Optical Sensing as a Means of Monitoring Health of Multicomputer Networks

by

David L. Forbis

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE

in

Electrical Engineering

APPROVED:

Dr. Kent A. Murphy, Chairman

Dr. Richard O. Claus

Dr. Scott F. Midkiff

July 1994
Blacksburg, Virginia

Optical Sensing as a Means of Monitoring Health of Multicomputer Networks

by

David L. Forbis

Kent A. Murphy, Chairman

Bradley Department of Electrical Engineering,
Virginia Polytechnic Institute and State University

(Abstract)

The use of optical sensors to perform health monitoring in fault-tolerant multicomputers can allow the multicomputer to detect imminent failure in a particular section of the interconnection network due to damaging strain. This detection method allows the rerouting of critical data before data link failure occurs.

This thesis investigates the implementation of the extrinsic Fabry-Perot interferometer into an optical hybrid communications/sensing network. A testbed of personal computers, acting as nodes of a multicomputer, are used to monitor the integrity of the network to a high degree of accuracy. When a node determines that an adjacent data link is no longer reliable due to physical damage, communications are rerouted and the node is shut down.

Results of experiments with the testbed have shown that redundant nodes can be used to share computational loads, increasing the performance of the multicomputer, until network failure forces redundant nodes to assume full responsibility for computational tasks. Multicomputer performance suffers as a result of network damage, but full functionality is retained with no occurrence of errors or unknown conditions due to data link failure.

Acknowledgments

I would like to thank my chairman, Dr. Murphy, for his guidance and friendship over the past two years. I would also like to thank Dr. Claus for his endless supply of support and ideas, and to Dr. Midkiff for helping me get started on my work with this project back in my undergraduate career. Many thanks to these three for their time spent serving on my committee.

I would like to thank all of my friends at FEORC for their companionship, advice, and help, given unselfishly. My two years spent here were made more memorable by the dynamic personalities found in the lab.

I must express my thanks to Expresso Pizza and Millstone coffee for feeding me while I wrote this paper. Thanks also goes out to the Dave Matthews Band for feeding my ears during this time.

A special thanks to Asif Malik for helping me decide what would not be appropriate titles for this thesis.

Finally, and most importantly, I thank my parents, Dick and Sheryl, my sister, Holly, and my close friends Ajay and Shawn for their unending support and encouragement, without which this entire endeavor would have been much more difficult.

This thesis is dedicated to the Philadelphia Flyers, that they may return to the playoffs before the end of the twentieth century.

Table of Contents

List of Figures and Tables

1.0 Introduction and Background

The goal of this thesis is to perform physical fault detection on a multicomputer by using interferometric optical sensors to measure applied strain on interconnection data links. The main interest in research of this sort is to provide reliable optical interconnections between computers that may operate in harsh environments. An example of such a system would be found on an aircraft or other military vehicle which, by nature of its function, is expected to undergo some form of structural trauma in its operating lifetime.

This thesis will examine an application of the problem of dynamic fault tolerance, and provide an example of a practical solution. Damage will be inflicted upon a data link of an operational hybrid communication/sensing multicomputer and detected with the use of optical sensors. It will be shown that the multicomputer can reroute all communications functions and reallocate computational tasks around this faulty data link with no loss of information or functionality.

A limitation of this research is that damage will be inflicted upon a specific data link, meaning that the multicomputer will only monitor damage occurring in one location. It is assumed that there are no backup or redundant data links on the network; the node connected to the damaged data link will no longer be available to process any tasks.

The multicomputer will be faced with two computational tasks: communicating with a host computer and resolving the output of an optical sensor into meaningful strain measurements. These two tasks will be executed in parallel by two of the nodes of the multicomputer. Upon the detection of damage to the interconnection network, one of the

nodes will assume responsibility for the sequential completion of both tasks. It will be shown that the efficiency of the multicomputer drops with the reduction of computing resources available after network damage has occurred.

It will also be shown that the multicomputer can accurately convert the output of an extrinsic Fabry-Perot interferometer into a strain value and compare this calculated value to a predetermined maximum strain value, by use of an algorithm that continuously calculates strain based on the sinusoidal output of the interferometric sensor.

1.1 Fiber Optics

The fields of communications and sensing have been revolutionized by the advent of fiber optics, the science of transmitting information and data in the form of light through small strands of silica waveguides, known as optical fibers. In 1970, the first practical low attenuation fiber was developed at Corning Glass Works. [1,2] This low attenuation fiber marked the beginning of the optical communications era, in which single optical fibers have shown the ability to replace dozens, or even hundreds, of existing copper wire transmission lines. Optical fibers provide many advantages over copper wire, including, but not limited to, increased bandwidth, reduced attenuation, immunity to electromagnetic interference, and smaller size. [1]

The optical fiber is detailed in Figure 1.1. The refractive index of the core, n_1, is slightly higher than that of the cladding, n_2, to the extent that

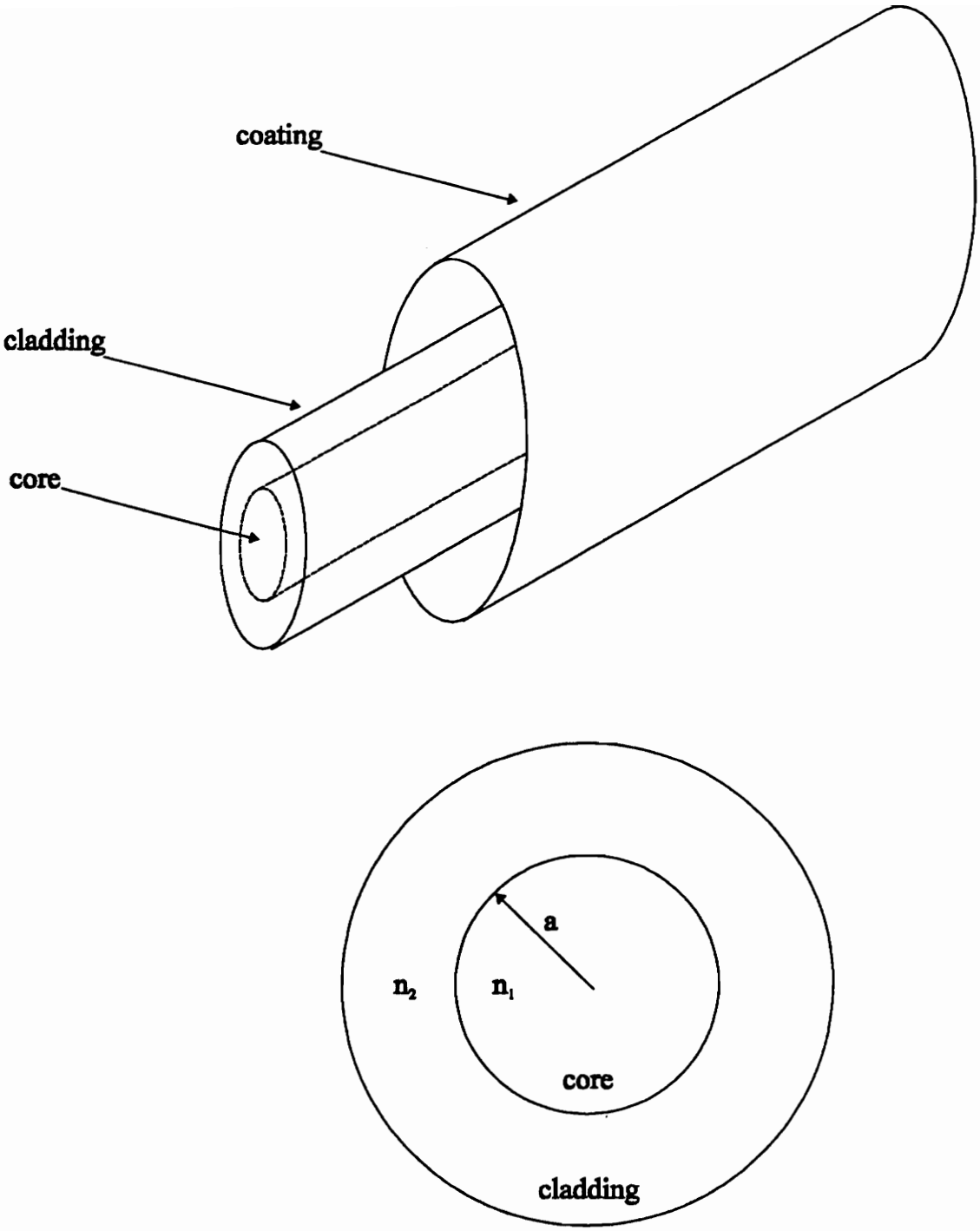


Figure 1.1. Optical fiber.

$$n_1 \approx n_2 \approx 1.467. \tag{1.1-1}$$

This approximation defines what is known as a weakly guiding fiber. [2,3] This dielectric waveguide supports the propagation of a discrete number of guided modes, each of which is a solution to the reduced wave equation, [4]

$$\nabla_t^2 \psi + \left(k_0^2 n^2 - \beta^2\right)\psi = 0, \tag{1.1-2}$$

where

$$\nabla_t^2 = \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial \phi^2}, \tag{1.1-3}$$

k_0 is the free space wavenumber, β is the propagation constant, and ψ is a Cartesian component of the fields E or H. The number of modes supported in a fiber can be determined by the normalized frequency V, [5]

$$V = \frac{2\pi a}{\lambda} NA = \frac{2\pi a}{\lambda}\left(n_1^2 - n_2^2\right)^{1/2}, \tag{1.1-4}$$

which is a function of core radius a, wavelength λ, and numerical aperture of the fiber.

When dealing with small core sizes, the V number becomes quite small. Below a V number of 2.405, only one mode exists. This mode is denoted HE_{11}, and will exist down to a point where the core radius is zero for step-index matched cladding fiber. [1] The HE notation indicates that both E and H fields are non-zero in the z-direction. [1] In weakly

guiding fibers, the HE_{11} mode is often referred to as the LP_{01} mode, stressing that the modal field is linearly polarized. Thus,

$$\psi = E_x \text{ or } E_y,$$

corresponding to the x or y polarization of the LP mode. The solutions to the wave equation for the LP^x_{01} mode are

$$\psi = A\begin{cases} \dfrac{J_0\left(\dfrac{ur}{a}\right)}{J_0(u)} & r < a \\ \dfrac{K_0\left(\dfrac{wr}{a}\right)}{K_0(w)} & r > a \end{cases} \qquad (1.1\text{-}5)$$

where

$$u = k_0 a\sqrt{n_1^2 - \overline{\beta}^2},$$

$$w = k_0 a\sqrt{\overline{\beta}^2 - n_2^2},$$

$$\overline{\beta} = \beta / k_0,$$

A is an arbitrary amplitude, J is the Bessel function, and K is the modified Bessel function.

One of the factors that limits the bandwidth of a data carrying optical fiber is dispersion. Dispersion can be classified into two categories: intramodal and intermodal dispersion. Intermodal dispersion only affects multimode fiber, as it is the dispersion caused by the different group velocity of different modes.[1] Intramodal dispersion manifests itself in the form of material dispersion and, of high importance to applications involving singlemode fiber, waveguide dispersion.

The delay introduced by intermodal dispersion in step-indexed fiber, in which n_1 is constant with respect to radius, easily derived from ray optics, is

$$\Delta t = t_{\max} - t_{\min} = z\frac{n_1}{c} - z\frac{n_1^2}{cn_2}, \tag{1.1-6}$$

where c is the speed of light in a vacuum, and z is the distance traveled by the light through the fiber. [6] Clearly, this dispersion is greater with large differences in refractive indices between the core and the cladding. In weakly guiding fibers, this dispersion effect is minimized.

Material dispersion is a physical property of all fibers used with sources that have a non-zero spectral width. Since all physical sources have some non-zero spectral width, material dispersion takes place with any combination of source and fiber. Since the refractive index is a function of wavelength, any deviation in wavelength will cause a change in signal propagation speed. If the spectral width of a source is given by $\Delta\lambda$, the spread in pulse width is given by [6]

$$\Delta t = -\frac{z\lambda}{c}\left|\frac{d^2 n_1(\lambda)}{d\lambda^2}\right|\Delta\lambda. \qquad (1.1\text{-}7)$$

It can be seen from Equation 1.1-7 that, as the signal propagates further down the fiber, the effect of material dispersion will increase linearly.

