WANG SYSTEMS NEWSLETTER NO.3

JULY, 1977

THINGS ARE STILL CHANGING

As mentioned in previous issues, the System Support Group's growth is based upon the growing marketplace and technology, as well as corporate requirements. We would, therefore, like to introduce the latest addition to our staff.

Charles Miller — System Consultant
Responsible for National Account technical sales support and technical information (proposal/continuing support).

APPLICATION BULLETINS

The following are new Application Bulletins released.

NUMBER	SUBJECT
21	T/C Support Utilities #1
22	EASYFORM

PRODUCT BULLETINS

The following are new Bulletins released.

NUMBER	SUBJECT
137	2228B and Option 62B Communications Controller
138	2270A IBM-Compatible Diskette Drive
139	2271 Printer
140	2281 Printer
141	2261W Line Printer
142	2263 Line Printer
143	PCS-II
144	2236, Interactive Terminal and 2236MXC (Multiplex Controller)
145	WCS/40
146	WCS/25

WANG LABORATORIES, INC.
ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL (617) 851 4111, TWX 710 343 6769, TELEX 94-7421

Printed in U.S.A
700-3137C
7-77-1.3M

MANUALS

The following is a list of new and reprinted manuals.

NEW MANUALS

Introducing LIFELINE (700-4155)
GBS – Mods I and II (Diskette) Operator's Manual (700-4163)
GBS – Mods I and II (Disk) Operator's Manual (700-4204)
Introducing GBS (700-4186)
2200 – Management Planning System User Manual Diskette (700-4185)
2200 – Liquid Scintillation Data System (LSDS) (Cassette) Operator's Manual (700-4173)

REPRINTS

WCS/20 – Introduction and Installation Guide (700-3854A)
2231W – Line Printer Data Sheet (700-3821D)
2250 – I/O Interface Controller Data Sheet (700-3253C)
2216/2217 – CRT Display/Cassette Drive Data Sheet (700-3150C)
2200 – Sequential Analysis Operator's Manual (700-3578B)
2272 – Digital Drum Plotter User Manual (700-3869A)
GBS – Mods I and II (Diskette) Technical Guide (700-4072A)
Literature and Price List (700-3761C)
2200 – Diskette Demos Operating Instructions (700-3625A)
2262 – X-Y Digitizer User Manual (700-3735A)
2200 – BASIC Language Pocket Guide (700-3030D)

ERRATA SHEET

As software enhancements and/or problems are encountered, an Errata Sheet is released. As we receive the information, it will be published.

GBS – Mod I (Disk and Diskette System Manual – Update #1 (700-4142)
GBS – Update #2 (700-4207)

DID YOU KNOW?

- THAT TAM subroutines can be run on a "C" CPU. Option 2 is required and if you use the logging subroutine, Option 5 is required.
- THAT in a "T" CPU, variables used most often should be defined last in the body of the program. The search time through each item in the variable stack is 64 microseconds. To speed up any program, search time can be reduced by judicious ordering of the variable stack.
- THAT in the "T" CPU, statement lines referenced most often should be placed near the front of the program. Search time for numbered statements required about one millisecond for every 24 numbered statement lines, counting from the top of the program.
- THAT spaces within the program body consume execution time. 25 spaces require .44 milliseconds.
- THAT 2200A, B, C, S and T CPU had a 1.6 microseconds read/write cycle.
- THAT ERR 02 frequently occurs as a result of software problems in the subroutine/FOR loop stack. This stack is limited to any combination of 50 special functions, for/next loops or subroutine entries. The space required for each is:

Each Nested FOR/NEXT loop	–	20 Bytes
Each Nested Subroutine	–	4 Bytes
Special Function Entries	–	4 Bytes

The important thing to remember is that this information is cleared from this stack only when a subroutine return is made or the FOR/NEXT loops are completed.

