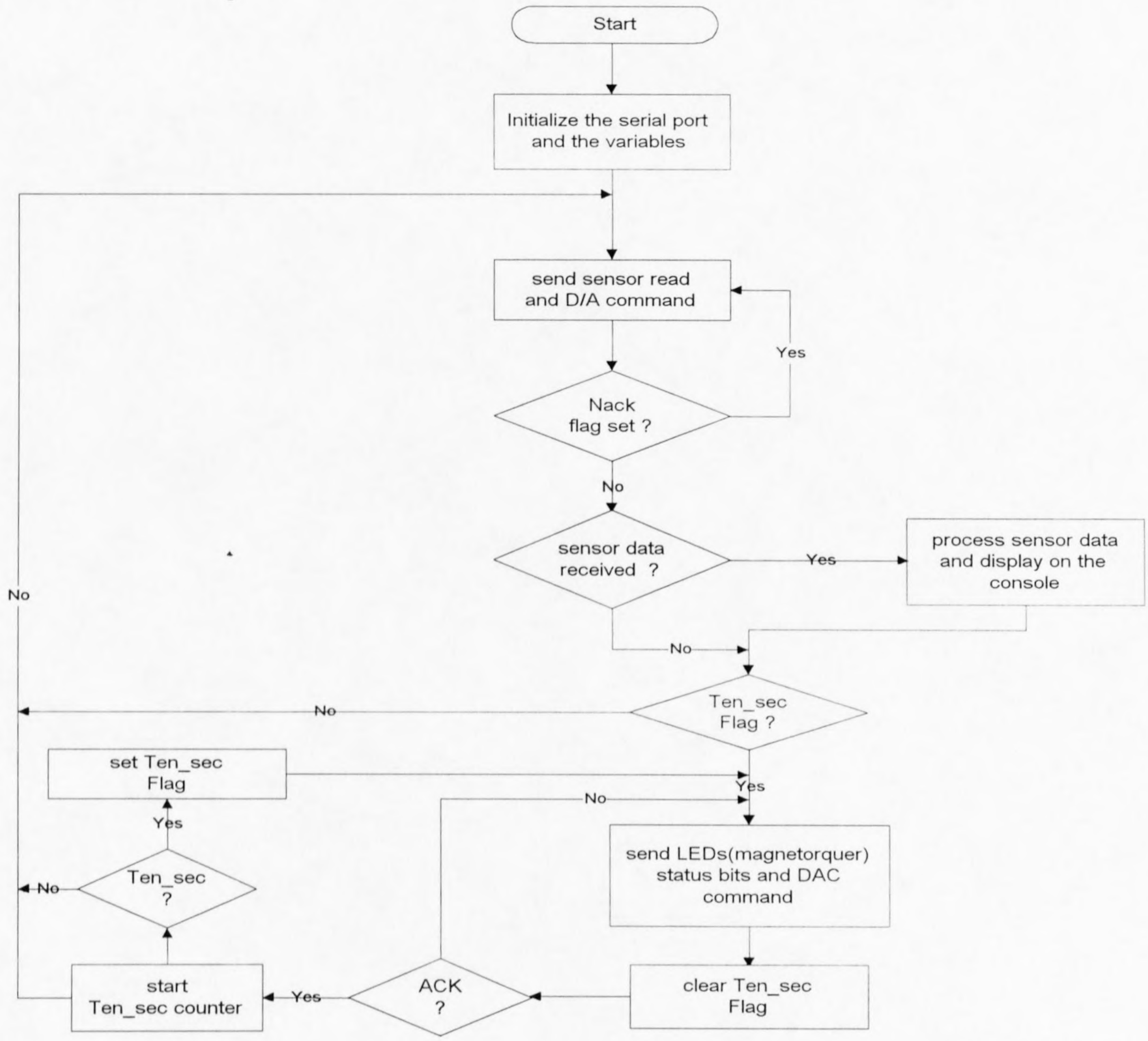


Figure 4.5 ACP Simulation Program Flow Diagram

The program is written in Java and is represented by the flow diagram in figure 4.5. The program stays in the loop sending out messages to the slaves. The control data for LEDs (magnetorquers) is send every ten seconds and for DACs the control data is send every second.

The incoming bytes to the PC serial port are handled after the serial port event is invoked. This Java program is also able to convert the A/D conversion bytes into the decimal format. The sensors' data is displayed on the console to verify the measurements done at the A/D input channels.

The following chapter will look at the implementation of the communication protocol between the master and the two slaves. The communication data rates between the slaves, the master and the horizon /fine sun sensor simulator, are also discussed.

4.9 Communication Protocols

This section discusses the message formats of the protocol implemented for testing the sensor and actuator interface test circuit. The messages discussed here are the acknowledgement messages from slave1 and slave2 going to the PC and messages from the PC to slave1 and slave2. The master initiates communication with the slaves. Data from the sensors is read first, then the simulated actuators are communicated with.

4.9.1 Messages from slave1 to the PC (Master)

On the bus network shown in figure 3.1 there is only one master (PC). Since there is one master, slave1 and slave2 can send messages containing sensor data and acknowledgements to the PC without giving the PC's address. There would be chaos if the slaves had more than one master on one RS-485 network.

The message structure for this protocol is as follows:

Start Byte	Slave 1 Address	Type Byte	Data	Checksum	
1	**1**	**1**	**n**	**1**	**: Byte Count**

Figure 4.6 Message Structure from Slave 1 to Master Protocol

In the case of slave1 message format, two types of messages exists. The first is the message that carries all the sensor data. The second type of the message is the one that carries only the A/D conversion (magnetometer) results when the Horizon / Fine Sun sensors are off.

The overhead percentage in the message can be given by the following equation [8]:

$$\text{Overhead Percentage} = 1 - N/(C + N) \quad 5.1$$

where N, is the number of data bytes and C is the number of control bytes.

