CONTENTS

Introduction 3
Everything you need for supervision and control 4
Powerful CRT console 6
IMPAC control package 8
FOX 1 FORTRAN and MAX 10
And more programming capability 13
Comprehensive software security 14
FORTRAN statements and assembler instructions 15
Summary of FOX 1 software characteristics 18
Companies are fast realizing that the most substantial rewards of process computer systems lie in the area of plant supervision. An on-line computer can perform a broad range of supervisory functions that maximize plant efficiency while simultaneously performing data acquisition and control. Such activities as process optimization, management reporting, adaptive control, utility efficiency, and material balancing can cut costs dramatically while boosting production and improving quality.

To date, however, computer implementation of advanced engineering concepts and other supervisory activities by process engineers has been impeded by a programming barrier. Becoming familiar with computer language is only one of the problems. Many programming languages are inadequate for the process environment and call for roundabout programming approaches. And the effort and detail involved in incorporating new programs into the system are tedious and time consuming. As a result, managements have had to either forego highly profitable supervisory benefits, or pay a fortune in programming man-hours while waiting months or years for results.

FOX 1 software was designed to bridge the gap between concept and computer, between highly creative and profitable operating plans and their effective execution by the computer. The FOX 1 system is on line and performing while others are still being programmed.

FOX 1 software is fast. Fast because it employs easy languages that speed process programming as never before. Fast because it's advanced, automates many functions, relieves engineers of a multitude of system details. Fast because it's more fully programmed for the process environment, freeing plant personnel from writing many programs and providing a range of programming shortcuts and alternatives.

In short, FOX 1 software is fast because it liberates process engineers from a lot of programming, and simplifies the rest. The results: far less programming man-hours, fast implementation of process supervision, and fast return of its economic rewards.

FOX 1 software. Fastware.
FOX 1 SOFTWARE

A PROCESS CAPABILITY WITH EVERYTHING YOU NEED FOR SUPERVISION AND CONTROL.

FOX 1 software is complete, safe, and ahead of the art. It includes every facility you'll need for all plant activities, from total plant supervision down through supervisory control, tuning, and regulatory control. It provides immediate safe response to all supervisory and control demands. And, together with the powerful array of FOX 1 equipment features, its design advances give the FOX 1 system outstanding capability for today and tomorrow.

AN ARRAY OF TIME-SAVING PROGRAMMING FACILITIES

A strong collection of programming tools simplifies every phase of new program development, greatly accelerating implementation of new supervisory schemes, management reporting, and advanced control.

The powerful and efficient FORTRAN IV language has been greatly expanded by several convenient process extensions and such features as bit and byte handling, mixed-mode arithmetic and file management statements. In addition, a FORTRAN Optimizer improves efficiency to a level approaching assembly language programs.

Another high-level language capability is MAX, a macroprocessor that lets you define and use your own application language to speed programming and enhance communications.

The FOX 1 Assembler offers the sophistication of machine-level symbolic coding plus extensive pseudo-operators to handle extra-complex problems.

Once a new program has been written, on-line testing and debugging of the program are remarkably safe, simple, and efficient. Virtually all program implementation functions, such as linking new programs into the system, have been automated. Process engineers do less because the system does more.

In addition to all of these software features, an extremely efficient system generation package permits the user, when making plant and system changes, to restructure hardware assignments and software linkages in a fraction of the time required by other systems.

ADVANCED, PROCESS-READY SUPERVISION AND CONTROL

As soon as it arrives, the FOX 1 system is ready to control your plant, thanks to IMPAC, a powerful data base generation and control package. The requirements of your process are described on simple fill-in-the-blanks forms and loaded into the computer. This can be done off line or directly at the FOX 1 CRT console with your forms displayed on the screen. Changes to the system also are made through IMPAC's easy fill-in-the-blanks forms or displays.

FASTER COMMUNICATION VIA CRT

The highly advanced FOX 1 CRT-based console simplifies man-machine communication because of the speed with which information can be presented, comprehended, and changed. Program development tools like FORTRAN, the MAX macroprocessor, the assembler and, in particular, the background debugging system can all display their outputs on the CRT.

The CRT console also gives operators and engineers fast access to all programs and plant information stored in the computer. For the first time, a CRT console is fully supported with standard keyboard input and display-generating software for total plant supervision and control.

HIGHLY RESPONSIVE REAL-TIME OPERATION

The heart of the FOX 1 software is a flexible responsive operating system which manages all supervision and control functions via a priority-structured, foreground/background multi-programming scheme. Within this environment, new supervisory programs can be developed on line safely while at the same time plant control proceeds smoothly under constant surveillance. The operating system performs many system functions, including input/output handling, task scheduling, and event response.

The Real-Time Executive is the primary element of the operating system and allocates the major system resources: computing time, core and bulk storage, and access to I/O devices. It consists of interrupt handlers, queue handlers, clock handlers, program requests, entrances and exits, and coordinates the following operating system programs in handling system activities.

The Input/Output Control System handles operation of all peripheral I/O devices such as the teletype, line printer, typers, and card and paper tape readers and punches. These devices all operate independently, using the internationally recognized ASCII code, and can back each other up if required.

The Automatic File Manager simplifies the job of data handling. It manages all storage of, and access to, plant data and programs in bulk storage. As a result, engineers need not encumber themselves with details when storing or retrieving data. The file manager creates, modifies, and repacks files, and generates all file directories.

