

2.0 Command and Data Handling Subsystem

The Command and Data Handling Subsystem is the ‘brain’ of the whole autonomous CubeSat. The C&DH system consists of an Onboard Computer, OBC, which controls the operation of the CanSat. The OBC has software installed that manages the programs written to handle various tasks; for example, a program whose function is to create a telemetry stream will read the status of the payload sensors and then encode the telemetry stream. The same program can further control the flow of the data from sensors to the temporary memories inside the microcontroller in the event of communication restrictions, such as the blocking of communication signals between the CubeSat and the ground station.

This section begins with a discussion of the C&DH Subsystem of the previous CanSat group. Further subsections discuss the requirements and constraints for the CubeSat followed by the choice and evaluation of microcontrollers. The last subsection discusses the modifications made to the hardware and software of the previous CanSat.

2.1 Background

The previous CanSat had an OBC for C&DH. The primary component of the OBC was the microcontroller, AT90S4433. The secondary components were EPROM and RAM. These components were used for storing the software and were built-in to the microcontroller. The software was written in C language that performed only one loop to run all subsystems. A separate function was written in C language to make the OBC accept commands from the ground station and to override any operating function during

the falling phase of the CubeSat. All programming was done with the help of an STK 500 programming board, shown in Figure 1.

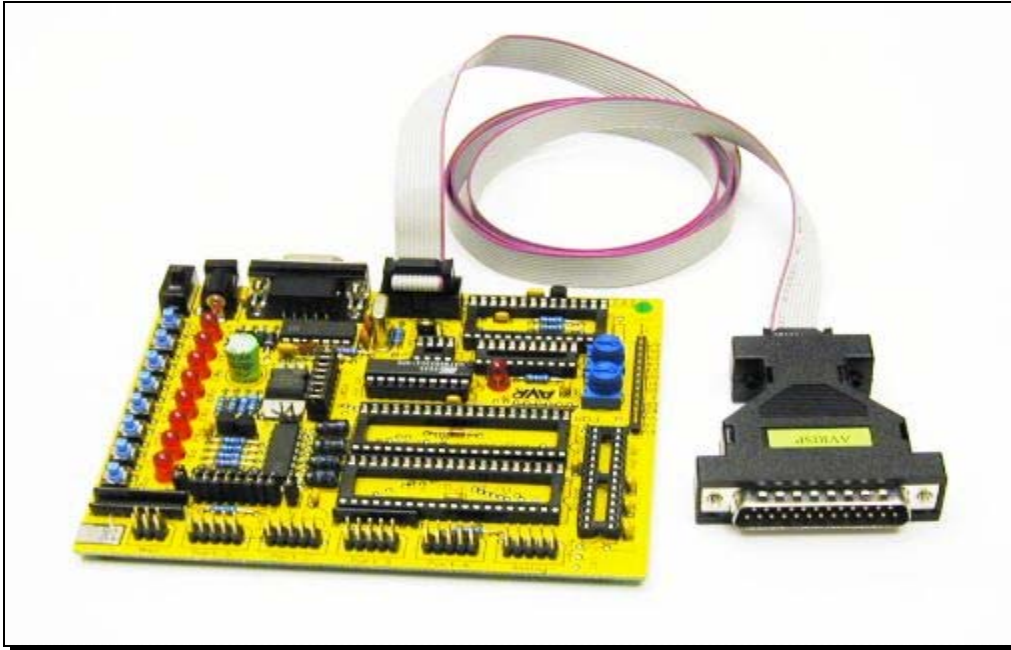


Figure 1: STK 500 AT89S/AT90S Series Flash Microcontroller Starter Kit [“Compass Lab,” 2003].

The following paragraphs discuss the advantageous as well as the flawed C&DH design considerations of the previous CanSat.

Advantageous Design Considerations:

- Control of Payload Sensors
- Control of Communication Subsystem
- Ability to transmit Telemetry Stream to the ground station during the falling phase of the CanSat

- Provision to store data from the accelerometer sensor during the ascending phase of the CanSat in RAM, so that data can be retrieved later (this provision failed and no data was stored).
- Control of quartz clocks that assisted in timing the data collection from temperature, pressure, and accelerometer sensors.

