RF Automation Demonstration

R. B. Crow
R. F. Systems Development Section

The radio frequency (RF) automation demonstration was conceived to gather data for a life cycle study to evaluate the cost and operational effectiveness of automation. The RF automation demonstration, as originally envisioned, performs a very important function of "qualifying" the system in that it checks both the necessary conditions (i.e., that the receive/transmit system can be configured and calibrated) and the sufficient conditions (by automatically acquiring the carrier and subcarrier). Continuous monitoring also allows real-time repair of those "necessary conditions" (relay operation, references, etc.) so that the system can be maintained at a higher level of readiness than is now possible.

I. Introduction

The RF (radio frequency subsystem) automation demonstration was conceived to gather data on an operational system so that criteria for managing DSN automation could be developed. The RF "subsystem," consisting of the microwave subsystem, two Block IV receivers and one exciter, two Block IV Subcarrier Demodulator Assemblies (SDAs), the noise-adding radiometer (NAR), and the 400-kW S-band operational transmitter located at DSS 14, was chosen since the hardware evolution made automation possible, and the administrative control for the development of this equipment was contained primarily within one section. Each subassembly was capable of either manual or computer control (selected by a front panel configuration switch). A computer was used to control each type of subassembly while system control and coordination were performed by the RF demonstration computer. The RF automation demonstration, as originally envisioned, performs a very important function of "qualifying" the system, in that it checks both the necessary conditions (i.e., that the receive/transmit system can be configured and calibrated) and the sufficient conditions (by automatically acquiring the carrier and subcarrier). Continuous monitoring also allows real-time repair of those "necessary conditions" (i.e., relay operation, references, etc.) so that the system can be maintained at a higher level of readiness than is now possible.

II. Demonstration Plan

The RF automation demonstration was managed by developing a functional requirements document describing the scenario for the RF automation demonstration, software standards, a development schedule, and manpower assignment.

Each subassembly controller software design was reviewed to assure compliance with the functional requirements before further development was undertaken. Maximum checkout at JPL was attempted to minimize the DSS 14 installation and checkout time.

Figure 1 is a functional block diagram of the RF system automation demonstration configuration. Each of the major subassemblies shown was controlled by its own computer; system control and coordination were performed by the RF demonstration computer, which also served as the operation interface.

The remainder of this report defines the original demonstration ground rules and reviews the actual demonstration performed at DSS 14. Observations and data on the experiment are listed. The report concludes with a summary of the important data of this demonstration, a proposed plan for future work, and criteria for future automation implementation.

III. Demonstration Ground Rules

A functional requirements document for the RF automation demonstration was written, setting forth the following ground rules:

(1) All software would be top-down design, implemented in a high-level language using structured programming.

(2) Available research and development (R&D) computers would be utilized (three PDP-11/20 minicomputers and one Intel 8080 microcomputer; see Fig. 1).

(3) The hardware interfaces between existing hardware and subsystem controllers would not be changed from those existing prior to this demonstration.

(4) The hardware interfaces between subsystem controllers and RF demonstration controller would be new installations utilizing the DSN standard 14-line interface and standard teletype.

(5) The interface between the operator and the RF demonstration controller would be simple English words.

(6) The RF demonstration computer would accept all configuration messages from the operator and distribute them to the correct subsystem controller. The RF demonstration controller would receive the necessary feedback from the subsystem controller and perform all necessary coordination between subsystem controllers to effect automatic configuration, calibration, acquisition, and failure backup.

(7) The RF demonstration controller would appear transparent to each subsystem controller in that diagnostic messages from each subsystem controller would be printed at the operator terminal exactly as sent by the subsystem. In this way, the design of diagnostics in the subsystem controller would not impact the design of the RF demonstration controller.

(8) Polling of the interfaces would be used to communicate between computers.

IV. Description of the RF Automation Demonstration

The RF demonstration controller was programmed to execute five phases of operational sequences which are described in the following paragraphs. Figure 2 is a flow diagram describing the demonstration software.