The Console System Software enhances man-process communications. It controls all communication between the computer and up to six independent CRT/keyboard consoles. It services demands for use of the keyboard, trend recorder and console printer, and controls all data and command transfers involving these devices.

The Background Job Processor automatically sets up sequences of tests, executions and language processing required for developing and running supervisory programs, relieving the engineer of these tasks. It manages and time shares all background activity on the basis of priorities and job statements provided by the process engineer. It also includes facilities for safely testing programs under simulated conditions.
A NEW LOOK AT YOUR PROCESS THROUGH A POWERFUL CRT CONSOLE THAT'S FULLY SOFTWARE SUPPORTED

The FOX 1 System contains a new CRT-based console that provides a window through which the engineer or operator can instantly access all plant information. In other words, supervision and control of all plant operations is centralized in a keyboard-and-display unit about the size of a television set. The console offers both alphanumeric and graphic capabilities.

USED FOR ALL FUNCTIONS

The FOX 1 console can be used with practically all supervisory and control functions as well as for program preparation and testing. Many console functions and displays have already been provided by Foxboro; in many cases, no additional console programming will be required after the system arrives at the plant.

The Console System Software and the Automatic File Manager together form a highly effective data entry and retrieval capability for supervisory calculations, management reporting, program preparation, tuning, control initiation and control changes, process scanning and alarming.

MANY DISPLAYS PROVIDED

FOX 1 console software provides a number of standard displays and display directories, including:
- Plant directories – names of units in the plant
- Process unit displays – all measurements for a unit
- Loop displays – identifies records linked together for supervision or control
- Measurement displays – e.g., all temperatures, all flows, etc.
- Supervisory and control data base record displays – records showing all data associated with each process function.
- Supervisory program directory – names of background and supervisory programs.

With the aid of this versatile display capability, plant personnel can "zoom" in on a variable or see it in context of overall plant operations – in alphanumericics and graphics, judging its effect on the entire process (see illustration at right). Display formatting is easy and very flexible, allowing the user to add his own displays through the keyboard.

EASIER PROGRAMMING

Program preparation and debugging is fast and easy when using the FOX 1 console. Its pushbutton operation and instant full-screen display permit quick retrieval and changes of single statements or entire program segments, bypassing the more complex and tedious procedures associated with conventional paper tapes or card processing.

RAPID PINPOINT DIAGNOSIS

Investigation of plant upsets is quick and effective with the CRT console. By pressing the appropriate flashing alarm button, the operator displays complete process information for a particular plant area, giving him all relationships. The specific alarming variables blink for immediate identification of the problems.

PROCESS CHANGES

The FOX 1 console also provides a fast and simple means of implementing new approaches to supervision and control or modifying existing schemes through fill-in-the-blanks displays. It's far more convenient than existing procedures because it eliminates the need for cumbersome and time-consuming card or tape input.
These four displays, just a few of the many available with the FOX 1 CRT console, illustrate the powerful communication capability programmed into the system. Plant situations can be viewed at any level – from the plant manager's overview down to the process unit – zooming in to individual measurements for immediate comprehension of process situations.
Development of a working program for supervisory and regulatory control can be a prolonged effort lasting many months and involving programming of the functions, generation of the data base, and integration of all elements into a smooth, safely operating system.

Or it can be a relatively simple, overnight task – with IMPAC.

IMPAC is a comprehensive software system for process monitoring, supervisory and regulatory control, and data base generation. Requiring only the entry of basic process information on fill-in-the-blanks forms or CRT displays, the IMPAC system does the rest:

- Data base generation.
- Scanning of analog and digital inputs over a wide range of frequencies.
- Filtering of process input data and conversion to engineering units.
- Calculation of supervisory and control outputs using either standard or user-defined algorithms.
- Real-time updating of process information on the console displays.
- Linkage of standard scan and control software to user-written programs.
- On-line modifications or additions to the control scheme.

RAPID DATA BASE GENERATION

After receiving the user’s process information, IMPAC’s data base generator handles all process input and output hardware assignments and parameter specifications, creates individual records for each scan, computation, supervisory or control function, separates them by their scanning periods, and saves them in bulk storage awaiting their use by IMPAC’s scan and control package.

Each record consists of a number of fields that contain information such as the alarm limits, input source, control or supervisory algorithm and all linkages. The records provide the information for all plant monitoring and control activities as well as for logging and display. Data base records will frequently be linked together in building-block fashion to form loops.

The data base generator also creates messages, cross reference tables, and the loop structure directory which links scan, computation, supervision and control records. It also produces all the necessary documentation: listings, core and drum location maps, reports and diagrams of individual loops, plus reports sorted by record or loop identification, instrument type, digital inputs, valve inputs, and so on.

A WEALTH OF CONTROL FACILITIES

Once the system is on stream, IMPAC’s scan and control package examines each record at its predefined frequency and performs the required functions. The scan and control package contains the following set of standard supervisory, calculation and control algorithms:

- Multiply/Divide
- Measurement sum
- Measurement select
- Integration
- Action
- Output summing
- Fan out
- Limit
- Three-term feedback
- Pure integral
- Pure proportional
- Error squared
- Ratio
- Bias
- Lead/lag
- Deadtime
- Parabolic
- Auto Select
- Bang-bang

BROAD CONTROL FLEXIBILITY

IMPAC offers great versatility in supervisory and direct control strategy. Its standard algorithmic building blocks can be combined in a variety of ways, or new algorithms can be formulated using FOX 1 FORTRAN and easily incorporated into the system.

