THE STAR (SELF-TESTING-AND-REPAIRING) COMPUTER: AN INVESTIGATION OF THE THEORY AND PRACTICE OF FAULT-TOLERANT COMPUTER DESIGN*

Algirdas Avižienis,** George C. Gilley, Francis P. Mathur, David A. Rennels, John A. Rohr, David K. Rubin

Jet Propulsion Laboratory, California Institute of Technology
Pasadena, California, USA

1. Introduction: Chronology and Rationale

This paper presents an overview of the theoretical results and design experience obtained in a continuing investigation of fault-tolerant computing which is being conducted at the Jet Propulsion Laboratory. Initial studies (1961-65) [1] led to the conclusion that dynamic [2] (also called "standby") redundancy offered the greatest promise in the design of fault-tolerant digital computer systems. The dynamic redundancy approach requires a two-step procedure for the elimination of a fault: first, the presence of a fault is determined; second a corrective action is taken (e.g., replacement of failed unit, repetition of program, reconfiguration of systems, etc.). The alternative to the dynamic approach is static [2] ("masking") redundancy, which was already being utilized in existing component-redundant [3,4] and triple-modular redundant (TMR) [4,5,6] computers. The static method depends on the permanently connected structure of the computer to mask the occurrence of faults and is based on the assumption that faults are statistically independent events affecting single components or logic elements [6].

Early analytic studies of dynamic redundancy with idealized series-parallel system models [7,8,9,10] indicated that mean life gains of an order of magnitude and more over a non-redundant system could be expected from dynamically redundant systems with standby spares replacing failed units. This gain compared favorably with the mean life gain of less than 2 in the typical TMR systems. Other identifiable qualitative advantages of the dynamic over the static redundancy were [1,11]:

(a) greater ability to handle catastrophic (non-independent) faults which is especially important for densely packed microelectronic circuitry;

(b) survival of system until all spares of one type are exhausted;

(c) ability to eliminate errors which are caused by transient faults by the use of program rollback;

(d) ready adjustability of the number and type of spare units;

(e) utilization of the potentially lower failure rate of unpowered components in spare units;

(f) avoidance of the circuit-related problems of static redundancy: increases in fan-out, fan-in, power requirements, and the need for isolation and synchronization of separate channels;

(g) facilitation of the checkout of spare units - use of standard diagnostic programs.

The attainment of the apparent advantages of a dynamically redundant system had been shown to depend very strongly on the successful execution of the detection and replacement operations [9,10]; these observations have been later formalized as the concept of coverage [12].

The second phase of the investigation (1965-1970) was focused on the identification and solution of the problems involved in the design of a general-purpose digital computer possessing the properties attributed to the abstract model of a dynamically redundant computing system. Three major areas of investigation were: (a) an investigation of fault-detection methods; (b) a study of computer architecture with emphasis on partitioning into subsystems with minimal interconnection requirements; (c) a study of the "hard-core" problem - the alternate technologies and logic organizations for implementing the detection and switching functions.

The choices among feasible alternatives in all three areas are strongly affected by assumptions on the available component technology and on the computing tasks to be required of the computer. In order to retain contact with the practice of computer design, it was decided to design and construct an experimental general purpose digital computer which would incorporate dynamic redundancy, i.e., fault detection and replacement of failed subsystems, as integral parts of its structure. The design objectives have been carried out and the system, called the STAR (Self-Testing-And-Repairing) computer, has been operating since March 1969. The modular nature of the STAR computer has allowed a systematic expansion and modifications which are still being continued.

The first objective of the STAR computer design was to study the class of problems which are encountered in transforming the theoretical model of a self-repairing system into a working computer. State-of-the-art integrated circuit and memory technology was employed in the design. The choice of computing tasks was determined by the needs of the supporting agency - NASA and the Jet Propulsion Laboratory. The STAR computer characteristics were chosen to satisfy all predictable requirements of a spacecraft guidance, control, and data acquisition computer which would be used in the very long (10-years and more) unmanned missions exploring the outer planets of the solar system [13].

The second objective of the STAR computer was to provide a tool for laboratory studies of fault-tolerant computing, including the injection of transient as well as permanent faults of catastrophic nature. Very extensive displays of registers, manually controlled clocking, and provisions for convenient modification

*This paper presents the results of one phase of research carried out at the Jet Propulsion Laboratory, California Institute of Technology, under Contract No. NAS 7-100, sponsored by the National Aeronautics and Space Administration.