The use of the RETURN CLEAR statement will cancel the last subroutine called when an exit from the routine is required before encountering the return statement. This will help to avoid ERR 02.

- THAT in order to clear the FOR/NEXT loop stack for an exit prior to completion, the example below can be used.

```
10 FOR X = 1 TO 10
20 1F X = 5 THEN 40
30 NEXT X
40 Y=X:X=10:  NEXT X:  X=Y (then continue processing)
```

- THAT the "VP" CPU can be more accurate and can produce different results in numeric calculations than the "T" CPU. This can be a problem in using a hashing technique for locating sector addresses numerically.

  Solution: calculate hashing alphanumerically, and the results will be the same on all Wang CPU's — regardless of the system.

  The program used for the disk index is an example:

```
10 DIM N$8,L$1,H$2
20 PRINT
30 INPUT "NO OF INDEX SECTORS", S
40 INPUT "NAME", N$
50 XOR (STR(N$,2), N$   :REM -- EXCLUSIVE OR EACH BYTE OF NAME
60 L$=STR(N$, 8, 1)     :REM -- L$=EXCLUSIVE OR OF EACH BYTE
70 H$=HEX(0000)
80 ADDC(H$,L$): ADDC(H$,L$): ADDC(H$,L$)  :REM -- H$=3*L$
90 ADD(STR(H$,1,1), STR(H$,2,1))  :REM -- ADD 2 BYTES OF H$
100 H=VAL(H$)             :REM -- CONVERT RESULT TO NUMERIC
110 H=H-INT(H/S)*S        :REM -- REDUCE HASH VALUE (NO OF
120 PRINT "HASHES TO SECTOR", H
:RUN

NO OF INDEX SECTORS? 24

NAME? JOHN

HASHES TO SECTOR 9
```

- THAT because of the technique used to assign a location in the index sector, the "LS" value should not be equal to the number 3 of any factor of the number 3. By using other values, you gain more efficient use of the index.

WANG SLANG

ASCII

American Standard Code for Information Interchange.
A seven bit coding system which is used in paper tape as well as telecommunication. It is a standard in the industry with the exception of certain manufacturers. An optional eighth bit is used for parity.

CHAINING

When the CPU cannot contain all programs required for execution, through a series of systems, required modules are chained and automatically called into the CPU. Segments, total programs, or routines can be chained based on the command. (See Basic Programming Manual.)

CORE
A tiny doughnut of magnetic material. Of course, the Wang 2200 CPU's uses MOS chips for memory, but the term is used in conjunction with the working or temporary storage of the CPU.

MICROSECOND
$1/10^6$ seconds or one millionth of a second – μs

MILLISECOND
$1/10^3$ seconds or one thousandth of a second – MS

ROM
Read Only Memory – that portion of the CPU which is hardwired, containing such things the BASIC instruction set, trigonometric functions, interpreter, etc., the contents of which cannot be altered.

BITS AND BYTES

RPL CHANGES

The following changes to RPL will enable it to run on either a 2200T or 2200VP. The problem is that when the locator array produced by MAT SORT, sorting a one dimensional array, is referenced by other than MAT MOVE, the results are different. Perhaps the simplest way to resolve this incompatibility is to make the array being sorted two-dimensional. But in the case of RPL, the arrays being sorted are referenced throughout the compiler, and it would be difficult to change every reference to two dimensions. Therefore, the following changes are made to reference the first byte of the locator array on a 2200T, or the second byte on a 2200 VP.

The common variable S3 is added in the RPL compiler, consistent with ISS usage. S3 = 2 indicates a 2200T, and S3 = 3 indicates a 2200VP.