Flawed Design Considerations:

- Wiring between the OBC and other subsystems, see Figure 2.
- No repairing provisions.
- No additional Input and Output ports for additional payload sensors.

Overall, the design of the C&DH subsystem was very impressive, as it was simple in design and had about a 95 % success during the final field test.

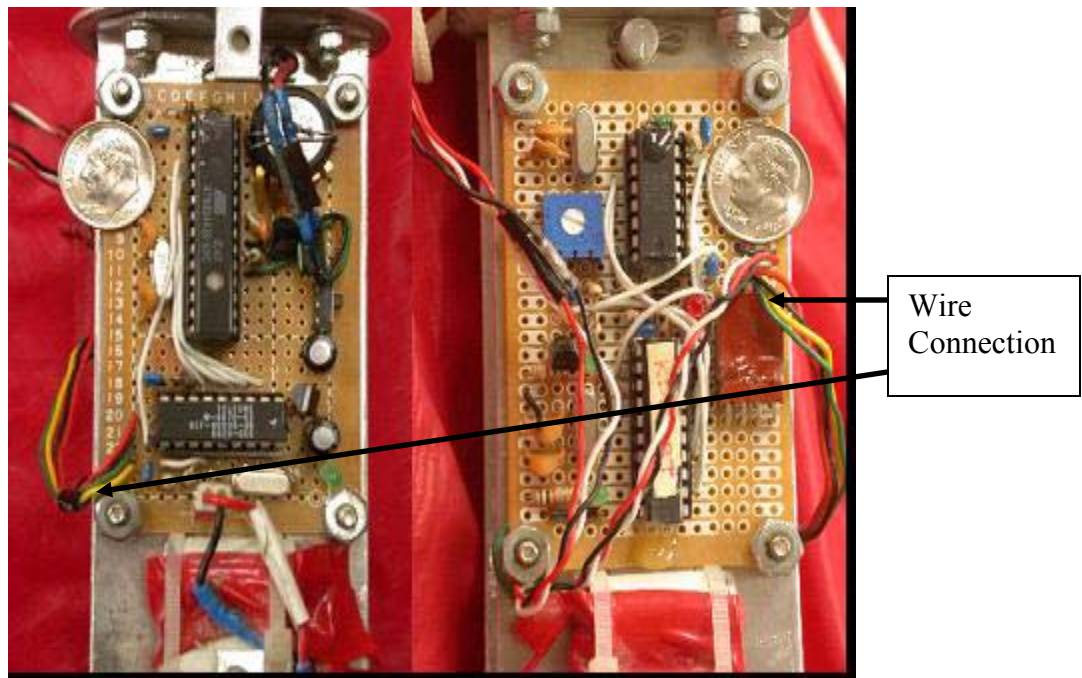


Figure 2: Wiring for connecting different Subsystems in Previous CanSat [Campbell and others, 2003].

2.2 Requirements and Constraints

The objective of the C&DH subsystem is to provide the CubeSat with operation sequences to various subsystems. Because of the size restrictions of the CubeSat, the C&DH subsystem needs to be efficient, small, lightweight, and easy to integrate with all of the other subsystems in CubeSat. This subsystem should be able to perform several tasks, as listed below:

- Subsystems control
- Communication with the ground station
- Data and software storage in allocated memories
- Fault detection and management
- Telemetry stream generation
- Data uplink and downlink feature.

In addition to the aforementioned objectives expected of the CubeSat, the Satellite Solutions group tried to meet additional design constraints, as listed below:

- Easy installation and repair provisions
- Provisions for additional subsystems for future work
- Efficient programming of the microcontroller (multiple loops instead of single loop code).

Each of the above requirements has different demands concerning the OBC and is also critical for efficient operation of the CubeSat. Hence, the proper evaluation of all options available for components that make up the OBC was necessary and are discussed in the next section.

