

DESIGN OF FAULT-TOLERANT COMPUTER SYSTEMS USING GRADED COMPONENTS

S. SANYAL
COMPUTER SYSTEMS & COMMUNICATIONS GROUP
TATA INSTITUTE OF FUNDAMENTAL RESEARCH
BOMBAY - 400005 , INDIA

1. INTRODUCTION : Fault tolerant computer design essentially means providing correct execution of program inspite of fault-presence. Utilization of very high quality, heavily screened components, modules etc. provide an answer to this. By reducing the probability of failure, higher availability of the computer systems is ensured. On the other hand, using redundancy in terms of coding, duplication etc. it is also possible to achieve the same result with average quality components. The third approach is translating the problem of error detection and subsequent correction in the domain of time. Repeated diagnostics are run on the system to check 'correct input producing known correct output'. In case of non-conformity, error signals are generated which triggers manual intervention. At this point error locating tests help finding faults at a reasonable resolution level and replacement of faulty modules again ensures correct operation.

A slightly different approach is explored here to optimize certain parameters. We call this 'a graded approach'. Here a combination of all three approaches were made to achieve a design which is essentially quite simple but still provides sufficient fault tolerance. In case of inability to mask the fault, indications are provided about fault-presence.

Basically the graded design approach advocates different techniques at different level. Redundancy and coding is applied to a modest degree. Explanation of this phenomenon is as follows - if the modern totally self-checking checkers are studied, it can be seen that though algorithmic correctness is achieved, overall circuit complexity increases. Here the submission is that with increased quantity of hardware, probability of failure becomes non-trivial. Hence an optimum balance of all parameters is the desired goal. It will be seen with the help of a design example that a step by step evaluation and selection of proper redundancy, either in hardware or in time can be chosen.

Judicious choice of certain very highly screened components at critical areas will increase the overall reliability. On top of this, diagnostics and finer resolution microdiagnostics are run at periodic interval to improve the overall confidence in the performance. This total approach is believed to be of higher interest from the practical design viewpoint aswell.

2. BASIC SYSTEM IMPLEMENTATION:

The basic system configuration is shown in Figure 1. Processing units were designed using bit slice microprocessors (4 of them to have a $4 \times 4 = 16$ bit processor). One point of importance is that though the design example is shown with a 16 bit processor, the basic design philosophy holds for any similar design.

Data from the outside world (memory, peripheral devices) are brought into the processor through the DataIn Bus and brought out from the processor through the DataOut Bus. There is a ROM sitting on the DataIn Bus from which some of the vector addresses and constants (required for arithmetic operations, interrupt servicing) are obtained. DIn Register and DOut Register are the buffer registers between the internal Data In/ Data Out Bus and the external Data Bus.

The control section is microprogrammed. All the control information is stored in the ROM in the form of bit patterns. Like any memory, this storage can be addressed (called microaddress), output of this unit generates control information to the whole computer. The machine code consists of opcodes which when decoded (by a ROM called IR Decode ROM) generates starting microaddress of a pertinent microroutine. Subsequent microaddresses are generated from the control memory next address field . Another additional facility exists to alter the microaddress according to microbranch inputs (which could be condition code outputs etc.). When a conditional microbranching is desired, the least significant three bits of the next address field is kept zero and they are 'OR'ed with the conditional branch inputs. So, depending upon the condition inputs being '0' or '1', next microaddress will be modified accordingly.

3. SYSTEM IMPROVEMENT FOR INCREASED RELIABILITY:

As mentioned earlier, the basic emphasis is to increase overall system reliability without going to any overall extreme design measure. For convenience, we shall typify two grades of components : Type A and Type B. Type B conforms with standard, commonly available integrated circuits, components etc. whereas Type A components are screened for higher temperature, higher humidity and other parameters. These are used for high reliability design requirement.

We shall describe the design strategy adopted for individual modules (refer to Figure 1).

3.1 PROCESSING UNITS : The basic processing units are designed around Bit Slice Microprocessors. Associated multiplexers etc. are not shown in the diagram. Initially it was thought that Arithmetic Codes (An + B Type) are to be used for producing redundancy to get better reliability. On a reassessment it was finally decided to steal time from main processing job to run microdiagnostic routines in an aperiodic way. Basically our approach is slightly different in the sense that the running of the microdiagnostic routines is initiated by the operating systems whenever it is 'idling'. This process is run at a low priority so as not to disturb any real time processing needs.