The equation above shows that an increase in the number of data byte leads to the decrease of the overhead percentage. This means when slave1 transmits only the A/D conversion data, the overhead percentage in the message increases.

4.9.2 Messages from the PC to the Slaves

The message that the PC sends to slave1 does not contain any data bytes. This message only requests slave1 to perform sensors reading and transmit data to the PC. The message that the PC sends to slave2 contains the data bytes. These data bytes carry the control bits for the LEDs (magnetorquers). The D/A conversion bytes are also carried in the data field of the message.

Start	Slave Address	Type Byte	Size	Data	Checksum	
1	**1**	**1**	**1**	**n**	**1**	**: Byte Count**

Figure 4.7 Message Structure for Master to the Slaves Protocol

The checksum byte is the single byte addition of all the bytes in the message ignoring rollover. This type of structure is suitable for the FSA driven programs running in the slaves. The type byte field is for the type of command given to the controller. Two types exist for slave2. The first type is for the magnetorquer and D/A conversion command sent every ten seconds. The second type is only for the D/A conversion sent every second.

4.9.3 ACK Messages from the Slave to the PC

The ACK messages from the slaves to the PC have the following structure:

Slave Address	Type Byte	ACK	Checksum	
1	**1**	**1**	**1**	**: Byte Count**

Figure 4.8 ACK Message Structure

In this message structure the field, type byte, represents the type of message being acknowledged. This message structure contains no data field. The slave address field identifies the origin of the ACK message. The NACK message takes the same form as the ACK message.

Chapter 5

Discussion on the Results of the Study

5.1 Hardware and Software Analysis

In this chapter the conversion, reading and transmission rates are presented. The transmission rates, includes the total time it takes for the micro-controller to read the data from the simulated sensors and transmit this data to the PC. The communication protocol between the PC and the two slaves is also discussed in this chapter.

5.1.1 Magnetometer

The dual tracking power supply was used to simulate the magnetometer.

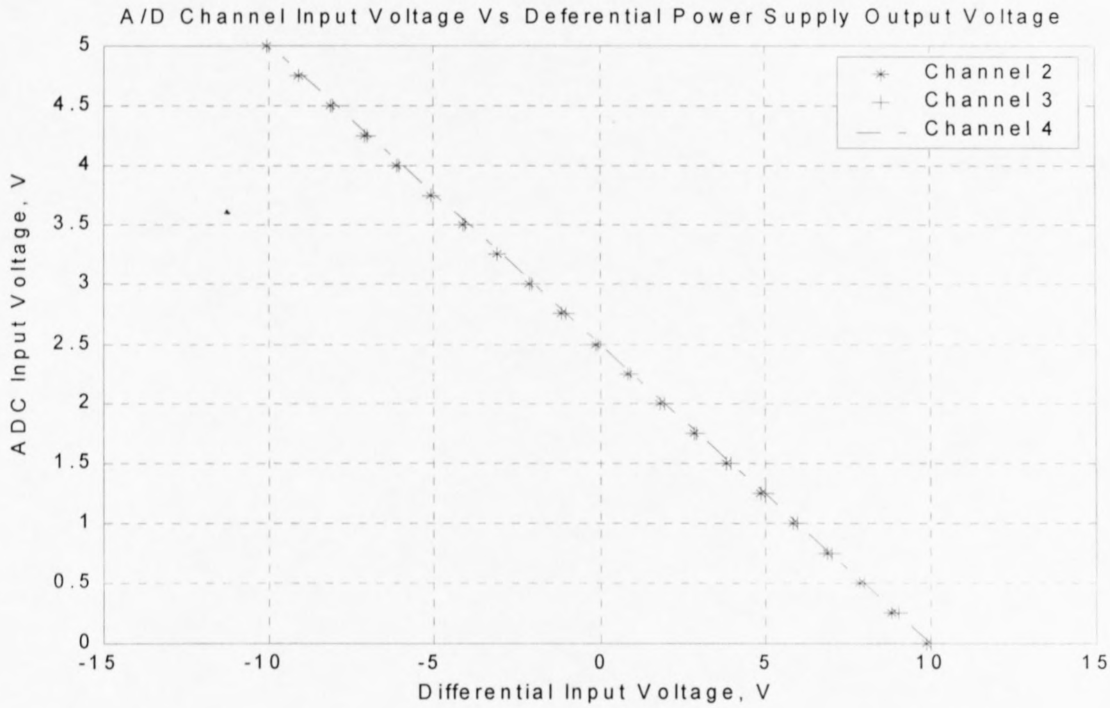


Figure 5.1: ADC Single Ended Channel Input Voltage Vs Differential Input Voltage

The Op-Amps (see Appendix B.1) were employed to change the differential voltages into single ended voltages. Figure 5.1 is the plot of the measured single ended voltages versus the corresponding measured differential voltages. The graph shows that the single ended voltage changes linearly with the change in the differential voltage over the range, -10V to +10V. The range of change for the single ended voltage over this differential voltage change range is 5V to 0.

To convert the analogue voltages to digital format the micro-converter (see section 3.1) was used. The selection of the channel to be converted is done via the internal multi channel multi-plexer. Writing to the ADC's SFR, ADCCON2 controls the multi-plexer and selects the channels (see Table 5.1 below). Three channels are employed to convert the simulated magnetic field strength and the fourth one converts the analogue voltage proportional to the temperature of the magnetometer housing.

Table 5.1 Bytes for Channel Selection

CHANNEL	**ADCCON2**	**INPUT (SIMULATED)**
Channel 2	00000010	X Magnetic Field Component
Channel 3	00000011	Y Magnetic Field Component
Channel 4	00000100	Z Magnetic Field Component
Channel 6	00000110	Temperature of Sensor Housing

The period for converting all four channels, plus the time taken to load the A/D data into the transmission buffer is, 155μs, which translates to 38.75μs per channel. It should be noted that the increase in the number of channels converted results in the increase in total conversion cycle time. The ADuC812 has the total of 8 A/D conversion channels. The conversion cycle takes 140μs (35μs per channel) to complete without the data being loaded into the buffers. This shows that the loading of buffers can be done within the ADC conversion subroutine without impacting too much on the time taken to complete the reading of A/D channels.