2.3 Options and Evaluation

The OBC is made up of several components, such as microcontrollers, capacitors, resistors, voltage regulators, LEDs, memories (RAM, EEPROM, and ROM), and timers. Out of all of these components, memories, timers, and microcontrollers are the most important. Cost and ease of fabrication of these electronic components are the major factors that result in various configurations in which an OBC can be designed. The requirements restrained our team to choose components that result in a small and light-weight C&DH subsystem. We chose a microcontroller that has built-in memories and timers; this makes the microcontroller the most critical component in C&DH subsystem. To assist in selecting the right microcontroller (processor) and to meet the requirements of the OBC, a set of minimum specifications was developed and is listed below:

- Data and Nonvolatile Program Memory:
 - EEPROM (*Electrically Erasable Programmable Read Only Memory*) – min. 8 kB
 - Flash - minimum 512 bytes
 - SRAM (*Static Read Only Memory*) - minimum 512 bytes
- Desirable features
 - High processing speed – more than 4 MHz
 - In-built Analog/Digital Converters
 - Programmable UART (*Universal Asynchronous Receiver-Transmitter*)
 - Master/Slave SPI Serial Interface
 - Controllable I/O pins
 - Programmable timers, especially the watch-dog timer
 - Minimum 16 bit architecture
 - Avoid Ball-grid-array (BGA) microcontrollers (difficult to solder)
- Low Power consumption – less than 10 mA and voltage less than 5 volts
- Size – should fit on a 5 cm x 5 cm Printed Circuit Board

- C compiler - must be available for the processor (microcontroller)
- Operable in temperature range 0 – 40° C; however, wider range is preferable.

Next, several microcontrollers were investigated that best met the specifications set above. Table 1 below lists the microcontrollers (that meet the minimum specifications) that were examined along with the reason that they were rejected or selected.

Table 1: Microcontrollers list and the reason for their rejection.

Company	Reason for Rejection or Selection
Motorola	Hard to solder because of BMG configuration
Hitachi	Programming skills very limited
Intel	High power consumption
Microchip	Nothing wrong
PIC	Nothing wrong
ATMEL	Nothing wrong

Table 1 indicates that the Microchip, PIC, and ATMEL microcontrollers meet our requirements. All three have similar features, but ATMEL was picked because the previous CanSat group used an ATMEL microcontroller with satisfactory performance. In addition, all the programming accessories were available to Satellite Solutions. Finally, an ATMEL microcontroller, model number Atmega163 (Figure 3) was selected at an expense of only \$6.42.

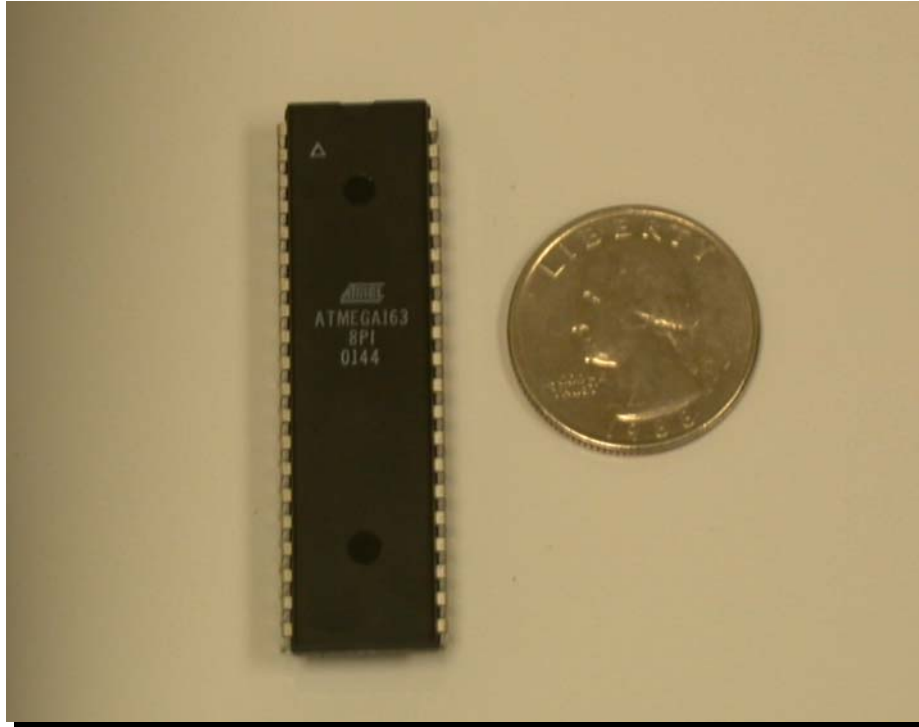


Figure 3: ATmega163 ATMEL microcontroller.