Another factor that affects intramodal dispersion is waveguide dispersion. The propagation constant of a mode, β, is dependent upon wavelength. Again, using a source with non-zero spectral width, β will not be an exact number for a particular mode; instead, it will have a range proportional to the spectral width, Δλ. This non-discrete value of β is the cause of waveguide dispersion. Again, this will result in an increase in signal dispersion as the transmission length increases. If the dependence of n_1 on λ is ignored, the effect of waveguide dispersion on group delay, specifically for the LP_{01} mode, can be expressed in terms of V-number and normalized propagation constant b as [3]

$$\frac{d(Vb)}{dV} = b\left[1 - \frac{2J_0^2(ua)}{J_{-1}(ua)J_1(ua)}\right]. \qquad (1.1\text{-}8)$$

The pulse spread caused by the effect of waveguide dispersion is given by [1]

$$\Delta t = -\frac{n_2 z \frac{n_1 - n_2}{n_1}\Delta\lambda}{c\lambda} V\frac{d^2(Vb)}{dV^2}. \qquad (1.1\text{-}9)$$

Putting aside the assumptions of independence of n_1 and λ,

$$\Delta t = -\frac{z}{2\pi c}\left[\lambda^2 \frac{d^2\beta}{d\lambda^2} + 2\lambda \frac{d\beta}{d\lambda}\right]\Delta\lambda \qquad (1.1\text{-}10)$$

expresses the total delay imposed by material and waveguide dispersion on signals in singlemode fibers. [6]

Material dispersion can be approximated as zero at an operating wavelength around 1300 nm, [7] as the rate of change of the core index of silica fibers is almost zero between wavelengths of 1300 and 1320 nm. This dispersion becomes much greater as the operating wavelength drops.

As a means of quantitatively comparing these forms of dispersion, three cases relevant to the optical network used in this thesis can be examined. When germanium-doped silica fibers are operated at a wavelength of about 820 nm with a source of spectral width of 5 nm, the time delay associated with material dispersion is about -0.6 ns/km. The other form of intramodal dispersion, waveguide dispersion, can be found to be about -0.0487 ns/km at 1300 nm operation, and -0.408 ns/km at 820 nm operation. Clearly, when this silica fiber is used with a 1300 nm source, waveguide dispersion is the main source of dispersion. However, when the operating wavelength decreases, material dispersion can be just as strong a contributing factor to dispersion. If, in addition to the above mentioned dispersion figures, intermodal dispersion is considered for the case of a 9 μm core diameter matched with an 820 nm wavelength source, resulting in four mode operation, the total dispersion, intermodal and intramodal, is approximately equal to -18.7 ns/km. Operating over a distance of 10 km, for example, the maximum transmission rate would be

about 5.3 MHz. This same fiber, operated in the singlemode range at a dispersion reducing wavelength of 1300 nm produces a total dispersion of -0.0487 ns/km. Again, over a length of 10 km, the maximum transmission rate would be over 2 GHz. Clearly, intermodal dispersion is the dominant form of dispersion in multimode communication systems.

The effects of dispersion must be taken into account when developing any high speed optical network. In multimode applications, weakly guiding fiber is essential to minimizing intermodal dispersion. Applications that require singlemode fiber again can make use of weakly guiding fiber to reduce waveguide dispersion. In silica fibers, material dispersion can be reduced by choosing an operating wavelength around 1300 nm; silica exhibits low dispersion effects at this wavelength. [1] Fiber geometries other than matched cladding can be used to reduce dispersion effects. In addition to the selection of fiber and operating wavelengths, dispersion can be reduced by opto-electronic repeaters to refresh the desired sharp edges of the transmitted data in a similar manner as repeaters are used to combat attenuation. [1]

1.2 The Extrinsic Fabry-Perot Interferometer

The extrinsic Fabry-Perot interferometer, or EFPI, is an optical fiber sensor that is used to measure displacement, from which strain can be derived. It can be used in many of the same applications that use electrical strain gages, while providing the advantages of optical sensors over electrical sensors, namely, reduced electromagnetic interference, reliable operation at high temperatures, small size, and multiplexing ability.

The EFPI, shown in Figure 1.2.a, is composed of two fiber segments, cleanly cleaved, inserted and bound into a small segment of hollow tubing. The length between the two bonding points, L, is known as the gage length, which is used as a constant to determine strain along with a measured change in gap separation, s. The output of an EFPI is monitored as in Figure 1.2.b, with back-reflections monitored by a photodetector.

The EFPI determines changes in the gap separation by producing an intensity modulation created by the interference between two Fresnel reflections, R_1 and R_2. Reflection R_1 can be treated as a reference signal. Reflection R_2 will interfere with R_1, producing a measurable signal at the end of the fiber. As the gap separation changes, the relative phase of R_2 with respect to R_1 will shift, changing the intensity of the output signal that is received at the photodetector, [8]

$$I = A^2\left[1+\frac{2ta}{a+2s\tan\left(\sin^{-1}(NA)\right)}\cos\left(\frac{4\pi s}{\lambda}\right)+\left(\frac{ta}{a+2s\tan\left(\sin^{-1}(NA)\right)}\right)^2\right], \qquad (1.2 - 1)$$

where A is an arbitrary amplitude, t is the transmissive coefficient of the air-glass interface at the end of each fiber, and NA is the numerical aperture of the fiber. The output intensity cycles though a series of fringes as the gap separation increases or decreases, as shown in Figure 1.3. The cosine term in Equation 1.1-2 represents the cycling of the fringes, and shows that, at an operating wavelength of 1300 nm, one fringe corresponds to

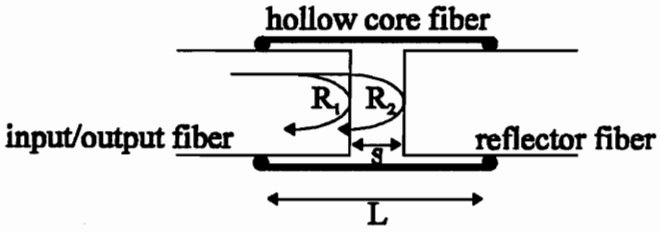


Figure 1.2.a. The extrinsic Fabry-Perot interferometer.

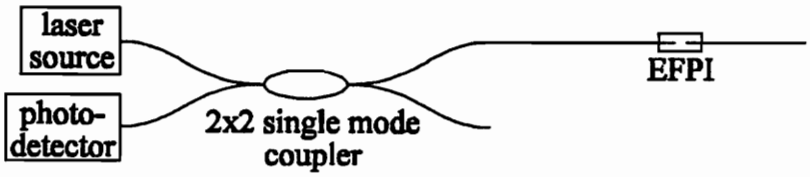


Figure 1.2.b. Operational EFPI setup.

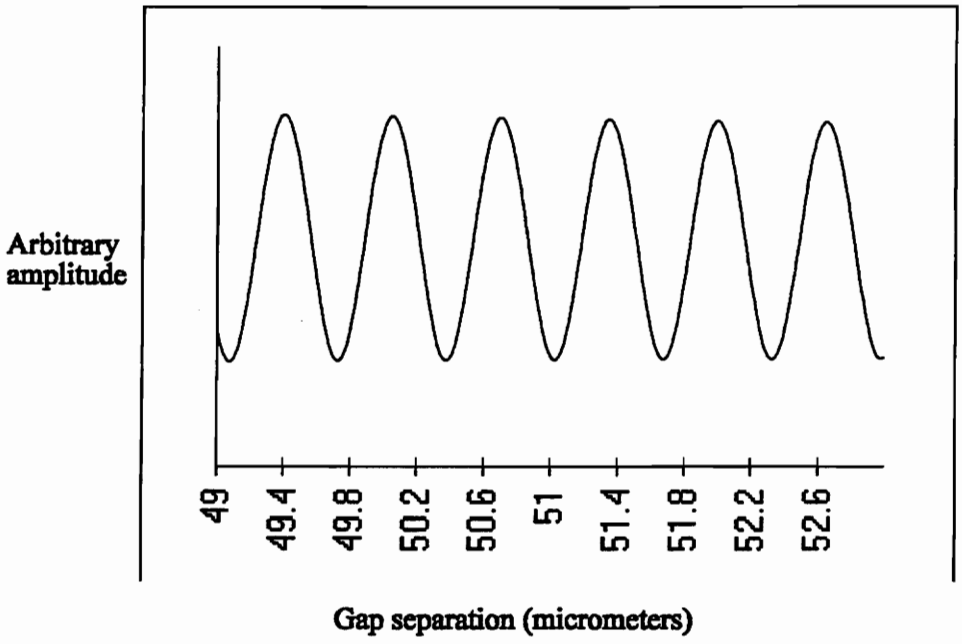


Figure 1.3. EFPI fringes.

0.65 μm displacement between the fiber endfaces. The coefficient of the cosine term decreases as the gap separation increases, accounting for the signal loss attributed to distance and light that is no longer coupled back into the core of the input/output fiber. The squared term in conjunction with the constant 1 accounts for the change in offset of the signal as the gap separation increases.

At its simplest, displacement can be calculated by counting the number of fringes that the output has passed through. A more accurate model of measuring displacement can be formed by solving Equation 1.1-2 for gap separation as a function of intensity. This more accurate model is discussed in Chapter 3, and allows for resolving displacement for intensity changes that do not encompass an entire fringe.

By knowing the change in displacement and the gage length of the sensor, each given in micrometers, the actual strain experienced by the EFPI can be calculated by

$$\varepsilon = \Delta s / L, \tag{1.2-2}$$

the resulting number being a unitless quantity known as microstrains.

The EFPI can be used in many different applications to determine strain, sense vibration, measure temperature changes, and many other physical perturbations that can be transferred by contact to the sensor. One of the main disadvantages that the EFPI poses is the computationally-intensive signal processing that is required to extract a strain measurement from a change of intensity.

1.3 Parallel Processing

The purpose of parallel processing is to increase the performance of a computer system by permitting operations or tasks to be executed or performed simultaneously. [9] The concept of parallel processing arose from the design of the Gamma 60 Computer, [10,11,12] which explored speedups provided by the simultaneous execution of instructions, and was further developed by Lehman in 1966 [10,11] to suggest that methods for exploiting parallelism of any given task should be studied, as the growth of parallel processing machines would play an important part in the future of computing.

In 1966, Flynn [13] suggested a classification of computer systems, in which any given system could be grouped according to the presence of one or more instruction and data streams. Computers that were only capable of executing one command on one datum were classified as single instruction stream, single data stream, or SISD. An example of this machine would be the IBM-PC. A single instruction stream, multiple data stream, or SIMD, computer is capable of executing one command on a series of data, such as vector processing. The CM-1 Connection Machine is a SIMD architecture that contains 65,536 processors, all fed by the same instruction stream. [20] A multiple instruction stream, multiple data stream (MIMD) computer, such as the Intel iPSC, is a good example of a multiprocessor structure [9] in which several independent computers, synchronized by a common operating system, each execute individual programs and share data. A MISD, or multiple instruction stream, single data stream, system is not a good example of an effective machine. Few applications can be found for a machine that is capable of performing many functions on a single piece of data. Figure 1.4 provides a summary of Flynn's taxonomy.

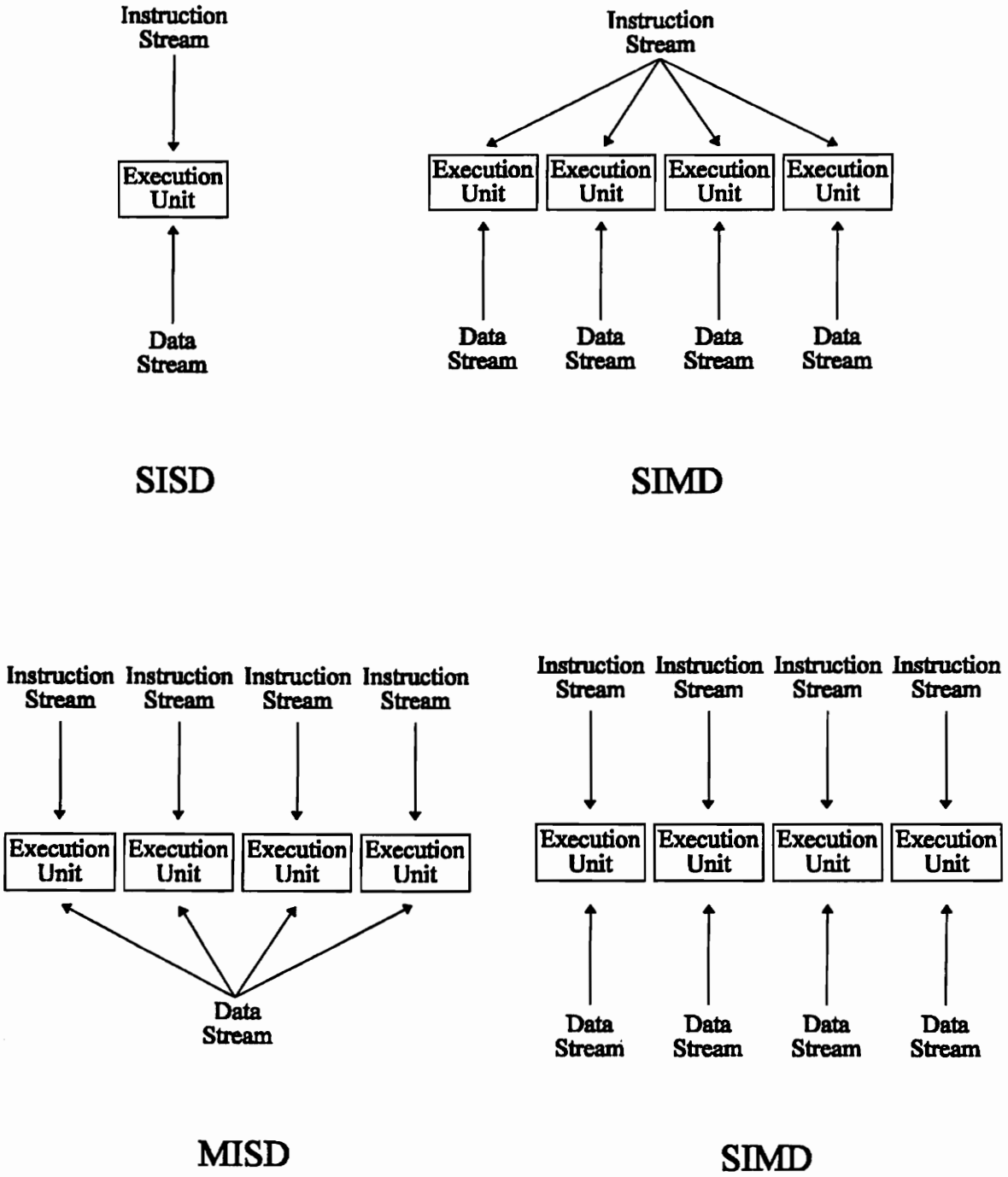


Figure 1.4. Flynn's taxonomy of parallel architectures. [9,13]

Two memory structures of a multiprocessor can be defined: shared memory and distributed memory. [14] In a shared memory architecture, each processor has access, through an interconnection network, to one of several global memory units. In a distributed memory system, or multicomputer, [15,16] each processor, with local memory, is connected to surrounding processors by an interconnection network.

