



Apr 26th, 2:00 PM - 5:00 PM

Paper Session I-C - Rubicon System for Space Shuttle Vehicle Health Management

T. R. Fowler

Rockwell International, Launch Support Services, Kennedy Space Center S. B. Wilson, NASA / TV-GDS, Kennedy Space Center

A. C. Melichar

Rockwell International, Launch Support Services, Kennedy Space Center S. B. Wilson, NASA / TV-GDS, Kennedy Space Center

Follow this and additional works at: <http://commons.erau.edu/space-congress-proceedings>

Scholarly Commons Citation

T. R. Fowler and A. C. Melichar, "Paper Session I-C - Rubicon System for Space Shuttle Vehicle Health Management" (April 26, 1994). *The Space Congress® Proceedings*. Paper 5.

<http://commons.erau.edu/space-congress-proceedings/proceedings-1994-31st/april-26-1994/5>

This Event is brought to you for free and open access by the Conferences at ERAU Scholarly Commons. It has been accepted for inclusion in The Space Congress® Proceedings by an authorized administrator of ERAU Scholarly Commons. For more information, please contact commons@erau.edu.

Abstract

Rubicon System for Space Shuttle Vehicle Health Management

T.R. Fowler, A.C. Melichar, Rockwell International, Launch Support Services, Kennedy Space Center
S. B. Wilson, NASA / TV-GDS, Kennedy Space Center

An important function performed by Space Shuttle firing room engineers is the detection and analysis of system malfunctions and failures. Timely identification and resolution of anomalous conditions is a key element of efficient orbiter test and processing. Presently problem resolution is a largely manual activity relying heavily upon the engineer's expertise and memory.

There are currently efforts in progress at Kennedy Space Center to improve this process by automating fault detection. Several software prototypes called advisory systems are under development which perform automated failure analysis. These systems monitor Space Shuttle telemetry and continuously check for off-nominal conditions.

To date advisory systems have been developed independently of one another to diverse standards and requirements. They are consequently single-purpose, requiring their own individual workstations, telemetry processors, and supporting software procedures. However, as these projects move out of the prototype stage into an operational environment this approach becomes unworkable. Part of the solution is to assemble individual advisory systems into a comprehensive system for vehicle health management.

The Rubicon (Reasoning Based on Intelligent Computing and Networking) project attempts to do this by providing a vehicle for practical introduction of advanced diagnostic software into an operational environment. Rubicon takes the form of a workstation-based software system giving the engineer access to an array of vehicle health management software and data analysis tools. Rubicon includes graphical user interfaces and controls for advisory system operation and a set of utilities for real time data analysis and display. The engineer can conveniently access a powerful set of fault analysis tools at a single location.

Rubicon is being developed by NASA / TV-GDS and Rockwell / Launch Support Services at KSC. It is a step toward operational vehicle health management. Rubicon demonstrates several important points. It shows that multiple advisory systems can run cooperatively on a single workstation. Also, common functions supplied by Rubicon can preclude the need for their redevelopment for future advisory systems. Finally it is possible to automate data analysis tasks that are now performed manually. When completed it will serve as a new data resource for the firing room engineer.

Rubicon System for Space Shuttle Vehicle Health Management

T.R. Fowler, A.C. Melichar
Rockwell International, Launch Support Services
Kennedy Space Center

S. B. Wilson
NASA / TV-GDS
Kennedy Space Center

Introduction ...

An important function performed by Space Shuttle firing room engineers is the detection and analysis of system malfunctions and failures. Timely identification and resolution of anomalous conditions is a key element of efficient orbiter test and processing. Presently problem resolution is a largely manual activity relying heavily upon the engineer's expertise and memory. Collecting relevant data and supporting documentation is also a manual operation.