The total A/D conversion rate of 155μs (6.45kHz) together with the parallel data reading discussed in the next section, and the transmission of data to the PC takes less than one second to complete. This will be shown in the next section.

5.1.2 Horizon / Fine Sun Sensor

To simulate the five Horizon / Fine Sun sensor registers, the parallel port of the PC was used (refer to section 3.3.1 for an explanation on the organization of the registers). Port0 of slave1 was used to read the parallel digital data from the PC. Five lines from port2 were used to select (address) register 379H of the parallel port. It should be noted that port2 can be configured to address 8 registers if required, but under this condition the PC's parallel port can not be employed to simulate the horizon / fine sun sensor since the parallel port has only five input pins [11]. Setting one of the five lines low corresponds to addressing one of the horizon/fine sun sensor registers.

The horizon / fine sun sensor data was simulated with the use of the Java program. Only one line at the time is set low to prevent the PC from sending more than one different byte before the data is read by the micro-controller.

Addressing port 379H of the PC was done as shown in table 5.2.

Table 5.2 Horizon / Fine Sun Sensor Simulation Addresses

SIMULATED REGISTER	ADDRESSES FROM PORT 2	BYTES OUTPUT BY THE JAVA PROGRAM
D_REG_0	7FH	00110001
D_REG_1	BFH	00110010
D_REG_2	DFH	00110011
D_REG_3	EFH	00110100
D_REG_4	F7H	00110110

The manipulation of the digital data from the PC (horizon/fine sun sensor registers) into 2-bytes for each sensor is done by the program running in slave 1. In total, there are 6 bytes produced by the manipulation, 2-bytes for the fine sun sensor and 4-bytes for the two horizon sensors.

The subroutine to read and load the digital parallel data into the transmission buffer takes 280ms to execute. This 280ms-subroutine period is 35000 times slower than the rate at which all five pins of register 379H are polled by the Java code. The five pins are polled at the rate of 125kHz. This polling rate ensures that no register 379H pin can go low and high again without being detected. Each one of the five lines stays low for 42.5ms.

The micro-controller takes 750ms from the time it receives the first byte of the message from the PC, to the time that the micro-controller transmits to the PC the last byte (checksum) of the data message. This 750ms period can allow slave1 to send data to the PC every second if it were the only slave on the RS-485 network.

The test RS-485 network has two slaves connected to it. Slave 2 receives the control data for the actuator simulators and sends back an acknowledgement. The execution periods for slave2 code are presented in the next two sections.

5.1.3 Magnetorquer

The interface between the ADuC812 (slave2) and the magnetorquer is one way. This means the ADuC812 sends control bits to the magnetorquers and the magnetorquers send nothing in return. The magnetorquers are interfaced to the 80C51 via the MOSFET drivers. There are 6 MOSFET drivers on SUNSAT, one for each coil and two coils per axis.

To test the validity of the control bits, the circuit in Appendix B.3 was constructed. This circuit was interfaced to port 2 of the micro-controller. Another similar circuit but with only four LEDs, was interfaced to the upper port 3 nibble pins of the same micro-controller (slave 2). With this interface the validity of all the control bits can be tested. When the LED is off logic 1 state at the port pin is represented and when the LED is on logic 0 is represented.

Table 5.3 Magnetorquer Simulator Control Bits

Port 2 (Slave 2)	Simulated Control Bits	Port 3 (Slave 2)	Simulated Control Bits
P2.0	X1A	P3.4	Z1A
P2.1	X1B	P3.5	Z1B
P2.2	X2A	P3.6	Z2A
P2.3	X2B	P3.7	Z2B
P2.4	Y1A	-	-
P2.5	Y1B	-	-
P2.6	Y2A	-	-
P2.7	Y2B	-	-

Every ten seconds the PC (simulating the ACP) sends the message with control bits. On receiving the correct message the micro-controller sends back the acknowledgement packet. The program running in slave 2 has the counter that allows the magnetorquers to be on for up to 8 seconds over a period of ten seconds. The 8seconds period is not fixed and has a resolution of 10ms. To have the magnetorquers switch on for periods less than 8 seconds, slave2 code should have a function that calculates the period according to the message received from the master (PC). The control of the time counters is done within the 10ms Timer 0 Interrupt Service Routine. The updating of port2 and 3 is done simultaneously. The micro-controller takes 12ms from the time it receives the first byte of the message from the PC, to the time that the micro-controller transmits to the PC the last byte (checksum) of the ACK/NACK message

5.1.4 Reaction Wheels

The input variable of the controller for the required reaction wheel speed comes from the D/A converters. The D/A converters output voltages that range from 0 to V_{REF} according to the ADuC812 specifications document [12]. The measured voltage range on DAC0 and DAC1 pins for V_{REF} equals to 5.40V starts at -0.14V to 5.38V. To protect the DACs from short circuits, the

Op-Amps is employed (see Appendix B.1). The measured output voltage range of the Op-Amps is -2.50V to 3.00V.

Figure 5.2 shows the divergence of the measured DAC outputs from the calculated outputs.