Since the microdiagnostic routines run much faster than a machine code level diagnostics, less time is required to ensure average reliability. System and processor status is preserved and restored at the beginning and end of this operation. Any error detected from these 'correct code input should generate correct code output' tests interrupts the normal run. This in turn initiates running of fault locating tests through operator intervention.

3.2 CONSTANT ROM : The ROM sitting on the DataIn Bus provides vectors, constants etc. This is checked with the help of parity bits. Size of this ROM is 32 word x 16 bits. Normally two 32 word x 8 bit ROM would have been sufficient. In this design another 32 word x 8 bit ROM is used. Out of this 24 bits, 5 bits of each word store the address of that word itself. For example, when location 5 is addressed, the corresponding address bits are compared with the stored 5 bits of address. Any non-conformity indicates hardware failure. Two bits are used as parity bits (one for 10 bits and another for 11 bits, one bit remains unused). Parity is checked through hardware and error, if any, is indicated.

One interesting aspect is that the checking circuitry (parity checkers, associated SSI circuits) are designed using Type A components. This ensures higher reliability by reducing the probability of failure of the checkers. By selectively choosing only some critical components to be of higher reliability, overall cost factor is kept optimum.

3.3 DATA IN AND DATA OUT REGISTERS, ADDRESS REGISTER (AR), DATA IN AND DATA OUT BUS, DATA AND ADDRESS BUS :

The diagnostic routine which checks the Processor also ensures proper working of these registers and the buses. To ensure additional reliability another basic factor was considered. These are higher reliability card edge connectors and bus connectors. Earlier while designing standard commercially usable computer systems it was observed that this area was one of the main source of unreliability. Additionally, some sockets are used for the LSI chips (bit slice processors, PROMs). These are also chosen to be of Type A quality.

3.4 DIAGNOSTIC DEVICE ON THE BUS - All the tests and measures described so far do not help checking the control lines on the bus although a minimal checking through proper response of the memory is assured.

To handle this, a special device is designed. This device is memory mapped. On being addressed from the processor this microprogrammed device goes through a dummy set of sequences to ensure proper working of the bus. All control signals are checked. The components used here are of Type A.

3.5 INSTRUCTION REGISTER (IR) DECODE ROM AND MAIN CONTROL ROM : These are provided with address and data parity in the same way as described in section 3.2 for the constant ROM .

3.6 INSTRUCTION REGISTER (IR), MICRO BRANCH MULTIPLEXER, NEXT ADDRESS MULTIPLEXER , MICRO DATA BUFFER : Type A components are used for these modules to ensure high reliability.

3.7 TIME DOMAIN TESTING (MICRODIAGNOSTICS) : Whenever the operating system is 'idle', microdiagnostic routines are initiated to check different parts of the system, as described. To satisfy easy switching over from the normal operating mode to the microdiagnostic mode, all status are saved and retrieved. Advantage of this level of diagnostic are twofold:

(a) lot many fields of the microdata register can be active in parallel in this horizontally microprogrammed machine, hence speed of operation is high.

(b) resolution of fault location is also very high as we can isolate faults at a very fine level.

4. CONCLUSION : A new approach of mixing different type components of different reliability for the design of an optimally reliable computer system at a moderate cost was undertaken and fulfilled. For a medium range reliability requirement this type of design methodology will definitely make its mark.

REFERENCES :

1. Design of a High Reliability Self Diagnosing Computer using Bit Slice Microprocessors : S.Sanyal and P.V.S.Rao, accepted for publication in the Journal of Microprocessors and Microprogramming, North Holland.
2. An Integrated Approach to Automated Computer Maintenance : F.J.Hackl and R.W.Shirk, IEEE Conference Record on Switching Circuit Theory and Logical Design, 1965, pp 289-302.
3. Design of a Self-Checking Microprogram Control : R.W.Cook, W.H.Sisson, T.F.Storey and W.N.Toy; IEEE Transaction on Computers, Vol C-22, No 3, March 1973, pp 255-262.
4. Fault Tolerance : G.Bona and I.Erenyi , Preprint (KFKI-1986 - 75/M) from Central Research Institute for Physics, Hungary.