The interconnection network is an important aspect of the multicomputer, as performance can be limited by inefficient data exchange between processing units. [16] Decreased efficiency can be caused by many factors, including but not limited to the amount of network traffic and protocol efficiency. [15] In an interconnection network that has high activity, processor-to-processor messages may have to wait for the network to be free before data may be exchanged.

Protocol efficiency is an important factor in interconnection network performance. The amount of time that is required to move a message from one point to another can be characterized by two values: setup time and data transmission time. This communications time can be expressed as

$$T = \alpha + n\beta \,, \qquad (1.3\text{-}1)$$

where α represents setup time, n the number of bits in the message, and β the throughput in bits per second. [16] It can be seen from Equation 1.3-1 that not only is transmission bandwidth an important factor in communications efficiency, but setup time is critical, as it is constant for any message length. Efficiency can be increased by sending a few large data packets instead of several small packets, increasing the ratio $n\beta/\alpha$.

As computing power grows with the increase in number of processing elements and data streams, the importance of the interconnection network which allows processors to share data begins to show. A powerful multicomputer can be rendered useless if the interconnection network is too slow to properly handle the message traffic required by the computer or application.

1.4 Fault Tolerance

As multicomputers develop into more complex machines, it is essential that hardware and software failures be dealt with in such a way that the operation of the machine as a whole is unaffected, or marginally affected, by such a failure. The concept of fault tolerance allows computers to reconfigure, repair, or correct a significant portion of errors that may occur under normal, or abnormal, operating conditions.

One of the most common forms of fault tolerance is hardware redundancy. Redundancy is the replication of critical processing and support hardware, such that the extra hardware is not used for normal operation. These redundancies can be classified as static or dynamic.[17]

Redundant components may all be constantly active in a static redundancy scheme. In a triple modular redundancy model, [18] three processors perform identical calculations, and a voting mechanism chooses the correct answer. This system is error-free as long as two out of the three processors have produced a correct result.

Dynamic redundancy takes a different approach. Faults must first be detected, and then backup or redundant systems must be activated to take over the tasks previously performed by the faulty processing unit. The GF11 computer, developed by IBM, is used to perform simulations of quantum chromodynamics, which may take a full year to run. [19] It can be easily seen that a failure during such a time consuming simulation can result in losses of extreme amounts of time and money. Therefore, the GF11 has 64 redundant processors that may be used to replace any of the 512 primary processors should a hardware fault be detected. It can be deduced that dynamic redundant systems may be more cost effective, as they require fewer redundant elements to achieve fault tolerance. However, dynamic redundant fault detection schemes will invariably be more complex than those associated with static redundant systems.

2.0 MIL-STD-1773 Hybrid Communications/Sensing Testbed

A multicomputer network consisting of three personal computers connected by an optical fiber network has been constructed. This network, or testbed, was used to perform fault tolerance and parallel processing experiments based on data provided by interferometric optical sensors. Intercomputer communications over the optical network were performed by use of the MIL-STD-1773 communications protocol. This chapter will provide a brief description of the MIL-STD-1773 protocol as well as an overview of the testbed construction and operation.

2.1 Overview of MIL-STD-1773

As avionics technology grew beyond analog instrumentation and controls, the need for a centralized digital control system arose to allow aircraft components and on-board computers to communicate. Developed in 1968 by the Society of Automotive Engineers, MIL-STD-1553 was the communications protocol that, in 1973, the Air Force chose to adopt for use in the F-16 fighter aircraft. [21] After a few years of field testing, revisions to the original MIL-STD-1553 led to the release of the upgraded protocol, MIL-STD-1553B. Accepted by many other countries and NATO, the standard is still in use today, incorporated into many different types of aircraft, missiles, satellites, and tanks. In the commercial sector, MIL-STD-1553B has found use in applications such as oil wells and subways. [21]

In 1988, MIL-STD-1773 was recognized as the fiber optic equivalent of MIL-STD-1553. The operation of the protocol was unchanged, only the physical interface was changed

from twisted pair copper wire to optical fiber. No specifications were set in MIL-STD-1773 with regards to types of transmitters, receivers, or fiber that was to be used in a network; any configuration that is able to transmit and receive data in accordance with MIL-STD-1553B is acceptable for implementation.

The topology of a MIL-STD-1773 network consists of one bus controller (BC) and one or more remote terminals (RTs), all connected by a common data bus, as shown in Figure 2.1. All nodes, BC and RT, are capable of receiving any data placed on the bus, however, only those nodes whose addresses are specified by the command word will store the data supplied by the bus. The BC is responsible for directing all communication over the data bus via the command word. An RT is incapable of initiating communications, requiring the command of the BC to begin transmitting or receiving data.

As many as 31 remote terminals may be active on the data bus at one time, each with a unique five-bit address. Each RT has 30 memory locations, known as subaddresses, that are accessible to the protocol for data storage or retrieval by the BC. [22] Only one BC may be present on the bus at any one time, although other processors may take over the job of BC if so instructed by the acting BC, or other systems.

The format of the command word issued by the BC is shown in Figure 2.2. [23] In this command word, an RT address is specified. The RT that is addressed is instructed by this command word to transmit (or receive) a specified number of words from (or to) a specified subaddress. A parity bit at the end of the word is used for error detection.

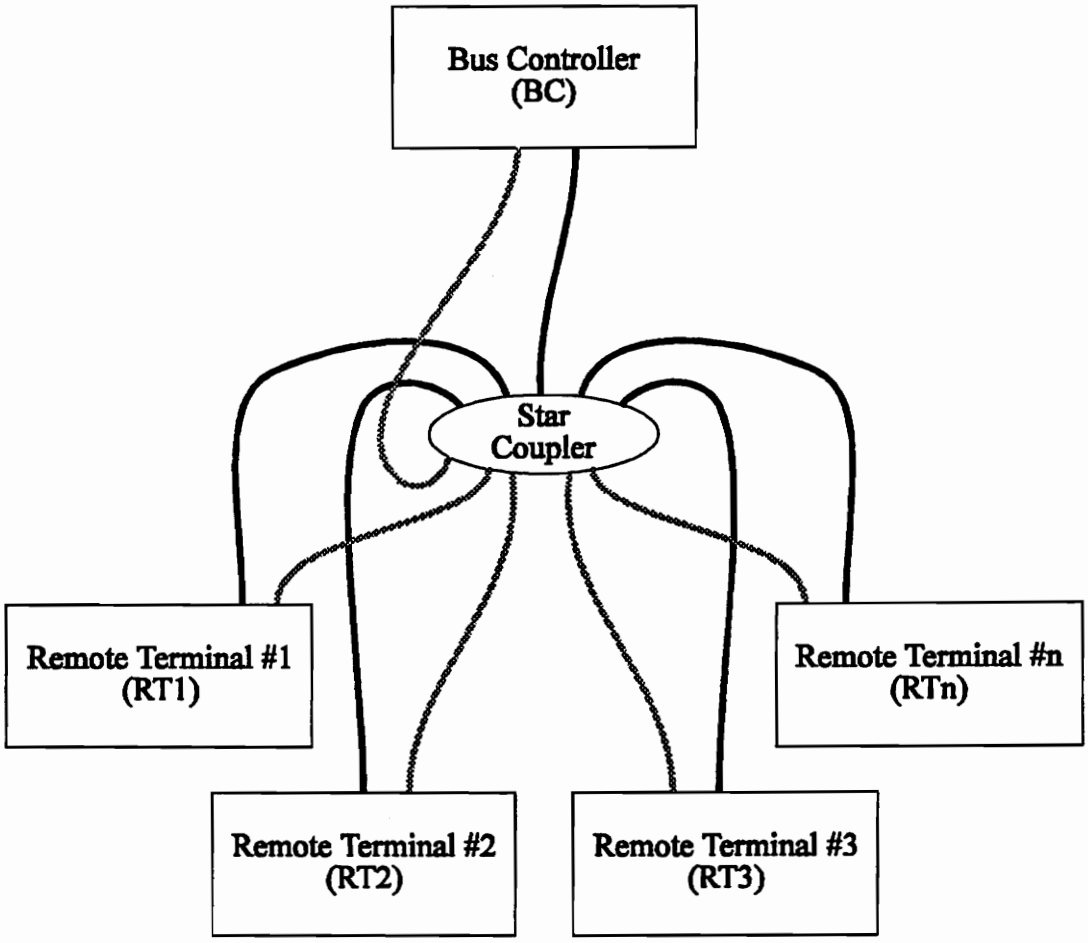
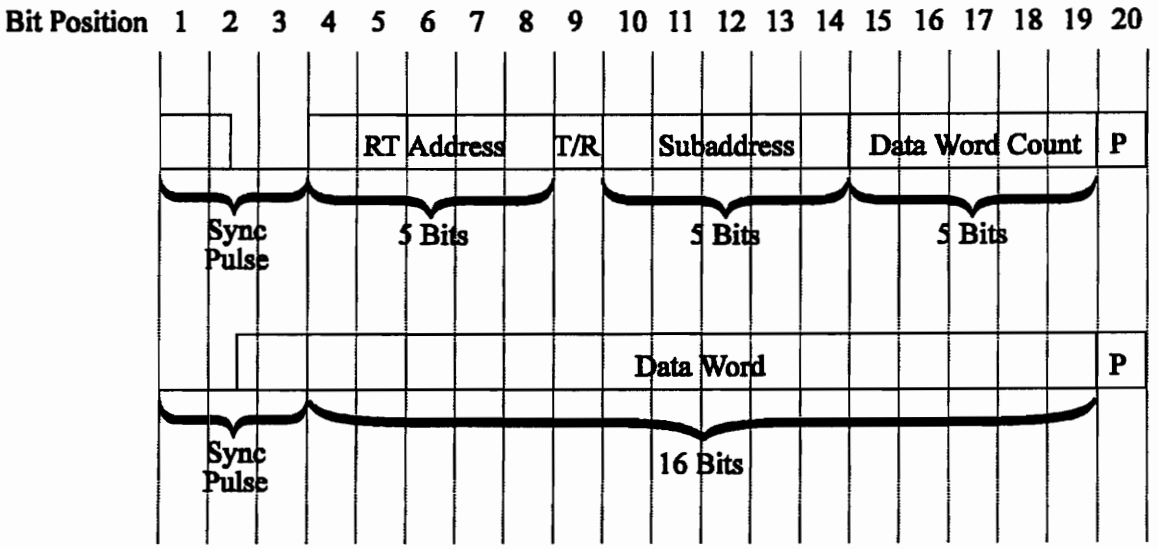


Figure 2.1. MIL-STD-1773 topology.



Sample Command Word

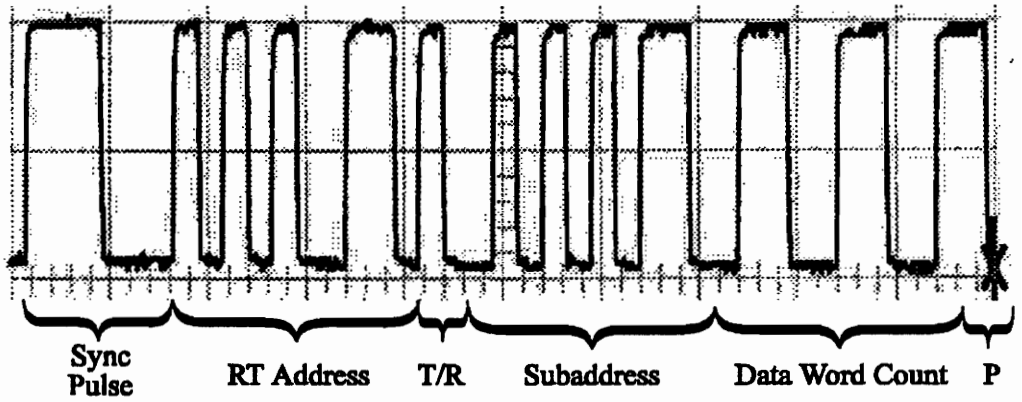


Figure 2.2. Command and data word structures.

