Design Considerations for A Gas Microcontroller

by

Douglas A. Ritter
Purdue University Getaway Special Group
West Lafayette, Indiana 47907

Abstract

This paper discusses some of the design problems that we are now addressing in considering a microcontroller for our upcoming GAS payload. We will be using microcontrollers to run our experiments and collect and store the data from those experiments. Some of the requirements for a microcontroller are to be small, lightweight, have low power consumption, and high reliability. This paper will discuss some of the solutions that we have developed to meet these design requirements. At present we are still in the design stage and our final design may change from what we have now. We are continuing to look for new integrated circuits that will do what we need all in one package.

1. Introduction

Microcontrollers will be used in our GAS payload to automatically run our experiments. There are only 3 controls, operated by the astronauts, that can be used to control the payload. When the command is received to start an experiment the microcontroller must prepare the experiment and then run it. The microcontroller must also collect the data from the experiment and store it for use back on earth. The microcontroller will be in control of almost everything inside the payload.

In designing our microcontroller we considered a number of design criteria. Some of these were reliability, power consumption, weight and compactness. High reliability was very important to us and played an important part in the decision for the basic design. There are two ways of assuring high reliability. One is to have three or more parallel processors, all checking one another. This was determined as too difficult to implement on our scale. The next best alternative was to have a separate microcontroller for each experiment. If one controller failed it would not jeopardize all of the other experiments. Our design goal was to develop a microcontroller that with a few modifications or additions could be used with many different experiments.

2. Power Consumption

Designing a microcontroller with a low power was very important, as batteries are heavy and take up space. By limiting power consumption we can reduce the size of batteries that are required and utilize the weight and space saved for the experiments.

In considering available integrated circuit technologies, it was determined that CMOS was the best choice. Most of the popular microprocessors and microcontrollers are available in a CMOS version. Several companies are also making the 74HC series of digital integrated circuits which will also be needed in the microcontroller. This new generation of integrated circuits combine the low power requirements of CMOS with the high speed of the older TTL circuits.

There is, however, one major drawback to CMOS circuits. They are very static sensitive. If the devices are mishandled or if the circuit takes a static shock, the electronics can easily be damaged. We decided that with proper circuit board and enclosure design, static electricity should not be a problem for us.

3. Size

Because of the small size of the GAS canister, space is at a premium. Reducing the size of the electronics would leave valuable space for the experiments. Besides making the layout of the circuit boards as efficient as possible, reducing the number of integrated circuits used is the best way to reduce the size of the electronics. Reducing the device count was the major factor in selecting the circuits we plan to use.

A side benefit of reducing device count was that it would also reduce power consumption. A device that replaces several integrated circuits will have a lower power consumption than the total of all the individual devices, so wherever possible, we tried to find circuits that would do as much as possible in a single package.

4. Device Selection

Taking these design criteria into consideration, we had to make our selection of the devices that we were to use for our microcontroller. Most of the popular microprocessors are available in CMOS versions, so power consumption is not much of a problem. However, microprocessors require support chips for timers, input and output, and memory, which leads to increased device count. We then looked at microcontrollers. These devices have RAM, I/O, and timers all in one package. The particular microcontroller that we looked at in detail was the INTEL 80C31. This device is of the CMOS variety, has 4 8 bit I/O ports, and 128 bits of random access memory for temporary data storage. This seemed to fit the requirements for what we needed.

However, we identified several problems with this device. It has on-board mask programmable read only memory which we could not use. The chip can be programmed to use external RAM, but we lose the use of 3 I/O ports for address, data and control lines. This left us with 1 I/O port and internal timers, so we still have a reduced device count.

For our program memory, we have chosen a CMOS PROM device from Harris. It is an HM-6616 CMOS PROM package with 2048 by 8 bit memory arrangement.

On our previous GAS package that we sent up (GAS 009), EEPROMS were used for data storage. When the data was read out after the return of the canister, some of it seemed to have been altered and it was suspected that cosmic radiation could change a few bits here and there. This is very undesirable in the program memory because if a bit is changed, it could result in the microcontroller locking up. We decided, therefore, to use a PROM device. In a PROM, selected conductors are actually destroyed like a fuse to create the ones and zeros and it would not be possible for cosmic radiation to change them.

