Automatic Microwave Configuration Control

J. G. Leflang
R. F. Systems Development Section

Several radio frequency subsystems were recently operated at DSS 14 using minicomputers for data input, calibration, control, monitoring, and failure backup and diagnosis. As part of that test, the microwave configuration control portion of the antenna microwave subsystem was interfaced with a PDP-11/20 computer.

I. Introduction

Recently, practical computer control and monitoring of several radio frequency (RF) subsystems was demonstrated at DSS 14. A portion of the antenna microwave subsystem, the configuration control group, was included in the demonstration. The equipment in the configuration control group provides a means of controlling and reading 120 switch positions. Approximately 80 switch positions are presently used.

II. Hardware Design

Switch control is manually accomplished by actuating a push-button that completes a dc return path for one coil of a set of relays which control the 110-volt ac power applied to the microwave switch motors. The computer interface utilizes an open collector peripheral driver connected to the dc return path for each relay in a wire-OR configuration (Ref. 1). Thus, either manual push-buttons or the peripheral drivers may be used to actuate relays (and thereby switches) in the configuration control hardware.

Switch position indication is accomplished by using a 24-volt dc signal to illuminate the manual push-buttons which correspond to the switch position. The original technique for controlling and monitoring switch positions used opto-isolators for noise rejection and for shifting to transistor–transistor logic (TTL) levels (Ref. 1). Field tests using the original design equipment proved that the opto-isolators were unnecessary, and the equipment used for the RF demonstration used lower cost resistive voltage dividers for shifting from 24 volts dc to TTL levels, and did not use opto-isolators in the peripheral driver circuits.

The computer interface assembly is composed of a DSN standard interface, a function decoder, and five 24-bit shift registers. The shift registers provide 120 parallel outputs which drive all five switch groups (bays) within the configuration control hardware. The function decoder reads three bits of each input byte to determine the required function.

Of the eight possible function codes, five are used. The functions are:

(1) Output six data bytes which contain information on power supply status and the state of a computer input enable/disable flip-flop.

(2) Output 24 data bytes which contain information on the output state of all shift registers.

(3) Enable relay driver outputs.

(4) Disable relay driver outputs.

(5) Output 24 data bytes which contain information on all switch position indicators.

The use of a three-bit function code of value zero causes the shift registers to accept configuration data.

III. Software Design

The configuration control software was designed as two modules that are part of a nonreal-time sequence of processes which combine the receiver and microwave software in one computer. Coding was done in Digital Equipment Corporation's RT-11 BASIC.

Configuration commands are received as strings of two-digit numbers. Each number represents a macro configuration which is a useful functional portion of some configuration required for operations. One of the two software modules, the input message processor, segments the command string and validates each command. Valid commands are stored in a prime command stack if not specifically identified as a failure backup command by the prefix "FB." After the command string is stored, the stack is scanned one command at a time, and the corresponding macro is overlayed into an array which is then shifted in 24 bytes into the hardware registers. The registers are tested by reading back into the computer and comparing them to the macros specified in the prime command string. If the registers are properly loaded, an enable command is given to drive the switch control relays. This process is repeated for all storage cells in the command stack. Any register error is reported as a macro failure.

The second module provides a periodic monitor of switch positions specified by the command string. Switches not specified are not monitored. This module reports any discrepancies, loads the failure backup command, if any, into the prime command stack, reloads and tests the shift registers, enables the relay drivers, tests the switch position indicators for correct position, and reports the backup action to the monitor console.

IV. Results

The configuration control hardware operated reliably and precisely as intended. All circuits appeared to have noise margins which are adequate for operation in the station environment. In spite of the 305-meter (1000-foot) cable runs, opto-isolators were not required, but it should be pointed out that noise, not ground loops, is the only concern because the relay coils and contacts used for indicators are not grounded to the antenna.

Similarly, the software functioned well. However, core size restrictions prevented the use of extensive diagnostic routines. Also, the macro configuration tables were originally stored in magnetic tape virtual memory because of the shortage of memory. However, the long access time associated with reading a file from the tape prompted a software modification which stored the macro tables in memory. Prior to the modification, it was possible to consume many minutes performing diagnostics and failure backups.

At first, manual override of a computer-generated configuration required that the switch in question be moved twice because the computer would attempt to restore the original configuration once after any manual change. This proved to be very annoying, and the software was changed to permit human operators to move switches at any time. However, any manual operation which alters a computer-specified configuration still is detected by the computer and is identified on the display console.

V. Conclusions

Automatic configuration control of the microwave subsystem is feasible. We have developed a simple, compact, and reliable hardware design that interfaces the configuration control group with any computer which has a DSN interface and can support the necessary software. The use of software instead of hardware is very desirable for the purposes of establishing and testing functional combinations of switch positions (modes). The resulting hardware is low cost, and the software provides the flexibility necessary to adjust to changing mission requirements without large monetary expenditures.

The software functioned fairly well, but it did prove to be unreasonably slow and to require a large amount of memory to store the system executive and the basic interpreter (21.2 kbytes).

Because of low speed and large memory requirements, BASIC is not attractive as a programming language for use in the microwave subsystem. There are other languages which offer speed and compactness, and the use of one of these will permit a hardware/software package which will prove to be of substantial value to operations and will not carry a prohibitive price tag.

Reference

1. Leflang, J. G., "Automation of Microwave Configuration Control," in *The Deep Space Network Progress Report 42-21*, pp. 59–64, Jet Propulsion Laboratory, Pasadena, Calif., June 15, 1974.