

(14) / CW / 11 / 33

2001 NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM

JOHN F. KENNEDY SPACE CENTER
UNIVERSITY OF CENTRAL FLORIDA

Intelligent Systems for Self-Healing Electronics

Carl D. Latino Ph.D.
Associate Professor
Electrical and Computer Engineering Department
Oklahoma State University

ABSTRACT

For long duration missions it is imperative to be able to monitor and record critical information. The data acquisition systems used must therefore be fault tolerant. This usually meant having redundant copies of critical channels. Since each channel usually consists of various components, the parts count, cost, weight and complexity of the system could be very high. The Advanced Data Acquisition System (ADAS) has been developed as a proof of concept. The purpose was to demonstrate an architecture where individual spare parts can replace defective ones to repair a channel. By so doing entire channels do not need replication. This reduces the need of total redundancy and reduces the parts count. This has the added feature that in addition to spare parts, good components of a failed channel can be used as spares in another channel. In addition to reducing parts count and cost, this configuration, with an intelligent decision maker, can improve the reliability of the overall system. Another unique feature of ADAS is that it uses reconfigurable analog filters. These components can be programmed, by the smart system to meet the specific needs of the part they are to replace. This way one part can serve as spare for many different components. The hardware was built and now serves as a platform for developing intelligent algorithms. Another related project was a wireless data acquisition system. I was invited to participate in the meetings and issue suggestions. A brief description of this system will also be included.

Intelligent Systems for Self-Healing Electronics

Carl D. Latino Ph.D.

1. INTRODUCTION

The objectives of the summer 2001 project entailed the development of a self-healing system. Such a system is intended for cases where repair or replacement of failed parts is not possible, cost effective or practical. In space flight, for example, data collection is critical and must be taken regularly. If a data acquisition system fails, critical data could be lost endangering the mission or even risking lives. Generally this problem is dealt with by inclusion of redundant systems. This approach results in increased weight, cost and system complexity. Furthermore, if a single component fails, an entire sub-system may become unusable. The objective of this summer's project was to help in the development of an intelligent system capable of reconfiguring itself to switch out defective components and switch in good replacements. The system, as designed, includes flexible analog components manufactured by Lattice Semiconductor Corporation. These unique parts contain discrete analog components that can be configured into a broad range of filters and amplifiers. With this flexibility, one component can serve as the spare for a broad range of parts. To make the system intelligent, it is necessary to have all the necessary hardware on board and controlled by an embedded computer with appropriate software. The system is part of the ADAS (Advanced Data Acquisition Systems) objective. The prototype system consists of a central processor capable of performing the decisions and other processing duties, components and a means for interconnecting the components. The data acquisition system that was built consisted of the following components:

Switching matrix [1] – This is a programmable x-y matrix which can interconnect analog signals for system reconfiguration. The actual parts used were Mitel MT8806 and permitted the programmable connection of 4 rows (columns) and 8 columns (rows). The matrix consists of thirty-two switches corresponding to the cross points. These switches are closed by writing a logic “1” at the appropriate address and opened by writing a “0”.

Programmable filters [2]– These devices, manufactured by Lattice Semiconductor Corporation are programmable by writing digital data to the device. The basic building block is an instrumentation amplifier with programmable gains and feedback. There are several such blocks per integrated circuit. Programming is performed by serially loading a binary file into the device. For more detail refer to ispPAC documentation. The device can be reprogrammed while in the circuit.

Digital to Analog Converter [3]– D to A converters are used to change digital data to analog signals.

Sample and Hold – These components are used in conjunction with Analog to Digital Converters to stabilize the input while the Analog to Digital converter digitizes the sample.