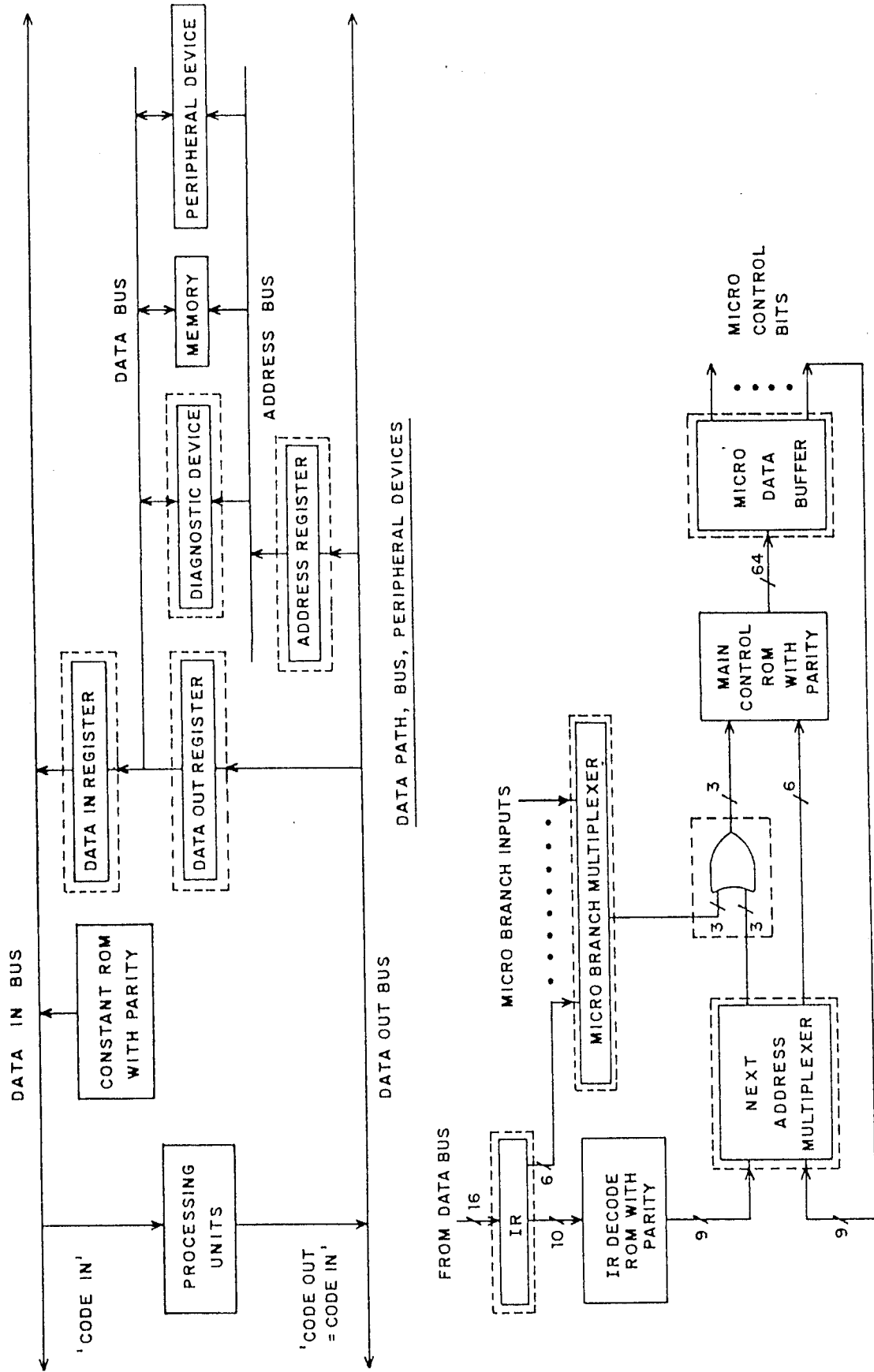


Fig. 1. SCHEMATIC DIAGRAM OF THE COMPUTER
ALL MODULES INSIDE DOTTED BOX ARE DESIGNED WITH TYPE A COMPONENTS.

CONTROL SECTION