There are currently efforts in progress at Kennedy Space Center to improve this process by automating fault detection. Several software prototypes called advisory systems are under development which perform automated failure analysis. These systems monitor Space Shuttle telemetry and continuously check for off-nominal conditions. When such occurrences are detected the system attempts to determine the cause and advise the firing room engineer of its findings. Advisory systems can thereby provide efficient and consistent analysis of vehicle anomalies.

Advisory systems have been developed for various orbiter systems, e.g., data processing and main propulsion, by various organizations. To date these projects have been developed independently of one another to diverse standards and requirements. They are consequently single-purpose, requiring their own individual workstations, telemetry processors, and supporting software procedures. However, as these projects move out of the prototype stage into an operational environment this approach becomes unworkable. Systems developers have to re-invent software previously created elsewhere. Part of the solution is to assemble individual advisory systems into a comprehensive system for vehicle health management.

The Rubicon (Reasoning Based on Intelligent Computing and Networking) project attempts to do this by providing a vehicle for practical introduction of advanced diagnostic software into an operational environment. Rubicon takes the form of a workstation-based

software system giving the engineer access to an array of vehicle health management software and data analysis tools. The workstation will be located at the firing room console or other site with access to orbiter telemetry. Rubicon includes graphical user interfaces and controls for advisory system operation and a set of utilities for real time data analysis and display. The engineer can conveniently access a powerful set of fault analysis tools at a single location. **Figure 1** is a schematic representation of Rubicon.

Rubicon is being developed by NASA / TV-GDS and Rockwell / Launch Support Services at KSC. To date, a working prototype has been completed and two advisory systems have been incorporated into the Rubicon architecture. The sections to follow describe development of the prototype and work planned for the future.

Rubicon Goals and Requirements . . .

Three general goals were defined for Rubicon at the start of the project. First, Rubicon would allow the engineer convenient access to a variety of advisory systems. This required an architecture to be developed that would permit multiple advisory systems to run on a single workstation. Second, Rubicon would include tools or utilities that automate common data analysis tasks. Third, Rubicon would make available to the advisory system developer a series of standard software procedures that perform functions required in common by all advisory systems.

Requirements for Rubicon were developed from the basic goals as follows::

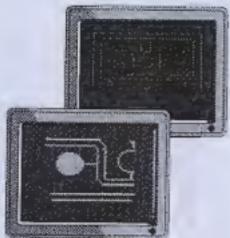
1. Rubicon must allow multiple advisory systems to run on a single workstation. It should not limit the type or number of advisory systems which can be run simultaneously; rather, this should be bounded only by the capabilities of the hardware.
2. Rubicon must not diminish performance of an advisory system. Imposition of procedures, code, or interfaces between an advisory system and the data it requires should be avoided.
3. Rubicon should not impose arbitrary standards on advisory systems. No unnecessary constraints should be placed on developers.



Common Telemetry Link



Common Data Base Interface



Access to Multiple Advisory Systems



Access to Multiple Utility Functions

Figure 1: Rubicon System For Vehicle Health Management

4. Rubicon should not become a burden for the console engineer. Rubicon will not require any operator action or intervention. Similarly, Rubicon should require little or no formal training.
5. Rubicon will make available to developers a set of procedures for performing common functions but whenever possible their use will not be required. An advisory system may perform its own routines if desired. There will be some limits to this however such as when a procedure used by one advisory system interferes with that used by another.
6. Rubicon will maximize use of capabilities of the workstation and UNIX operating system rather than developing new software to perform similar functions. An example is communications among advisory systems which has been implemented using UNIX message queuing.

These requirements have been developed into a software architecture to support advisory system development and implementation.

Rubicon Architecture . . .

Rubicon is comprised of several software elements which together form an architecture for advisory system development. In general, Rubicon provides essential services to advisory system developers with very low overhead. Rubicon provides:

1. A common user interface
2. Common access to real time orbiter telemetry
3. Common access to data bases containing problem histories, documentation, text and graphical information
4. Message processing among advisory systems and Rubicon
5. Common access to utility functions which provide tools for viewing and analyzing data

Specifics of each element are given in the following sections.