To store data collected during the flight, we decided to use an EEPROM device. We wanted to keep the power supply at 5 volts and some of the new EEPROM chips come with single 5 volt supply requirements. The device we are testing is a NCR 52832, a 4096 by 8 bit low power EEPROM memory package. One thing to note about EEPROMS is it takes as long as 10 milliseconds to permanently write data. This is much longer than the clock cycle of the microcontroller device, so the chip must be buffered from the microcontroller. Further, the microcontroller must go into a delay loop while the device burns in the data. The particular device we chose has a 16 word register, so 16 words can be written at one time. During the write cycle the EEPROM requires a lot of power, so writing 16 words at a time will save power.

We are now looking at several analog to digital converter packages. We are looking at a few CMOS A/D devices that have an eight channel multiplexer on board. This will allow 1 circuit to read eight different analog inputs. This again combines several chips into one and reduces the device count. This chip will be used to collect temperature data or voltage readings throughout the canister.

The rest of the support devices that we will use are from the 74HC series of integrated circuits. These are CMOS versions of the 7400 and 74LS series of TTL chips.

5. The Microcontroller Design

We are presently working on 4 experiments that require microcontrollers. There will be two other experiments in our payload, but they are passive. For reliability each experiment will have its own microcontroller. The basic design of our microcontroller that will be common for all the experiments will include a microcontroller circuit, a PROM for program memory, and an EEPROM for data storage. This will serve as the foundation of our microcontrollers. Other circuits will be added as required by the different experiments.

The first experiment we are working on will measure the noise, shock, and vibrations that our payload experiences during launch. We are planning on using a new kind of structural support for our canister using graphite fibers. We intend to measure the stresses involved during lift off to determine the effectiveness of our new design. We will need to add a high speed analog to digital converter to the basic microcontroller to measure the outputs from the accelerometers.

For the second experiment, the microcontroller will be taking and storing temperature data taken at various points in the canister. The purpose is to measure the effectiveness of the thermal blocks whose job it is to maintain the temperature in the canister. The microcontroller will require an A/D converter, but because the sampling rate will be slow it does not need to be high speed. We may need to use a high resolution convecter depending on the accuracy of the measurements that we need.

The third experiment is the materials processing experiment. In this experiment tin and zinc carbonate are melted to form a "foam metal". The microcontroller for this experiment will be interfaced with a switching circuit to control the heating of the metal. It will also have an A/D converter circuit for temperature measurement inside the oven.

The fourth experiment will be this data telemetry experiment. The microcontroller for this experiment will collect data and status information from the other microcontrollers through a serial communication network. This microcontroller will be interfaced to a speech synthiser device to produce English speech. The output from this circuit will go to a radio transmitter to transmit the data back to Earth. This is so we can get data and the status of the canister on a real time basis. This will be the only microcontroller that does not require the EEPROM in the basic design because no data will be stored by this microcontroller.

Not all of the experiments will run immediately when turned on. The data telemetry and the materials processing experiments will be run only after certain prerequisites are met. The astronauts will determine when these prerequisites are met and will signal for the experiments to commence through the use of the GCD relays. The microcontrollers will have to be able to read these switches. The I/O port on the microcontroller circuit will be used for this.

6. Conclusion

We recognize that the design described here is probably not the best design possible. Design requirements will probably change from payload to payload. It is hoped, however, that what we have described are the problems to be solved and how we have chosen to meet them. We would like to hear about any ideas on how to meet these requirements in a better fashion. We are always open to ways of doing things better.

One of the new ideas that we are considering is that a new microcontroller device that has an auto reset feature. It is a timer that counts down. If this timer is not reset by the program (because, for example, the microcontroller has locked up), the timer will force the device to reset itself. This may be a desirable feature in the event that the cosmic radiation changed a bit inside the microcontroller circuit, or if there was just noise on the line causing a lock up. The search goes on.

TABLE OF DEVICES USED

Device	Manufacturer Number	Manufacturer
Microcontroller	80C31	INTEL
CMOS PROM	HM-6616	HARRIS
EEPROM	52832	NCR CORP
A/D Converter	AD7581	ANALOG DEVICES

MICROCONTROLLER BLOCK DIAGRAM