MODULE	SOURCE PROGRAM LINE	COMPR. PROGRAM LINE	CHANGE
RPL	60	60	Add S3, after Q1$8
	5052	5050	Add MAT SORT C2$() TO T1$(), T2$()
	5054	5050	Add S3 = 3 – VAL (T2$(1))
RLS306BA	7170	7170	Change VAL(D9$(D7)) to VAL(STR(D9$(D7), S3-1))
RLS307BA	5850	5830	Change VAL(D9$(D7)) to VAL(STR(D9$(D7), S3-1))
	6850	6830	Change VAL(D9$(D7)) to VAL(STR(D9$(D7), S3-1))
RLS309AA	2190	2170	Change VAL(BØ$(E5)) to VAL(STR(BØ$(E5), S3-1))
RLS311AA	590	200	Change VAL(S3$(P)) to VAL(STR(S3$(P), S3-1))
	1660	1640	Change VAL (S3$(1)) to VAL(STR(S3$(1), S3-1))
	1700	1690	Change VAL (S3$(D)) to VAL (STR(S3$(D), S3-1)A

LIMITS

There appears to be some confusion regarding the LIMITS statement. This is written to alleviate some of that confusion and, hopefully, to prevent future problems.

Consider a disk with one index sector (sector zero) and a data file from sectors one to five. Sectors one, two, and three have single sector records in them and sector four has a software trailer record created by a DATASAVE DC END statement.

If we open the file and do a LIMITS using device number preceded by a D SKIP END, the values assigned to the variables in the LIMITS would be: Starting equals 1 (actual sector address); Ending equals 5 (actual sector address); and the third variable would be current sector address equal to 4 (actual sector address).

If a LIMITS is done after opening the file, but not preceded by a D SKIP END, the Starting would equal 1, the Ending would equal 5, but the current sector address would equal 1.

These two LIMITS statements read the device table in memory, not the disk.

The other form of LIMITS is done by not using the device number but instead the file name. This form of the LIMITS reads the disk index, not the device table in memory. The values read from the disk are placed into the device table, destroying the current values in the device table.

Using the same example as above, open the file and do a LIMITS with the file name: The values entered into the variables would be: Starting equals 1; Ending equals 5: and Used equals 5. Several problems arise with this form of LIMITS. One problem is the destruction of the values in the device table. A way around this problem is to open the same file using a different device number. A second problem arises if no DATASAVE DC END was executed. In this case, the catalog index would not be updated. The result of this would be that the third value in the LIMITS, Used, would always equal 1. A solution to this problem is to open the same file using a different device number. Then do a LIMITS using this new device number and the file name. This will destroy the values in the dummy slot but preserve the values in the actual device table slot.

A good programming practice is to always put a software trailer record at the end of the valid data records.

A more important note to this is that all of these figures are on a model "T" CPU. The same things do not hold true for the "VP" processor.

On the "VP" processor the LIMITS statement using the file name is not a destructive LIMITS, as it is on the "T" processor.