BEGIN. To begin a mission, the RF demonstration controller polled the subassembly controller to determine which of the subassembly devices were under computer control. The RF demonstration controller then sent a message to the operator indicating which devices were under computer control. The transmitter controller was then directed to turn on the filaments and coolant system, and to check all monitor facilities and the crowbar safety device. During this time, the Block IV SDAs were directed to calibrate so as to determine their operational status. When the transmitter became ready for precalibration (PRECAL) and both SDAs had been internally calibrated, the operator was notified to load the mission configuration into the RF demonstration controller's paper tape reader and type 'CAL' to begin PRECAL.

PRECAL. The mission configuration data were read in from a paper tape. The RF demonstration controller then sent all configuration messages to the appropriate subassembly controllers. After the messages had been received and acknowledged by the subassembly controller, the RF demonstration controller then sent calibration commands.

The noise-adding radiometer (NAR) was used to automatically measure system noise temperature. These data were sent to the receiver controllers where they were available to determine the signal power in the automatic-gain control (AGC) calibration process.

The RF demonstration controller coordinated the calibration phase by monitoring the status and issuing calibration commands at the appropriate time (i.e., after RCV 3 had completed calibration, SDAs 5 and 6 were commanded to calibrate with reference to RCV 3).

The operator was kept apprised of the progress of the calibration sequence by messages from the subassembly controller (through the RF demonstration controller) which were displayed on the operator's CRT control unit.

When all subassemblies were calibrated, the RF demonstration controller issued a standby command. The operator was then prompted to start the mission (by typing 'START').

OPERATE. When the operator typed a 'START' command, the transmitter was turned on and an automatic carrier acquisition sequence was initiated. The operator (and the RF demonstration controller) was notified by message generated by the receiver controller when each receiver had acquired. The RF demonstration controller automatically commanded the appropriate SDA to acquire once its input receiver was in lock. The operator was informed when each SDA acquired lock.

During this phase of the mission, the noise-adding radiometer automatically measured the open-loop gain of the maser preamplifier and front end of the S-band receiver. Changes in gain (greater than 0.02 dB) were sent to the receiver in order that the signal power versus AGC calibration could be corrected so that a more accurate carrier power estimate was available.

Each subassembly controller maintained a constant monitor on the "necessary" conditions for operation. If a failure was observed, the subassembly controller automatically reconfigured the appropriate subassembly to the predetermined (part of the original configuration input) backup configuration. The operator (and the RF demonstration controller) was notified of the configuration change and automatic acquisition was initiated. The operator then received the appropriate diagnostics/instruction to repair the failed unit.

The RF demonstration controller coordinated reacquisition (i.e., when a receiver failed, not only must it be functionally replaced with its predetermined backup, but the necessary configuration changes must be made to the SDAs to accept a new receiver input) and subsequent reacquisition of the carrier and subcarrier.

The operator terminated the mission by typing 'HALT' which automatically started the post-calibration (POSTCAL) sequence.

POSTCAL. The operator's 'HALT' command caused the transmitter to turn off. When the transmitter controller confirmed that the transmitter was in standby, the RF demonstration controller initiated a post-calibration sequence by first configuring the microwave equipment and then issuing post-calibration commands to the transmitter, receiver–exciter, and SDA subassembly controllers.

A similar sequence was followed as in pre-calibration (i.e., the RF demonstration controller coordinates the receiver and SDA calibration sequence). Noise temperature was automatically obtained from the NAR as in pre-calibration.

When all subsystems had completed post-calibration, the operator was prompted to type 'RESTART' to run another mission or type 'END' to secure the RF subsystem.

END MISSION. The transmitter controller was commanded to secure the transmitter (i.e., an orderly turnoff sequence in which the coolant system cools the power amplifier before turning itself off). All other subsystems were placed in a "standby" configuration.

V. Demonstration Results and Observations

The receive channel (i.e., the antenna microwave subsystem, Block IV receiver–exciter, Block IV SDA, see Fig. 1) successfully demonstrated automatic configuration, calibration, and acquisition (both carrier and subcarrier) of a simulated signal during the first tests. The transmit channel was not tested during this time (due to a hardware failure of the 400-kW S-band operational transmitter). However, a later demonstration was run with the 400-kW R&D transmitter, where automatic configuration, calibration, and output power control were successfully demonstrated.