RT-to-RT communication takes place when the BC issues a receive command to an RT, then issues a transmit command to another RT. This dual command has the effect of telling one terminal to begin listening, then telling another terminal to begin transmitting. BC-to-RT and RT-to-BC communications are simpler, requiring only one command word, since the BC is involved in the data exchange. The BC simply issues a receive (or transmit) command to the RT, then the BC either transmits (or receives) the data to (or from) the RT. A 20-bit status word is written to the bus by an RT involved in communications. The status word precedes the data words when the RT is transmitting, and follows the data words when the RT is receiving. Thus, in RT-to-RT transfers, two status words are reported, one by each RT.

Data to be transmitted data is encoded in Manchester format, which aids in keeping clock synchronization throughout the transmission period. [1] Data bits are represented by state changes in Manchester code, a low-to-high transition representing a logic zero and a high-to-low transition representing a logic one, as shown in Figure 2.3.

Data packets can be any size up to 32 words, with each word being composed of 16 bits of data with an additional four bits for synchronization and odd parity check, as shown as bit P in Figure 2.2. Only one packet may be transmitted in each session. Data is transmitted at a rate of 10^6 bits per second (1 Mbps). Although the transmission rate may be high, actual throughput is significantly lower due to the small packet size imposed by the standard, the setup time required for the BC to initiate communication with the RTs, and the processing time required for the receiving node to retrieve the data from or store the data to the subaddress.

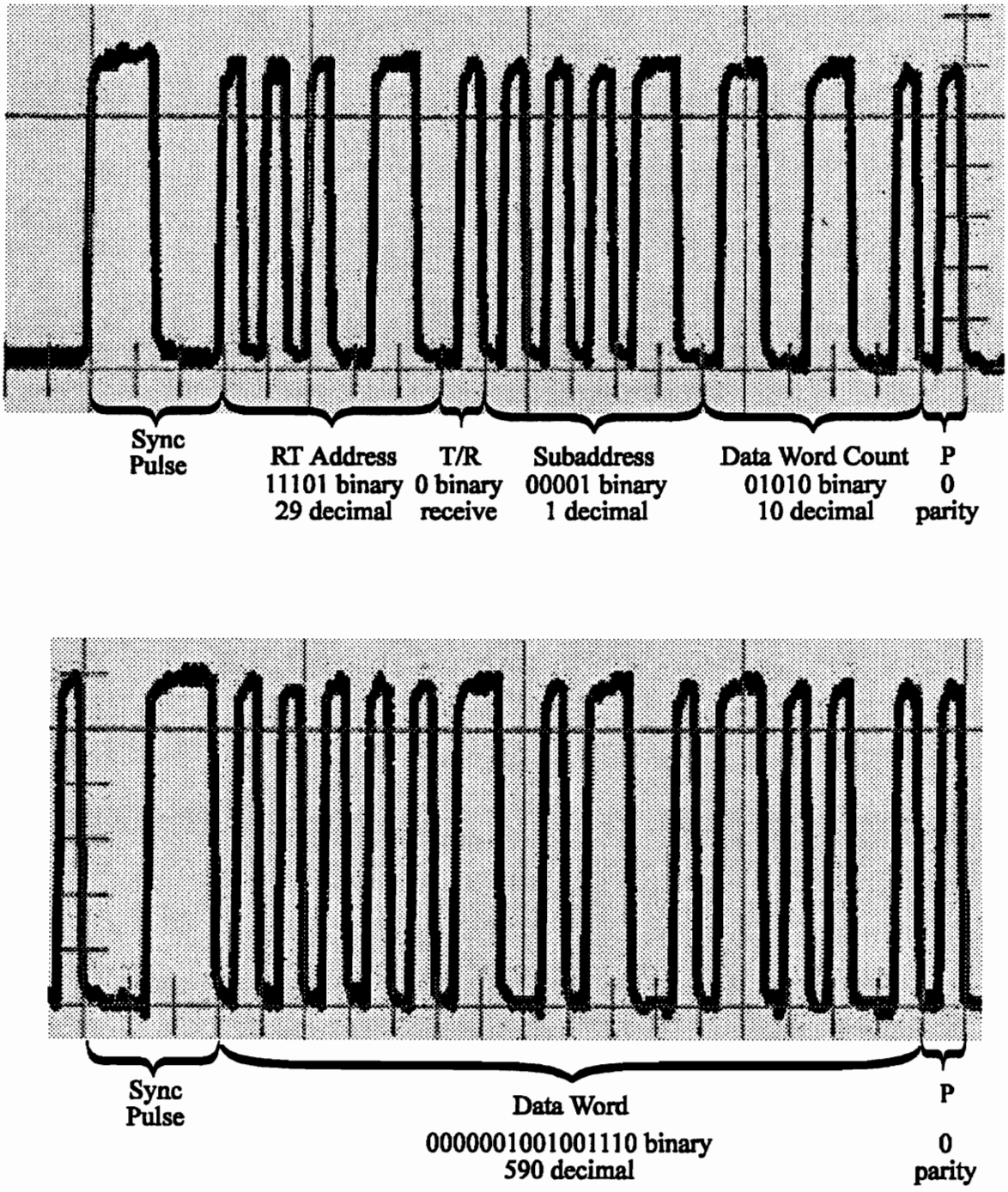


Figure 2.3. Manchester coding example of command and data words.

Error detection is performed by the receiving RT. As specified in Paragraph 4.4.1.1 of the MIL-STD-1553B definition, the RT sets the message error bit in the status word if any of the following cases occur:

- The synchronization pulse is invalid.
- The Manchester coding is invalid.
- 16 bits of data are not presented.
- Parity of the received word is not odd.

If the message error flag in the status word is set, it is the responsibility of the application program to retransmit a request.

Using the form of Equation 1.3-1, the time required to transmit a packet is

$$T = \alpha + 2\times 10^{-5} n, \tag{2.1-1}$$

where n is the number of 16-bit words being transmitted and α is the setup time. The value of β is found by taking into account a data rate of 10^6 bits per second and the additional four bits that are added to each data word by the protocol for synchronization and parity checking. Specifically, neglecting processing and transit time, α is a representation of the time required to transmit command words and status words. The setup time for BC-to-RT communications is shorter than the setup time for RT-to-RT communication, as two command words and two status words must be transmitted by the BC in the latter case. Neglecting processing time, α can be found to be 40 μs for BC-to-

RT communications, and 80 μs for RT-to-RT communications. Experimental values for α are discussed in Chapter 4.

2.2 Hybrid Communication/Sensing Testbed

A testbed for performing experiments on hybrid communication and sensing over the same optical fiber was designed and constructed by McHenry, et al.[24,25] This testbed consisted of three personal computers (PCs) connected by multimode fibers, over which MIL-STD-1773 was used to transmit data. Intensity-based air-gap sensors were used to determine information about external strains applied to the data links.

For this thesis, the testbed has been modified to permit the use of interferometric sensors to collect data. The addition of interferometric sensors required rebuilding the interconnection network to make use of singlemode fiber, which is the necessary medium for interferometry. Since EFPIs were to be employed as the sensing devices, the network had to be designed in such a way to allow the EFPI to operate in its most efficient reflective mode. To allow operation in reflective mode, photodetectors used for sensing were placed on the same side of the EFPI as the source used for sensing. A diagram of the optical interconnection network is given in Figure 2.4. Note that communication is performed at a wavelength of 820 nm, while sensing is performed at 1300 nm, thereby preventing crosstalk between the communications and sensing signals. As can be seen in Figure 2.5, there are effectively separate optical paths for the communication and the sensing signals. This separation is provided by different types of photodetectors for

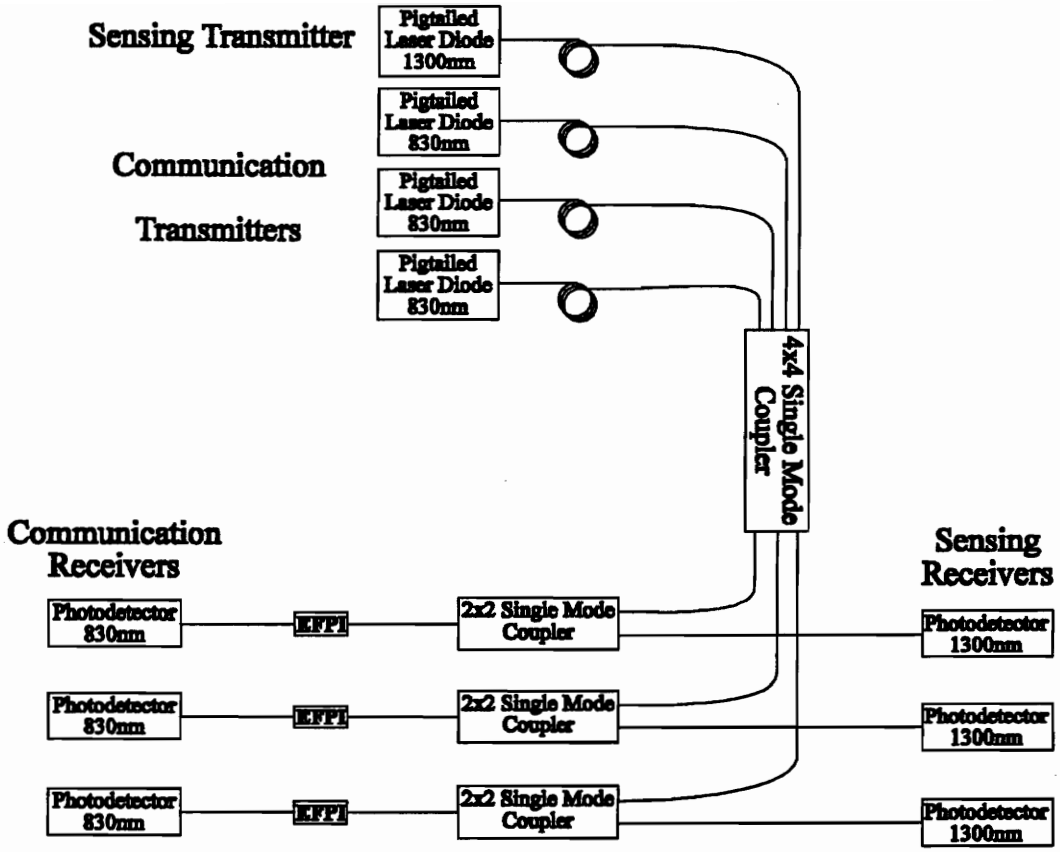


Figure 2.4. Hybrid communication/sensing testbed with EFPI sensors.

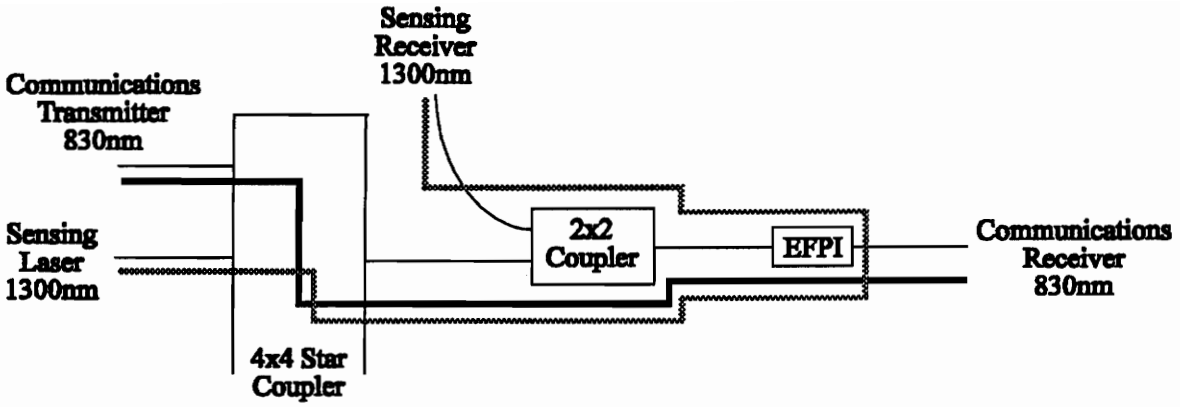


Figure 2.5. Effective optical paths of communication and sensing signals.

communications and sensing. Effectively, the EFPI allows transmission of communications data while reflecting the sensing signal, which is split off to a photodetector by a 2x2 coupler.

The use of singlemode fiber necessitated the use of laser sources for communications and sensing, because LED sources are not powerful enough to inject sufficient power into the 9 μm (singlemode at λ=1300 nm) core diameter fiber used in the network. A power budget for the network, shown in Table 2.6, indicates that 1 mW of injected power is sufficient to allow all nodes to successfully perform communications functions.