The data structure is also characterized by great flexibility. New supervisory or control schemes can be implemented on line in a fraction of the time possible with other systems. All file organization, linkage assignments, and cross-reference modifications are handled automatically. Just display on the console screen the existing record or a blank format and you can make any change or addition you desire. You can add a new record or whole loops; delete any specified record, its linkages, and all references to it; or change any record definition or response parameter. Or you can display any record on the console and simulate the operation of a loop without producing a process output. If you want a hard copy record of changes, a full screen of information can be reproduced on the FOX 1 console printer in less than 20 seconds.
Shown here are hard-copy and console display fill-in-the-blanks IMPAC forms for process supervision, data base generation, control, and other functions such as logging and display. Though Foxboro does provide a standard format, the fill-in-the-blanks form is not restricted to this format. You can easily tailor the form to suit your own needs.

<table>
<thead>
<tr>
<th>IDENTIFICATION</th>
<th>SCAN AND ALARM INFORMATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.T.Y.P.E. = S</td>
<td><strong>B.N.A.M.E.</strong> = F.S.C.401</td>
</tr>
<tr>
<td>C.A.S.T. = 10</td>
<td></td>
</tr>
<tr>
<td>B.F.R.E.Q. = 1</td>
<td></td>
</tr>
<tr>
<td>B.UN.I.T. = T/P/H</td>
<td>B.F.A.Z.E. =</td>
</tr>
<tr>
<td>B.N.M.O.D. =</td>
<td></td>
</tr>
<tr>
<td>B.F.O.R.M. =</td>
<td></td>
</tr>
<tr>
<td>B.M.E.S.G. = HEATER STEAM</td>
<td>B.M.E.S.G. =</td>
</tr>
<tr>
<td>FLOW CONTROLLER</td>
<td>D.*</td>
</tr>
<tr>
<td></td>
<td>B.C.H.T. = FLOW MEAS.</td>
</tr>
<tr>
<td></td>
<td>M.T.A.G. =</td>
</tr>
<tr>
<td></td>
<td>B.C.M. E. =</td>
</tr>
<tr>
<td></td>
<td>B.C.A. B. =</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ANALOG INPUT INFORMATION</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>M . INP =</td>
<td>02</td>
<td>Multiplexer Address</td>
</tr>
<tr>
<td>M . WIN =</td>
<td>1</td>
<td>Gain Code</td>
</tr>
<tr>
<td>M . REL =</td>
<td></td>
<td>Relay (0) or General (1) Multiplexer Nest</td>
</tr>
<tr>
<td>M . PM N =</td>
<td></td>
<td>Allowance Code (blank = none, 1 = +3 pct, 2 = +4.8%)</td>
</tr>
<tr>
<td>M . O C D T =</td>
<td></td>
<td>Open Circuit Detection for Relay Inputs (blank = none, Y = Yes)</td>
</tr>
</tbody>
</table>

---

**GREEN VALLEY PLANT**

**JUNE 10**

**TIME 1745**

**PROCESS DISPLAY**

**9 OPERATOR'S BAM DISPLAY**

**BNAME** FSC401 **BLOCK ID**

**BMESG** HEATER STEAM FLOW CONTROLLER

**BTYPE** CONTROL **BFREQ** 10 **PROCESSING INTERVAL IN SEC**

**BINP1** FSC400 -SETPOINT SOURCE

**BINP2** FSF401 -MEASUREMENT SOURCE

**BUNIT** TP/H -ENGINEERING UNITS

**CLREF** 50.0 -LOW SETPOINT LIMIT

**CFUNC** PID -CONTROL ALGORITHM

**PBAND** 50.5 -PROPORTIONAL BAND

**RESET** 1.2 -INTEGRAL CONSTANT

**DERIV** DERIVATIVE CONSTANT

**DLAG** 0.0 -ALARM DEADBAND

**DLAG**

**BLIM** LOW OUTPUT LIMIT

**BHLIM** HIGH OUTPUT LIMIT

---

**ALARM INFORMATION**

**CHDEV** 10.0 -LOW DEVIATION LIMIT

**CHDEV** 15.0 -HIGH DEVIATION LIMIT

**CDDEV** 2.0 -ALARM DEADBAND

**BSUPE** 2 -SUPPRESS ALARMS INDIC

**BOUT** FV101 -OUTPUT ID

---
In addition to offering fully programmed software for primary supervision and control, the FOX 1 system provides an advanced programming capability featuring two powerful programming languages – FOX 1 FORTRAN and the MAX macroprocessor. These capabilities give process engineers new levels of simplicity in programming for such high-potential profit-producing functions as management reporting, adaptive control, modelling, and optimization.

FORTRAN and MAX are supported by a host of system software, discussed in following pages, which speed the incorporation of user programs into the system with unprecedented ease.

POWERFUL FOX 1 FORTRAN

Several features make FOX 1 FORTRAN the most powerful FORTRAN in process control. FOX 1 FORTRAN not only contains all of the elements of ANSI standard FORTRAN IV, it also includes real-time extensions for the process control environment: mixed-mode arithmetic, bit and byte manipulation, scaled-fraction data, alphanumeric statement labels, automatic data conversion, a system COMMON area, and file access statements – all enable the system to handle the many different types of data and calculations encountered in process supervision and control. It gives an engineer all the tools he needs to implement the most sophisticated calculation techniques for analysis, simulation, or on-line optimizing.

