DRAFT II: JBB/pj
1/27/72

CONF-720607--2

QUALITY ASSURANCE TECHNIQUES FOR ON-LINE PROCESS COMPUTER SOFTWARE*

J. B. Bullock
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37830

Experience in the operation of an on-line process control computer at the High Flux Isotope Reactor (HFIR) has indicated that a major consideration in the planning of such systems should be software quality assurance. The following practices have been implemented at ORNL and have proven to be most helpful in maintaining the software integrity in the HFIR system.

1. Maintain complete listings of all system programs in a clearly indexed file or notebook. This file should include a list of the previous version of any revised or corrected program.
2. Maintain a copy of all system program source images (cards, tape, etc.), clearly labeled by name and revision date or number.
3. Maintain a logic block diagram of all complex programs showing their relationship to other programs in the system. The more complex programs should use both the serial logic and the parallel logic forms described below.
4. Require appropriate designer comments in the comment fields of each program to clearly document the intent of the adjacent computer instruction groups.
5. Maintain a duplicate ~~or nearly duplicate~~ copy of the source image of all programs in a separate room or building in the event of local damage or disaster.

---

*Research sponsored by the U. S. Atomic Energy Commission under contract with the Union Carbide Corporation.

6. Establish and rigorously enforce a software change procedure which should include specifications of:

   a. person(s) responsible for reviewing and approving change requests,

   b. person(s) responsible for effecting the change,

   c. the method or test required to verify the anticipated effects of the change,

   d. documentation changes, including new program listings, new logic diagrams, and new source copy.

Many of the items specified above are reasonably obvious, common sense techniques which many users have no doubt recognized and implemented. However, the practice of constructing both serial logic and parallel logic block diagrams is probably unique with ORNL and merits further consideration.

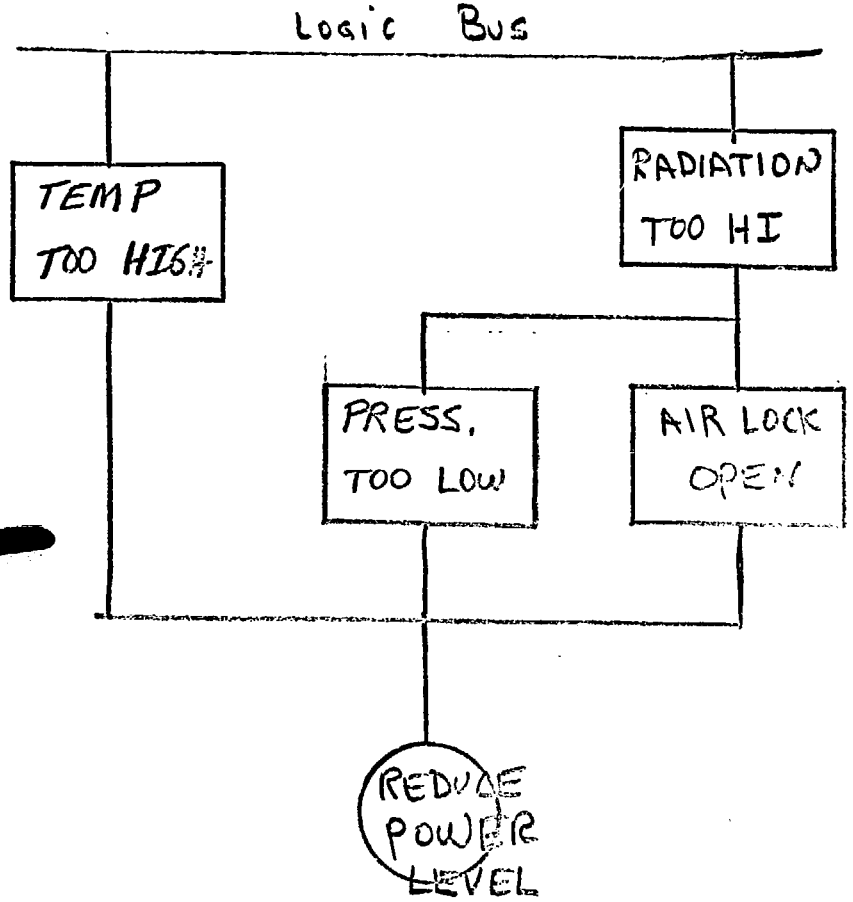
The serial logic diagram is the conventional logic diagram usually drawn by computer programmers showing the yes-no decision blocks and the functions being performed as the logic flows in a manner roughly synchronous with the program instructions. The term "parallel logic" is applied to a logic diagramming technique developed for conventional control system design and has been in use for many (1-3) years at ORNL. In the past the method has been used most extensively for showing the functional criteria or design objectives in conventional relay logic control system design. However, the technique has now been used for block diagramming complex computer program logic with great success. The principal advantage of the parallel logic diagram is illustrated with a highly simplified algorithm by displaying the identical logic functions using both the serial and the parallel logic diagram form. It is readily apparent that the parallel scheme is ideally suited for rapidly ascertaining the multiple paths or conditions to a given end state. Whereas