2. THE ADAS SYSTEM

The prototype system was designed by NASA and printed circuit boards built. The system architecture is described in the NASA document[4]. The printed circuit boards were manufactured and populated but needed to be tested. Since the assembled system was available for only two weeks, the prime objectives were to ensure the board was defect free and that all the components interacted correctly. The system was designed to allow analog signals to be read in, routed to appropriate components and output to the desired locations. Among the many duties, the processor calibrates the data paths, analyzes the received data and decides on how and when to reconfigure the system. The intelligence necessary to perform these functions have not yet been developed. The initial tasks required making a few corrections to the hardware and generating the software needed to ensure that all components interfaced correctly. At this writing, the components were individually tested and communications between all parts were verified. A voltage reference component did not operate as expected. It later turned out that the part was unsuitable and a different part will be needed. Other than this, the system operated correctly. Accomplishing these goals was a large step in the direction of demonstrating the capabilities of a working system. Future goals include the incorporation of software to analyze the incoming data for alarm conditions or to determine if component aging or failure is a possibility. The analysis of incoming data might be done using intelligent algorithms such as Neural Nets and other tools. This would evaluate the health of the device being measured as well as monitoring its own health. The system was designed capable of injecting known signals at desired inputs, routing these signals in a variety of paths for the purpose of locating possible system defects. The type of problems that this system will be asked to deal with include broken paths, degrading components and evaluation of the components being sensed. These are issues that are just now being addressed. Other topics to be addressed include search algorithms for faulty path detection and fault location. Knowing this, selection of alternate paths must be determined. This problem is further complicated when multiple paths must all be working. Other topics of importance include the development of models, which based on component reliability cost and other factors, suggest necessary redundancy.

3. SYSTEM RELIABILITY

Cost analysis of Advanced Data Acquisition System.

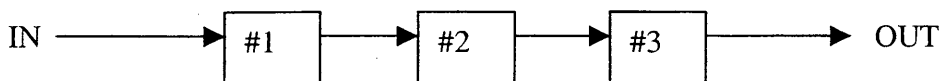


Fig. 1 Single Channel Single Path

To ensure that a sensor system is functioning, all components in the data path must be working. I.e. If a sensor system consists of three components operating in series, all must

be working for the system to work. If probability of failure for the components is p_1 , p_2 and p_3 , (where $0 < p_i < 1$ for $i = 1, 2$ or 3) the probability that the system works is:

$$P(\text{system works}) = (1-p_1)*(1-p_2)*(1-p_3)$$

That is, the system works, if each of the three components works.

To ensure that the data is collected, an identical redundant system might be employed as shown in Fig. 2.

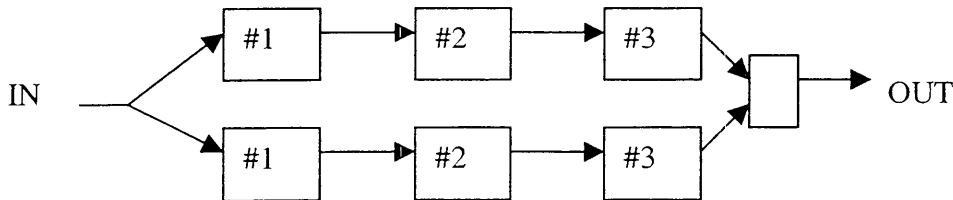


Fig. 2 Single Channel Double Path

If the redundant system has the same reliability as the original, the probability of the overall redundant system working is:

$$P(\text{redundant works}) = 1 - (1 - P(\text{system1 works})) * (1 - P(\text{system2 works}))$$

This basically states that the system will work if the redundant systems have not both failed. The increased reliability comes at a cost in that twice as many components are needed and additionally a means of deciding which of the systems to operate. Neglecting, for the moment, the means of deciding and assuming the switching component won't fail, let's see what benefit is realized by this redundancy.

Example: If failure probabilities are: $p_1=0.1$, $p_2=0.15$ and $p_3=0.3$

$$P(\text{system works}) = 0.9 * 0.95 * 0.7 = 0.598$$

$$P(\text{redundant works}) = 1 - (1 - 0.598) * (1 - 0.598) = 0.8387978$$

This increases reliability by about 40% while increasing cost by about 100%.

Another possibility is that of making the individual components interchangeable. That is, that each of the three components can be switched in and out as shown below:

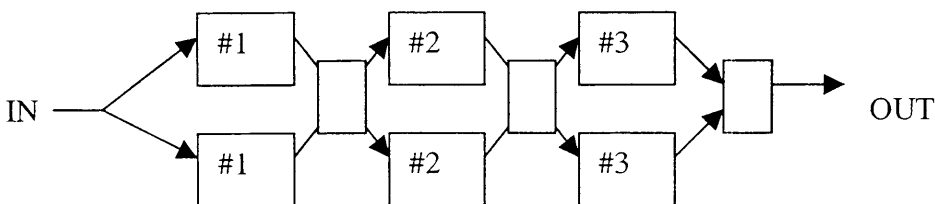


Fig. 2 Single Channel Changeable Parts

The cost is further increased, but so is the reliability. This system works if at least one of each of the redundant components is working.