User Interface

Rubicon is operated through a base user interface (**Figure 2**). When Rubicon is not handling requests from users or advisory systems it is primarily involved in display update. This has been kept to a minimum so as not to monopolize CPU resources. Some general information - active data sources, telemetry formats, vehicle identification and time - is displayed at the top. The center area of the base display contains a representation of the orbiter and important system components. The diagram gives power status of each unit. Its inclusion as the base display is arbitrary but demonstrates a useful graphical presentation of orbiter data. When an advisory system is active its display will overlay the orbiter diagram as shown in **Figure 3**. Although it is desirable for advisory system displays to fit in this space there is no size constraint imposed by Rubicon. The lower part of the base display contains operating controls. From the base display the user can call up several advisory systems, the number being limited only by the capabilities of the hardware.

Buttons in the lower right of the base display (**Figure 4**) operate individual advisory systems. Below each button is a status "light" that gives current state of the advisory system. Rubicon provides for background and active modes of operation. This was implemented so that advisory systems may run but not be displayed. This allows the operator to watch one system while running another in the background. Systems running in background mode can still detect and analyze faults and report results through Rubicon's message processing.



Figure 4: Advisory System Execution Controls

DATA PROCESSING SYSTEM POWER STATUS

- INDICATES LRU POWERED ON
- INDICATES LRU POWERED OFF

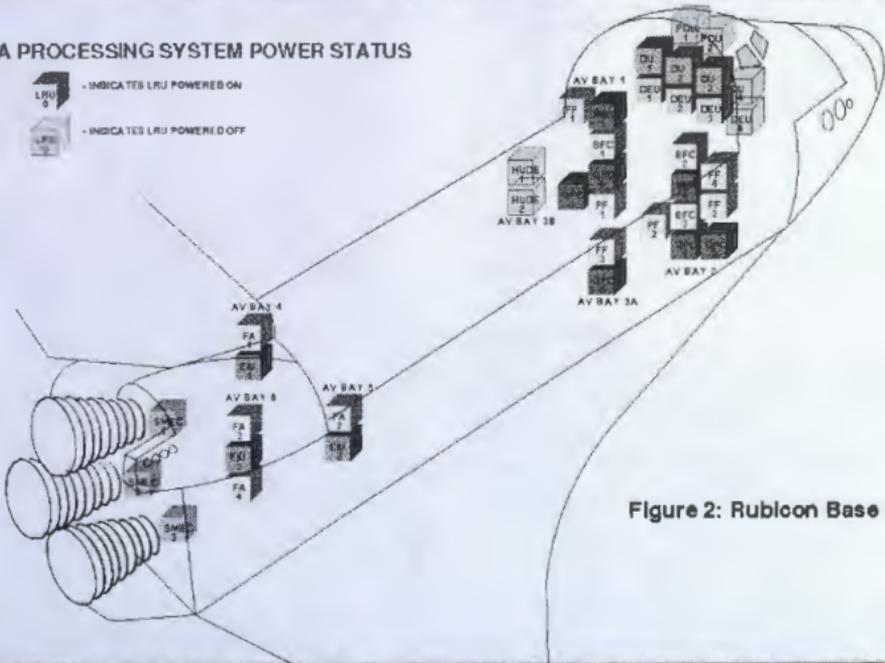


Figure 2: Rubicon Base Display

AVSTN MESSAGE:

1.000 BUBBL	AIRWAY	AIRWAY	DATA RANGE	EXIT	INACTIVE	INACTIVE	INACTIVE	INACTIVE	INACTIVE

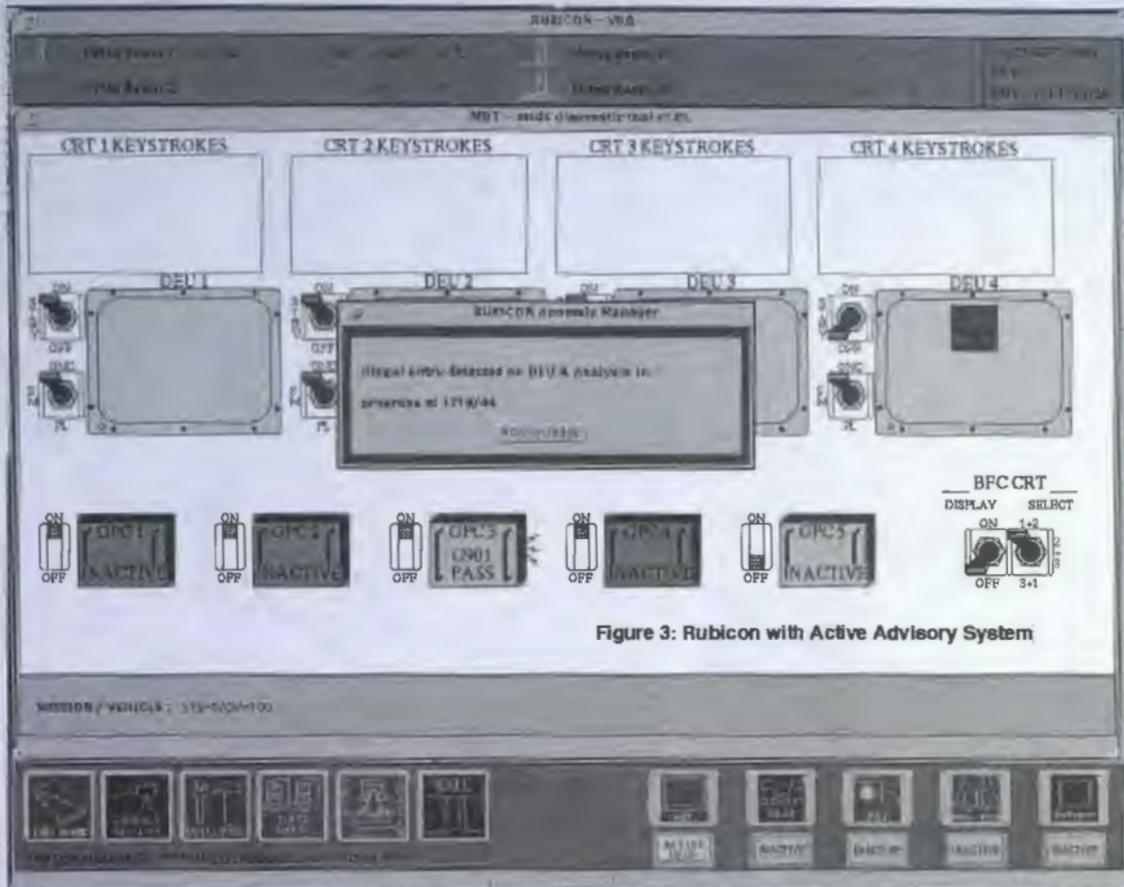


Figure 3: Rubicon with Active Advisory System

Data Acquisition

For vehicle health management programs to function they must have access to real time vehicle data. Advisory systems determine when an error has occurred by continuously monitoring a range of measurements and starting a fault analysis procedure when off-nominal values are detected. Most advisory systems will also drive a display which uses real time telemetry to show current system status. Rubicon acquires telemetry via ethernet link to the PCGOAL network. PCGOAL is a NASA developed network to distribute real time orbiter test data. Data are then made available to advisory system through a standard data read function.

Data Base Interface

All advisory systems and certain utility functions will require access to a data base. When an advisory system analyzes the cause of an anomaly it must retrieve results - data and text output - to present to the user. Systems also have requirements to access problem histories or other program documentation. ORACLE data base software has been incorporated into Rubicon to manage the data base and an interface has been added to facilitate user inquiries. Provision of a data base interface by Rubicon eliminates the requirement for each advisory system developer to build one anew.