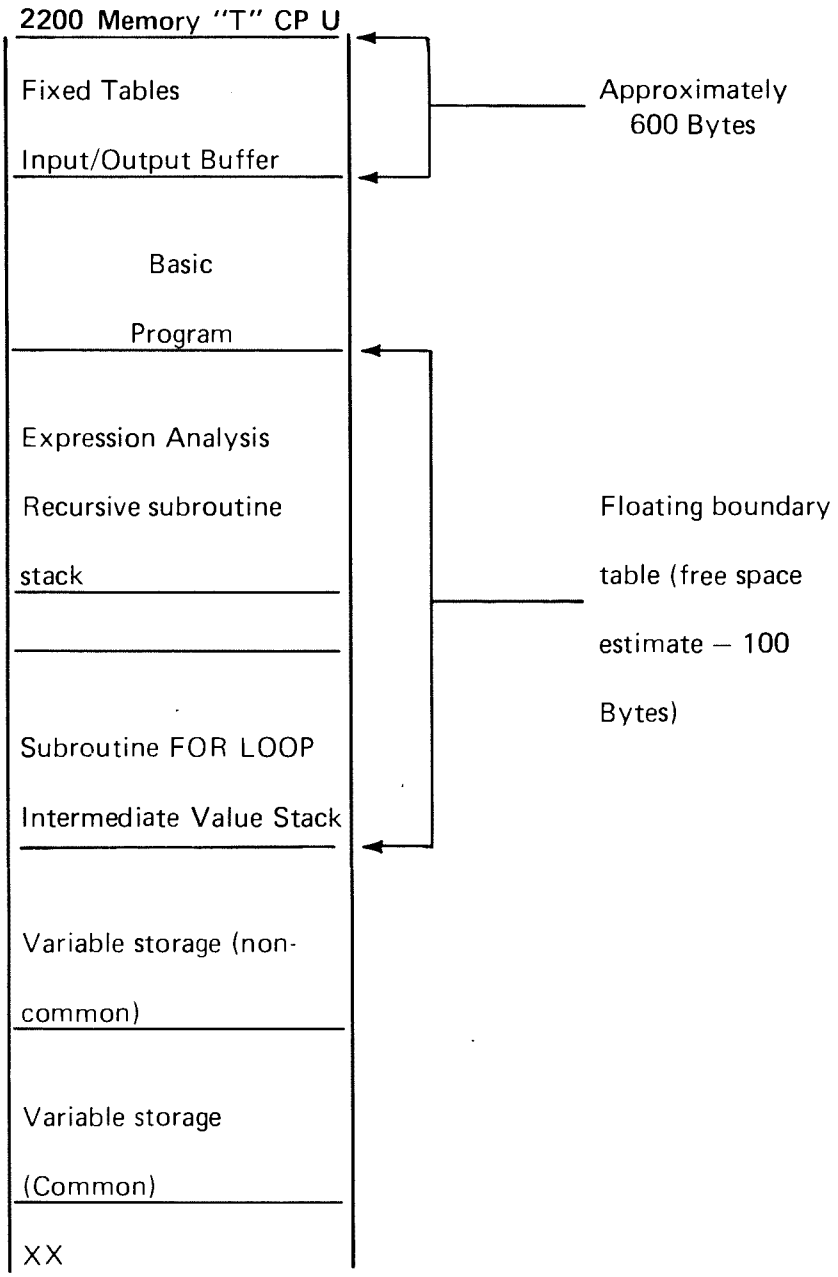
WANG POLICY

The following policy statement reflects the current philosophy of Wang Laboratories, Inc. in regard to use of non-Wang supplies with Wang equipment.

> Wang Laboratories, Inc. cannot be responsible for equipment failures and/or damages caused by supplies purchased by customers from sources other than Wang Laboratories. All supplies provided by Wang Laboratories, Inc. are produced to rigid specifications and are subject to close quality-control specifications. Specifications of supplies produced and furnished by Wang are available upon request. However, since it is impossible for Wang to exercise quality control over supplies produced by other companies, Wang cannot guarantee performance of its equipment with these supplies.
>
> It has been Wang Laboratories' experience that even various batches of supplies from other manufacturers can range in quality from satisfactory to unsatisfactory, with the latter being extremely detrimental to the equipment's operation.
>
> In the event supplies not furnished by Wang are used and a service call results from failure of the equipment to operate properly due to the poor quality of these supplies, Wang Laboratories may void all warranties made to the customer. In addition, Wang Laboratories may charge the customer for the service call (travel, labor, and parts) for the specific failure caused by the supplies. This policy also applies to customers with equipment covered by our Maintenance Agreement Contract.

"T" CPU

BASIC programs and fixed tables and buffers are stored starting from address 0 in memory upward. Variable storage is allocated starting at the highest memory address downward. When a given program is run, these two areas become fixed in size. Between the variable storage area and the BASIC program are two dynamic tables. The table starting from the BASIC program area and going upward stores recursive internal subroutine return information during mathematical expression analysis. The other stores BASIC subroutine and FOR/NEXT loop information, intermediate calculation results, and address required during the execution of BASIC statements, this table starts at the variable storage area and moves



downward. If at any time during the execution of the program, the end of one of these tables meets the end of another, an ERROR 02 (Table Overflow) will occur.