The main features of Atmega163 microcontroller are listed below:

Data and Nonvolatile Program memory

- 16 kB of in system Programmable Flash
- 1024 Bytes of SRAM
- 512 Bytes of Programmable EEPROM

Peripheral Features

- One 8 bit timer/counter
- One 16 bit timer/counter
- 8 channels and 10 bit Analog/Digital Converter
- Programmable watch-dog timer
- Programmable Serial UART
- Master/Slave SPI serial interface
- 32 programmable I/O lines

Power Consumption

- Maximum Current Consumption 5.0 mA
- Maximum Operating Voltage 5.5 V

Processor Speed: 8 MHz

Physical Size: 52.71 mm x 13.97 mm x 4.83 mm

2.4 On-Board Computer Design: Hardware and Software

Each of the requirements mentioned earlier has different demands concerning the OBC. The fact that the computer has to operate in an atmosphere with high gravitational forces acting on the system as a whole also implies that the system has to be designed in a special way. This section describes how these requirements affect the design of the OBC as well as the theoretical outline for the computer. The system described in this section is depicted in Figure 4.

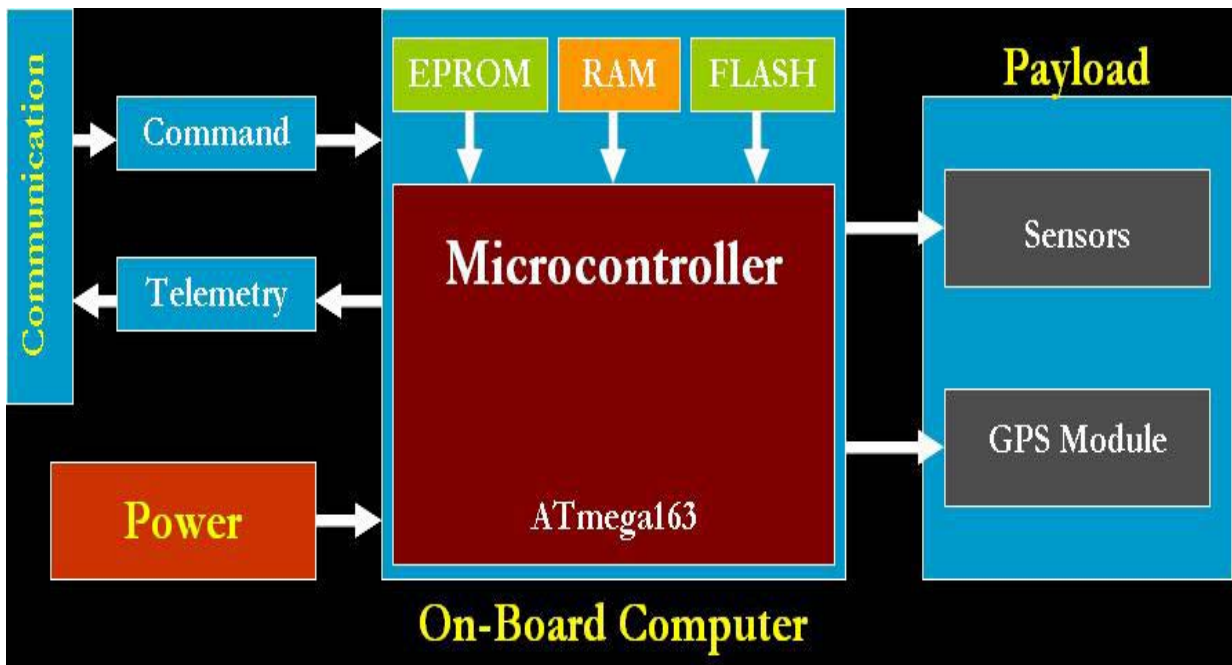


Figure 4: Block Diagram for C&DH Subsystem (middle) and other Subsystems.