**Also, Associate Professor, Computer Science Department, School of Engineering and Applied Science, University of California, Los Angeles, California.

Reprinted from *FTCS-1*, 1971, pp. 92-96, © IEEE
0-8186-7150-5/96 $5.00 © 1996 IEEE
Proceedings of FTCS-25, Volume III

of subsystems were incorporated into the experimental STAR computer breadboard. An automatic fault-injection and response-observation system is being designed to supersede the currently used manual methods.

During the studies of fault-tolerant architecture and the design of the STAR computer, concurrent investigations were being conducted in other closely related areas of fault-tolerant computing, including studies of software, reliability prediction, and extension of dynamic redundancy to peripheral devices [14]. At the same time, a complete redesign of the STAR computer was performed to match the exact requirements of a control computer for the "TOPS" Thermoelectric Outer Planet Spacecraft [18]. This effort led to the evaluation of additional fault-recovery techniques. The results of the efforts described above are summarized in the following sections of this paper.

2. STAR Computer Architecture

The STAR computer is a replacement system which provides to the user one standard configuration of functional subsystems with the required computing capacity. The standard computer is supplemented with one or more spares of each subsystem. The spares are held in an unpowered state and are used to replace operating units when a permanent fault is discovered. The principal features of the STAR system which are used to implement error diagnosis and recovery are listed below:

(a) All machine words are encoded in error-detecting codes to provide concurrent fault diagnosis.

(b) The computer is subdivided into a number of replaceable functional units. This decentralization of the system allows simple fault location procedures and simplifies system interfaces.

(c) Fault detection, recovery, and replacement are carried out primarily by special purpose hardware. In the case of memory damage, software is used to augment the recovery hardware.

(d) Transient faults are identified and their effects are corrected by the repetition of a segment of the current program; permanent faults are eliminated by the replacement of faulty functional units.

(e) The replacement is implemented by power switching. The information lines of all units are permanently connected to the busses through isolating circuits; unpowered units produce only logic "zero" outputs.

(f) The error-detecting codes are supplemented by monitoring circuits which serve to verify the proper internal operation of the functional units.

(g) The "hard core" test-and-repair processor (TARP) is held to a small size and protected by complete replication and replacement by spares.

The functional units of the STAR computer communicate through two 4-wire information busses by exchanging error-coded 32-bit data and instruction words in the form of eight 4-bit bytes. Three powered test and repair processors (TARP) check all bus information for proper coding and receive status signals from each unit in order to validate the functioning of all units in real time. If a unit is found to have a permanent fault, it is replaced. Similarly, if one TARP is found defective by the other two (by monitoring disagreement indications), it is replaced by a standby spare from a set of spare TARPs.

The use of modified modulo 15 residue codes [21] gives an error coverage of over 95% at an increase of less than 20% in functional unit complexity. In order to further increase error coverage, three general techniques are employed. They take advantage of the byte-serial nature of the STAR machine to multiplex functional unit status information to the TARPs through two status monitor lines from each active unit. Status information is generated by the following means:

(a) Direct monitoring of critical control signals within the units.

(b) Error monitors within a unit, which are periodically checked with special test op-codes.

(c) Each unit which normally operates singly can be run in duplex for critical computations, giving essentially 100% coverage.

The TARPs do not need to contain permanent non-volatile storage. Critical information, such as the system memory assignment, power, and "rollback" status as well as the "hard-core" executive program is always kept in duplicated read-write memories which are tagged in such a way that in case of a catastrophic transient fault (e.g., transient power failure) the TARPs can locate this information and bring the system back into operation. The TARPs thus are reduced to simple finite state automata which (when in agreement) replace defective units and which, when they disagree, start at a common reset point, gather the information required, and bring the system into operation.

3. STAR Computer Software System

The STAR Computer Software System is partitioned into two separate subsystems. The first subsystem, the programming subsystem, consists of an assembler, a loader, a functional simulator, and a simple executive program to coordinate the operation of the other three components. The second subsystem, the operating subsystem, consists of the resident executive program and the applications programs [14,17].