Several observations made during the RF automation demonstration effort are listed below to document some of the more interesting characteristics.

(1) *Actual Time to Automatically Control the Receive Channel.* The BEGIN phase took 6 to 7 minutes to poll all controllers and internally calibrate the Block IV SDAs.

The PRECAL phase took 6 to 8 minutes to input and distribute the mission configuration data and an additional 10 minutes to calibrate receivers 3 and 4 (the receivers were calibrated in series) and 0.5 minute to calibrate SDAs 5 and 6.

This performance was far slower than the original estimate of 5 minutes for a complete PRECAL sequence. Future effort will be directed at reducing the PRECAL and configuration time.

(2) *Intra-computer Communications.* Software implementation of data transfer over a 14-line interface was used. This routine was written in both RT-11 (DEC) BASIC and PL/M (high-level language used in the Intel 8080). The BASIC software data transfer was very slow (6 bytes/second). The PL/M software, when transferring data to the Block IV SDA, operated at 600 bytes/second and should operate at 300 bytes/second if transferring data to another 8080 microcomputer.

Polling was used in lieu of interrupts in an attempt to simplify the software development and determine if polling performance was adequate for real-time hardware control of this type. It was found, however, that polling is probably the largest contributor to the poor communication speed, since as much as one minute was spent waiting for some of the controllers to process a message before accepting new data. This characteristic became a serious problem when it was necessary to continue handling low priority (diagnostic) messages while operational "action" messages were left waiting for several minutes.

Both 14-line and serial interfaces were used during the demonstration. No known noise or communication failures were noted. One hardware failure occurred, and a possible "start up" problem in the serial interface hardware was noted.

(3) *Program Size/Development Time.* Table 1 indicates the size of each program and the approximate software development effort. Several factors should be considered when reviewing these data:

(a) The RT-11 operating system had software bugs in its "overlay" option that forced the RF demonstration program to be rewritten to achieve a reasonable software reliability.

(b) The PDP-11's with the large number of unibus loads (i.e., Dectape, several digital input/output (I/O) interfaces, analog-to-digital converter (ADC), teletype (TTY), etc.) were unreliable. Random system failures occurred with no known hardware failure to explain them. During the course of the demonstration, several hardware failures did occur (i.e., memory failure, TTY failure, paper tape reader failure). The PDP-11's poor reliability record stood out in stark contrast to the 8080 microcomputer which has operated *without* failure since its purchase in December 1974.

(c) The RT-11 operating system required 10.5 kilowords of core. The core requirement of the programs (listed in Table 1) required "chaining" in all PDP-11 computers except the receiver–exciter controller where 24 kilowords of core existed.

(d) Proper structuring of the software from both the instructional and data viewpoints was hindered by the use of RT-11 BASIC and the lack of memory. For example, in the receiver–exciter controller, sets of bits had to be packed into floating point variables conserving core, but creating awkward, time-consuming BASIC constructs to manipulate them.

(4) *Operational Effectiveness.* The demonstration pointed out the need for more flexible operator controls so that nonstandard operations could be handled (i.e., it should be possible: (a) to change the receiver's acquisition characteristics by merely typing "RCV 3, Δf = -100 kHz" and lower the original S-band acquisition window by 100 kHz, (b) to change the receiver doppler rate, and (c) to change SDA symbol rate and modulation index by similar high-level operator inputs, etc.).

(5) *High-Level Language.* RT-11 BASIC was used for all controllers except the SDA controller (which used PL/M). The BASIC language was slow, had no capability for bit manipulation (required in configuration control), and was not well suited for structured programming.

PL/M is a high-level language currently available as a cross compiler installed in the Univac 1108 to generate code for Intel 8080 as compiled object code. PL/M provides a more favorable speed and memory characteristic than does the RT-11 BASIC. The present PL/M operating system contains a two-pass compiler and an 8080 simulator. This facility is flexible and was used with good results for correcting a software problem while at DSS 14.

(6) *Top-Down, Structured Design.* This design method produced efficient readable code. Changes were

easily made (due to modular code that resulted from using a structured design).