The EFPI is an ideal choice as an interferometric sensor to implement. Its structure lends itself well to hybrid sensing and communications as it permits light to propagate through, as opposed to a Michelson interferometer which completely reflects the optical path at the sensor head. The addition of a Fabry-Perot cavity in the communications link has the potential to significantly increase dispersion and attenuation of the transmitted signal, as a highly reflective cavity would cause a reduction in drop-off of the trailing edges of square pulses. This loss of edge definition is caused by the pulse traveling through the cavity being reflected back and forth several times. If the Fabry-Perot cavity is low finesse, i.e. small Fresnel reflections at the end faces of the fiber, communications data is not attenuated more than 5dB. Fresnel reflections for the standard EFPI are 4%. Signal dispersion caused by the cavity is not a function of distance, as the signal spread only occurs once with each Fabry-Perot cavity. A low finesse cavity, in which only one extra reflection is significant, keeps dispersion down on the order of 0.4 ps. This dispersion does not affect most systems, even higher speed SONET systems. [1] Any reflections

Table 2.6. Power budget for singlemode optical network.

Device	Associated Loss
4x4 Singlemode Coupler	- 6.1 dB
2x2 Singlemode Coupler	- 3.2 dB
Splice Tubes (4)	- 0.4 dB
EFPI	- 5 dB
Total Losses seen by communications system	- 14.7 dB
Minimum power for communications to operate	- 25.4 dBm
Required transmitter power	- 10.7 dBm = 0.085 mW

generated in Fabry-Perot cavities employed as in-line sensors on a communications link must not contribute significantly to the forward transmission on the data fiber.

The implementation of EFPI sensors provides the network with dramatically more sensing power than previously used intensity-based sensors, as illustrated in Figure 2.7. [25] The increased resolution provides the computers with the ability to sense smaller perturbations to the data link. The sensors can be constructed to measure strain, temperature, or vibrations.

In addition to requiring a laser source to inject sufficient power into the singlemode fiber, interferometric sensors require a source that has a coherence length greater than the optical sensing path, which in the case of the EFPI, is twice the gap separation. A 1300 nanometer wavelength laser is used to provide power to all sensors in the network, providing the needed power and coherence length to successfully make the conversion from multimode air-gap sensors to singlemode EFPIs.

2.3 Modified Remote Terminal Operation

The operation of the remote terminals was modified from traditional "dumb terminal" operation by altering the program executed by the bus controller in an attempt to improve network performance and reduce RT dependency of the BC. Subaddress 1 of each RT is designated as a status register and is polled periodically by the BC via a transmit command directed at subaddress 1. This status information provides the BC with critical data on the

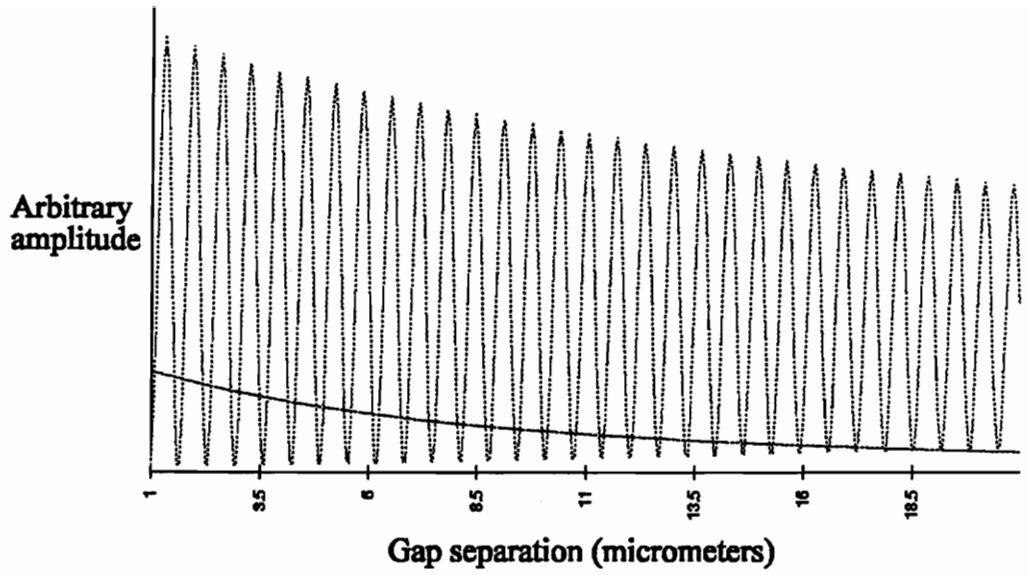


Figure 2.7. Response of EFPI in contrast to air gap sensor.

readiness of the RT to receive or transmit data, shown in Figure 2.8. If a particular RT is ready to transmit data to the BC, or to another RT, the contents of subaddress 1 are altered to contain three values: 1) the number of data words to be transmitted, 2) the subaddress which holds the data to be transmitted, and 3) the destination of the data. This creation of a status register provides an important improvement of the operation of the RTs, as it effectively allows the RT to request service from the BC, instead of waiting for service at intervals determined by the BC. The reliability of the protocol is also increased with this change. Previously, the BC was required to "guess" when valid data was available for transfer from the RT, as no status information was provided by the RT. Now, the BC is told when such data is ready for transmission.

Offset within Subaddress 1 (byte position)	
0	RT Request Transmit Flag/Destination Indicator 0 when not ready, otherwise provides destination address (1 for BC)
1	Number of Data Words to be Transmitted by RT Total number of data words ready for transmission
2	Subaddress in which Outgoing Data is Stored Location of outgoing data words
3	NOT USED
4	NOT USED
5	RT Ready to Receive Flag/SA Indicator 0 when not ready, otherwise points to SA for incoming data

Figure 2.8. Map of subaddress functionality.

2.4 Overview of Experimental Goals

In the process of redesigning the existing MIL-STD-1773 sensing/communications testbed, four primary goals were established:

- Demonstrate interferometric sensing in hybrid communications/sensing applications.
- Perform real-time EFPI fringe counting and strain resolution.
- Demonstrate an implementation of a dynamic redundant fault-tolerant network.
- Demonstrate parallel processing to reduce computational load on a single computer.

The introduction of interferometric sensing into a computer interface provides the ability to quickly decode the otherwise difficult-to-process fringes that are produced by a Fabry-Perot interferometer. The accuracy of the EFPI is a desirable feature that can increase the sensitivity of measurements performed by the testbed, allowing more subtle and delicate tests to be performed, such as vibration analysis and potential vibration dampening.

With a direct link to an EFPI, a computer can be used to process the EFPI output while such intensity changes occur, as opposed to post-processing captured data. The fringe processing algorithm that was developed for use with this system is detailed in Chapter 3.

Since MIL-STD-1773 is a communications protocol that is used commonly in military applications, it would be natural to assume that a system designed to operate in a military

environment would at some point be subjected to damage that is beyond normal operational strain and aging. Such a system would undoubtedly employ redundant systems as backups to reduce the chance that catastrophic failure to one element would result in mission failure. The application of dynamic redundancy requires some form of damage detection to provide the multicomputer network with information as to when rerouting of tasks is required due the occurrence of physical faults on the interconnection network. Chapter 3 will detail experiments in which a data link containing an in-line EFPI was placed under stress. After the resulting strain level surpassed a preset tolerance level, the data link was determined to be unreliable, and communications were rerouted to an alternate data link before loss of critical data could occur due to data link failure.

To establish parallel processing, two major tasks were developed and implemented in the testbed. The first task involves an RT processing and storing data transmitted from the bus controller. The second task is to monitor the strain that is experienced by the data link that the RT relies on to provide data from the BC. These two tasks are divided among the two active RTs, as detailed in Chapter 3. After a point in which the data link becomes unreliable, both tasks are moved to one RT, rerouting traffic from the failing link. It was shown that the network still performed both tasks, although the completion of the pair is slowed by the reduction in computing resources.

3.0 Testbed Operation and Functions

The establishment of fault tolerance of the multicomputer required the implementation of a fringe counting algorithm that would process the output of the interferometric sensors into a discrete strain level. Given that strain level, the multicomputer could reorganize itself according to the health of the interconnection network. The configuration of the testbed setup and program logic used to reorganize the multicomputer is presented in this chapter.

3.1 Physical Configuration and Program Flow

To implement fault detection with the MIL-STD-1773 testbed, the configuration in Figure 3.1 was designed and constructed. Two computers, the remote terminals, are equipped with extrinsic Fabry-Perot interferometers. These two computers act as slaves of the third computer, the bus controller. The RTs operate together to receive a string of data from the BC and, at the same time, process sensor data containing information about the current strain in the fiber. These two subtasks together comprise a single unit of operation of the network, hereafter referred to as a task.

While the network is operating in an "undamaged" state, both RTs are available to lend resources in cooperation toward completion of the task. After a fault has been detected, only one RT will still be available for use in the network, and it will be required to process both subtasks.

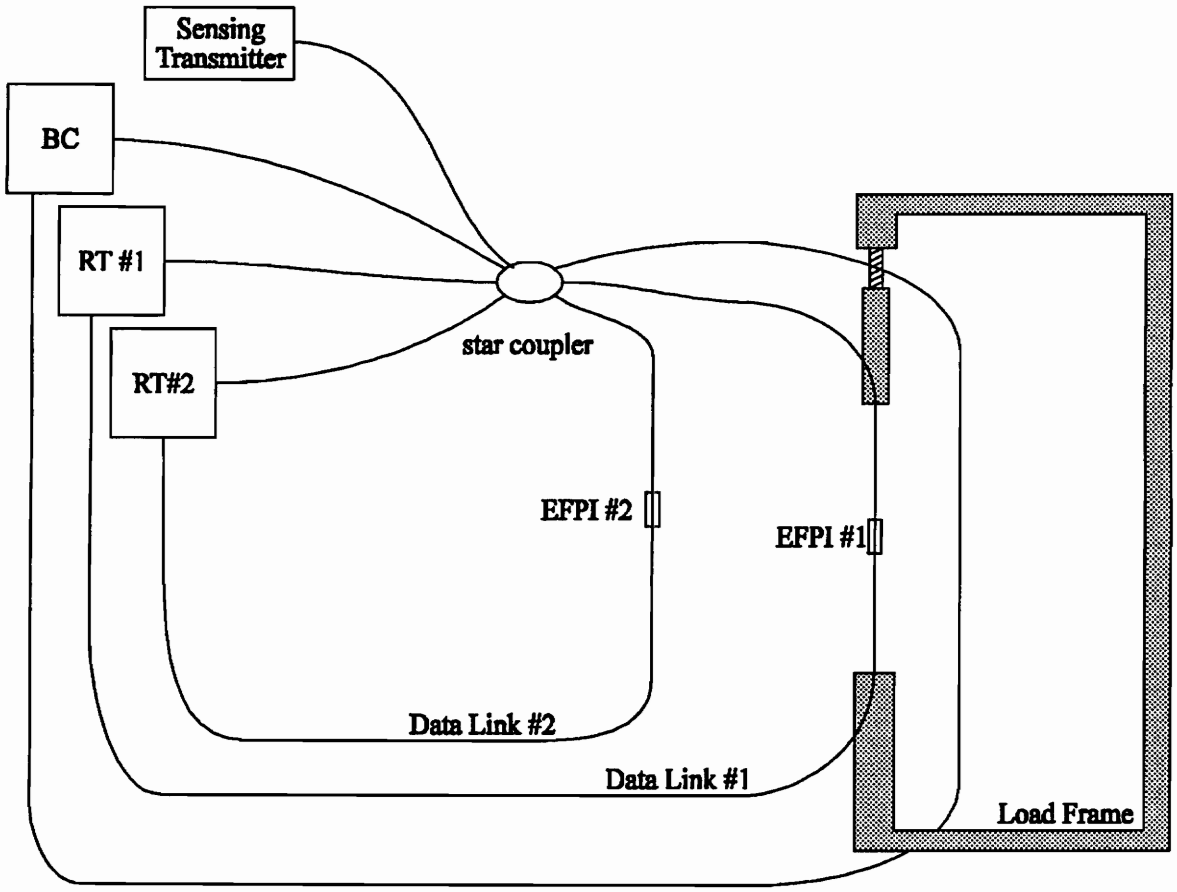


Figure 3.1. Experimental setup with data link #1 attached to load frame.

Initially, RT#1 is dedicated to Subtask #1, the reception, processing, and storage of the data packet transmitted by the BC. RT#1 also transmits data collected from EFPI#1 to RT#2. Subtask #2, in which the sensor data is collected and processed into strain information, is allocated to RT#2. Once failure of link #1 is imminent, both subtasks #1 and #2 are taken over by RT#2.

The BC transmits the packet of data to RT#1, which signifies the beginning of the task. Upon reception of this data packet, RT#1 requests permission to transmit the most current state of EFPI#1 to RT#2. RT#1 then processes the data packet received from the BC, which involves retrieving it from the subaddress in which it was received, and stores this data into a file on a fixed disk.

Upon reception of the sensor value from RT#1, RT#2 begins processing the sensor data into a strain value. The processing algorithm is detailed in Section 3.2. The sensor data and calculated strain are stored in a file. If the resulting strain value is above the acceptable strain levels tolerated by the data link, RT#2 transmits a signal to the BC notifying the controller that the link is no longer reliable. At this point, the BC issues a shutdown command to RT#1, and no longer acknowledges it as part of the network.