TEXT ATOMS-"T" CPU

HEX		DESCRIPTION
80	=	LIST
81	=	CLEAR
82	=	RUN
83	=	RENUMBER
84	=	CONTINUE
85	=	SAVE
86	=	LIMITS
87	=	COPY
88	=	KEYIN
89	=	DSKIP
8A	=	AND
8B	=	OR
8C	=	XOR
8D	=	TEMP
8E	=	DISK
8F	=	TAPE
90	=	TRACE
91	=	LET
92	=	DRAM
93	=	DIM
94	=	ON
95	=	STOP
96	=	END
97	=	DATA
98	=	READ
99	=	INPUT
9A	=	GOSUB
9B	=	RETURN
9C	=	GOTO
9D	=	NEXT
9E	=	FOR
9F	=	IF
A0	=	PRINT
A1	=	LOAD
A2	=	REM
A3	=	RESTORE
A4	=	PLOT
A5	=	SELECT
A6	=	COM
A7	=	PRINTUSING
A8	=	MAT
A9	=	REWIND
AA	=	SKIP
AB	=	BACKSPACE
AC	=	SCRATCH
AD	=	MOVE
AE	=	CONVERT
AF	=	PLOT(SEL.)
B0	=	STEP
B1	=	THEN
B2	=	TO
B3	=	BEG
B4	=	OPEN
B5	=	CI
B6	=	R
B7	=	D
B8	=	CO
B9	=	NOT USED
BA	=	OFF
BB	=	BACKSPACE
BC	=	VERIFY
BD	=	DA
BE	=	BA
BF	=	DC
CO	=	FN
C1	=	ABS(
C2	=	SQR(
C3	=	COS(
C4	=	EXP(
C5	=	INT(
C6	=	LOG(
C7	=	SIN(
C8	=	SON(
C9	=	RND(
CA	=	TAN(
CB	=	ARC
CC	=	#PI
CD	=	TAB(
CE	=	DEFFN
CF	=	ARCTAN(
DO	=	ARCSIN(
D1	=	ARCOOS(
D2	=	HEX(
D3	=	STR(
D4	=	ATN(
D5	=	LEN(
D6	=	RE
D7	=	#
D8	=	%
D9	=	P
DA	=	BT
DB	=	G
DC	=	VAL(
DD	=	NUM(
DE	=	BIN(
DF	=	POS(
E0	=	LS=
E1	=	ALL
E2	=	PACK
E3	=	CLOSE
E4	=	INIT
E5	=	HEX
E6	=	UNPACK
E7	=	BOOL
E8	=	ADD
E9	=	ROTATE
EA	=	$
EB	=	NOT USED
EC	=	NOT USED
ED	=	NOT USED
EE	=	NOT USED
EF	=	NOT USED
FO	=	EDIT KEY
FF	=	STMT ATOM

"HEAT"

Did you ever notice how warm it gets when there is an operating WCS/30 system in a room that has only 100 square feet of area. The reason is that the system does generate some heat. Listed below are some figures for your reference.

Equipment	BTU/HR (heat)
2221W	Approx. 870
2270	Approx. 770
2231W	Approx. 440
2226	Approx. 225
2230/2260 Disk	Approx. 2,730
T-CPU	Approx. 690
Workstation	Approx. 400

Here is the formula used to compute the above:

BTU/HR = No. of watts per peripheral X 3.4129

Try it! It will help in determining whether or not the customer has adequate air conditioning and ventilation for his equipment.