$$P(\text{switched works}) = ((1-p_1^{n_1}) * (1-p_2^{n_2}) * (1-p_3^{n_3})).$$

Where n_i ($i = 1, 2$ or 3) is the number of redundant sub components. If $n_i = 2$ for all i , then:

$$P(\text{switched works}) = ((1-0.1^2) * (1-0.15^2) * (1-0.3^2)) = (1-0.01) * (1-0.0225) * (1-0.09) = 0.8806297$$

This further increases reliability (by about 47% over original) but requires greater complexity. This is in the form of switching components, which could also fail, as well as require more intelligence to decide on the switching arrangement. An advantage of such a system is that higher redundancy can be applied to most critical components only incrementally increasing the cost of the overall system. Using the above as an example, since device 3 is most prone to failure, what happens if three copies of it are used (instead of just two).

$$P(\text{switched new works}) = ((1-0.1^2) * (1-0.15^2) * (1-0.3^3)) = (1-0.01) * (1-0.0225) * (1-0.027) = 0.9415964$$

This represents a sizeable increase in reliability (increase over original of about 57%, and an increase over redundant by about 7%) with a marginal increase in cost. To make an intelligent decision, it is necessary to decide factors such as component cost, system weight, importance of the data etc.

Another way of looking at the problem is that probability of the system working must be at least some value x . If $x = 0.9$ and there is no redundancy, then $(1-p_1) * (1-p_2) * (1-p_3)$ must exceed 0.9. If all p_i are equal, $(1-p_i)^3 < 0.9$, therefore $(1-p_i) = 0.9^{(1/3)} = 0.965489$ and $p_i = 0.034511$. This is the ideal case and in reality, there is a "weak link" which makes the numbers look worse. That is, if one component is more prone to failure, the overall system performance is affected mostly by this component. In general, if a signal must traverse three components, the most favorable situation is where the probability of successfully traversing each of the three component is the same. Roughly speaking, if a component is more apt to fail, then it must exist in greater number. As a simple example, if $p_1 = 0.1$, $p_2 = 0.2$ and $p_3 = 0.3$, and there are n copies of unit 1, how many copies of units 2 and 3 are needed for an optimal system?

Example:

Let $n = 2$, with the above error probabilities. What is the probability of a working system?

Answer:

If there are two copies of unit #1, the probability that it works is:

$$(1-0.1^2) = 0.99$$

To get the same probability for unit #2 want $(1-0.2^k) \geq 0.99$.

$$\text{If } k=3 \text{ then } (1-0.2^3) = 0.992$$

Similarly for unit #3, will need about three of those.

$$(1-0.3^m) \geq 0.99 \quad \text{if } m=4, (1-0.3^4) = 1-0.0081 = 0.9919$$

A system thus configured would have a probability of success of:

$$0.99*0.992*0.9919 = 0.9741251$$

If each of the different units cost the same, is there a better way of configuring the system as to increase reliability or reducing cost?

Removing one of the two units #1 will drop the reliability to:

$$P(1,3,4) = ((1-0.1^1)^* (1-0.2^3)^* (1-0.3^4)) = 0.9*0.992*0.9919 = 0.8855683$$

Removing one of the three units #2 will drop the reliability to:

$$P(2,2,4) = ((1-0.1^2)^* (1-0.2^2)^* (1-0.3^4)) = 0.99*0.96*0.9919 = 0.9427017$$

Removing one of the four units #3 will drop the reliability to:

$$P(1,3,4) = ((1-0.1^2)^* (1-0.2^3)^* (1-0.3^3)) = 0.99*0.992*0.973 = 0.9555638$$

As expected, the more reliable system has the greatest effect on reliability. That is, removing a reliable component will have a greater effect on reliability than removing a less reliable component. Now for the second part of the test, seeing if the removed component is replaced by one of the other units, i.e. remove one of 4 units #3, and replace it with one unit #2.

$$P(2,4,3) = ((1-0.1^2)^* (1-0.2^4)^* (1-0.3^3)) = 0.99*0.9984*0.973 = 0.9617287$$

This is a lower probability of success than the P(2,3,4) configuration. Other combinations using a total of 9 components will be lower than 0.9617287. This illustrates (but does not prove) that for a given cost, the best configuration balances the reliability of each leg.