Now, operating as the only processing unit, RT#2 takes over the task previously performed by RT#1, that is, receiving and storing communications data issued by the BC. RT#2 now also takes measurements from EFPI#2 and performs calculations to determine strain exerted on link #2. Since there is no third processor able to take over in the event of failure of link #2, no action is taken based on the strain measured on link #2.

A flowchart of the operation of the network is shown in Figure 3.2. As can be seen in the flowchart, the parallelism of the task is lost after link #1 has failed. The drop in performance of the network after failure will be detailed in Chapter 4. Subaddresses within the RTs are dedicated to individual purposes, allowing the RT to easily determine what function is to be performed on incoming data.

The actual testing procedure begins when the software has been loaded into the three network computers and sensor sampling rates are initialized for the two RTs. These sampling rates tend to be about three to five times faster than the rate of task initiation, which is determined by an on-board clock on the BC. Data link #1 is suspended in a load frame, as shown in Figure 3.1, which allows the operator to manually increase the amount of strain in the data link.

Each time the BC initiates a task, the strain level of the data link is measured and compared with a predetermined threshold. When the strain produced by the load frame reaches this threshold, the data link is shut down, detouring message traffic from a potentially unstable path before critical data is lost.

3.2 Fringe Counting Algorithm

The output of an EFPI is an optical intensity that varies as a function of the separation between the two fiber end faces that comprise the Fabry-Perot cavity. This intensity variation, as discussed in Chapter 1, cycles through a series of fringes as the size of the gap between the two fiber endfaces increases or decreases, as described in Equation 1.2-1.

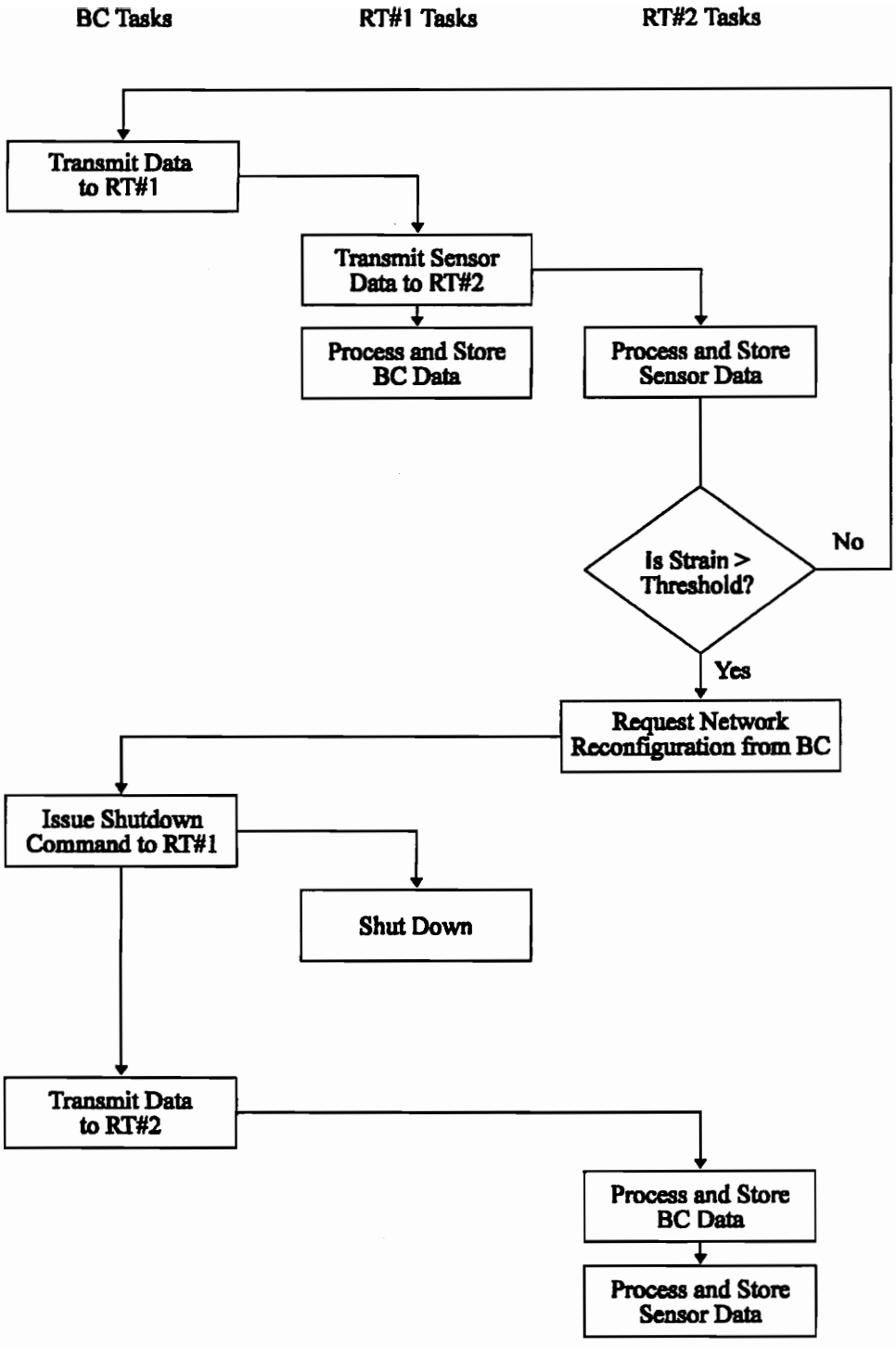


Figure 3.2. Flowchart of network operation.

Resolving strain from this intensity involves first solving Equation 1.2-1 for gap separation as a function of intensity, and then relating that separation to strain, as in Equation 1.2-2.

Equation 1.2-1 can be broken down into two equations, representing a static section and a dynamic section. The static section represents the envelope of the fringes, or how the fringe contrast and offset changes with displacement. The static section can be represented as

$$f_1(I_N) = \frac{4.5 + 0.3034s'}{8.82}\left[I_N - \frac{4.41^2}{(4.5 + 0.3034s')^2}\right], \qquad (3.2\text{-}1)$$

where s' represents a current estimate of gap displacement and I_N represents normalized intensity, provided by knowing the initial fringe contrast of the EFPI. The initial fringe contrast is found by a calibration routine within the RT program. Function $f_1(I_N)$ is able to make use of an estimate of gap separation since the value of $f_1(I_N)$ changes slowly as I_N changes, as opposed to the dynamic section,

$$\Delta s = \frac{2}{9.6664}\cos^{-1}\left(f_1(I_N)\right), \qquad (3.2\text{-}2)$$

which changes rapidly with respect to I_N. The value Δs represents the change in gap separation corresponding to a change in normalized intensity I_N. After Δs has been determined, the estimate s' can be adjusted before the next calculation is made.

The fringe counting algorithm makes use of a noise filtering scheme which rejects values of I_N that have not changed significantly since the last calculation. This prevents amplification noise arising from optical to electrical conversion from appearing as false changes in gap separation.

3.3 Fringe Counting Algorithm Verification

The fringe counting algorithm was tested with previously acquired data from the MIL-STD-1773 testbed to check for accuracy. Sample plots of captured EFPI data and resulting calculated gap displacement are shown in Figures 3.3.a through 3.3.d. It can be seen that the rate of change in calculated gap displacement is proportional to the speed at which fringes occur.

It is important to verify that the change in gap displacement as exactly one peak-to-peak fringe cycle occurs is 0.650 micrometers when the EFPI is interrogated with a 1300 nm wavelength source, as determined by Equation 1.2-1. The distance between peaks was measured from Figure 3.3.a and for selected peaks from Figure 3.3.c. These distances were plotted in Figure 3.4 along with the constant of 0.650 to demonstrate the accuracy of the fringe counting algorithm. The flat line represents the constant 0.650, which the algorithm should match. The data series corresponding with the calculations diagrammed in Figure 3.3.a closely matches the theoretical value of 0.650. The data series corresponding with the calculations diagrammed in Figure 3.3.c, however, is significantly lower than the theoretical value. This discrepancy is probably due to the fact that the

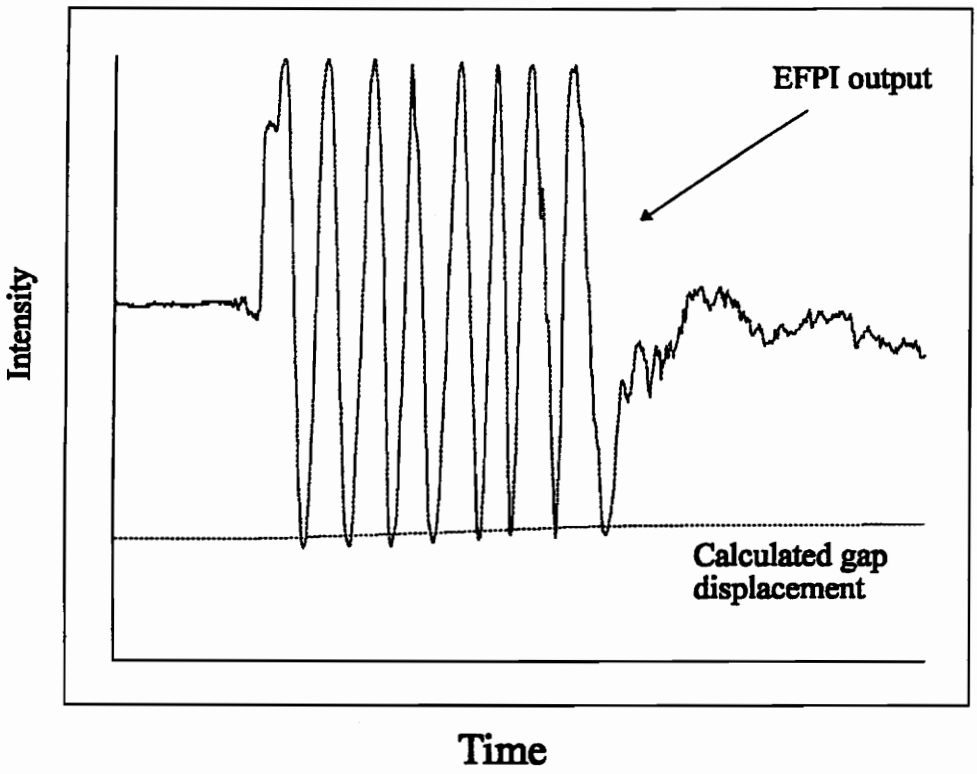


Figure 3.3.a. EFPI output and corresponding gap separation.

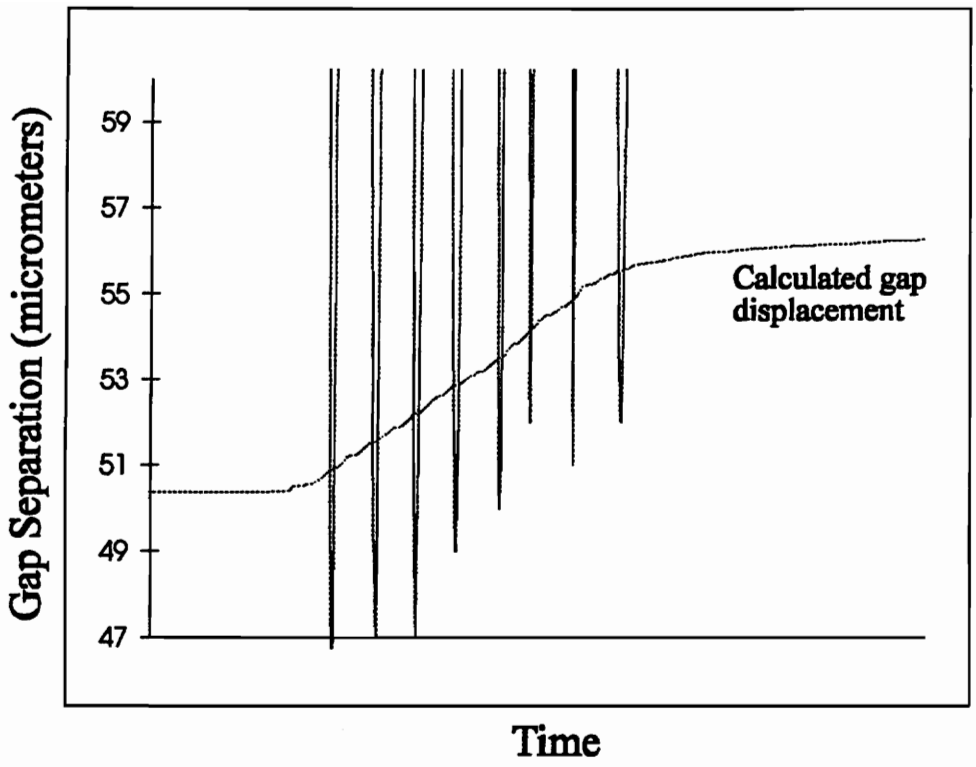


Figure 3.3.b. Detail of Figure 3.3.a.