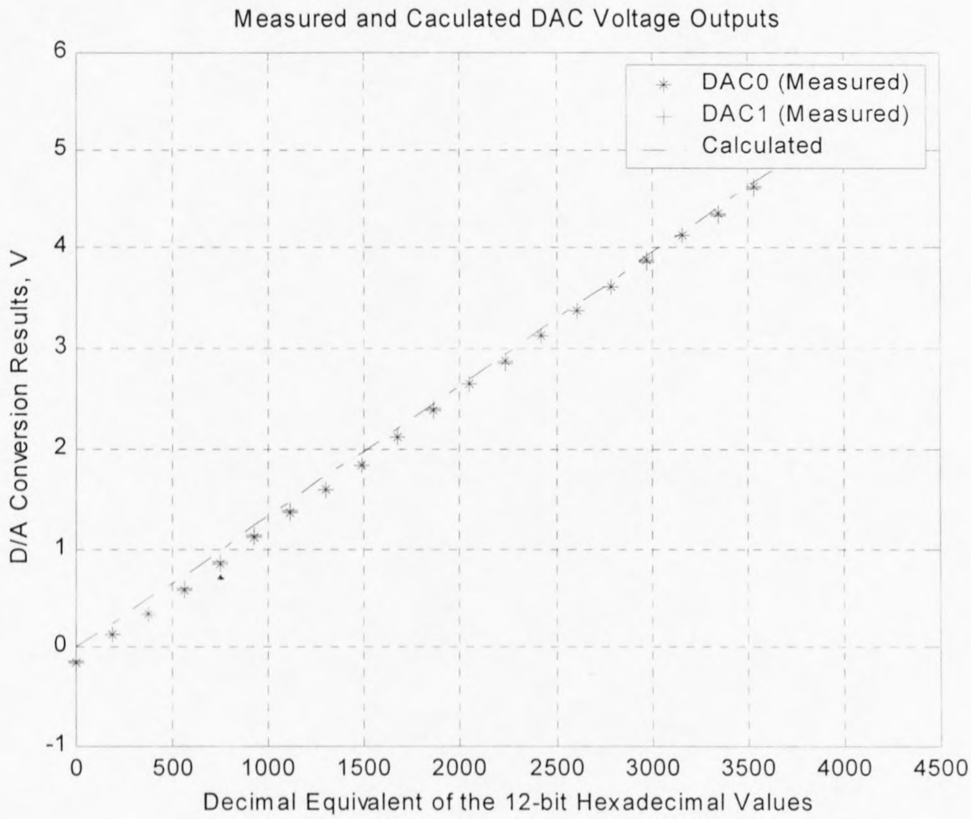


Figure 5.2 Comparison between Measured and Calculated DAC Outputs

The DAC outputs were calculated from the 12-bit hexadecimal values. On the average the measured values differ from the calculated values by 77mV. The 77mV value means that on the average the difference between the measured and the calculated analog outputs is within 1.43%. The 1.43-% difference is partly due to the load caused by the Op-Amps connected at the DAC pins and also the measured V_{REF} was 5.37V dc.

Figure 5.2 shows that at high voltage outputs the difference between the measured and the calculated DAC outputs becomes smaller, and at these voltages the ADuC812 will work reasonably well to produce the calculated high reaction wheel speeds. For fast maneuvering the reaction wheels have to attain high speeds that are functions of high voltages.

Chapter 6

Conclusion

Three simulated sensors and two actuators were interfaced to the test circuit. These sensors and actuators combined represent the analog and digital attitude determination and control components. Horizon /fine sun sensor are digital outputting devices whereas the magnetometers are the analog outputting sensors. The reaction wheels are analog actuators and the magnetorquers are digitally controlled.

On SUNSAT the magnetometer data is requested every second. This means all four A/D channels are read in less than one second. The ADuC812 can read four A/D channels within a period of 155μs, and is also capable of converting eight A/D channels. The total conversion time increases to 315μs for eight channels. With, the total conversion time of 315μs the A/D data can still be send to the ACP every second. The test circuit was designed for the analog input voltage resolution of 4.8mV (see figure 5.1). The magnetometer on SUNSAT has a resolution of 2.4mV or output swing of ±5V if the 12-bit A/D converter is used. This finer resolution can be obtained by changing the ratio of the resistors connected to the Op-Amps (change the ratio from 1:4 into 1:2 and the input voltage range from ±10V into ±5V). In general magnetometers with a wide range of output swing voltages can be interfaced to this interface test circuit.

The horizon/ fine sun sensor data was simulated using the Java written program. Like all other sensors on SUNSAT, the horizon / fine sun sensors are read every second. This requires that all five registers be accessed in less than one second. The ADuC812 takes 280ms to read the digital parallel data from the PC. This 280ms period, added to the total time taken for all four A/D channels to be converted amounts to less than one second. Both analog and digital sensors can be read and the data be transmitted to the PC (simulated ACP) every second.

When the simulated registers are addressed one line at a time stays low for 42.5ms. This is enough time for the horizon / fine sun sensor simulation program to detect a low pin since the pins are polled at the rate of 125kHz. The ADuC812 is capable of accessing more than five registers. The PC can not be used to simulate more than five registers. The PC's parallel port has five input pins [11]. To simulate more than five registers, more than one PC's parallel port is needed.

For interfacing the magnetorquers, only the status of the coil (direction of current flow) was tested. The magnetorquers are switched on over a ten seconds period for a maximum duration of 8 seconds as is done on SUNSAT. These periods can be modified in the slave2 code to suite the attitude control requirements of the particular satellite. By using port2 and higher nibble of port3, the status of six magnetic coils can be controlled. The magnetorquer control bytes are received every ten seconds. Again this period can be modified in the network master code (simulated ACP), to satisfy the particular satellite's control requirements.

On SUNSAT, every second during pointing and stabilization the reaction wheels are commanded. The speed of the reaction wheels is the function of the control voltage. The control voltage is the output from the D/A converters. The relationship between the calculated D/A output and the measured D/A outputs is shown in figure 5.2. The size of the control voltage depends on the D/A control bytes received from the PC. Two control bytes are needed for each D/A conversion channel. While the A/D converters process the speed feedback the current reference from the D/A converters of the same ADuC812 chip can be output to the motor drive system.