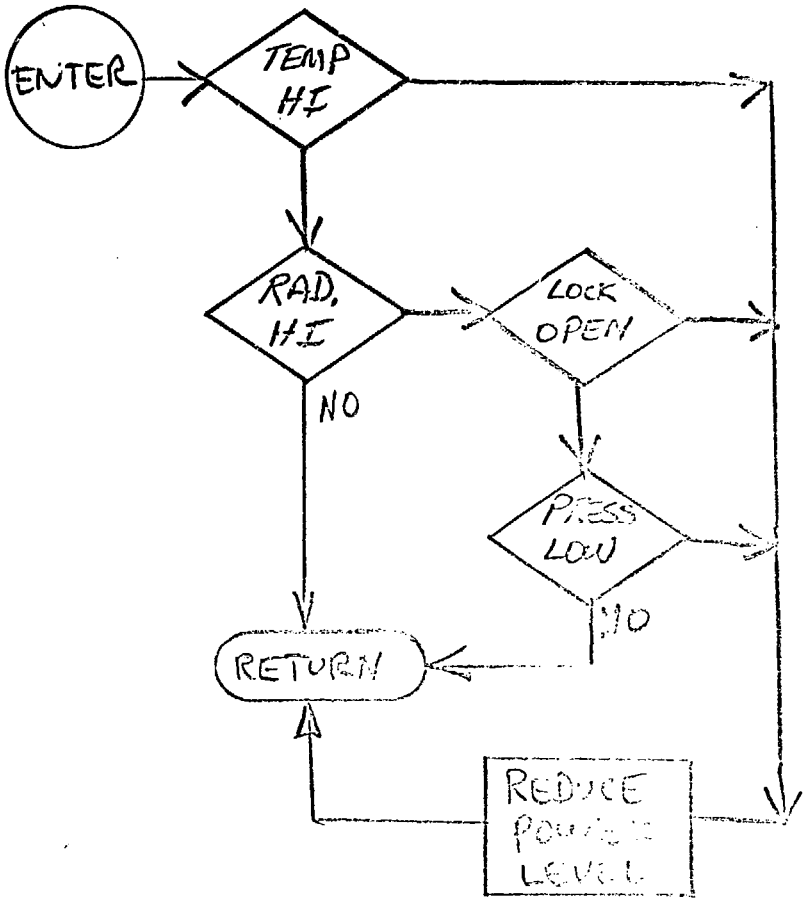
the serial logic diagram defines the programming steps more exactly and is consequently more useful than the parallel diagram in verifying that the actual program instructions will perform the desired functions.

When applied to complex programs, the two techniques represent a very efficient and compatible method for clearly documenting the design intent, the functional criteria, the significant variables, and the overall control system strategy. Documenting these important factors in a clear, simple manner will result in an information base on which a detailed design, a high quality review, and operational debugging can be planned, because, not unlike the conventional system design, a major detriment to the assurance of high quality in software systems is the basic man-to-man interface.

F. T. Binford and E. N. Cramer, Editors, The High Flux Isotope Reactor, USAEC Report ORNL-3572 (Rev. 2), June 1968, p. 125.