VI. Summary of the RF Automation Demonstration

Lack of operational reliability was the single most troublesome factor in demonstrating automation feasibility. The importance of reliable hardware and software with built-in fault isolation and diagnostic capability was recognized and will be incorporated into future designs.

A much better understanding now exists of the needed operator controls and the environment in which they are used.

The RF automation demonstration served as an excellent "breadboard" in that it offered a stage where several concepts turned out quite well (i.e., top-down, structured software design, use of 8080 microcomputer utilizing the PL/M language, etc.), while others turned out quite poorly (i.e., BASIC was a poor language for this demonstration, polling did not work satisfactory, the PDP-11's were unreliable, etc.).

VII. Recommendations

Automation should be implemented when:

(1) The automation controllers to be implemented are reliable enough so that the added burden imposed on the station is more than offset by improved performance and station availability.

(2) The system is flexible enough to permit the operators to handle all nonstandard events that are presently accomplished.

Other general recommendations are:

(3) Polling should not be used; instead the system should be interrupt driven.

(4) Future interface hardware design should contain built-in fault isolation aids to facilitate system repair.

(5) Failure analysis and automatic functional backup should save valuable mission data while the subassembly diagnostic should minimize down-time and so should be actively pursued.

(6) RT-11 BASIC is too slow for a real-time control system. This observation coupled with the poor reliability of the PDP-11 hardware and software indicate the PDP-11 should not be used again as the controller.

(7) The PL/M language used by the 8080 appears efficient in time and memory and should be used in another demonstration.

(8) Adequate memory should be planned for.

(9) In the foreseeable near future it seems wise to concentrate on providing the operators with improved tools (i.e., useful macro commands, RCV 3:Δf = −100 kHz, DOP = 200 Hz) without trying to solve costly operational problems that occur infrequently (i.e., it would be possible to conduct an automatic organized search of carrier, subcarrier, symbol rate in the event of major trauma to the spacecraft; however, our current state of development does not as yet offer this capability as a cost-effective option).

(10) Software standards should be created using top-down, structured design with a high-level language so that eventually one software sustaining engineer could maintain the RF subsystem software.

VIII. Future Plans

A second RF automation demonstration is presently planned for April 1976. This demonstration will follow the basic plan established for the first demonstration but will emphasize system reliability, operator flexibility, and improvement of the automation data base. A preliminary study will be undertaken to determine the current stage of hardware automation in each major subassembly and will present management options for future automation.

Table 1. RF demonstration program size/development time

Controller name	Total number of BASIC or PL/M statements	Core requirement in 1000 16-bit words	Software development time, man days			Computer type/language
			Design	Code	Debug	
Demonstration configuration	1250	14	40	15	25	PDP-11 BASIC
RF demonstration	1050	10	35	20	60	PDP-11 BASIC
Receiver–exciter	1000	10	20	10	25	PDP-11 BASIC
Transmitter	700	10	50	15	50	PDP-11 BASIC
SDA	650	9 (8-bit words)	30	15	10	8080 PL/M
Microwave	200	4	10	2	5	PDP-11 BASIC