Message Processing

Rubicon can both receive messages from and send messages to advisory systems. Message processing involves identifying the message type and contents and invoking the proper handler. Advisory systems can ask Rubicon to perform some standard function, such as retrieving data base records or displaying text messages to the user. Presently six types of messages have been incorporated:

1. General Advisory Messages
2. Anomaly Detected Message
3. Anomaly Resolution Retrieval
4. General Data Base Access
5. Stop Processing Message
6. Inter-Advisory System Message

Rubicon's message processing is accomplished using UNIX interprocess communication commands. Should an advisory system want to communicate it needs only to assemble the message text and execute a message send procedure. For example, when a fault is detected, an advisory system needs only to send a message to Rubicon; all announcements to the user and data base accesses will then be performed automatically.

Utilities

Rubicon has incorporated several procedures, called utilities, which perform services useful for general vehicle health monitoring. These functions perform no fault analysis but rather allow the operator more efficient access to and improved display of supporting data. Rubicon presently has three utilities:

1. Multiplexer / Demultiplexer (MDM) Channelization
2. Log Book
3. Switch Scan

MDM Channelization - When an orbiter component has failed it is often useful for troubleshooting to determine the related MDM card and channel. Rubicon can retrieve and graphically display MDM channelization data and card complement. Presently this information is not conveniently accessible to the console operator. A button on Rubicon's utilities panel brings up a screen that gives immediate access to MDM data. This window (**Figure 5**) contains an orbiter schematic showing only MDM's and a scrolling list of measurements. Each MDM is a mouse target. Clicking on any one brings up a sub-window showing the card complement of that MDM. Clicking on a card brings up another window listing all the measurements on that card. The operator may also click any measurement contained in the scrolling list. This will bring up the MDM card and channel of that measurement and all other measurements on that card.

Log Book - Another feature which advisory systems have in common is a procedure to log key events. Examples are occurrence of faults, power status changes, or keyboard entries. There is no need for every advisory system to implement its own log. Rubicon provides a function which allows any system to record a time-tagged event to the log. The event can also be tagged with the name of the advisory system recorded it. A panel button causes the log book to be displayed. It can be printed if desired. This function will relieve the firing room engineer of manually recording general console activity.

ENTER BOARD # (0-255)
 Press [ENTER]
 ENTER BOARD #

Press [ENTER]
 Press [ENTER]

MDM Channelization Data Base

MSID / NOMENCLATURE

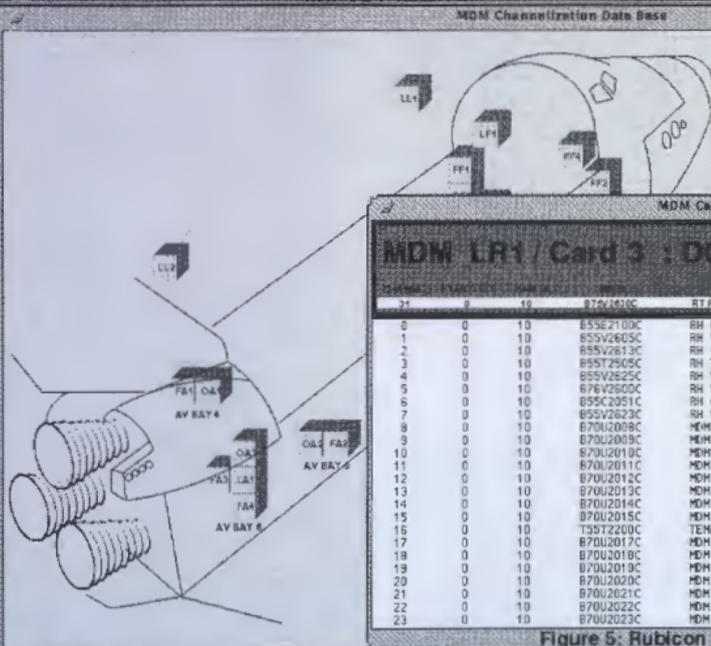
870U2027C MDM LR02 SPARE 'A'
 870U2028C MDM LR02 SPARE 'A'
 876U2029C MDM LR02 SPARE 'A'
 870U2030C MDM LR02 SPARE 'A'
 875K3085X MDM LOCK VERIF TEST DISC LL01-ODOL
 875K3086X MDM LOCK VERIF TEST DISC LL01-4DOL
 875K3087X MDM LOCK VERIF TEST DISC LL02-ODOL
 875K3088X MDM LOCK VERIF TEST DISC LL02-4DOL