The availability of system COMMON areas (in addition to program COMMON) and file handling statements give the FOX 1 FORTRAN user direct access to all information in the data base.

In addition to the language extensions, the FOX 1 FORTRAN library contains process input/output subroutines and calls consistent with those recommended by the Purdue Workshop on Standardization of Industrial Computer Languages.

Another very valuable feature of FOX 1 FORTRAN is its object code Optimizer, used during FORTRAN compilation to improve the efficiency of the generated code. It rearranges the coding produced by the compiler for better execution and removes any redundant steps uncovered. The resulting programs approach the efficiency of assembly language programs written by experienced programmers, minimizing both core storage and execution time.

An INCLUDE feature allows information prestored in files to be automatically inserted in programs.

Included in the System Library are the standard FORTRAN mathematical subroutines such as trigonometric functions (SIN, COS, ARCTAN, SINH, COTAN, etc.), logarithmic functions, exponential routines, and mode conversions (FIX, FLOAT, INT, REAL, CMPLUX, etc.).

MAX – A PROGRAMMING MILESTONE

Programming ease was an overriding consideration in designing FOX 1 software and one outcome of this objective was an all-new programming capability, the MAX macroprocessor. Using MAX, engineers can define and use their own plant-language statements – such as “Close Valve (V341)” or “Analyze Stream (A).” Once defined in a program, the statement can be used time and again without further definition, further reducing programming effort, and greatly improving communications between personnel.

In translating a program, the MAX macroprocessor puts the definition of the macro statement – a list of FORTRAN or assembly instructions – in a table, and then automatically inserts the instructions into the program whenever it finds the macro statement. The original program itself may combine macro statements with either FORTRAN statements or assembly language instructions for purposes of efficiency.
```assembly
72  J=IA(IP)+P
   00275 53604435
   00276 53000322
   00277 43001110
   00300 44600007
   00301 53200007
   00302 12600007
   00303 43200040
   00304 44600007
   00305 53504435
   00306 24000312
   00307 43000007
   00310 43200040
   00311 44600010
   00312 53600007
   00313 44600007
   00314 10514427
   00315 44324001
   00316 53504035
   00317 25000077
   00320 15000275
   00321 34000275
   00322 25000000
   00323 01300000
   00324 45600087

   73  C  ...  SCALING AND CONVERSION  ...
   00354 53604427
   00355 12609986
   00356 44314001
   00357 53504425
   00358 25000000
   00359 34000077
   0035a 44600007
   0035b 53504405
   0035c 25000000
   0035d 12600000
   0035e 44600010
   0035f 53600007
   00360 44600007
   00361 10514427
   00362 44324001
   00363 53504035
   00364 25000077
   00365 34000275
   00366 25000000
   00367 01300000
   00368 45600087

   73  C  ...  SCALING AND CONVERSION  ...
   003ba 53604427
   003bb 12609986
   003bc 44314001
   003bd 53504425
   003be 25000000
   003bf 34000077
   003c0 44600007
   003c1 53504405
   003c2 25000000
   003c3 12600000
   003c4 44600010
   003c5 53600007
   003c6 44600007
   003c7 10514427
   003c8 44324001
   003c9 53504035
   003ca 25000077
   003cb 34000275
   003cc 25000000
   003cd 01300000
   003ce 45600087
   003df 53333777
   003e0 13323880
   003e1 44233222
   003e2 44233222
   003e3 44233222
   003e4 44233222
   003e5 44233222
   003e6 44233222
   003e7 44233222
   003e8 44233222
   003e9 44233222
   003ea 44233222
   003eb 44233222
   003ec 44233222
   003ed 44233222
   003ee 44233222
   003ef 44233222
   003f0 44233222
   003f1 44233222
   003f2 44233222
   003f3 44233222
   003f4 44233222
   003f5 44233222
   003f6 44233222
   003f7 44233222
   003f8 44233222
   003f9 44233222
   003fa 44233222
   003fb 44233222
   003fc 44233222
   003fd 44233222
   003fe 44233222
   003ff 44233222
```

A RICH ASSORTMENT OF FEATURES MAKES FOX 1 FORTRAN EASIER, MORE EFFICIENT

An extension of ANSI-standard FORTRAN, FOX 1 FORTRAN offers several features which simplify programming for the process engineer. Some of these are illustrated here.

1. Packed data -- Use of a single memory location for several items of data, efficiently conserving computer storage.
2. Fractional variables -- For dealing directly with instrument readings.
3. System COMMON area -- Gives programs immediate access to system data.
4. Nondecimal numbers -- For expressing logical information in efficient packed forms.
5. Input/Output error labels -- Provide escape routes should a device malfunction.
6. File input/output -- Allows easy access to and manipulation of plant data in bulk storage.
7. Alphanumeric labels -- Provide the clarity and readability of names, rather than numbers, when identifying program statements.
8. DO-loop parameter expressions -- Provide flexibility for repetitive operations.
9. Mixed-mode arithmetic -- Provide flexibility for dealing with the diverse sources and types of data found in the process environment.
10. Optional deletion of CALL (before subroutine name) -- To improve readability and reduce writing.
11. Alternate subroutine returns -- Allow a routine to provide a variety of responses depending on its inputs.
Besides FOX 1 FORTRAN, IMPAC, and MAX, FOX 1 software includes a number of software packages that further simplify the program development effort.

