

SIGMA COMMUNICATION

SUBJECT: Product Planning Position on TSD DOC. NO.: 333
AUTHOR: Al Bongarzone DATE: 5 March 1968
DISTRIBUTION: Sigma Series Notebook Holders FILE NO.: 10

I. INTRODUCTION

This document presents the Product Planning position on the development of a minimal time-sharing system that we can release as a product using TSD as the base. Marketing and Program Development are hereby requested to respond to this memo as soon as possible so that we can proceed with more detailed planning and implementation.

II. MARKET CONSIDERATIONS

Product Planning recommends that we develop a time sharing system for field release before the end of the third quarter of 1968. The base for this product will be the current TSD system being developed for SDS by System Concepts. This recommendation is based on the following assumptions:

- A. It now appears unlikely that we will field release UTS until mid 1969 at the earliest.
- B. There is presently no other backup for the UTS development effort in the event further slips are necessitated.
- C. SDS previously announced UTS and subsequently decommitted from an earlier delivery. We need some presence in the marketplace now.
- D. There is a need for software products with sufficient "sex appeal" to assist SIGMA 7 sales. The product must run only on the Sigma 7 using the MAP, and not run on the SIGMA 5.
- E. SDS has established itself as a leader in time sharing through the 940. We are not following up this advantage rapidly enough on the SIGMA 7.
- F. Our competitors are demonstrating time sharing (DEC, IBM) and remote batch operations (IBM, UNIVAC, CDC) while we are still struggling to get BPM operating.

In summary, we need to announce and demonstrate a time sharing software system as solid evidence of our intentions to be in the time sharing business with increasingly sophisticated systems.

III. PRODUCT GOALS

The product that we release should:

- A. Be a true superset of BPM and allow for interchange of data files and processor compilations and assemblies.
- B. Be compatible at the terminal user level with the planned command languages and control characters of UTS.
- C. Allow for 8 simultaneous on-line users without seriously degrading batch throughput. Our goal is 75% of normal batch throughput with 8 users.
- D. Allow for 16 additional users (24 total) as more memory is added and as the high speed RAD (7212) is added to the equipment configuration.
- E. Provide a SYSGEN capability that is independent of minor changes and corrections to BPM and TSD so that customers can add their own processors in the field and make their own new system masters.
- F. Contain, initially, only manual recovery procedures for the computer operator. No automatic recovery procedures need be provided beyond those currently in BPM.
- G. Response time for on-line user should be less than 5 seconds with simultaneous batch operation.
- H. The system must provide Symbol, Edit and Debug capabilities; FORTRAN IV-H on-line capability; and Meta-Symbol in the on-line compatible mode.
- I. TSD is the acronym for Time Shared Debug. This name is neither descriptive nor appropriate for the initial release of a line of standard SDS time-sharing software products. The product we will release will be essentially a batch monitor with a minimal amount of time-shared capability added. This minimal capability, however, will include more than a debug facility since we herein have proposed adding to the system originally contracted for with Systems Concepts. We need a name that more nearly characterizes the product than does TSD. We propose to call the new product Batch/Time Sharing 1.0 (B/TS 1.0). This name emphasizes the batch capability and highlights time sharing without an arbitrary delimiter such as Assembler (TSA) or Debug (TSD). The differences between this product and UTS are sufficiently large in design and implementation that this product should not be called UTS-1.

IV. PRODUCT DESCRIPTION

A. MONITOR AND MONITOR SERVICES

B/TS 1.0 is an extension of the BPM system which will allow up to eight users to edit, assemble or compile and debug programs. All the above mentioned steps will be handled on-line in a multiprocessing manner with the batch monitor. The total system is primarily designed, and optimized, for batch operation. As a result, on-line users may pay a slight penalty in response time. The batch processing monitor will normally operate at no less than 75% efficiency while there are eight simultaneous on-line users.