SCAP, the STAR Computer Assembly Program, is a traditional two-pass assembler incorporating machine instructions, pseudo-operations, macro facilities, and a unique COMPILE pseudo-operation which implements automatic compilation of arithmetic assignments within SCAP. The first pass of SCAP builds symbol and literal tables, expands macro and COMPILE instructions, and determines the length of the program. Between passes the header cards containing loading information are generated and the tables are transformed into a form appropriate for the second pass. The second pass of SCAP calculates addresses, constructs instructions, and converts data into internal (STAR computer) form. Finally, after the second pass various tables generated by SCAP are printed to aid in debugging and documenting the program.

LOAD, the STAR Computer LOADer, combines relocatable decks produced by SCAP into an absolute load module. Various tables generated by LOAD are printed. The output from LOAD can be input to either the STAR computer simulator or the STAR computer itself.

STAR, the STAR Computer Simulator, is a functional simulator. The simulator is used to debug programs before hardware is completed and to resolve conflicts during hardware debugging. The functional simulator is modular in nature. Each processor of the STAR computer is simulated by a module which contains the simulated registers and algorithms required to duplicate the functions of the processor. The simulator

includes trace and dump features which are useful for debugging and documenting programs.

SYS, the executive SYStem program, processes control cards and coordinates the use of SCAP, LOAD, and STAR. In addition, SYS provides the interface with the EXEC 8 system of the UNIVAC 1108 on which the programming subsystems run. Subroutines are included for printing, punching cards, reading cards, and other common functions.

The resident executive augments the diagnosis and repair features of the STAR computer hardware and controls the programs which run on the computer. While most of the diagnosis and repair capability of the STAR computer is contained in the hardware, certain functions require software assistance. Rollback or recovery points must be included in all programs run on the STAR computer. In addition, the resident executive must have a "cold start" capability in case of a catastrophic failure requiring a complete hardware reset. Memory replacement operations also require software assistance to load a newly-activated memory module. During normal operation the resident executive system must handle all interrupts and give control to the proper program at the proper time. Finally, during idle periods diagnostic programs can be run to check powered processors and examine failed processors.

4. Prediction of Reliability

The reference computer used to evaluate the relative reliability of the STAR was the nonredundant Mariner Mars 1969 (MM-69) computer. The MM-69 computer was chosen since it is a recent aerospace computer which has been tried and tested and whose component failure rates have been readily available.

Knowing the reliability of the MM-69 computer and by postulating comparative complexities in equipment, an estimate of the STAR computer reliability was developed. Under various simplifying assumptions the upper and lower bound on the reliability of the STAR computer were obtained relative to the reliability of the MM-69 computer. The reliability models of (1) the MM-69 computer, (2) a simplex computer having the same computational capability as the STAR computer, and (3) the STAR computer are shown in Fig. 1. The MM-69 computer (Fig. 1a) is assigned a complexity of unity. It is then assumed that the simplex computer (Fig. 1b) consisting of eight uniform processors is 8 x CF times as complex as the MM-69 computer, where CF is the relative complexity factor and is defined as the ratio of complexity (aggregate component count) of a single STAR processor to the complexity of the MM-69. For CF equal to 1/4 the simplex computer then is twice as complex as the MM-69 computer. Knowing this relative complexity the reliability of the simplex computer was then readily evaluated. The basic assumption being made that the simplex computer would be constructed from the same components, the same technology and procedures and the same packaging techniques as those used in the design and fabrication of the MM-69 computer. This basis of the equivalence in fabrication technologies is also carried to the STAR computer model shown in Fig. 1c.