**ASSEMBLY SYSTEM**

The FOX 1 Assembly System gives experienced programmers added capability through a rich instruction vocabulary. What's more, it not only checks for as many as 63 different error conditions, producing error diagnostic messages, but it also produces listings of the program for documentation and debugging. Twenty-nine directives or pseudo-operations including memory assignment, constant definitions and data packing are included in the assembler. The machine language output is relocatable, relieving the programmer of this task.

**COMPLETELY SAFE BACKGROUND DEBUGGING**

Once a new program or routine has been compiled or assembled, it is tested by the Background Debugging Monitor. Testing is performed under hardware and software surveillance to ensure that an undebugged program cannot inadvertently change some parameter beyond its fenced boundaries.

The program being debugged runs in the background, time-shared with other background programs. For simulation purposes, the new program can have all of its process outputs trapped and rerouted to a peripheral device. Any attempt to violate its established area is likewise trapped and prevented, and appropriate diagnostic messages are printed.

<table>
<thead>
<tr>
<th>IMPAC FILL-IN-THE-BLANKS FORMS</th>
<th>REGULATORY AND BASIC SUPERVISORY CONTROL</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONSOLE SOFTWARE IMPAC LANGUAGE</td>
<td>ADVANCED SUPERVISORY CONTROL</td>
</tr>
<tr>
<td>CHANGE MONITORING AND CONTROL STRATEGY</td>
<td>SYSTEMS ANALYSIS</td>
</tr>
<tr>
<td></td>
<td>WRITE PROGRAM</td>
</tr>
<tr>
<td></td>
<td>PREPARE SOURCE</td>
</tr>
<tr>
<td></td>
<td>COMPARE</td>
</tr>
<tr>
<td></td>
<td>TEST PROGRAM</td>
</tr>
<tr>
<td></td>
<td>LINK EDIT</td>
</tr>
<tr>
<td></td>
<td>LOAD</td>
</tr>
<tr>
<td></td>
<td>ADD NEW HARDWARE</td>
</tr>
<tr>
<td></td>
<td>FINAL SYSTEM</td>
</tr>
</tbody>
</table>

FOX 1 software is very comprehensive and provides time-saving features for every phase in the development of a program, as shown here.
Program testing can be conducted using either the engineer’s CRT console or the teletype. Hard copy output can be obtained from the line printer, the teletype, card punch, or paper tape punch. Input can be made from drum files, the CRT, the teletype keyboard, the card reader, or the paper tape reader.

Program testing usually requires displays of the memory locations occupied by the program and its associated data tables. The debugging monitor can produce these displays on line printer, teletype, card punch, paper tape punch, console printer, or the CRT. They can be expressed in ASCII, decimal integer, or single- or double-precision floating point, giving the engineer his choice of format.

The debugging monitor can stop a program as often as desired at selected locations and restart it from these or any other locations when ready. The stops can be selectively removed on line as they become unnecessary during successive executions. When problems are found, the contents of any addressable register or core location can be examined and nearly all of them can be modified.

Every instruction or just the program control instructions (branches, skips, and jumps), can be traced as they occur. The entire program or various segments of it can be traced. The trace display includes the instruction and its location, the data address and its contents before execution, and the contents of any register that changed during execution.

**LINK EDITOR**

Once tested, using the on-line debugging package, programs are easily added to the system, with the required linkages supplied by the link editor. The new program can access any data base values or have its parameters displayed and modified at the console. Existing programs and library subroutines can be used as required. The new program can be executed as the result of a scheduled request, a call from another program, a call from a point or loop being processed, or as a result of a process interrupt.

**ACCELERATED SYSTEM REGENERATION**

System generation is first accomplished at the time a system is installed and establishes the basic software modules. Later, however, plant expansion or major modification may dictate generation of a new software organization for greater efficiency. With most computer systems, this operation requires going off line and off control for as long as 24 to 36 hours. And with some computers, regeneration may require going to a vendor’s computer center far from the control site. The FOX 1 system generator is automatic, efficient, and can perform on line in the FOX 1 system without disturbing the control of the process. Only at the moment of changeover to the new configuration must the FOX 1 be taken off line and then only for a very short time.
## FOX 1 FORTRAN STATEMENTS AND FOX 1 ASSEMBLER INSTRUCTIONS

### FOX 1 FORTRAN

**DATA TYPES**
- Integer
- Real
- Double Precision
- Scaled Fraction
- Double Fraction
- Packed Data
- Complex
- Hollerith

**SPECIFICATION STATEMENTS**
- Dimension
- Common
- Equivalence
- Type
- Implicit
- Data
- Syscommon
- Dsymbl
- Include

**ARITHMETIC, LOGICAL AND CONTROL STATEMENTS**
- Arithmetic Assignment
- Logical Assignment
- Go To
- Computed Go To
- Assigned Go To
- Arithmetic If
- Logical If
- Continue
- Do
- Pause
- Stop
- End

**INPUT/OUTPUT**
- Read, Formatted and Unformatted
- Write, Formatted and Unformatted
- Formats: Repetition Groups
- Rewind, Backspace, Endfile
- Decode, Encode