The major component of B/TS 1.0 is the SDS Batch Processing Monitor. Principal elements of the Batch Processing Monitor are employed by B/TS 1.0 without modifications. These include CCI, SORT, the Supervisor, etc. Some changes were made to the Batch Processing Monitor. The changes are:

1. A clock routine monitor was added to the batch monitor to call B/TS at specified intervals.
2. A series of changes were made to BPM to prevent it from locking out the B/TS processor for periods that would degrade on-line response time.

The principal elements of B/TS 1.0 that have been added to BPM are:

Scheduler

The Scheduler examines each user's status in a round-robin sequence to select the next user who is ready to run. If such a user is found, the swapper is called, otherwise the scheduler returns to Batch for a quantum.

Swapper

The swapper determines whether a swap must occur before the desired user can be run. If the new user is not in core then a swap is initiated and control returns to Batch until the swap is completed.

COC Routines

The teletype interface routines buffer messages both to and from the user console on a character basis. Status information is maintained by the COC routines for use by the scheduler and the user's program. In addition it

should be noted the user program sends and receives all characters in EBCDIC and the COC routines handle the conversion to and from ASCII.

File Interface

All file manage requests initiated by a B/TS 1.0 user programs are passed to the batch monitor after the following checks have been made.

1. The syntax is legal.
2. A disc file is being used.

This facility provides a common file access for both Batch and B/TS 1.0 programs.

B. PROCESSORS

The processors currently available in the B/TS 1.0 system are Edit, Symbol and Debug. Additionally, FORTRAN IV-H and Meta-Symbol are to be made available to the terminal user in a specialized manner as described below. Brief descriptions of the processors are provided.

Edit

The Text Editor works with the BPM File Management system to update the contents of a user's symbolic file. It will employ user supplied line numbers for identification of the line to be modified or for modification referencing. The Text Editor, as described in the UTS planning specification, performs the following functions:

Create a new text file

Insert or add one or more lines at a specified location in the text file

Delete one or more lines at a specified location in a text file

Replace one or more lines at a specified location in a text file

Automatically generate sequence numbers for simple continuous text file input

Resequence a text file

Name or rename text files (via BPM)

Symbol

The Sigma 5/7 Symbol was modified for use in an on-line mode under B/TS 1.0. The modifications included symbol table generation, provision for line printer or teletype output, and allowance for diagnostic error output to the teletype. The Symbol assembler itself will not be conversational in response to terminal input, except in the case of the selection of input options. This assembler will also operate under BPM.

Debug

The Interactive Debug program is used to load, control execution of, and modify user object programs. The Debug program assumes responsibility for loading user object programs and preserving the Symbol tables. It will not be physically resident when the Symbol processor (or any non-debug program) is executing. Debug working storage for a user program will be controlled and allocated when Debug loads the user program.

The debugging functions required are those described in the TSD design specification:

Control the Start location of program execution

Dump (open) one or more locations in the user's memory space with format options

Change the contents of one or more of a user's memory location

Insert instructions within existing user code (patches)

Install or remove a breakpoint

Search for masked values within bounded areas of a user's program

Define new symbols or change symbols (labels) within a user object program

FORTRAN IV-H

FORTRAN will be used in an on-line mode under B/TS 1.0 as follows:

The user may create source programs from his terminal using the Text Editor. Alternatively, an existing source program from cards may be used by submitting the deck as a job to be run under BPM. In either case, the source file created resides in permanent file storage. The user can then call from a terminal and have the FORTRAN subsystem compile the source file as a time-sliced on-line operation. The FORTRAN IV-H compiler will be modified to permit I/O to and from the Teletype, and user-program read and write run-time routines for the teletype. The compiler will output to the teletype only diagnostic messages with associated source lines rather than the complete listing, and will accept only input options a-la SYMBOL.

The compiler generates an output file of the object program. If no errors are found, this file can then be called by the B/TS user to execute. This will cause the loader to load the user program and make a library search for all subroutines required for program execution. Included in the run-time package will be the FORTRAN IV-H debug routine which will be modified to allow control from the teletype, and debug output to go to the teletype. Any changes required in the source code of a FORTRAN program will require source file editing and a new compilation.