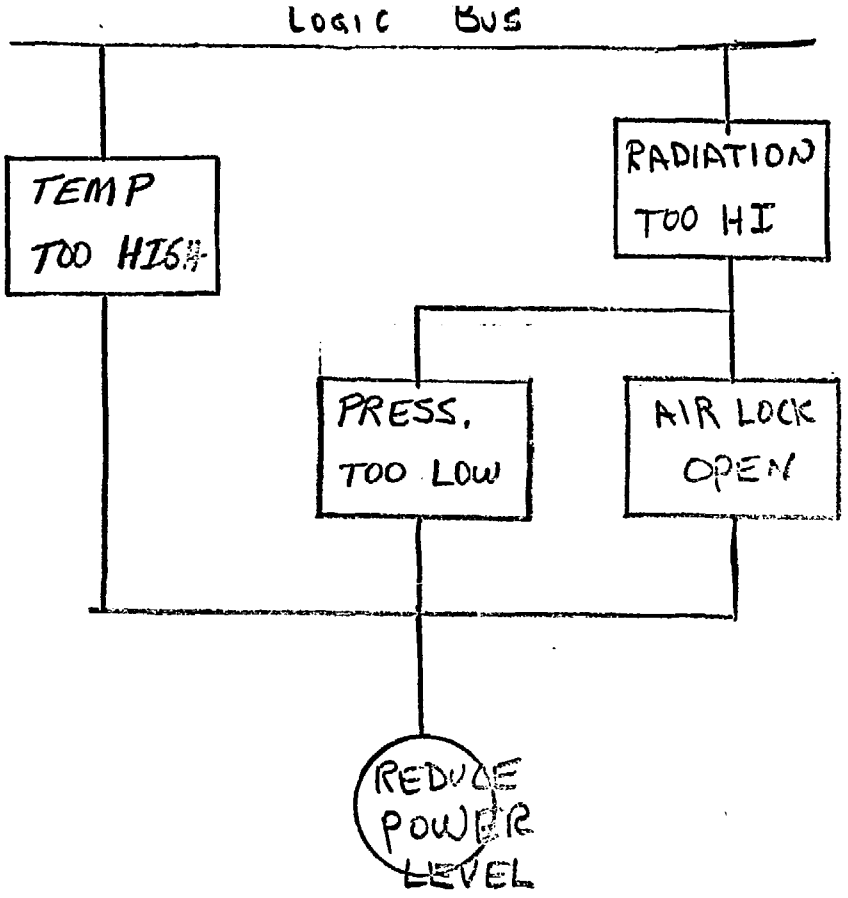
PARALLEL LOGIC DIAGRAM ← 14-121 → SEQUENTIAL LOGIC DIAGRAM

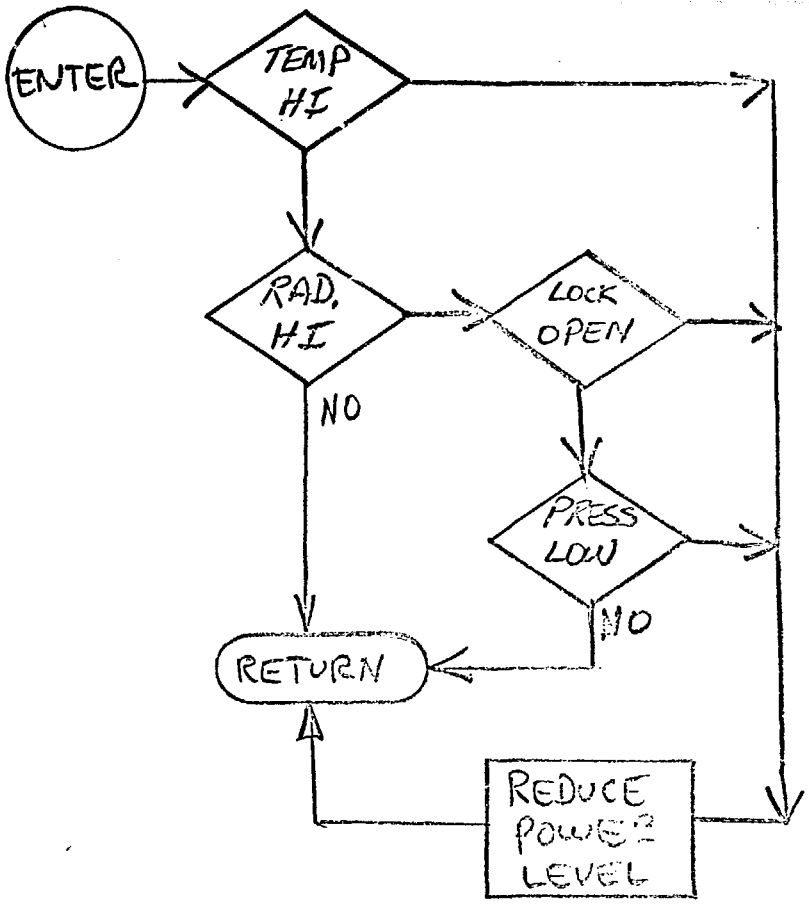




PW-1-001
8½x11
2-72 SLIDE
12-121

Figure — 1


LOGIC BUS
TEMP TOO HIGH
RADIATION TOO HI
PRESS. TOO LOW
AIR LOCK OPEN
REDUCE POWER LEVEL


ENTER
TEMP HI
RAD. HI
LOCK OPEN
NO
PRESS LOW
NO
RETURN
REDUCE POWER LEVEL

2

Figure — 1