The digital controlled sensors and actuators communicate with the ADuC812 using TTL level voltages. These voltages are very susceptible to noise. It is recommended that the testing of interfaces be done in areas with low electromagnetic noise interference. The bus network is less susceptible to noise since it uses the RS-485 voltage levels.

All the sensors and actuators can be interfaced to one ADuC812 with disadvantages. If the micro-controller becomes faulty, communication with all the sensors and actuators connected to the micro-controller can be disrupted. Also a faulty sensor can drive high currents into the micro-controller therefore damaging the micro-controller. Using more than one slave gives an option of distributing the sensors and actuators to avoid a total breakdown of communication between the sensors, actuators and the master (ACP) in the event of a micro-controller getting damaged. To address up to eight simulated horizon / fine sun sensor registers and use less number of pins on the micro-controller, a 3-to-8 decoder can be employed.

The advantage of using one ADuC812 is that the number of messages and acknowledgements transmitted between the master and the slaves are reduced. This can improve the communication time between the master and the slave and reduce the overheard percentage in the messages.

REFERENCES

1) Vladimir A. Chobokov : *Spacecraft Attitude Dynamics and Control* Krieger Publishing Company, Malabar, Florida, 1991.

2) James R. Wertz : *Spacecraft Attitude Determination and Control* Kluwer Academic Publishers, Dordrecht / Boston / London, 1999.

3) Robert F. Coughlin, Frederick F. Driscoll: *Operational Amplifiers and Linear Integrated Circuit Technology.* Prentice Hall, INC. Englewood Cliffs, New Jersey , 1983

4) W. H. Steyn: *Attitude Control Algorithms and Simulation Programs for Low Earth Orbit Spacecraft* (September 1990)

5) J. A. Connelly: *Analog Integrated Circuits* A Willey-Interscience Publication. John Wiley and Sons, 1975.

6) G. W. Milne, P.J. Bakkes, A. Schoonwinkel, J. J. du Plessis, J. H. R. Enslin, W. H. Steyn, S. Mostert, K. D. Palmer and D. M. Weber: *Motivation Paper for the Sunsat Unit*: Internal document: Department of Electronics and Electrical Engineering University of Stellenbosch, 1992.

7) X.C. Farr: *ICP software documentation for SUNSAT I* (29/09/1998) Internal document: Department of Electronics and Electrical Engineering University of Stellenbosch.

8) Elizabeth A. Nichols, Joseph C. Nichols and Keith R. Musson: *Data Communications for Microcomputers (With Practical Applications and Experiments)* Published by McGraw-Hill Book Company, 1982.

9) James R. Wertz and Wiley J. Larson: *Space Mission Analysis and Design* Third Edition, Published jointly by, Microcosm Press, Torrance, California and Kluwer. Kluwer Academic Publishers, Dordrecht / Boston / London, 1999.

10) Yashuki Dote and Sakan Kinoshita: *Brushless Servomotors Fundamentals and Applications.* Clarendon Press Oxford, 1990.

11) Lewis C. Eggebrecht: *Interfacing to the IBM Personal Computer.* Published by Howard W. Sams and Co.,Inc. 4300 West 62nd ST. INDIANAPOLIS, 1983.

REFERENCES FROM THE NET

12) *ADuC812 Specifications Document* : AnalogDevices (www.AnalogDevices.com). Downloaded on the 30th of May in the year 2000.

APPENDIX A

Design Calculations

A. Hardware Design Calculations

The voltage inputs to port 1 of the micro-converter must range from 0 to V_{REF}. V_{REF} is an internal reference voltage, which is 2.5 Volts. In the design the V_{REF} was overdriven by an external reference of 5V, so the input voltage can range from 0 to 5V dc. Refer to the schematic diagram in Appendix-B figure B.1 for the following calculations. Using the Op-Amp gates connected to jumper J16 and pin 1.2 of the micro-converter, the analog input voltage to pins P1.2, P1.3 and P1.4 were calculated as follows [3][5]:

$$V_{out1} = -(\frac{R4}{R1})(V_2 - V_3) \quad (1)$$

V_{out1} is an output voltage at pin 1 on IC4: A

V_2 is an input voltage at terminal 1 of J16

V_3 is an input voltage at terminal 2 of J16

The ratio of 1:4 was chosen so that V_{out1} can range from -2.5V to +2.5V. The maximum bipolar output voltage of ±5V is expected from the magnetometer. For maximum bipolar output voltage

$$V_{out1} = -(\frac{1}{4})*(-5 - 5) = 2.5 \text{ V} \quad (2)$$

and the magnetometer output voltage is 10Vpp.

The input voltage at the non-inverting side of the IC4: B is kept constant and is obtained from the voltage reference device that outputs 2.5 V dc when driven by 5V dc. The input to the inverting side of IC4: B is V_{out1}. The ratio between R_{16} and R_{13} (closed loop gain of the amplifier) is 1:1. Then in the same way as in equation (1), V_{out7} (output at pin 7 of IC4: B) was calculated as follows:

$$V_{out7} = -(\frac{R13}{R16})*(V_{out1} - 2.5 \text{ V}) \quad (3)$$

V_{out7} is also the single ended input voltage to P1.2 of the micro-converter. The calculation of the single ended input voltages to other three analogue-to-digital channels was done in the same way as above.

Gate IC3: C and IC3: D (refer to Appendix A) are used to convert the two single ended output voltages from two DAC output pins of the micro-converter into a differential voltage for use as the voltage inputs to the reaction wheels. The output of the voltage reference device (2.5V dc) driven by 5V dc is used as the input voltage to the non-inverting sides of both IC3: C and IC3: D. The input to the inverting sides is the DAC output voltage which ranges from 0-to-V_{DD} (see Appendix B.1). V_{out8} is the output voltage at gate IC3: C at pin 8 and was calculated as follows:

$$V_{out8} = -(\frac{R41}{R38})*(DAC1 - 2.5 \text{ Volts}) \quad (4)$$

where DAC1 is the single ended voltage from DAC1 pin.