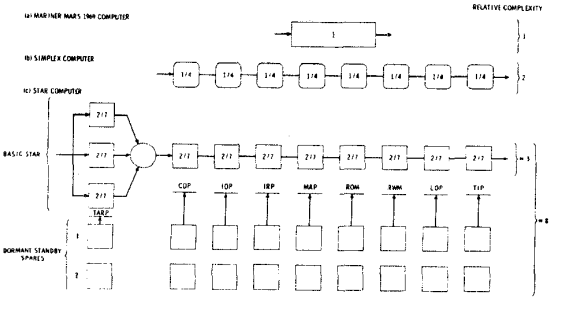

Fig. 1. Computer Reliability Models

The STAR model consists of eight processor types plus the Test and Repair Processor (TARP) in series reliability. All processors are allocated an equal number of spares. The reliability model applied to all processors except the TARP is the standby-replacement redundancy model with dormant spares [15]. The TARP was modeled as a hybrid-redundant (3,S) system [16]. The fault coverage factor in the STAR model is taken into account in two ways: (1) by considering the fault detector and restoration initiator as a separate processor (the TARP); and (2) by applying a Self-Testing Factor (STF) to the relative complexities of the processors. The simplex computer shown in Fig. 1b does not contain a processor corresponding to the TARP in the STAR computer since the simplex computer is a computationaly equivalent non-redundant machine and hence does not require 'test and repair' capabilities. Since 4 bits of the 32 bit STAR word are dedicated to error detection with the consequence that all full length hardware registers also have 4 bit positions for the check byte, a STF equal to 8/7 was chosen. This factor then takes into account the overhead due to the self-testing and repairing features within each of the STAR computer processors. A STF equal to 8/7 then states that each processor of the STAR computer is 8/7 times more complex than the processor of the simplex computer. For a CF of 1/4 the ratio 2/7 for the STAR processors results by applying the factor 8/7 to each of the simplex processors. Thus the overall complexity of the STAR relative to MM-69 is arrived at.

Knowing the mean-life of the MM-69, the reliability of the STAR computer was then evaluated by using the appropriate reliability equations [15]. By considering inverse dormancy factors K of unity and infinity, the lower and upper bound respectively of the STAR computer was arrived at with respect to dormancy of the spares. For the case of two spares and a complexity factor of one-third, the survival probability estimates are as shown in Fig. 2.

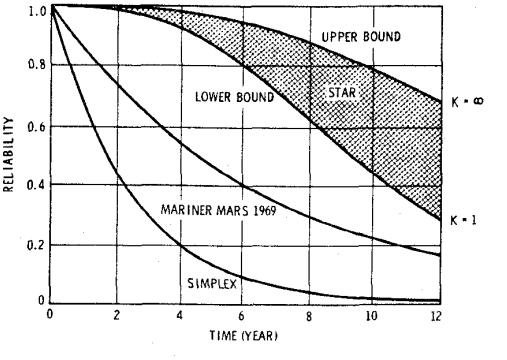

Fig. 2. Upper and Lower Bounds of the STAR Computer Reliability with 2 Spares and CF = 1/3

5. Extension of STAR Techniques to Peripheral Subsystems

The advantages of a Replacement System together with the capability of the STAR computer to maintain itself automatically led to the conclusion that a STAR-like computer could be used to effect the automatic maintenance of a larger, more complex system. A heuristic investigation was made of the implementation of the fully automatic maintenance of a simplified model of the JPL Thermoelectric Outer-Planet Spacecraft (TOPS) which is being proposed for the exploration of the outer planets [17,18].

The practical implementation of automatic check-out and continuous in-flight maintenance of the entire spacecraft imposes additional requirements on the spacecraft systems designers. From the beginning, the design of the systems must be influenced by the concept of automatic maintenance, including the interface with the computer. The design of each system must allow for algorithmically defined evaluation procedures with which all failures can be detected. Furthermore, each system must be organized into efficiently-sized replaceable units. The evaluation procedures must be capable of isolating detected failures to one of these units and then effecting the replacement of the failed unit.

The results of the investigation identified and quantized the capability required by the STAR-like computer (in terms of speed, storage, and instruction set) to effect the fully automatic maintenance of the "TOPS" spacecraft. Furthermore, the investigation showed that:

(a) The fully automatic maintenance of a complex, long-life spacecraft is feasible.

(b) All of the computer support requirements of the spacecraft systems can be defined well enough algorithmically to produce computer programs to satisfy these requirements.

The results of the investigation have systematically extended dynamic redundancy to various peripheral subsystems of a data processing system. In addition to the specific example of a spacecraft, the same principles are applicable to computer-controlled automatic maintenance of other complex data processing, communication, and control systems.

6. Design of the TOPS Control Computer

Because the STAR computer is a general-purpose machine, research-oriented, adaptable to unforeseen problems and varied applications, its capabilities are purposely as unconstrained as possible. The TOPS Control Computer Subsystem (CCS) could not have been a "shrunken" version of the STAR computer -- the TOPS weight and power constraints would not have been satisfied that way. Instead, the STAR design was simplified by preserving only those capabilities dictated by the TOPS functional requirements [18]. Aided by the study described in the previous section [17], the CCS design effort yielded a modest special-purpose digital computer with limited, none the less powerful, arithmetic and logic capabilities.

In spite of a transformation representing extensive simplification, none of the self-test and repair ability of the larger machine has been sacrificed. On the contrary, the CCS has expanded failure detection and recovery proficiency. A variety of advances have arisen from experience with the STAR computer. Too late to be reflected in its design, these have been incorporated into the CCS.