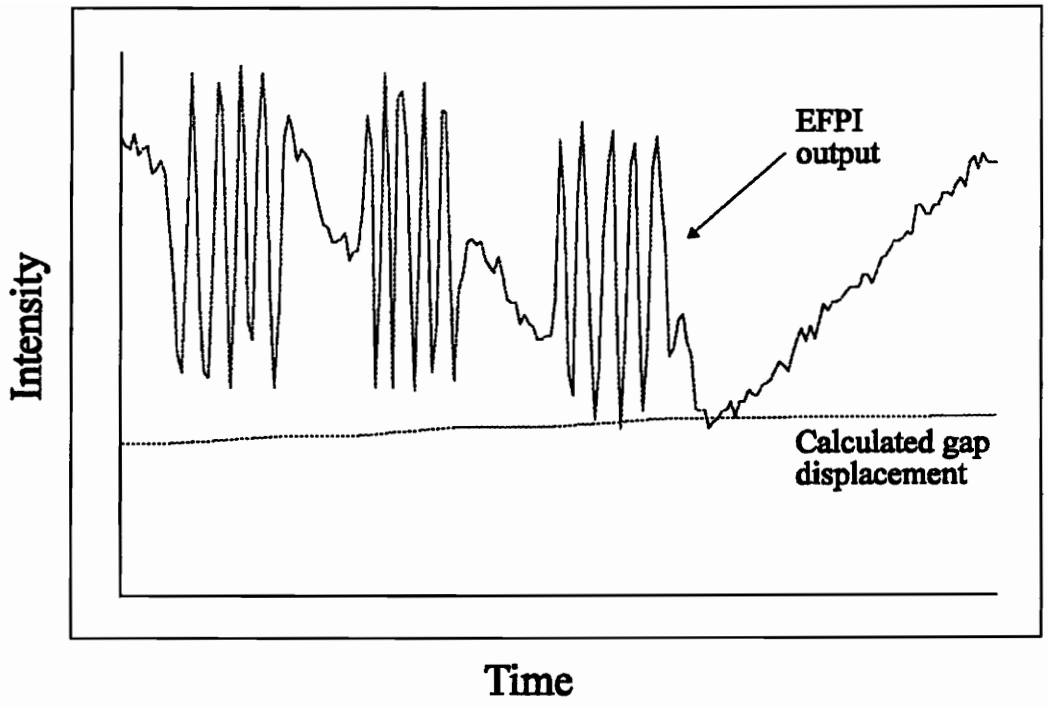


Figure 3.3.c. EFPI output and corresponding gap separation.

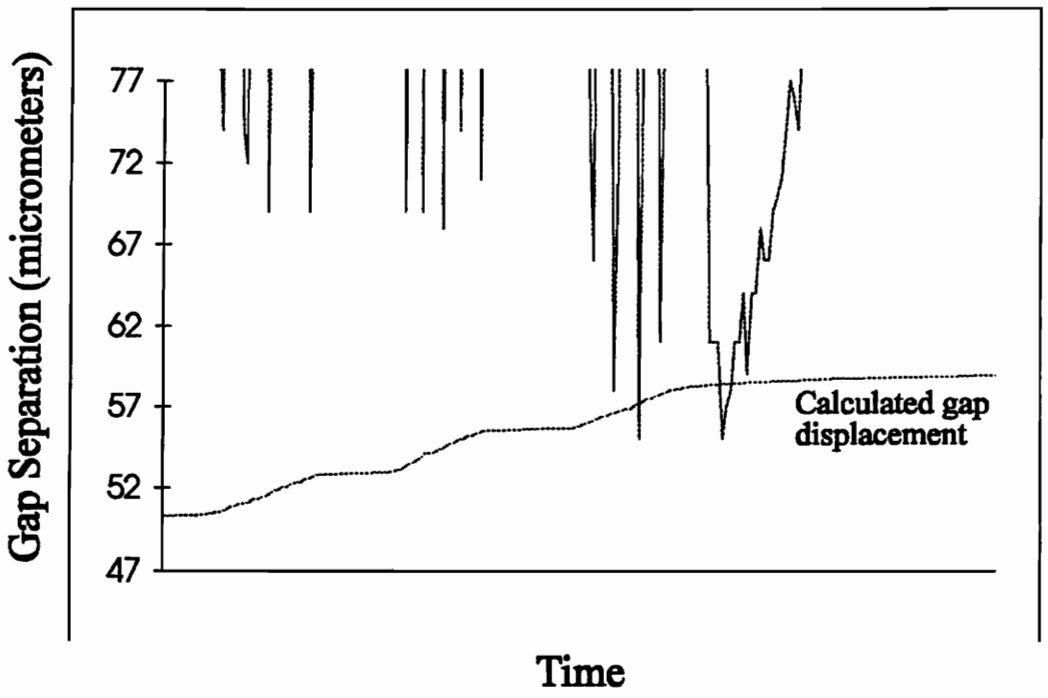


Figure 3.3.d. Detail of Figure 3.3.c.

data used in Figure 3.3.c was acquired at a sampling rate of 10 Hz, while a sampling rate of 20 Hz was used to acquire the data used in Figure 3.3.a.

The deviations shown in Figure 3.4 can be attributed to two sources of error, the first being the previously mentioned low sampling rate used to acquire data. The second may be due to the noise filtering scheme mentioned above. This filtering scheme has the disadvantage that no changes in gap separation are measured at the peaks and valleys of the fringe sinusoid, as the changes in intensity fall below the determined noise threshold. This loss of resolution at the peaks and valleys of fringes is an error that was unavoidable, and cannot be fixed without the use of very low noise optical-to-electrical conversion amplifiers.

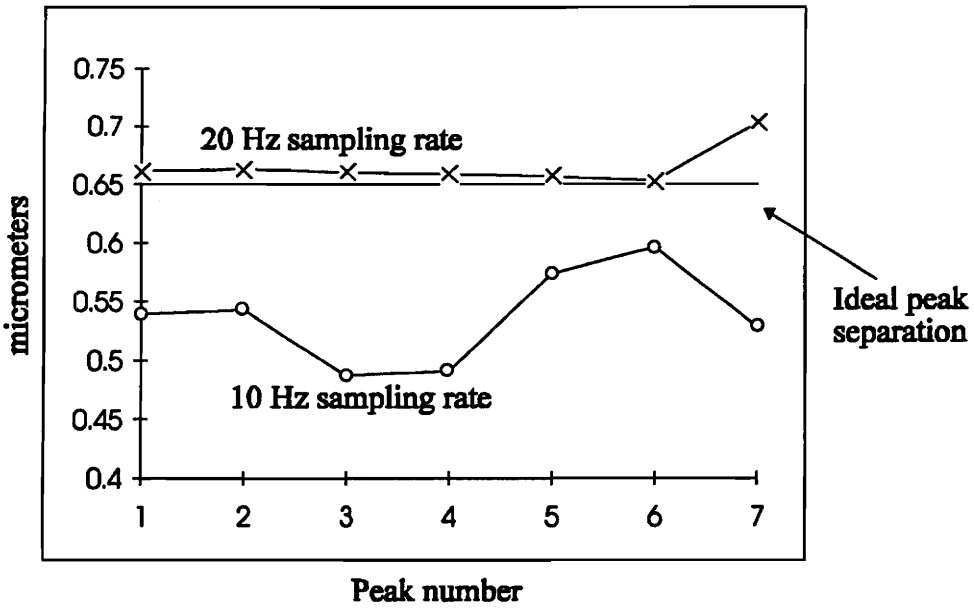


Figure 3.4. Peak to peak displacement comparison.

4.0 Experimental Results

The results of the experiments detailed in Chapter 3 show that EFPI sensors can be used to predict failure in sections of an optical network, allowing network reconfiguration before catastrophic failure occurs. This failure prediction can be taxing on computation facilities, but the effect of this added computational load can be reduced by allowing the use of redundant systems to help offload tasks from primary systems.

The fringe counting algorithm operated successfully, processing EFPI data into a strain value proportional to the tension level indicated by the load frame. A graph of sensor output and corresponding strain can be seen in Figure 4.1. As the strain reaches the threshold indicated, the rerouting of communications and reallocation of tasks occurs. The stream of data graphed in Figure 4.2 shows that no breaks or discontinuities occurred during the communication switching. The light markers in Figure 4.2 correspond to data packets received by RT#1 before data link failure has occurred. The dark markers correspond to data packets received at RT#2 after link failure and network reorganization has occurred. Thus, fault tolerance has been established, making use of dynamically redundant processors.

The division of the task into two subtasks that can be simultaneously processed by the two remote terminals provides an example of using the MIL-STD-1773 testbed as a successful parallel processing architecture. The time required to complete the task was measured from the transmission of communications data from the BC to the point at which the sensor processing subtask was completed. The sensor processing subtask was always the last subtask to finish; it was a more time consuming task than processing communications

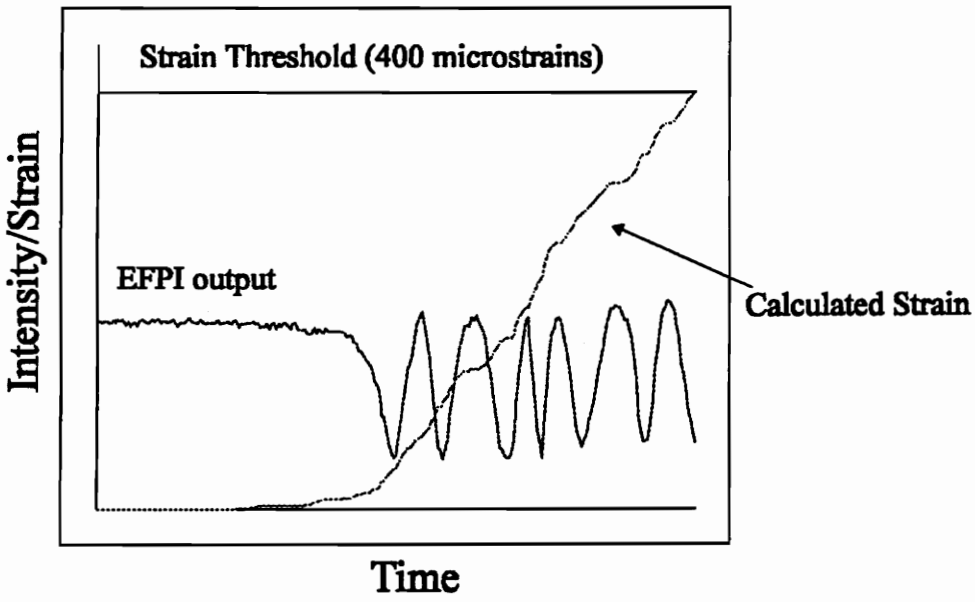


Figure 4.1. Sensor output, corresponding calculated strain value, and threshold cutoff.

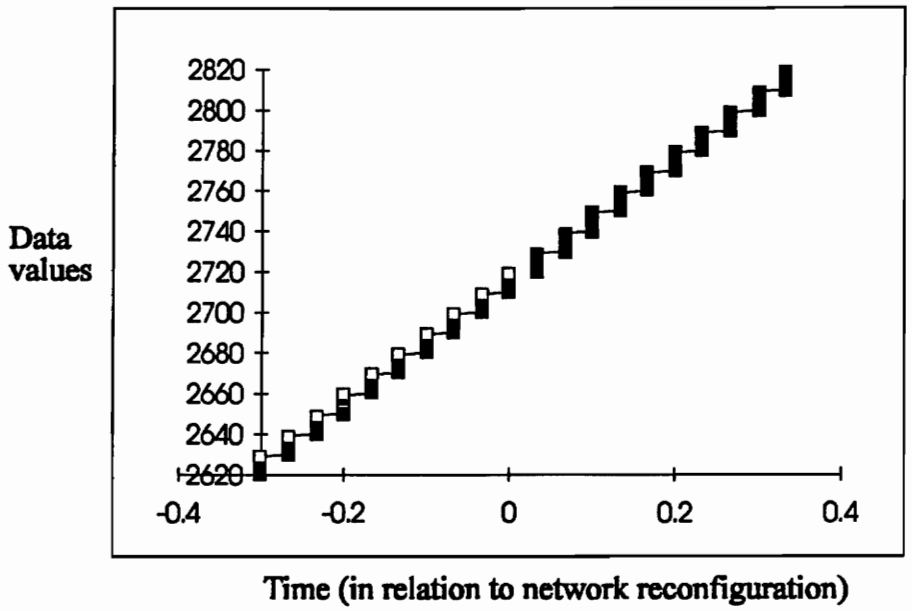


Figure 4.2. Verification of continuity of data flow.

data, and, in the single RT configuration (after data link failure), the communications data was processed before the sensor data was processed.

The actual times required to complete various sections of the overall processing task are given in Tables 4.3.a and 4.3.b. These times were measured with the aid of a logic analyzer, in which the smallest unit of measurement used was 0.02 ms. The logic analyzer was connected to the parallel ports of the three nodes. Each node sent codes to the logic analyzer through its parallel port at the beginning and end of tasks. The overall time required to complete the task, along with the times required to process each subtask, for both parallel processing and single processing cases are shown in Table 4.4. As can be seen, a 13.2 percent decrease in completion time was gained by the use of the redundant system to offload one subtask.