### FOX 1 ASSEMBLER INSTRUCTIONS

**LOAD GROUP**
- LDA 53 Load A Register
- LDE 52 Load E Register
- LLC 04 Load Logical Complement
- LDL 51 Load Long

**LOGICAL GROUP**
- AND 02 Logical And
- IOR 03 Inclusive Or
- XOR 05 Exclusive Or

**SHIFT GROUP**
- SHF 43 Shift
- NMS 41 Normalize Short
- NML 42 Normalize Long
- RLE 40 Rotate Left E Register

**FIXED POINT GROUP**
- ADD 10 Add To A Register
- ADL 11 Add Long To A, E Register
- SUB 12 Subtract From A Register
- SBL 13 Subtract Long From A, E Register
- MPY 14 Multiply
- DIV 15 Divide

**FLOATING POINT GROUP**
- FAS 30 Floating Add Short
- FSS 31 Floating Subtract Short
- FMS 32 Floating Multiply Short
- FDS 33 Floating Divide Short
- FAL 34 Floating Add Long
- FSL 35 Floating Subtract Long
- FML 36 Floating Multiply Long
- FDL 37 Floating Divide Long

**STORE GROUP**
- STA 44 Store A Register
- STE 46 Store E Register
- STL 45 Store Long A, E Register
- SNR 47 Store Normalized And Rounded
- EAM 54 Exchange A Register With Memory
- MST 55 Masked Store
- DEM 56 Decrement Memory

**BRANCH GROUP**
- BRU 22 Branch Unconditional
- BRN 23 Branch If Register A Is Negative
- BRZ 24 Branch If Register A Is Zero
- BSP 26 Branch And Save Place
INDEX REGISTER GROUP
LXA 60 LoadXA
LXB 61 LoadXB
SX A 62 StoreXA
SX B 63 StoreXB
AX A 64 AddToXA
AX B 65 AddToXB
CXA 66 CompareXAAndSkip
CXB 67 CompareXBandSkip
TIA 70 TestAndIncrementXA
TIB 71 TestAndIncrementXB
BDA 72 BranchAndDecrementXA
BDB 73 BranchAndDecrementXB

TWO-WORD GROUP
MOV 50 MoveMultiple
BIT 07 BitManipulation
CWM 20 CompareWithMemory
BSR 25 BranchAndSaveRegion

MISCELLANEOUS GROUP
RFI 16 ReturnFromInterrupt
GEA 01 GenerateEffectiveAddress
PIO 21 ProgrammedInput-Output
SPL 17 SetPriorityLevel
HLT 00 Halt
BYT 06 ByteManipulation

SHF GROUP
ASSEMBLED AS SHF INSTRUCTIONS
WITH LITERAL ADDRESSES
ALS ArithmeticLeftShortShift
ALL ArithmeticLeftLongShift
ARS ArithmeticRightShortShift
ARL ArithmeticRightLongShift
LLS LogicalLeftShortShift
LLL LogicalLeftLongShift
LRS LogicalRightShortShift
LRL LogicalRightLongShift
RLS RotateLeftShort
RLL RotateLeftLong
RRS RotateRightShort
RRL RotateRightLong

BIT GROUP
THESE MNEMONICS SPECIFY THE OP CODE
AND THE SUB-OP CODE
SKS Skip If Bit Set
SKR Skip If Bit Reset
SKU Skip Unconditional
BIT Bit No-Op
SKSS Skip If Bit Set And Set Bit
SKRS Skip If Bit Reset And Set Bit
SKUS Skip Unconditional And Set Bit
SBIT Set Bit
SKSR Skip If Bit Set And Reset Bit
SKRR Skip If Bit Reset And Reset Bit
SKUR Skip Unconditional And Reset Bit
RBIT Reset Bit
SKSC Skip If Bit Set And Complement Bit
SKRC Skip If Bit Reset And Complement Bit
SKUC Skip Unconditional And Complement Bit
CBIT Complement Bit

CWM GROUP
THESE MNEMONICS SPECIFY THE OP CODE
AND THE SUB-OP CODE
BZEQ Branch If Zero Is Equal To Memory
BZNE Branch If Zero Is Not Equal To Memory
BZGT Branch If Zero Is Greater Than Memory
BZLT Branch If Zero Is Less Than Memory
BZGE Branch If Zero Is Greater Than Or Equal To Memory
BZLE Branch If Zero Is Less Than Or Equal To Memory
BZSE Branch If Zero Sign Equals Memory Sign
BAEQ Branch If A Is Equal To Memory
BANE Branch If A Is Not Equal To Memory
BAGT Branch If A Is Greater Than Memory
BALT Branch If A Is Less Than Memory
BAGE Branch If A Is Greater Than Or Equal To Memory
BALE Branch If A Is Less Than Or Equal To Memory
BASE Branch If A Sign Equals Memory Sign
BEEQ Branch If E Is Equal To Memory
BENE Branch If E Is Not Equal To Memory
BEGT Branch If E Is Greater Than Memory
BELT Branch If E Is Less Than Memory
BEGE Branch If E Is Greater Than Or Equal To Memory
BELE Branch If E Is Less Than Or Equal To Memory