There are many similarities between the two machines. For instance, the CCS uses the same residue code employed in the STAR computer; it is composed of a subset of the STAR processor units; for the most part, its instruction set is a subset of the STAR's; it has the same word length and byte size. There are also numerous differences. Unlike the STAR computer, which uses a two-out-of-four code on its instruction operation codes, the CCS also utilizes a residue code on its op-codes. Inputs, outputs, interrupts and program rollbacks are handled in a manner more suitable to the TOPS application.

7. Present Research Activities

The STAR computer employs a balanced mixture of coding, monitoring, standby redundancy, replication with voting, component redundancy, and repetition to attain hardware-controlled self-repair and protection against transient faults. The principal objective of the design is to attain fault-tolerance for the widest possible variety of faults: transient, permanent, random, and catastrophic. The actual construction (rather than simulation) of the STAR breadboard has two significant advantages. First, the design process has uncovered interesting new problems and led to numerous improvements. Second, the computer serves as a vehicle for further experimentation and refinement of the recovery techniques.

Design of several improved "second generation" functional units is under way. They include a new Arithmetic Processor, a Control Processor for medium-scale integrated circuit implementation, and a "shared" Read-Write Memory module for the storage of automatic maintenance information from the spacecraft telemetry system. Analysis of automatic maintenance algorithms and design of a Command/Data bus for their implementation are under intensive study. The investigations of the STAR computer have stimulated new research efforts in fault tolerance. Current advanced investigations are concerned with the following areas:

(a) Hardware-software interaction in a fault-tolerant system with recovery, such as the interaction of the TARP and the operating system, the automatic verification and insertion of "rollback" points in all programs, and auxiliary diagnostic programs.

(b) Studies of advanced recovery techniques, i.e., post-catastrophic restart, TARP replacement schemes, recovery from massive interference, partial utilization of failed units.

(c) Advanced component technology, especially methods to attain bus and power switch (i.e., "hard core") immunity to faults.

(d) Formulation of a theory of fault-tolerance by interpretation of extensive experiments with the STAR breadboard as the instrument.

(e) The design of a "Super-STAR" computer with universal processor and storage modules, and their implementation by large-scale integration.

(f) Computational utilization of the spare units for supplemental tasks.

At the present time it is evident that the STAR computer design and construction effort has led to valuable new insights into the problem of fault-tolerant computing; further results in this field are expected from the research program in the future.

References

[1] Avižienis, A., "Design of Fault-Tolerant Computers," AFIPS Conference Proceedings, Vol. 31, (Fall Joint Computer Conference 1967), Thompson Books, Washington, D.C.; 733-743.

[2] Short, R. A., "The Attainment of Reliable Digital Systems Through the Use of Redundancy - a Survey," IEEE Computer Group News, Vol. 2, No. 2, (March 1968); 2-17.

[3] Lewis, T. B., "Primary Processor and Data Storage Equipment for the Orbiting Astronomical Observatory," IEEE Trans. on Electronic Computers, Vol. EC-12, No. 6, (December 1963), 677-686.

[4] Kuehn, R. E., "Computer Redundancy: Design, Performance, and Future," IEEE Trans. on Reliability, Vol. R-18, No. 1, (February 1969), 3-11.

[5] Anderson, J. E., and Macri, F. J., "Multiple Redundancy Applications in a Computer," Proc. 1967 Annual Symposium on Reliability, Washington, D.C., (January 1967), 553-562.

[6] Lyons, R. E., and Vanderkulk, W., "The Use of Triple-Modular Redundancy to Improve Computer Reliability," IBM Journal Res. Dev., Vol. 6, No. 2, (April 1962); 200-209.

[7] Reed, I. S., and Brimley, D. E., "On Increasing the Operating Life of Unattended Machines," RAND Corp. Memorandum RM-3338-PR, November 1962.

[8] Kruus, J., "Upper Bounds for the Mean Life of Self-Repairing Systems," Coordinated Science Laboratory Report R-172, University of Illinois, Urbana, Illinois (July 1963), AD-418 174.

[9] Flehinger, B. J., "Reliability Improvement Through Redundancy at Various System Levels," IBM Journal Res. Dev., Vol. 2, No. 2, (April 1958), 148-158.