Another comparison worth considering is how many times must a path be made redundant to have the same reliability shown above? The probability of a non redundant path working is: $P(1,1,1) = (1-0.1)*(1-0.2)*(1-0.3) = 0.504$

Two such parallel paths have reliability $P(2 \text{ paths}) = 1-(1-P(1,1,1))^2 = 1-(1-0.504)^2 = 0.753984$. This requires six components.

Three such parallel paths have reliability $P(3 \text{ paths}) = 1-(1-P(1,1,1))^3 = 1-0.496^3 = 0.8779761$. This requires nine components.

Four parallel paths have reliability $P(4 \text{ paths}) = 1- 0.496^4 = 0.9394762$. In spite of the fact that twelve components are used, the reliability is only 0.9394762. This is compared to the probability of 0.9741251 utilizing only nine units for the ADAS system.

In order to operate the system it is necessary to adopt an efficient way of selecting the paths. For example, if there are 2, 3 and 4 elements in each level, there are $2*3*4 = 24$ possible paths. The following simple algorithm illustrates an exhaustive search. This will find all possible good paths. Are there better methods? Probably yes.

```

For i = 1 to 2
  For j = 1 to 3
    For k = 1 to 4
      Is Path ijk valid?
      Yes? Then
      Add ijk as valid path.
      No? Continue
    Next k
  Next j
Next I
No valid path
END

```

If multiple channel and thus separate paths are needed, the algorithm becomes more complex. This is so, since each path must use unique components. The advantage of multiple paths is that it takes only a few redundant components to handle a relatively large number of paths. If the number of spares is insufficient to deal with the faults, shutting down the least critical data paths and using some of those parts for spares may allow the overall system to function albeit at a somewhat reduced rate. It may also be possible to collect data at reduced rate by different data through the same channel, but multiplexed in time.

4. CONCLUSIONS

The circuit which was constructed represents a platform on which reconfigurable, intelligent systems may be evaluated and tested. Our objective was assembling such a system for the purpose of demonstrating its potential. The complexity and cost of such a system is relatively high but it has many advantages. In order for it to be feasible, however, the system must demonstrate tangible benefits. When a series of components must all work for a system to work, it is reasonable to replicate the least reliable ones in order to increase reliability while optimizing usage of resources. Some of the objectives were to demonstrate that parts count could be reduced, reducing power requirements and increasing overall reliability. To realize this, some statistical data should be taken and component redundancy should relate to these numbers in order to attain desirable results. This way, the more critical or least reliable components would be replicated in the proper amount to attain overall system reliability. Cost may also be a factor to consider. To help in the determination of level of redundancy and degree of reliability, a model should be developed with this problem in mind. Armed with this model, the user could make estimates of resource needs based on various factors such as component reliability, parts cost, number of needed channels etc. As the number of channels increases, the benefits of this type of system become more pronounced. With ADAS, a few redundant components can support a large number of channels. In addition to the normal spares, good parts of a failed channel can also be used. In extreme cases, the least critical

channels can be disabled or use one channel to collect data from several different components. The possibilities are many. Another issue of great importance is that although the ADAS developed has redundant systems, it has been designed with a single processor. This single failure point can invalidate many of the advantages of the system. Replicating the central processor, however, is a more difficult problem than simply producing another copy. This problem is an important issue to address in order to make the system optimally reliable.

5. A POSSIBLE WIRELESS COMMUNICATIONS PROTOCOL

The following is a brief, high level description of how the Base Station and Sensor Stations might communicate in order to maximize data throughput and minimize lost data. Depending on the hardware design, frequency bandwidths, power requirements and other limitations, such a system may or may not be practical. It however illustrates a means whereby each Sensor Station is identical with every other with the single exception of station number.

There is one Base Station (BS) and multiple Sensor Stations (SS_i) where *i* is the number of the station. Assume there are *N* stations numbered 1, 2, 3, ... *N*.