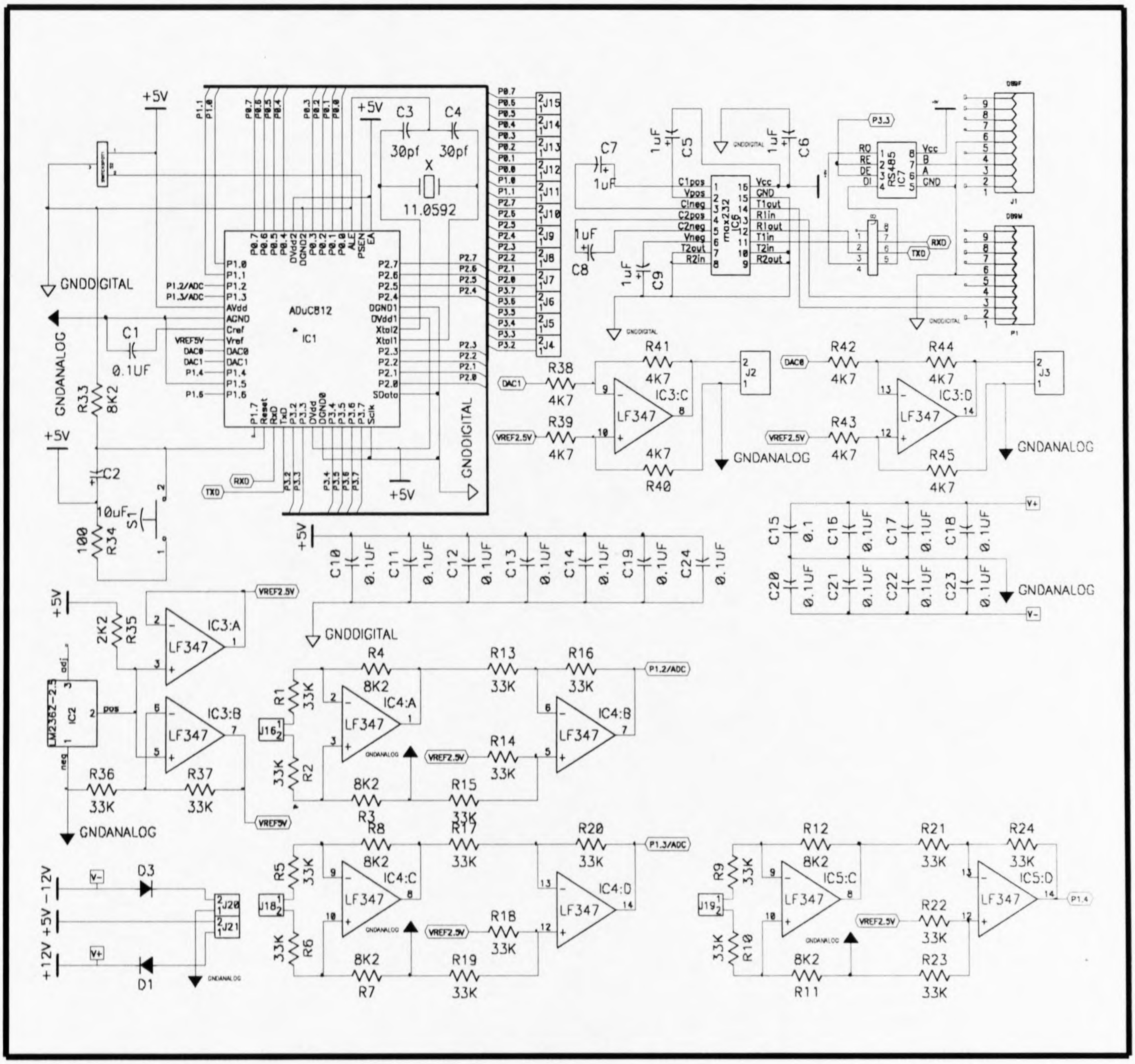
The ratio between R_{41} and R_{38} is 1:1; meaning that the closed loop gain of this Op-Amp is 1. V_{out8} according to calculations ranges from -2.5V to +2.5V dc when V_{REF} is 5V dc. IC3: C functions the same way as the gate assigned to DAC0 that is IC3: D.

IC3: C and IC3: D Op-Amps are also employed in the hardware to prevent the destruction of the micro-converter in the event of a short circuit at the output of the Op-Amp. If a short circuit occurs the output current will be limited to about 25mA (source: National Semiconductor LM347 Quad 741 Op-Amps Datasheet).