$GIO HOG MODE

Several months ago, ECN's were written for disk multiplexer products which allowed the use of a "HOG MODE" command to be set by a $GIO statement. The use of the $GIO "Hog Mode" greatly simplified programming of the "HOG MODE" by reducing the number of SELECT DISK statements necessary for multiple disk address programs where the operator is required to answer a prompt for the disk address. There are at least two new software packages about to be released which use the $GIO "Hog Mode." These software packages are ISS-3 for WCS Systems and KFAM-5.

The $GIO "Hog Mode" is programmed very easily by sending a CBS with OB8 ON to the multiplexer controller. For example:

```
10 $GIO/310 (4480, B$)
```

sets the controller addressed to 310 into the "Hog Mode."

To turn the "Hog Mode" OFF,

```
20 $GIO/310(4400, B$)
```

The general form is:

```
GIO/XYZ(44WØ, Arg 2)
```

where XYZ is the controller address and W=8 for ON, 0 for OFF

alternately,

```
$GIO#N(44WØ, Arg 2) and $GIO(73VØ 44WØ, Arg 2)
```

may also be used with the same result.

When the "Hog Mode" is set by a $GIO statement, it MUST be turned OFF the same way. Accessing the disk with a non-"Hog Mode" address will not turn the "Hog Mode" OFF. In fact, the $GIO "Hog Mode" was implemented so that only the primary disk addresses need to be used in a program, eliminating the need for a multitude of SELECT DISK statements.

Another $GIO sequence that is also used in disk access programs is one which will determine whether or not the disk is available for use. The sequence is used to check for a disk READY condition instead of allowing the CPU to "hang up" or respond with an error message by sending an initialization sequence to the disk. This is particularly useful in a large data entry program where an error message would require the program to be restarted from the beginning.

The sequence is as follows:

$GIO/XYZ (0600 0740 1262 70A0 4000 8601, B$)

where B$ must be at least 10 bytes, and XYZ is the disk controller address

The sequence 0600 0740 1262 sets a timeout condition (in this case 64 msec). If the disk is READY and responded to the sequence, the eighth byte of B$ will be HEX(00). If the disk did not respond, the eighth byte of B$ will be HEX(10). Also, the first byte will be HEX(CO) if the disk responds properly to the initialization.

If this sequence is used in a multiplexed system, the timeout should be lengthened and/or repeated several times to distinguish between a disk being accessed by another CPU and a disk problem. If the disk is OFF or in the LOAD mode, the eighth byte will be HEX(10), with one exception described below. If the disk is READY, and not being accessed or "Hogged" by another CPU in a multiplexed system, then the eighth byte will be HEX(00).

It is also possible to use the addressing variations discussed in the $GIO "Hog Mode" sequence; in fact, both sequences may be combined as follows:

$GIO/XYZ (0600 0740 1262 70A0 4000 8601 44WO, B$)

which will allow the program to check for the disk READY and an initialization response, then set/reset (depending upon W) the "Hog Mode"

The exception is in a multiplexed system using the 2230MXA; the response returned will not always be correct. This is because the disk READY/BUSY signal on the 6785 board is connected to ±0V, and always gives a disk READY condition to the board. If the disk is in the LOAD mode, the response will be returned correctly (byte 1=CO, byte 8=00), because the disk BUSY condition cannot be detected. If the disk is OFF, the eighth byte will still return correctly as HEX(00) for the same reason, but the first byte will be HEX(FF). If the disk is ON and another CPU is accessing the disk, the eighth byte will be HEX(10), as it should be. Note this is only applicable to the 2230MXA/B Multiplexer. All other multiplexers work correctly.

A final note – that only changes to this sequence should be in setting the timeout. Other experimentation with the sequence is not recommended and may result in serious loss of disk data integrity.

PRINTER CONTROL TAPES

The printer control tape used by Wang printers is available if ordered by part number.