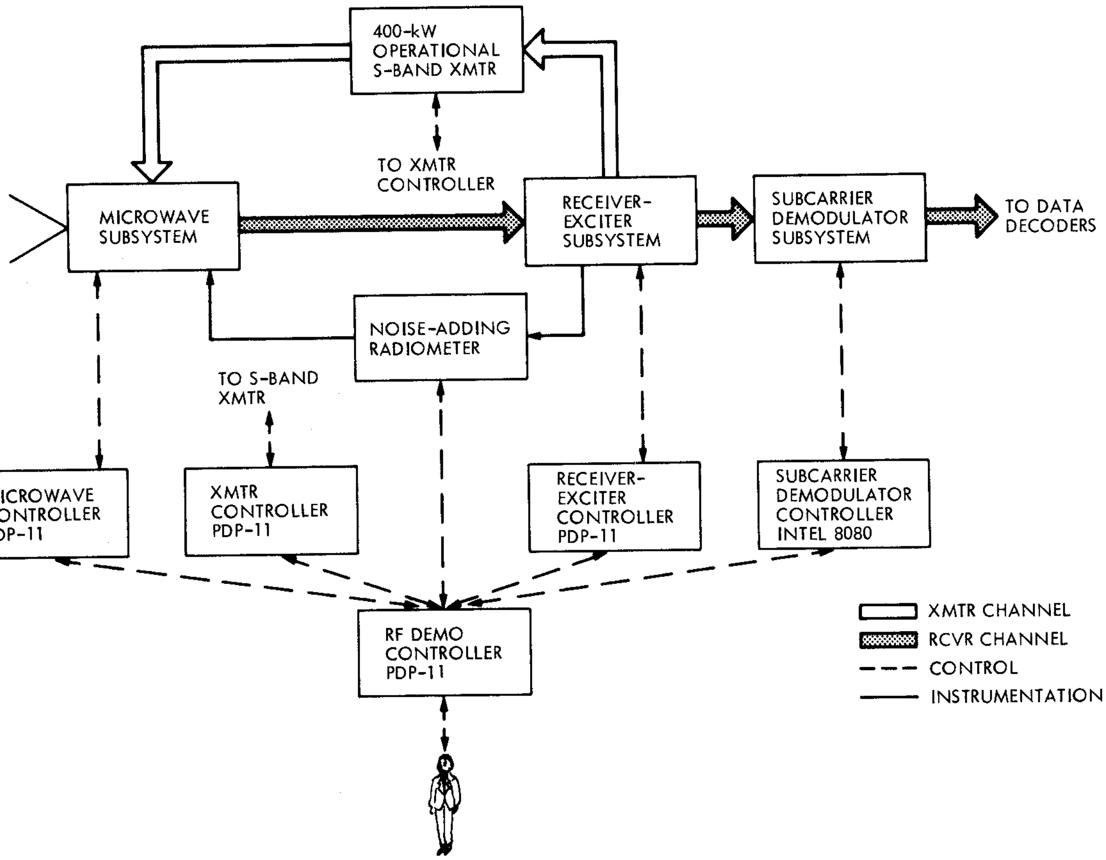

Fig. 1. RF automation demonstration functional block diagram

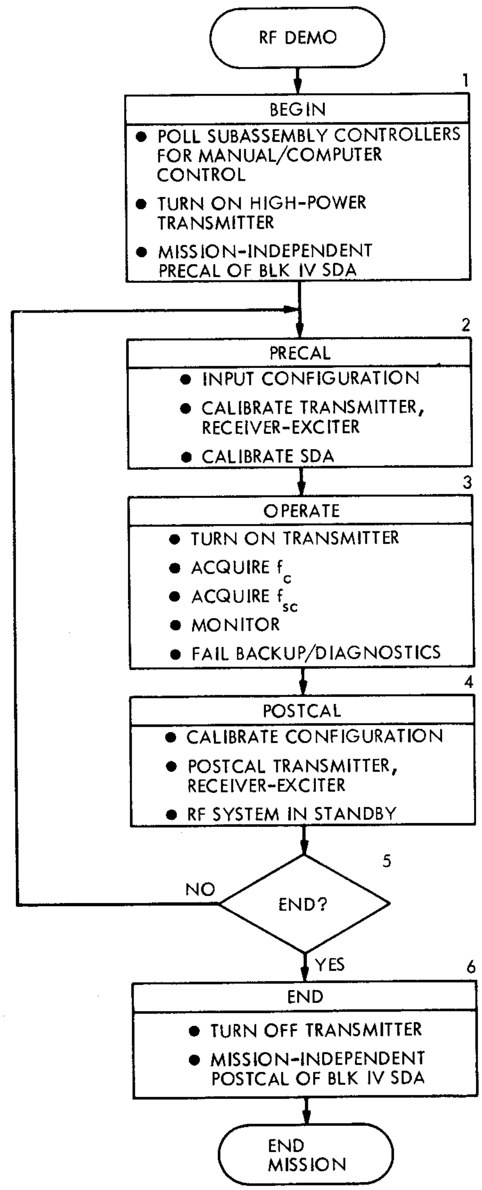

Fig. 2. Flow diagram of RF demonstration control program