APPENDIX B

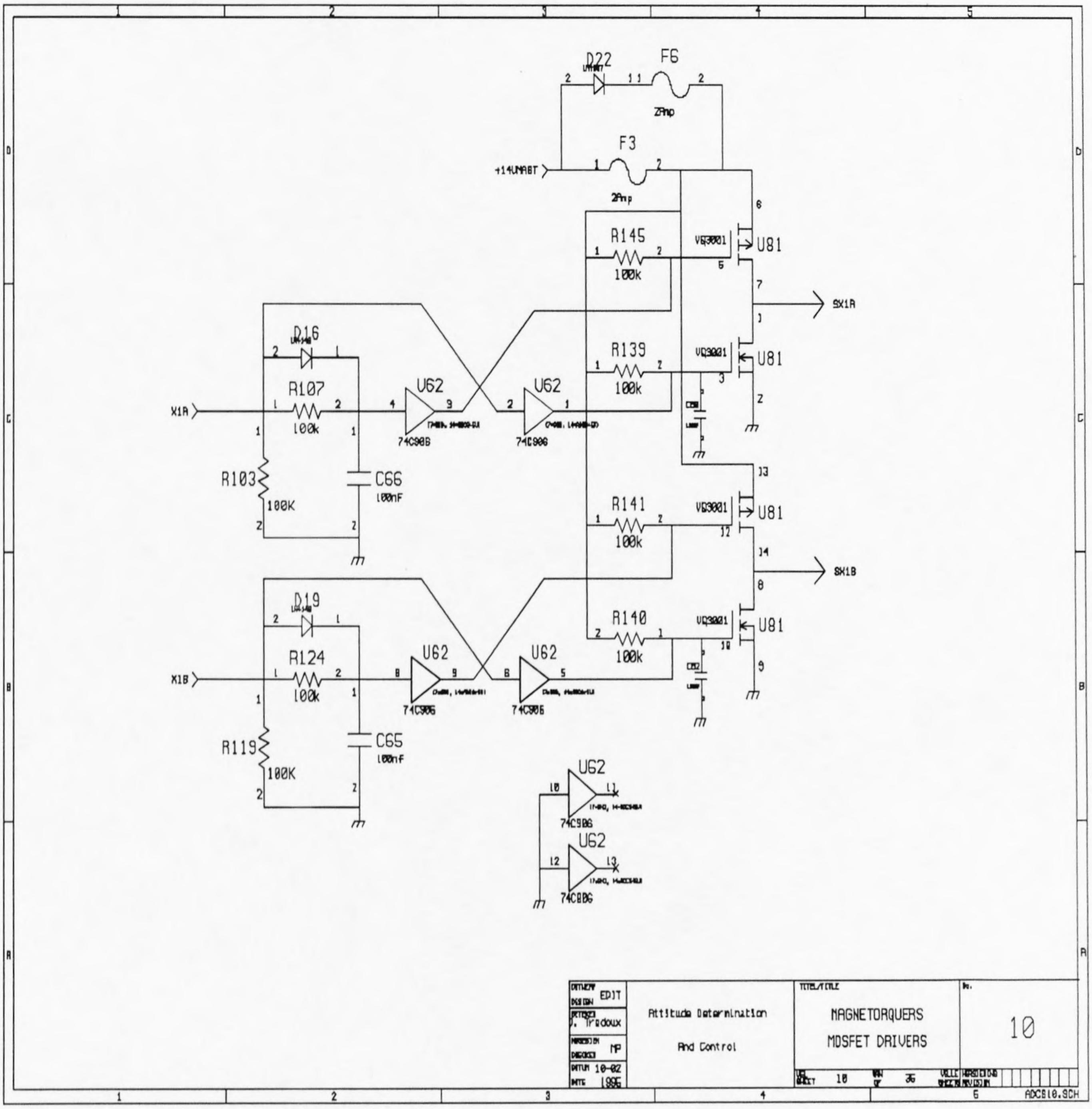
Schematics

The following schematic diagram is the representation of the interface test hardware. This interface hardware can interface both analog and digital attitude determination and control components.



B.1 Schematic of the Interface Hardware

This schematic diagram is for the MOSFET driver that drives the X1 magnetic torque coil. All six coils are driven by the MOSFETs the same as the one below. The schematic was copied from SUNSAT schematics document.

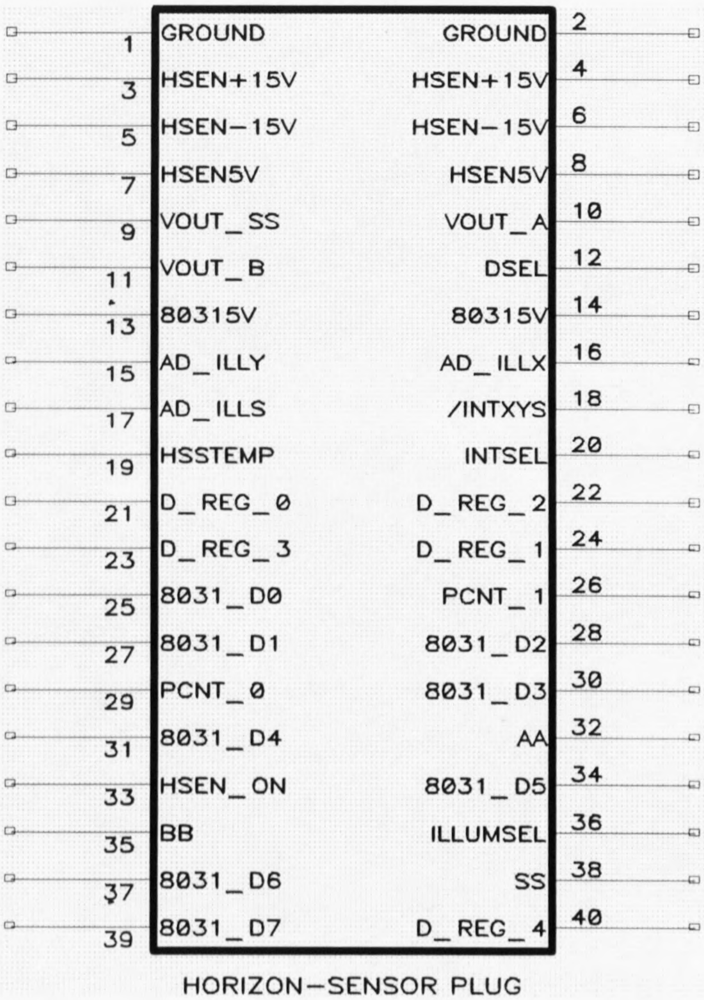


B.2 Schematic of the MOSFET drivers for the X1 Magnetorquer coil

The schematic diagram represents the test circuit for magnetorquer polarity status.