2221W Standard with EOD 11" - 701-9004
2221W Standard without EOD 11" – 701-9005

Consultant price per tape $6.00

For those consultants that require special printer control tapes for custom forms, a kit can be ordered from the following company:

Chester Farnharn Company
Computer Products
Box 163
Hingham, MA 02043
(617) 442-2323

Contents of Kit:

- 100 ft mylar control tape.
- Correction punch – 8-channel
- Splicer
- 500 opaque patches – black
- Marking pen

Price of kit – approximately $125.

CONSULTANT PROFILE

Elliot Rohde Associates
75 South Orange Avenue
South Orange, NJ 07079

The slogan "How to make Boxes and Profits" has been adapted by Elliot Rohde Associates and rightfully justified. The firm consisting of management consultants, industrial engineers, cost accountants, and system analysts specializes in the Corrugated Container Industry.

A package has been developed by the firm that provides the industry, as well as sheet plants, a fully integrated system that will cover their needs from "Top to Bottom." Some of the areas of the system that are available are such things as cost estimating and pricing, box specifications including scoring allowances, order entry, scheduling, statistics, production status, and all related accounting functions and requirements. Industry data is maintained on an area basis by Elliot Rohde Associates.

In 1973, Mr. Rohde saw the future opportunity in developing an integrated system for the industry. After eighteen installations to date, 85% of the organization's time is being devoted to Wang installations.

Through the requirements of the industry, several other packages have been developed, installed, and are available.

- Distribution/Warehousing – Shipping and Finished Goods Inventory System.
- Manufacturing – Cost Estimating and Pricing, Item File Retention, Purchasing and Raw Material Control, Order Entry, Scheduling and Work-In-Process, and Labor Productivity.
- Folding Carton Industry – All Systems.
- Printers and Lithographers – All Systems.

It hasn't been all work and no play at Elliot Rohde Associates, as the firm has developed some most interesting games and educational programs. One of the interesting programs is the Chess Tournament Directors' System as well as a Chess Game to be played with the computer. I wonder if the computer's opponent has a chance!

GENERAL BUSINESS SYSTEM (GBS)

In maintaining our ongoing support of GBS, this group of questions and their answers are published for your information. If you have specific questions not covered, they should be directed to Bob Soucy, Lowell, Mass.

Q. What detail documentation is available for vendors and users?

A. More specific documentation includes one Technical Guide for the disk version of GBS and one Technical Guide for the diskette version of GBS Mods 1, 2, and 3 inclusive.

Technical Guide

- Design Philosophy
- Backup
- Operational Functions
- Common GBS practices and routines
- Setup (files and software)
- Sample Data files

Finally for each module and each version of GBS there is a System's manual and Operator's manual.

System's Manual

- Record descriptions
- Terms and codes
- Subroutines and Utilities
- Program Narrative and block diagrams
- Screen Layouts
- Variable lists

Operator's Manual

- detail operating instructions

Q. What sales support documentation is there for GBS?

A. In a general nature there are three pieces of literature covering GBS Mods 1, 2, and 3.

Introducing GBS –

- general overview diagrams of each module
- GBS design philosophy configurations
- listing of individual module contents
- detailed timings for Mods 1, 2, and 3 comparing the disk and diskette versions for the "T" and "VP" CPUs.

GBS Sample Reports Manual –

- sample printouts of all of the significant output from the GBS system
- diskette file size restrictions
- general overview diagrams of each module
- contents of major files

Marketing Procedure

A marketing or sales brochure is currently being developed which is aimed at the first time computer user. This brochure has extensive references to GBS.

Q. In the "hard disk" version could key files and data files all be placed on the same disk platter?

A. Yes, but not without modifying nearly every program in the system. All select statements throughout the system would have to be changed.

Q. Why would the Accounts Receivable balance in the control sector of the Control File differ from the balance in today's record? Shouldn't they always be the same?