Communication delays imposed by the testbed setup and MIL-STD-1773 affect the processing times greatly. As can be seen in Table 4.4, the time required to complete the subtasks drops off considerably if communications latency is ignored. With no communications delay, it can be seen that the maximum speedup by parallel processing the two subtasks is

$$Speedup = \frac{4.32ms - 2.68ms}{4.32ms} = \frac{1.64ms}{4.32ms} = 38\%. \qquad (4.1\text{-}1)$$

Table 4.3.a. Processing times for parallel processing scheme.

Two Processors (before link failure)

Time	Event
0.00 ms	BC transmits to RT#1
0.40 ms	RT#1 receives from BC, begins data processing and storage, and transmits sensor data to RT#2
1.40 ms	RT#2 receives from RT#1 and begins evaluating strain information
2.22 ms	RT#1 completes data processing and storage (subtask #1 complete)
4.08 ms	RT#2 completes evaluating strain information (subtask #2 complete)

Event	Time required for completion
BC to RT#1 communication	0.40 ms
RT#1 to RT#2 communication	1.00 ms
RT#1 performs subtask #1	1.82 ms
RT#2 performs subtask #2	3.68 ms

Table 4.3.b. Processing times for single processor.

Single Processor (after link failure)

Time	Event
0.00 ms	BC transmits to RT#2
0.38 ms	RT#2 receives from BC and begins data processing and storage
2.20 ms	RT#2 completes data processing and storage (subtask #1) and begins evaluating strain information
4.70 ms	RT#2 completes evaluating strain information (subtask #2)

Event	Time required for completion
BC to RT#2 communication	0.38 ms
RT#2 performs subtask #1	1.82 ms
RT#2 performs subtask #2	2.50 ms

Table 4.4. Processing times of subtasks and tasks.

Event	Completion Time including Communications Delays	Completion Time not including Communications Delays
Subtask #1 (2 processors)	2.22 ms	1.82 ms
Subtask #1 (1 processor)	2.20 ms	1.82 ms
Subtask #2 (2 processors)	4.08 ms	2.68 ms
Subtask #2 (1 processor)	4.70 ms	4.32 ms
Entire Task (2 processors)	4.08 ms	2.68 ms
Entire Task (1 processors)	4.70 ms	4.32 ms

If the two subtasks were equal in weight, or required the same amount of processing time, the speedup calculated would reach its ideal maximum of 50 percent. With these two unequally weighted subtasks, the ideal maximum speedup is, as shown in Equation 4.1-1, 38 percent.

Communications delays measured are examined in Table 4.5, along with experimental values for the variables α and β described in Equation 1.3-1. According to the experimental values of α, the setup time associated with this implementation of MIL-STD-1773 can significantly reduce computing performance, considering that setup time accounts for 31.4 percent of the completion time in the parallel processing scheme, but only 3.8 percent of the completion time in the single processing scheme. This large discrepancy is due to the 1.1 millisecond setup time required for RT to RT transfers. A multicomputer with a more efficient interconnection network could greatly improve the performance of the parallel processing, moving performance closer to the ideal speedup shown in Equation 4.1-1. A graph of the performance of the parallel processing scheme employed here, along with the speedup associated with the MIL-STD-1773 network compared with the theoretical speedup associated with some other multicomputers [16] is shown in Figure 4.6. Note well that Figure 4.6 merely reflects communications speed; processing times were not taken into account and were assumed to be identical on the three multicomputer networks listed. It can clearly be seen that MIL-STD-1773, with its master-slave configuration of the BC and RTs is not as efficient as the interconnection networks employed by some other multicomputers, in which any node can transmit without permission of a single controller.

Table 4.5. Communications times and solutions to Equation 1.3-1.

Communications Times	$T = \alpha + n\beta$
BC to RT (10 data words): 0.38 ms	$0.38 = \alpha + 10(0.02)$ $\alpha = 0.18$ ms
RT to BC (10 data words): 1.3 ms	$1.3 = \alpha + 10(0.02)$ $\alpha = 1.1$ ms
RT to RT (10 data words): 1.3 ms	$1.3 = \alpha + 10(0.02)$ $\alpha = 1.1$ ms

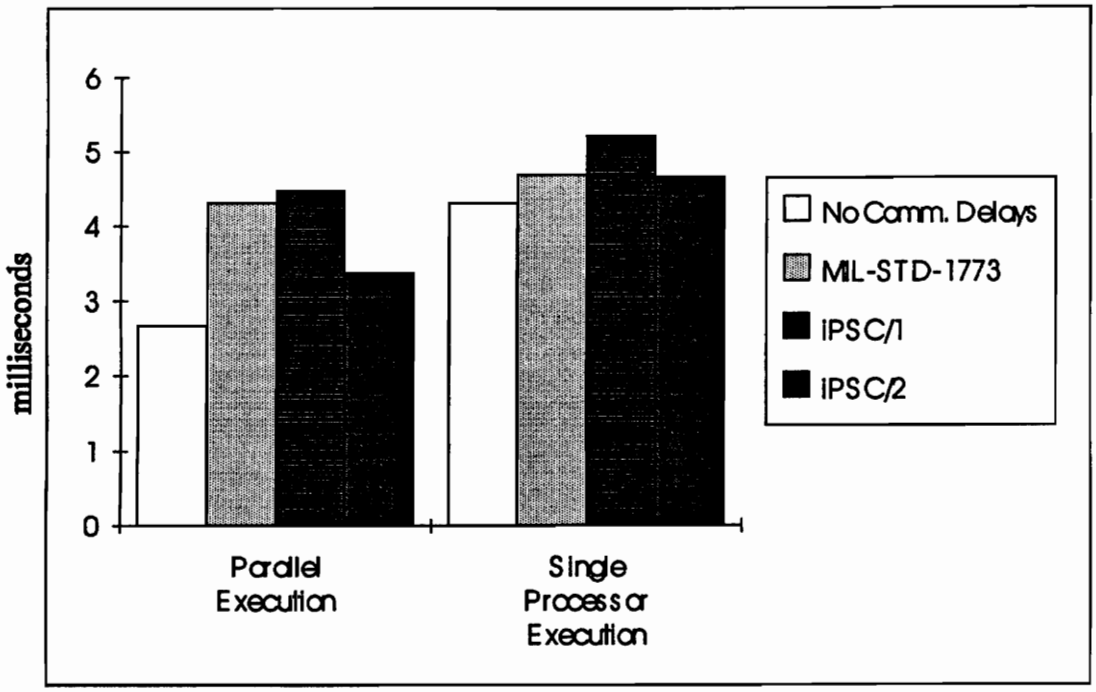


Figure 4.6. Comparison of MIL-STD-1773 with the case of no communication delay and with two alternate interconnection networks (see text).

5.0 Conclusions

It has been shown that the goals outlined have been met:

- EFPI output has been converted to a discrete strain level.
- Redundant systems have been employed prior to link failure to increase multicomputer efficiency.
- Fault tolerance of the multicomputer has been established.

The fringe counting algorithm developed has been shown to successfully convert the output intensity of an EFPI into a strain level. The accuracy of this algorithm is reduced if the sampling rate used is lower than 20 Hz.

The increase in time to complete the given series in tasks after data link failure has occurred verifies that parallel processing has been accomplished to increase efficiency of the multicomputer and verifies that the multicomputer continues to function correctly after reorganization has been necessary. It was shown that no data loss occurred at the time of data link failure.

It has been demonstrated that optical sensors can be successfully used to perform fault detection and aid in rerouting communications data away from failing network elements. Given the output of an EFPI, the multicomputer assessed the strain placed on a data link. When this strain level exceeded a preset tolerance level, unstable data links were isolated, and redundant systems were used to take over tasks before actual failure occurred, preventing loss of critical data. Although the efficiency of the multicomputer was be reduced due to the reduction of available resources, no loss of functionality occurred.

6.0 Future Considerations

There is still a great deal of work to be done in the field of hybrid sensing and communications. An optical sensor that would not affect the communications signal would be a great improvement over all sensors that have been tested with the MIL-STD-1773 testbed at Virginia Tech. A Bragg-grating sensor would be an ideal candidate for such requirements, as it would permit the undisturbed passage of communications signals. One drawback to using Bragg-gratings would be that it requires the capability to sense wavelength shifts.

Another possibility for experimentation would be in the area of multiplexing sensors. For example, a Mach-Zehnder interferometer, which operates transmissively, could be used in conjunction with an EFPI, which operates reflectively. Minimal crosstalk would occur if the EFPI were placed between the Mach-Zehnder interferometer and the source.

A more detailed analysis of strains placed on communications lines would be desirable. Temperature sensing could be easily accomplished with the current setup. The high sensitivity of interferometric sensors would make an analysis of vibration present near communications lines very feasible. With the availability of computing resources, active dampening of such vibrations would make an interesting study.

7.0 References

1. G. Keiser, *Optical Fiber Communications*, McGraw Hill, Inc., New York, 1988.

2. D. Gloge, "Weakly Guiding Fibers," *Applied Optics*, vol. 10, no. 10, pp. 2252-2258, Oct. 1971.

3. D. Marcuse, *Theory of Dielectric Optical Waveguides*, Academic Press, Inc., San Diego, CA 1991.

4. Class Notes, EE5154, Professor Safaai-Jazi, Spring 1993.

5. A. Flax, "Micro-Optical Fiber Devices used with Modal Domain Sensing", Masters Thesis, Virginia Polytechnic Institute and State University, 1988

6. A. W. Snyder and J. D. Love, *Optical Waveguide Theory*, Chapman and Hall, New York, 1983.

7. Product Information for SMF-28 Single Mode Fiber, Corning Incorporated.

8. K. A. Murphy, M. F. Gunther, A. M. Vengsarkar, and R. O. Claus, "Quadrature Phase-Shifted, Extrinsic Fabry-Perot Optical Fiber Sensors," *Optics Letters*, vol. 16 no. 4, Feb. 15, 1991.

9. Class Notes, EE5515, Professor Davis, Fall 1992.

10. M. Lehman, "A Survey of Problems and Preliminary Results Concerning Parallel Processing and Parallel Processors," *Proc. IEEE*, vol. 54 no. 12, pp. 1889-1901, Dec. 1966.

11. C. G. Bell and A. Newell, *Computer Structures: Readings and Examples*, McGraw-Hill Book Company, New York, 1971.

12. P. Dreyfus, "System Design of the Gamma 60," *Proc. WJCC*, pp. 130-133, May 1958.

13. M. J. Flynn, "Very High-Speed Computing," *Proc. IEEE*, vol. 54, no. 12, pp. 1901-1909, Dec. 1966.

14. H. S. Stone, *High-Performance Computer Architecture*, Addison-Wesley Publishing Company, Reading, Mass., 1992.

15. S. F. Midkiff and J. T. McHenry, "Multicomputer Networks for Smart Structures," *Proc. APDA/AIAA/ASME/SPIE Conf. on Active Materials and Adaptive Structures*, pp. 239-242, Nov. 1991.

16. X. Zhang, "System Effects of Interprocessor Communication Latency in Multicomputers," *IEEE Micro*, pp. 12-55, 1991.

17. T. Anderson and P. A. Lee, *Fault Tolerant Principles and Practice*, Prentice Hall International, Englewood Cliffs, NJ 1981.

18. V. F. Nicola and A. Goyal, "Limits of Parallelism in Fault-Tolerant Multiprocessors," *Dependable Computing and Fault-Tolerant Systems*, vol. 6, Springer-Verlag/Wien, New York, 1992.

19. D. A. Reed and R. M. Fujimoto, *Multicomputer Networks: Message Based Parallel Processing*, MIT Press, Cambridge, Mass., 1987.

20. W. D. Hillis, *The Connection Machine*, MIT Press, Cambridge, Mass., 1986.

21. D. D'Avino and C. deLong, *MIL-STD-1553 Protocol Reorganized*, Society of Automotive Engineers, Inc., Warrendale, PA, 1991.

22. MIL-STD-1553 Designer's Guide, ILC Data Device Corp., 1988.

23. Dept. of Defense Standards, MIL-STD-1553B Notice 2.

24. J. T. McHenry, "Multicomputer Networks for Smart Structures," Ph.D. Dissertation, Virginia Polytechnic Institute and State University, 1992.

25. J. T. McHenry and S. F. Midkiff, "Hybrid Sensing and Communication Fiber Optic Multicomputer Networks for Smart Structure Applications," *Smart Material Structure*, vol. 1, pp. 146-155, 1992.

Vita

David Forbis was born on April 14, 1970 in Urbana, Illinois. He attended Avon Grove High School in Pennsylvania before earning his BS in electrical engineering at Virginia Tech in Blacksburg, Virginia. He joined the FEORC group in 1992, and worked toward his MS, endeavoring to produce these pages. His parents currently reside in Carrboro, North Carolina. His long term goals include exploring the Great Pyramid of Giza in Egypt, becoming a season ticket holder of the Philadelphia Flyers, and seeing the Grateful Dead on New Year's Eve in 1999.