Meta-Symbol

The use of Meta-Symbol under B/TS 1.0 will be in an on-line compatible mode. Meta-Symbol will execute only in the Batch mode under BPM. However, the Meta-Symbol source file can be created on-line by the B/TS 1.0 Text Editor, or by Batch input or source cards, paper tape or magnetic tape.

The Meta-Symbol object program output will be stored in permanent file storage on the RAD, where it will be accessible to the B/TS 1.0 terminal user. The user will know when this condition has been achieved either by a normal job request completion report from the batch operation, or by calling for the loading of his new object program by file name from the terminal through B/TS 1.0. If the program has not been assembled, the file name used will not be found and his request will be rejected.

Once the user's Meta-Symbol object program is in permanent file storage, the user can call for loading it under the Debug subsystem for execution, debugging and modification. When reassembly is appropriate, the user will employ the on-line text editing function (or off-line card changes) to create a modified source program and submit control cards for a batch job run to the Data Center.

V. MINIMUM EQUIPMENT CONFIGURATION

<u>MODEL NUMBER</u>	<u>QUANTITY</u>	<u>DESCRIPTION</u>
8401	1	Sigma 7 CPU
8411	1	Two Additional Real-Time Clocks
8413	1	Power Fail Safe
8414	1	Memory Protect
8415	1	Memory Map
8416	1	Additional Register Block
8418	1	Floating Point Arithmetic
8421	1	Interrupt Control Chassis
8422	1	Priority Interrupt, 2 levels
8451	3	Memory Module, 4096 words
8452	9	Memory Increment, 4096 words
8471	1	Multiplexor IOP
7010	1	Keyboard/Printer, KSR/35
7120	1	Card Reader, 400 CPM
7201	1	RAD Controller
7204	2	RAD Storage Unit, 3.0 MB
7321	1	Magnetic Tape Controller
7322	2	60 KB Magnetic Tape Unit
7440	1	Line Printer, 600 lpm
7611	1*	Communications Controller
7612	1**	Format Group Timing Unit
7615	1***	Send Module
7616	1***	Receive Module

ADDITIONAL EQUIPMENT FOR 24 USERS

8451	1	Memory Module, 4096 words
8452	3	Memory Increment, 4096 words
8456	4	Three way Access
8485	1	SIOP
7211	1	RAD Controller
7212	1	RAD Storage Unit, 6 MB

*Controls up to 8 terminals

**One required for each class of terminal

***One required for each terminal

VI. DEVELOPMENT PLAN

Ideally, the first time shared system we release should be the initial step in a continuum of T-S monitor products. There is presently a discontinuity between TSD and UTS especially in the area of the command language and how the system is seen by the terminal user. TSD must be modified to insure compatibility at the user level with UTS. The changes to TSD will have to be made by Mike Levitt or someone from SDS working closely with him. Areas that require change include; Executive commands, Control functions, Edit functions and Debug.

A. PROGRAMMING

Program Development must immediately begin to estimate the manpower requirements and time schedules for modifying TSD into B/TS 1.0. Further discussions between System Concepts, Program Development and Product Planning will be required to define the precise nature and magnitude of the changes. These discussions must start as soon as possible, using the following as guidelines and starting blocks.

1. System Generation

It is still difficult to generate a system for TSD, particularly in the face of BPM modifications. The SYSGEN procedure for TSD must be smoothed out so that customers can make field modifications and additions to B/TS 1.0 easily.

2. System Recovery and File Maintenance

Appropriate procedures for file maintenance and for recovery from system failures must be established. Manual intervention and the recovery procedures of BPM are probably the most reasonable for the initial product release.

3. System Performance

It is advisable to cut down on the number of swaps and processor actions required to handle console requests, and thus get some insurance to guarantee our response-time and batch-degradation estimates. Some possible strategies are:

- a. Abbreviate the break-set of DEBUG to curtail the number of swaps;
- b. Abbreviate the break-set of EDIT (with particular attention to the tab function) to curtail the number of swaps;
- c. Cut down the average swap time by swapping only that portion of the user's area actually in use -- this may be difficult to do in general, but seems feasible for users who are editing or debugging.