A. It is quite likely that these two fields will differ. All programs which effect the balance in the control sector effect the balance in the daily sector as well but according to the current system date or batch date entered. The control record balance is either credited or debited whichever the case may be. (Invoices would add to and cash would reduce

the balance.) The daily balance is replaced each time that record is accessed (not added or subtracted). Thus, if the beginning balance for June 9 was $400 in the control sector and a batch of June 9th invoices was posted for $100 then the control balance would be $500 and the daily balance would be $500 for June 9th. Then if cash receipts for June 8th were entered worth $50, the control balance would be $450 and the daily balance for June 9th would remain $500 and the new June 8th balance would be $450. Thus, the daily balance is not always a true figure.

Q. Why are "Wash Accounts" necessary?

A. The Invoice Create program and the Shipping Confirmation program will produce erroneous results (and in some cases halt) if there are no "wash accounts" in the salesman and inventory files. Therefore, it is critical that these two records be established properly as outlined in the GBS Technical Guide. In both cases the system allows the operator to enter numbers which are not on file (with a message indicating this condition). The sales figures for these "non-stocked products", etc. must have a place to accumulate, that is the purpose of the "wash account". When the program does not find the particular product or salesman number from the invoice, it accesses a product of all Z's in the key or a salesman of all Z's in the key and adds to it. These records must be on file. They are not automatically generated and must be added through the file maintenance program like any other data record.

Q. What happens to a specific credit invoice entered into the system against a previously purged invoice?

A. The credit remains on file (for an open item customer) until a debit memo is issued against it. A credit given to an invoice after the invoice has been paid and purged from the file should be entered as a general credit through invoice create by entering a zero in the reference number field. This will then become a general credit when passed to the A/R file and purged out against the oldest unpaid invoice during the next month-end cycle.

Q. Are the GBS backup procedures foolproof?

A. No. The original GBS concept involved rigid backup procedures all under system control. Due to vendor requests this idea was abolished and more flexible routines were put into the system. Unmodified, the backup program (on the hard disk version) could accidentally destroy data or programs if the operator does not follow the prompts. Therefore, we suggest that every vendor who installs GBS be sure to put some type of checking routine (i.e. search the catalog for a known program or file before copying) into the backup program.

Q. Will GBS run with ISS Release 3?

A. No. GBS uses KFAM-3 which is contained in ISS Release 2.1, therefore, a vendor must have this earlier release of ISS to handle GBS file initialization etc. However, if the user has a need or desire to use the other ISS utilities in release 3, such as Copy/Verify, of course there is no problem in doing so.

Q. Why does the presence of KFAM-3 in GBS preclude running with a work station?

A. A problem with work station, KFAM-3 and GBS only occurs when both stations are accessing the same KFAM file at the same time. KFAM-3 does not read and update the KDR record or re-read it when ready to write a record. KFAM files will get mixed up if the same file is accessed from more than one station. If stations are accessing different files at all times, then there is no problem with GBS in relation to KFAM-3.

Please refer to the question "Does GBS support a work station or multi-user environment?" published in the WANG SYSTEMS NEWSLETTER No. 2 for further facts concerning this approach.

SOFTWARE PACKAGES AVAILABLE

You can establish contacts for future software business and get paid for it by becoming a licensed installing vendor for Wang Industry Marketing Systems.

WANG/CASH

$PARK

MORTGAGE MANAGEMENT

LIFELINE

AUTO/MATE III

PATIENT BILLING SYSTEM

TIME/CHECK

Please contact your local District Analyst for information. He will arrange for you to receive the software. After that, you are paid for each installation you perform. Naturally, when these users want future custom software, they will think of you FIRST.

GENERAL

Effective with the release of Wang Systems Newsletter No. 4, there will be a section dedicated to answer specific questions asked by our vendors. All questions should be directed to:

Wang Laboratories, Inc.
System Support Group
Professor L. O. Byte, KAA
One Industrial Avenue
Lowell, MA 01851
(617) 851-4111
TWX 710 343 6769
Telex 94-7421

Professor Byte will attempt to have all answers published in the following scheduled publication after the receipt of the question.