16
<table>
<thead>
<tr>
<th>BESE</th>
<th>Branch If E Sign Equals Memory Sign</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLEQ</td>
<td>Branch If A, E Is Equal To Memory</td>
</tr>
<tr>
<td>BLNE</td>
<td>Branch If A, E Is Not Equal To Memory</td>
</tr>
<tr>
<td>BLGT</td>
<td>Branch If A, E Is Greater Than Memory</td>
</tr>
<tr>
<td>BLLT</td>
<td>Branch If A, E Is Less Than Memory</td>
</tr>
<tr>
<td>BLGE</td>
<td>Branch If A, E Is Greater Than Or Equal To Memory</td>
</tr>
<tr>
<td>BLLE</td>
<td>Branch If A, E Is Less Than Or Equal To Memory</td>
</tr>
<tr>
<td>BLSE</td>
<td>Branch If A, E Sign Equates Memory Sign</td>
</tr>
<tr>
<td>SZEQ</td>
<td>Skip If Zero Is Equal To Memory</td>
</tr>
<tr>
<td>SZNE</td>
<td>Skip If Zero Is Not Equal To Memory</td>
</tr>
<tr>
<td>SZGT</td>
<td>Skip If Zero Is Greater Than Memory</td>
</tr>
<tr>
<td>SZLT</td>
<td>Skip If Zero Is Less Than Memory</td>
</tr>
<tr>
<td>SZGE</td>
<td>Skip If Zero Is Greater Than Or Equal To Memory</td>
</tr>
<tr>
<td>SZLE</td>
<td>Skip If Zero Is Less Than Or Equal To Memory</td>
</tr>
<tr>
<td>SZSE</td>
<td>Skip If Zero Sign Equates Memory Sign</td>
</tr>
<tr>
<td>SAEQ</td>
<td>Skip If A Is Equal To Memory</td>
</tr>
<tr>
<td>SANE</td>
<td>Skip If A Is Not Equal To Memory</td>
</tr>
<tr>
<td>SAGT</td>
<td>Skip If A Is Greater Than Memory</td>
</tr>
<tr>
<td>SALT</td>
<td>Skip If A Is Less Than Memory</td>
</tr>
<tr>
<td>SAGE</td>
<td>Skip If A Is Greater Than Or Equal To Memory</td>
</tr>
<tr>
<td>SALE</td>
<td>Skip If A Is Less Than Or Equal To Memory</td>
</tr>
<tr>
<td>SASE</td>
<td>Skip If A Sign Equates Memory Sign</td>
</tr>
<tr>
<td>SEEQ</td>
<td>Skip If E Is Equal To Memory</td>
</tr>
<tr>
<td>SENE</td>
<td>Skip If E Is Not Equal To Memory</td>
</tr>
<tr>
<td>SEGT</td>
<td>Skip If E Is Greater Than Memory</td>
</tr>
<tr>
<td>SELT</td>
<td>Skip If E Is Less Than Memory</td>
</tr>
<tr>
<td>SEGE</td>
<td>Skip If E Is Greater Than Or Equal To Memory</td>
</tr>
<tr>
<td>SELE</td>
<td>Skip If E Is Less Than Or Equal To Memory</td>
</tr>
<tr>
<td>SESE</td>
<td>Skip If E Sign Equates Memory Sign</td>
</tr>
<tr>
<td>SLEQ</td>
<td>Skip If A, E Is Equal To Memory</td>
</tr>
<tr>
<td>SLEN</td>
<td>Skip If A, E Is Not Equal To Memory</td>
</tr>
<tr>
<td>SLGT</td>
<td>Skip If A, E Is Greater Than Memory</td>
</tr>
<tr>
<td>SLLT</td>
<td>Skip If A, E Is Less Than Memory</td>
</tr>
<tr>
<td>SLGE</td>
<td>Skip If A, E Is Greater Than Or Equal To Memory</td>
</tr>
<tr>
<td>SLLE</td>
<td>Skip If A, E Is Less Than Or Equal To Memory</td>
</tr>
<tr>
<td>SLSE</td>
<td>Skip If A, E Sign Equates Memory Sign</td>
</tr>
<tr>
<td>TWBE</td>
<td>Three Way Branch On E Minus Memory</td>
</tr>
<tr>
<td>TWBL</td>
<td>Three Way Branch On A, E Minus Memory</td>
</tr>
<tr>
<td>TWBA</td>
<td>Three Way Branch On A Minus Memory</td>
</tr>
</tbody>
</table>

**PROGRAMMED I/O (PIO)**

**ASSEMBLED AS PIO INSTRUCTIONS WITH LITERAL ADDRESSES**

<table>
<thead>
<tr>
<th>RDA</th>
<th>Read Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Literal Address = Octal 03000 + Device Address</td>
</tr>
<tr>
<td>WDA</td>
<td>Write Data</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 01000 + Device Address</td>
</tr>
<tr>
<td>RDSK</td>
<td>Read Data And Skip</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 03400 + Device Address</td>
</tr>
<tr>
<td>WDSK</td>
<td>Write Data And Skip</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 01400 + Device Address</td>
</tr>
<tr>
<td>RST</td>
<td>Read Status</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 07000 + Device Address</td>
</tr>
<tr>
<td>RSTC</td>
<td>Read Status And Clear</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 13000 + Device Address</td>
</tr>
<tr>
<td>RSTCSK</td>
<td>Read Status And Clear And Skip</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 13400 + Device Address</td>
</tr>
<tr>
<td>WST</td>
<td>Write Status</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 05000 + Device Address</td>
</tr>
<tr>
<td>RILS</td>
<td>Read Interrupt Level Status</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 00000 + Device Address</td>
</tr>
<tr>
<td></td>
<td>Device Address .LT.24</td>
</tr>
<tr>
<td>CIO</td>
<td>Initiate Channel 1-0</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 00000 + Device Address</td>
</tr>
<tr>
<td>CISOK</td>
<td>Initiate Channel 1-0 And Skip</td>
</tr>
<tr>
<td></td>
<td>Literal Address = Octal 00400 + Device Address</td>
</tr>
</tbody>
</table>