MDM Card / Channel Data

MDM LR1 / Card 3 : DC Analog / P Differential

ST	0	10	870U2030C	RTFS SEP 1 (SW REV) VOL 15	EMDM
0	0	10	855E2100C	RH POWER R55 RCVR SIG STRENGTH A	BRS
1	0	10	855V2605C	RH VOLTAGE FMD THR PIN PIC CAP A	BPYRO
2	0	10	855V2612C	RH VOLTAGE FMD SEPM HOT PIC CAP A	BPYRO
3	0	10	855T2505C	RH TEMP SR0 R55 BATTERY	BRS
4	0	10	855V2625C	RH VOLTAGE R55 BATTERY NO 1	BPYRO
5	0	10	876V2600C	RH VOLTAGE OPERATIONAL BUS A	BELEC
6	0	10	855C2051C	RH CURRENT R55 BATTERY NO 1	BRS
7	0	10	855V2623C	RH VOLTAGE R55 PIC CAP A	BRS
8	0	10	870U2009C	MDM LR01 SPARE-AID CD 03 CH 08	BMDM
9	0	10	870U2009C	MDM LR01 SPARE-AID CD 03 CH 09	BMDM
10	0	10	870U2010C	MDM LR01 SPARE-AID CD 03 CH 10	BMDM
11	0	10	870U2011C	MDM LR01 SPARE-AID CD 03 CH 11	BMDM
12	0	10	870U2012C	MDM LR01 SPARE 'A'	BMDM
13	0	10	870U2013C	MDM LR01 SPARE-AID CD 03 CH 13	BMDM
14	0	10	870U2014C	MDM LR01 SPARE-AID CD 03 CH 14	BMDM
15	0	10	870U2015C	MDM LR01 SPARE-AID CD 03 CH 15	BMDM
16	0	10	755T2200C	TEMP ET R55 BATTERY B	TRS
17	0	10	870U2017C	MDM LR01 SPARE-AID CD 03 CH 17	BMDM
18	0	10	870U2018C	MDM LR01 SPARE-AID CD 03 CH 18	BMDM
19	0	10	870U2019C	MDM LR01 SPARE-AID CD 03 CH 19	BMDM
20	0	10	870U2020C	MDM LR01 SPARE 'A'	BMDM
21	0	10	870U2021C	MDM LR01 SPARE 'A'	BMDM
22	0	10	870U2022C	MDM LR01 SPARE 'A'	BMDM
23	0	10	870U2023C	MDM LR01 SPARE 'A'	BMDM

Figure 5: Rubicon MDM Channelization Utility



AV BAY 4

Switch Scan - Another utility gives access to orbiter panel and switch data. Knowledge of switch position is frequently required during routine orbiter operations. A diagram can be called up which shows all forward orbiter panels (**Figure 6**). The panel names are active mouse targets. Clicking on one will bring up a drawing of the panel and all switches thereupon. Switches which are instrumented are highlighted in color. The current position of each switch can be read as if one were looking at the actual panel. A set of buttons has been included to allow the operator to locate the panel for a given switch.

As with data bases, the range of utilities is essentially limitless. Additional ones are planned as resources become available.