[10] Griesmer, J. E., Miller, R. E., and Roth, J. P., "The Design of Digital Circuits to Eliminate Catastrophic Failures," Redundancy Techniques for Computing Systems, Spartan Press, Inc., Washington, D.C., 1962; 328-348.

[11] Avižienis, A., "An Experimental Self-Repairing Computer," Information Processing 68, Proceedings of IFIP Congress 1968, Edinburgh, Scotland, Vol. 2, 872-877.

[12] Bouricius, W. G., Carter, W. C., Schneider, P. R., "Reliability Modeling Techniques for Self-Repairing Computer Systems," Proc. of 24th National Conference of ACM, (Association for Computing Machinery, 1969), 295-309.

[13] Long, J. E., "To the Outer Planets," Astronautics & Aeronautics, Vol. 7, No. 6 (June 1969), 32-47.

[14] Avižienis, A. A., Mathur, F. P., Rennels, D., and Rohr, J., "Automatic Maintenance of Aerospace Computers and Spacecraft Information and Control Systems," Proc. of the AIAA Aerospace Computer Systems Conference, Paper 69-966, Los Angeles, (September 8-10, 1969), 1-11.

[15] Mathur, F. P., "Reliability Modeling and Architecture of Ultra-Reliable Fault-Tolerant Digital Computers," Ph.D. Thesis, University of California, Los Angeles, Computer Science Dept.; June 1970.

[16] Mathur, F. P., and Avižienis, A., "Reliability Analysis and Architecture of a Hybrid-Redundant Digital System: Generalized Triple Modular Redundancy With Self-Repair," AFIPS Conference Proceedings, Vol. 36, (Spring Joint Computer Conference 1970), AFIPS Press, Montvale, N. J., 375-383.

[17] Gilley, G. C., "Automatic Maintenance of Spacecraft Systems for Long-Life, Deep-Space Missions," Ph.D. Thesis, University of California, Los Angeles, Computer Science Dept.; September 1970.

[18] "TOPS Outer Planet Spacecraft," (Special Issue), Astronautics and Aeronautics, Vol. 8, No. 9, (September 1970).

[19] Avižienis, A., "A Study of the Effectiveness of Fault-Detecting Codes for Binary Arithmetic," Technical Report No. 32-711, Jet Propulsion Laboratory, Pasadena, California, (1965).

[20] Avižienis, A., "Concurrent Diagnosis of Arithmetic Processors," Digest of the 1st Annual IEEE Computer Conference, (Chicago, Ill., 1967), 34-37.

[21] Avižienis, A., "Arithmetic Error Codes: Cost and Effectiveness Studies for Application in Digital System Design," this Digest, 118-121.

Acknowledgments

The research and development of the STAR computer has been performed in the Spacecraft Computers Section of the JPL Astrionics Division, and recognition is due to most of the Section's members for support in their respective specialties. The STAR concept has been devised by A. Avižienis, who has directed the overall research effort. The hardware design is directed by D. A. Rennels, the software effort -- by J. A. Rohr, reliability estimation -- by F. P. Mathur, and the implementation of peripheral automatic maintenance -- by G. C. Gilley. Technical contributions to the design have been made by P. H. Sobel and A. D. Weeks, and consultation has been contributed by R. K. Caplette, E. Greenberg, G. R. Hansen, E. H. Imlay, G. R. Kunstmann, J. Nievergelt, J. J. Wedel, and L. J. Zottarelli. The STAR effort has been administered by J. R. Scull (Astrionics Division Manager), W. F. Scott (Spacecraft Computers Section Manager) and J. J. Wedel (Research Group Supervisor). The power switch has been developed by the Stanford Research Institute, Menlo Park, California, and a fault-tolerant Read-Only Memory has been designed by the M.I.T. Instrumentation Laboratory, Cambridge, Mass. under subcontracts from JPL. Construction of the computer was performed by J. Buchok, J. L. Cline, N. B. Funsten, J. C. Schooler and B. Stall. The design of the TOPS Control Computer is due to D. K. Rubin, with technical contributions by N. Deo, G. C. Milligan, and M. B. Vineberg.

A special acknowledgment is due to R. V. Powell, Manager for Electronics of the JPL Research and Advanced Development Program Office, and Messrs. F. J. Sullivan (Director, Electronics and Control), J. I. Kanter, G. A. Vacca, J. L. East, and T. S. Michaels of the NASA Office of Advanced Research and Technology, Washington, D.C., for their continued advice and encouragement of the STAR computer development.