2.4.1 Hardware

Figure 4 indicates a microcontroller with three built-in memories: EEPROM, Flash, and SRAM. These three types were chosen because this computer will operate in atmosphere where radiation sometimes leads to bit-flips in temporary memory (Flash), which results in unpredictable software errors. A programmable safety timer, called a watchdog timer, resets the processor if bit-flip occurs, and the processor reloads the boot-software from EEPROM to Flash. It is essential that the boot-software always works correctly, and therefore must be stored in a memory, where bit-flips do not occur [“The DTU sat,” 2003].

The Flash memory is used to store the main operating software and other subroutines, and a copy of this software will also be stored in EEPROM (permanent memory). The STK 500 starter kit helps in programming the memories in the microcontroller. Storage of software in this fashion is necessary in order to modify certain programming subroutines via the communication subsystem when the OBC is in the atmosphere [“The DTU sat,” 2003].

The SRAM temporary memory will be used to run the software and subroutines and also as a place to store the measured values from the payload sensors. Both Flash and SRAM suffer from bit-flips; therefore, an Error Detection and Correction Circuit (EDAC) consisting mainly of a watchdog timer and a delay interface between these temporary memories and the permanent memory is finalized for implementation in the hardware design [“The DTU sat,” 2003]. The CubeSat will be used as a testing platform for EDAC that is required by the FASTRAC nanosatellite project.

Next, in order to communicate with the ground station, the OBC must be connected to a radio or communication subsystem (see communication subsystem section for more details), as shown in Figure 4. The easiest way to implement communication is to use an UART, which converts signals between serial and parallel data and also handles the serial timing. The UART is a built-in peripheral inside the Atmega163 processor [“The DTUsat,” 2002].

The OBC will have a number of two different interfaces to other parts of the CubeSat: analog and digital. The payload sensors produce the analog signals, and all the other subsystems discussed in later sections produce the digital signals, as shown in Figure 4. Since the communication subsystem accepts only digital signals, the microcontroller will convert the analog signals to digital signals with the help of an Analog to Digital converter (ADC) [“The DTUsat,” 2002].

Next, two Real Time Clocks (RTCs), one 8-bit and the other 16-bit, as part of the microcontroller, will assist in scheduling tasks or operations to CubeSat’s subsystems [“The DTUsat,” 2003].

The delay interface is also of special interest. As mentioned earlier, this device is used along with watchdog timer in the EDAC. The purpose of this device is to find errors in the hardware and software design and to correct them by uploading new software to the Flash memory. The delay interface consists of measuring points (provided by the timer) for important signals and an interface to the processor and the flash memory.

As mentioned above, operating software is required to run the hardware. The following section discusses the features of such an operating software.

2.4.2 Software Design

The ATMEL Company requires C language for programming the microcontroller; therefore, Satellite Solutions assembled the required equipment for programming the microcontroller. The equipment basically consists of a desktop computer, the STK 500 starter kit, and an AVR studio software. The C language program is written, compiled, and then downloaded into the microcontroller with the help of AVR Studio software. The compiler converts C language into machine language (binary) that a microcontroller understands. In the end the STK-500 starter kit will be used to download the machine language from AVR Studio into the microcontroller. Mistakes are common while programming the microcontroller, but the problem can be fixed quickly because the ATMEL microcontroller's memory can be completely erased and rewritten 100,000 times.

The Satellite Solutions team failed to write complete C language code to meet the software requirements for CubeSat, as planned in the midterm report. Upon attempting to program the microcontroller, the team realized that this endeavor required much more knowledge of microcontrollers than was previously thought. However, Satellite Solutions performed the necessary research and laid out the software requirements, which will be robust, tested, and free of infinite loops.

The main operating software will be divided into several subroutines for the efficient operation of all of the CubeSat's subsystems. The reason for writing several subroutines is to avoid a major disadvantage which is inherent in an OBC. In an OBC different subsystems rely heavily on one another; as a result, a fatal error in a program without subroutines can result in a failure in execution of the parts of the program written

for other subsystems. This failure in software can render the entire CubeSat inoperable. In addition the written code should be of minimum length to ensure that the program fits within 512 Bytes of EEPROM memory. The requirements made the programming very difficult for the Satellite Solutions team, but with the help of Shaun Stewart (advisor) and Mr. Pascal (graduate student at Santa Clara University), we were able to obtain some previously written subroutines, which need to be modified for the ATmega163 microcontroller. The codes are attached in Appendix B of this report.