**MISCELLANEOUS GROUP**

<table>
<thead>
<tr>
<th>NOP</th>
<th>No-Op</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Assembled As A BRU to PC + 1</td>
</tr>
<tr>
<td>RFS</td>
<td>Return From Subroutine</td>
</tr>
<tr>
<td></td>
<td>Assembled As A BRU With A Literal Address Of 0</td>
</tr>
</tbody>
</table>
## CHALLENGE:

Compare these Fox 1 Software Features with any other process software.

<table>
<thead>
<tr>
<th>Package or Subsystem</th>
<th>Functions</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job Processor</td>
<td>Job stacking. Operator messages. I/O assignments for background programs.</td>
<td>Control of background program execution without operator intervention. Interface with operator through teletype or CRT console.</td>
</tr>
<tr>
<td>Console Software</td>
<td>Keyboard input. Display generation. Standard displays. New display definition.</td>
<td>Interfaces with all other software packages to provide most effective man-machine interfacing. Plant information, control schemes, system status, language processing, debugging messages can all be displayed on the CRT.</td>
</tr>
<tr>
<td>IMPAC</td>
<td>Data base generation. Basic supervisory and control action including data acquisition, calculation, basic supervision, and control. In-line addition, modification or deletion of any part of the control scheme.</td>
<td>Employs fill-in-the-blanks forms on paper or CRT screen. Modifiable format for application tailoring. 19 calculation, supervision and control algorithms. Extensive self-checking features. Ability to add new algorithms easily. Linkage of records in building-block fashion to form loops.</td>
</tr>
<tr>
<td>Package or Subsystem</td>
<td>Functions</td>
<td>Characteristics</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FOX 1 FORTRAN</td>
<td>All ANSI standard FORTRAN IV features.</td>
<td>Standardization of ANSI FORTRAN plus extensions to make it easy for the process engineer to express solutions to plant problems. Many data types to mirror the process environment and ability to mix types within calculations, avoiding lengthy conversion.</td>
</tr>
<tr>
<td>MAX</td>
<td>MACRO statements. Conditional translation.</td>
<td>Allows generation of customized language statements to express application-dependent activities naturally. MAX translates into either FORTRAN or assembly language.</td>
</tr>
<tr>
<td>Assembler</td>
<td>29 directives for memory assignment, data packing, constant definitions, etc. Complete listings and cross-references.</td>
<td>Intended for the experienced programmer who finds it desirable to deal directly with the language of the computer.</td>
</tr>
<tr>
<td>Link Editor</td>
<td>Preparation of translated program for execution.</td>
<td>Links programs and subroutines. Obtains functions from subroutine library. Finds locations of common data. Checks for and flags missing information. Prepares program in proper format for entry into memory.</td>
</tr>
<tr>
<td>System Generator</td>
<td>For revising software structure. Inserts all initial system parameters. Establishes all linkages for startup. Prepares initial hardware assignments.</td>
<td>Operates on-line. System need only be taken off line for short period for actual changeover.</td>
</tr>
<tr>
<td>Diagnostics</td>
<td>Peripheral device testing. Memory testing.</td>
<td>Available to insure proper operation of the system.</td>
</tr>
</tbody>
</table>
OPERATING COMPANIES

ARGENTINA
Foxboro Argentina S. A.
Avda. R. S. Peña 570
Buenos Aires

AUSTRALIA
Foxboro Proprietary Limited
Maroondah Highway
Lilydale, Victoria 3140

BRAZIL
Foxboro Brasileira Instrumentacao Ltda.
Caixa Postal 30.770
Sao Paulo, SP

CANADA
The Foxboro Company, Limited
707 Dollard Avenue
LaSalle, Quebec 650

CARIBBEAN
Foxboro Americana, Inc.
P. O. Box 4726
San Juan, Puerto Rico 00936

FRANCE
Foxboro France S. A.
B. P. 249
62-Arras

GREAT BRITAIN
Foxboro-Yoxall Limited
Redhill, Surrey

ITALY
Foxboro Italia S.p.A.
Via G. Fara, 39
I 20124 Milano

JAPAN
(Yokogawa Electric Works, Ltd.
2-9 Nakacho
Musashino-shi
180 Tokyo

MEXICO
Foxboro, S. A.
Apartado Postal 7-938
Mexico City 7, D. F.

THE NETHERLANDS
Foxboro (Nederland) N. V.
Koningsweg 30
Soest

UNITED STATES
The Foxboro Company
Neponset Avenue
Foxboro, MA 02035

WEST GERMANY
Foxboro Deutschland G.m.b.H.
Ross-Strasse 112
4000 Duesseldorf

Worldwide Facilities
Approximately fifty percent of all Foxboro products are sold outside of the United States. It is significant, therefore, that the same products, the same services, and the same facilities which are available to customers in the U. S. are also available to customers in other countries. Training, flow calibration, systems engineering, instruments and supplies, panel fabrication, installation and startup assistance, and numerous other Foxboro benefits are available in over 100 countries. Of the more than 125 Foxboro sales and service offices, over 70 are located outside of the U. S.