Normal Operations (All stations can receive from and transmit to BS)

BS tells SS_i to broadcast its data.
BS goes into listen mode and receives the data from SS_i
At the same time all other stations SS_j ($j \neq i$) receive the data
Each station purges all old data not requested by BS
This is repeated for all values of *i* from 1 to *N*

Example (Normal operation):

BS goes into Transmit Mode, all SS_i in Receive Mode
BS tells SS₁ to send its data and goes into Receive Mode
SS₁ goes into Transmit Mode, sends its data and returns to Receive Mode
All SS_i ($i \neq 1$) and BS receive the data
BS goes into Transmit Mode
BS tells SS₂ to send its data and goes into Receive Mode
SS₂ goes into Transmit Mode, sends its data, all SS₁ data is purged (since BS already received it). SS₂ returns to Receive Mode.
Process repeats. ...

Lost stations (One, or more SS_i can hear, but cannot communicate directly with BS)

BS broadcasts for SS_i to broadcast its data.
BS goes into listen mode but fails to receive the data from SS_i

At the same time all other stations SS_j ($j < I$) listen for and some receive the data
Station SS_i is added to the list of requested stations (at BS)
BS broadcasts for SS_k to broadcast its data and that of lost stations
 SS_k broadcasts all requested data that it has
At the same time all other stations receive the data they do not already have and add it to
their memory. Data not requested is purged from memory.
BS updates its request list and repeats the process.

Example: One BS, 4 SS. Assume SS_2 can broadcast to 1 and 4 but not to BS or SS_3 .
Assume also that SS_2 cannot hear BS.

Operation:

BS broadcasts for SS_1 to send data from a Request list (at first this only contains SS_1).
All stations hear request except SS_2 .
 SS_1 broadcasts its data. BS and all stations hear and record data.
BS broadcasts for SS_2 to send data. All stations that hear purge the SS_1 data stored
internally.
BS does not get response from SS_2 adds SS_2 to request list
BS broadcasts for SS_3 to send data from the Request list (that now contains SS_2 and
 SS_3). All stations hear request except SS_2 .
 SS_3 only has its data (not that of SS_2) and sends only that.
All listening devices add SS_3 data to their memory
BS hears SS_3 data and removes SS_3 from request list.
BS broadcasts to SS_4 for data (SS_2 and SS_4 are on the list)

6. FUTURE PROJECTS

The ADAS project, initiated by NASA and continued during the summer 2001 revealed the need for several issues to be addressed. To turn the prototype into a fully functional system, many critical issues need to be addressed. Among these is the need for redundancy in the imbedded processor. Since the processor, in the present system, is a single point of failure, its failure could make the entire system non-functional. Selecting the processors and duties to be performed by each is still an open problem. Another very interesting and important problem is that of intelligent software. Intelligence can be subdivided into the operational and evaluational. The operational controls the switching and reprogramming of the components while the evaluational determines the health of the system as well as that of the parameter being sensed. Health evaluation can be performed with Neural Networks and other intelligent means. Neural Networks have been successfully applied to the evaluation of the Gaseous Hydrogen Flow Control Valve. With Neural Networks, the ADAS could not only collect data but catalog system degradation over time. Another very useful tool would be a model. This could take cost, component reliability and other factors into account for deciding how many components are needed and where. All of the afore mentioned seem like ideal student projects which can be undertaken at The Oklahoma State University.

7. REFERENCES

- [1] Mitel MT8806 8X4 Analog Switch array
- [2] Lattice Semiconductor ispPAC
- [3] Analog Devices AD5541 %volt Serial-Input Voltage-Output 16 bit DAC
- [4] Intelligent Sensors & Instrumentation Advanced Data Acquisition Technologies Design Architecture Document. NASA internal document, January 2001

8. ACKNOWLEDGEMENTS

The work undertaken this summer was of great educational benefit to me. I hope that I contributed positively to the development of an intelligent data acquisition system of the future. Many people made the summer not only productive but very enjoyable. Whenever I needed answers or resources, there was always someone there to provide help and support. In mentioning the few I am sure I will be omitting the many who have made this summer very special for me. I hope I will be forgiven for not having mentioned you by name. My deepest thanks to Jose Perotti, my NASA Colleague. The fine talents of Bradley Burns and Tony Eckhoff, of Dynacs, Dr. Carlos Mata and Angel Lucena, of NASA and others helped bring the system to its present level. Without this support there would be many more unsolved problems than there are today. Dr. Ramon Hosler and Cassie Spears worked very hard to make the summer truly enjoyable for all the faculty and their families and guests. Their professionalism and friendship will long be remembered. Good luck and fondest wishes to Dr. Hosler on his retirement.