4. System Implementation and Maintenance

It is advisable that B/TS 1.0 permit DEBUG to be available in "master" mode to facilitate system implementation.

5. System Processors

- a. FORTRAN IV-H and Meta Symbol are to be modified as described above (pages 6 & 7). Many of the changes required to handle these processors and their associated object packages will carry over (except in fine detail) to UTS. They should, therefore, be done with some care. A decision on how FORTRAN IV-H object programs (and library routines) will be handled must be reached. They could either be loaded and debugged under the control of DEBUG, or loaded at the executive level and debugged via the FORTRAN IV-H run-time debugging routines. The latter tack conforms to general practice and is more realistic if the run-time debugging package can be used. In any event, it is advisable to have the loader (currently under the control of DEBUG in TSD) be available at the executive level in B/TS 1.0.
- b. DEBUG should be modified to operate with an abbreviated set of break characters. The extent and nature of the changes will be ironed out with System Concepts.
- c. EDIT differs from the proposed UTS editor in that both an input and an output file are required. This approach was the simplest for interfacing with BPM, and should not be abandoned at this stage. However, it would be an act of conformity to have the ASSIGN commands available at the EDIT level. To minimize EDIT swaps, the tab, character-erase, and line-erase functions should be extracted from EDIT or handled in some resident portion of either EDIT or the system.

The manner in which EDIT handles its command verbs should be changed so that only two forms of a command are recognized -- the full word itself, and an abbreviated form (a single letter in general usage). The dollar-sign, rather than a null character, should be used to denote "current line number", in conformity with SYMBOL, META SYMBOL and DEBUG. Discussions with System Concepts are required to determine which of the control functions and verbs can be modified to conform with UTS. In the event that conformity is not feasible, the "literal next" control function and the verb FILE may have to be expunged to prevent UTS-incompatibilities from being frozen into the files.

- d. System Control Functions: An austere set of control functions and an associated modus vivendi for UTS are described in PR-68-7016. B/TS 1.0 must conform to these conventions particularly with respect to the interrupt, monitor escape, character erase, and line erase functions.
- e. System Executive Commands: TSD currently recognizes the first of two characters of executive command-verbs, types out the remainder of the verb and awaits a period. This protocol is awkward and sometimes annoying. It must be changed to conform to the general UTS philosophy of handling commands: verbs are recognized in either their complete form or in an abbreviated form by the executive and by all processors except those that are, by nature, interactive on a character basis. The command repertoire may be increased if the loading and debugging of FORTRAN IV-H programs is made available at the executive level or via a new sub-executive. A facility for "dumping" and reloading core images under appropriate identifications should be included in the repertoire.

B. DOCUMENTATION

Documentation required for B/TS 1.0 includes:

Functional Specifications

Technical and Maintenance Manual

User's Reference Manual

C. COMPUTER TIME

It is felt that current computer availability is insufficient for meeting important product schedules such as UTS. There are a large number of jobs competing for available time. It is clear that a second computer is needed almost immediately. Also needed for checkout is the high speed RAD (7212) and its controller (7211). If a decision is made to proceed with the development of B/TS 1.0 as a releasable product, sufficient computer time must be made available immediately for checkout if the scheduled release date is to be met.

D. SCHEDULE ADJUSTMENTS

In terms of relative priority in the processor area, Product Planning feels that the modifications to FORTRAN IV-H for on-line operation are more critical than the development of the 8K FORTRAN. If there is a tradeoff of manpower resources required, the FORTRAN IV-H effort should get the resources.

It is recognized that Program Development has limited resources to apply to the time sharing monitors. Part of the response to this document should be an assessment of the impact that the development of B/TS 1.0 will have on other developments such as the BPM improvements and UTM. In the monitor area the order of priority should be BPM improvements, B/TS 1.0, and UTM in that order. If it is at all possible we should not lose the momentum and enthusiasm which has been built up in the UTM effort. But it must be recognized that we may have to take a short term setback in that effort in order to bring about the market improvement to be derived from this earlier release of B/TS 1.0.