Summary . . .

Rubicon is a step toward operational vehicle health management at KSC. The prototype demonstrates several important points. It shows that multiple advisory systems can run cooperatively on a single workstation. Also, common functions supplied by Rubicon can preclude the need for their redevelopment for future advisory systems. Finally it is possible to automate data analysis tasks that are now performed manually. Additional development of Rubicon is ongoing. When completed it will serve as a new data resource for the firing room engineer.

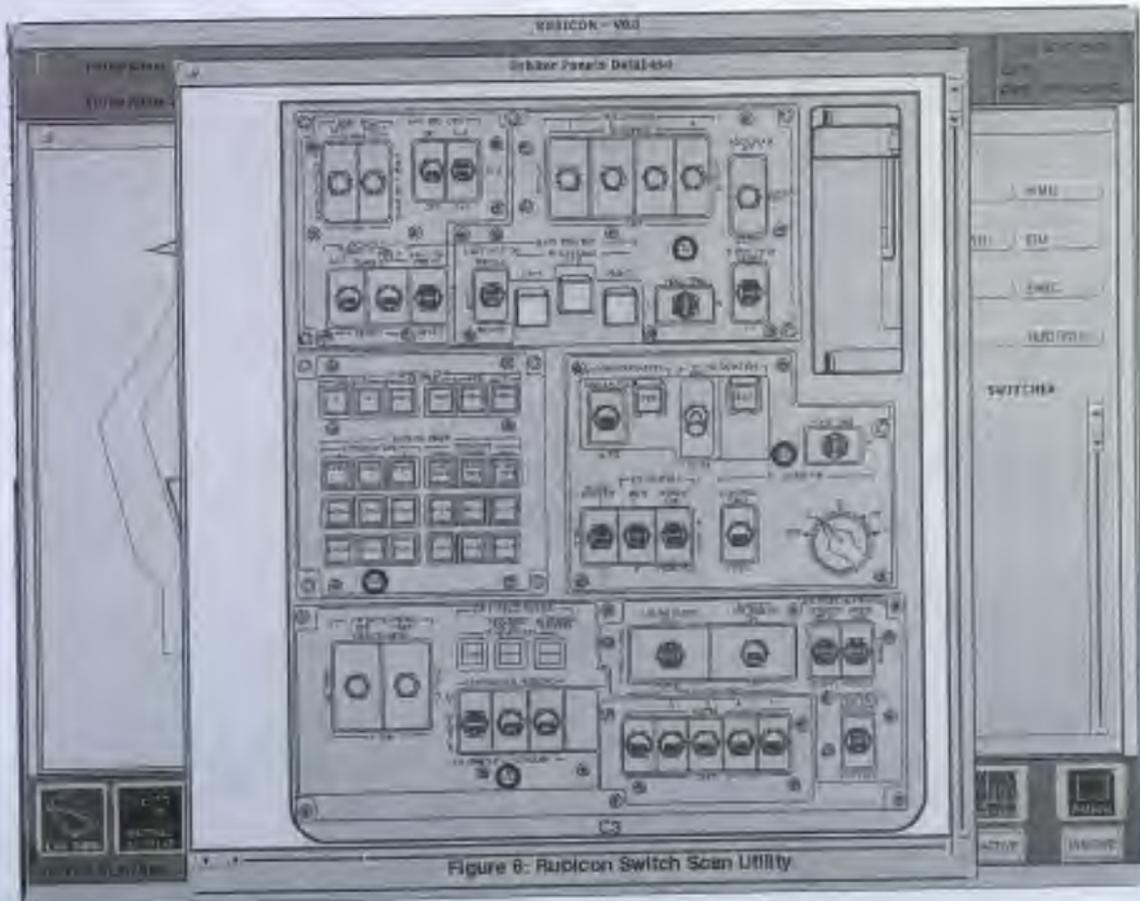


Figure 6: Rubicon Switch Scan Utility