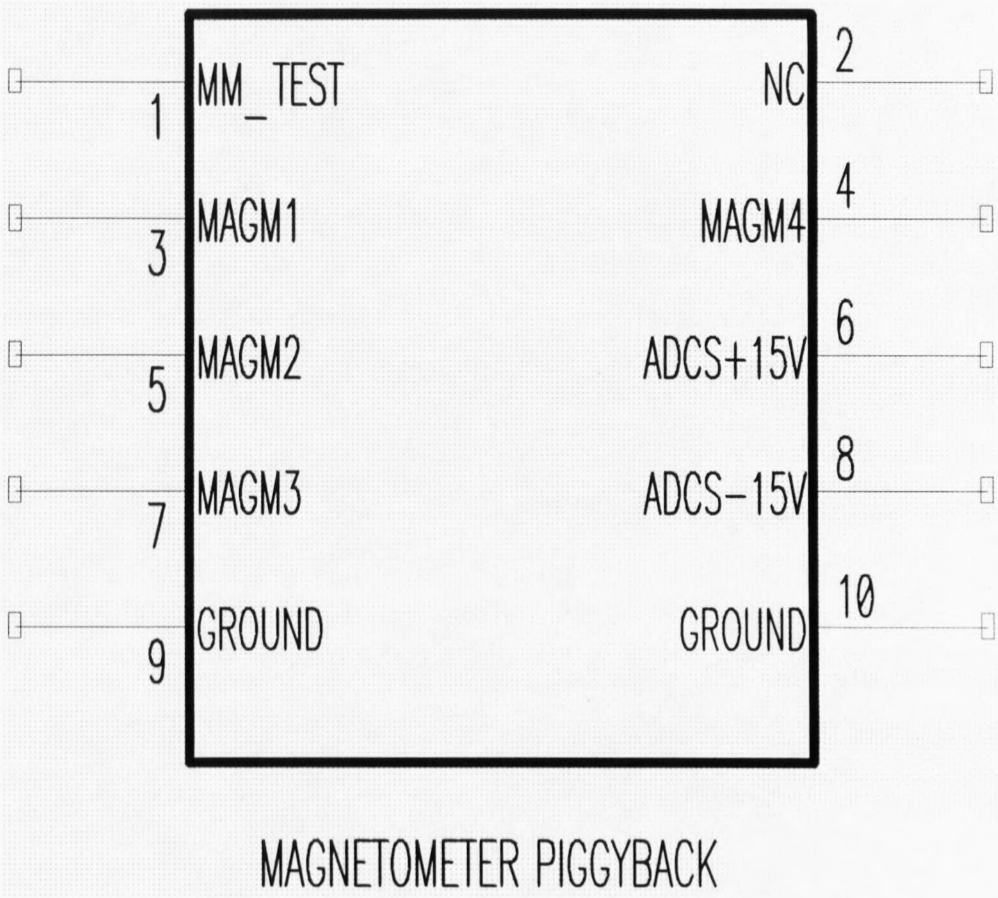
B.3 Schematic of the Magnetorquer Status Test Circuit

The schematic below, is the representation of the Horizon / Sun Sensor Piggyback Plug. The plug interfaces the Horizon / Sun Sensors to the ADCS of SUNSAT.



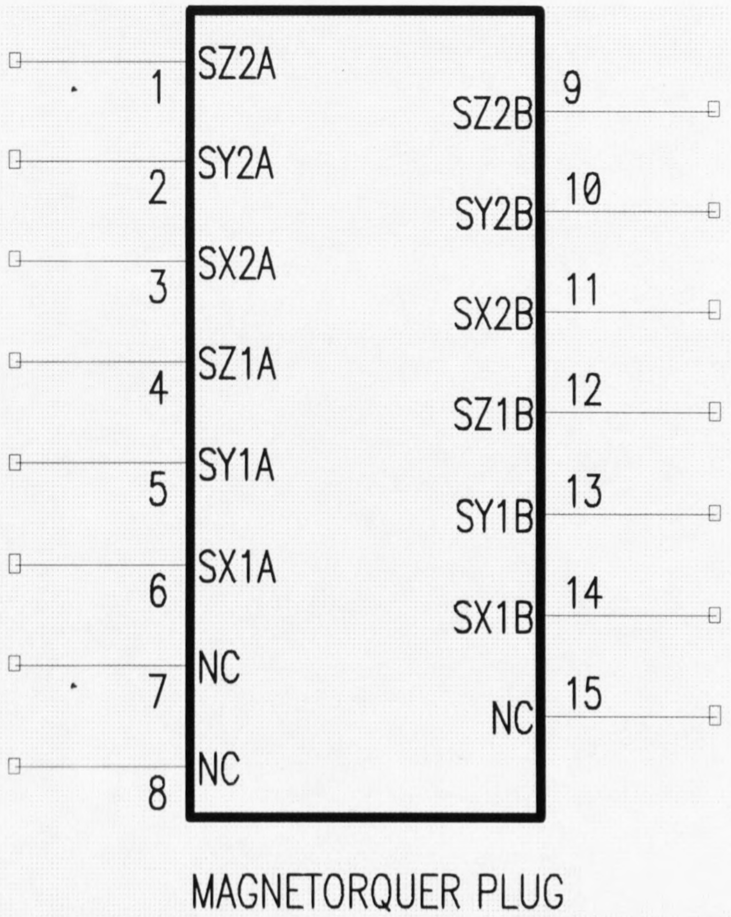
B.4 Schematic of the Horizon/ Sun Sensor Piggyback Plug

The schematic diagram below is the representation of the Magnetometer Piggyback Plug. The plug interfaces the Magnetometer to the ADCS of SUNSAT.



B.5 Schematic of the Magnetometer Piggyback Plug

The schematic diagram below, is the representation of the Magnetorquer Piggyback Plug. The plug interfaces the magnetorquer coils to the ADCS of SUNSAT.



B.6 Schematic of the Magnetorquer Piggyback Plug