

ELLIOTT 900 SERIES SIMULATOR

STANDARD SIR SUBROUTINE LIBRARY.

Elliott distributed a set of standard subroutines for the 900 series machines as SIR source code tapes for the programmer to append to their programs. The subroutines divided into seven broad groups:

1. basic paper tape character I/O and error handling
2. integer and single length fixed point fraction I/O and mathematical functions
3. double length fixed point fraction arithmetic, I/O and mathematical functions
4. floating point number arithmetic, I/O and mathematical functions
5. a Shell sort routine for sorting files held in store
6. utility programs for editing and debugging programs
7. subroutines for input-output operations on peripherals including line printers, card readers and graph plotters.

For convenience the first four of these groups have each been packaged as a single file for use in the simulator. If desired the user can edit the source files to build smaller packages containing just the routines required.

A summary of the contents of each group and method of use for the various routines follows. There is also a demonstration program for each group in the directory 903SIR to show the routines in practical use.

Note that the QFINOUT routines in FLKPAK does not work on a 920M model. (The ACD QFMATH subroutine library contains an amended version that will work on the 920M.)

(Note: in the following unless otherwise, stated subroutines are called using the standard convention

11 X

8 X+1

where X is the name of the subroutine).

Summary of SSYS1 routines.

QDOUT, QDIN specify I/O devices to be used
+1 Paper tape, +3 Teletype

QCHOP routine to output character from accumulator
>= 0, interpret as SIR internal code
< 0, interpret as telecode

ELLIOTT 900 SERIES SIMULATOR

QCHIN routine to read character to accumulator
blank, erase carriage return ignored
halt causes wait via QPAUSE
@ gives internal code 32
! gives internal code +1

QPAUSE subroutine to pause execution, continue by entering at 9.

QSTOP label for a dynamic stop (8 ;+0)

QERROR subroutines to print an error message and pause, continue by entering at 9.

Error number in accumulator on entry
Word following call gives address of parameter block

word[1]: address of e.g. link location of calling routine

word[2]: 0 n - number of locations in group following

word[3]: alphanumeric code
further words ignored

Summary of single length fraction routines (SGLPAK).

Includes SSYS1.

QIN1: routine for integer or fraction input
QIN1+2: scaling factor for mixed/fractional numbers
QIN1+125: 0 after integer input,
+1 after fraction input

Error 1: format error

Error 2: integer or mixed number overflow

Error 3: scale error - mixed number out of range

Error 4: negative scaling factor

QOUT1: routine for integer output

QOUT1+2: routine for fraction output

Call followed by parameter word /f n or f n:

/ = newline, otherwise sameline

f = width 6-f digits

n = scaling factor

integer occupies 7-f positions, fraction or mixed 8-f

Error 1: number cannot be express in specified format
after scaling

ELLIOTT 900 SERIES SIMULATOR

QATAN: routine to calculate $[1/\pi]$ arctan $[x/y]$
argument x in QATAN+89
argument y in QATAN+90
result in accumulator

QEXP: routine to calculate $\exp [2^p x]$
x in accumulator
p in QEXP+53
result in accumulator and QEXP+54
x saved in QEXP+52, p overwritten

QLN: routine to calculate $1/16 \ln x$
argument in accumulator
result in accumulator and QLN+52

QSIN: routine to calculate $0.5 \sin \pi x$ and $0.5 \cos \pi x$
x in accumulator
result $0.5 \sin \pi x$ in accumulator and QSIN+67
result $0.5 \cos \pi x$ in QSIN+68

QSQRT: routine to calculate square root - double length
working
argument [l.s.] in QSQRT+3
argument [m.s.] in accumulator
result in accumulator and QSQRT+45
argument saved in QSQRT+4 [m.s.]
and QSQRT+3 [l.s.]

QSQRT+2: routine to calculate square root - single length
working
argument in accumulator
result in accumulator and QSQRT+45
argument saved in QSQRT+4

Summary of double length fraction routines (DBLPAK).

QDLA double length fraction interpreter.

11 QDLA+20

8 QDLA+21

followed by a sequence of words interpreted as instructions
for a pseudo 900 machine with double length arithmetic.

ELLIOTT 900 SERIES SIMULATOR

Numbers held in 2 words. First word is most significant half.
Bit 18 of second word must be 0.

p - pseudo double length accumulator - QDLA+16,17
pseudo B register - QDLA+18
modified address field of pseudo instruction - QDLA+19
pseudo sequence control register (S.C.R.) and link for exit -
QDLA+20

Special Instructions (all other function similarly to normal
900 series instructions but with double length arithmetic).

(0, 10, 11 work on single length operands).

```

0 0      exit QDLA, set hardware accumulator p[m.s]

0 x      load pseudo B-register
3 x      store l.s. half of pseudo accumulator
6 x      set scaling factor
          6 x = 10^x, 6 8192-x = 10^-x
15 2048  input number from paper tape
15 2052  input number from teletype
          max 10 digits in input. Will be treated as
          an integer if no decimal point. Use scaling
          factor to input mixed numbers
          integers in range -17179869184 to
          +17179869184
          fractions in range -10^m to +10^m where m
          is scaling factor
15 6144  output number to paper tape
15 6148  output number to teletype
15 6144/6148 followed by parameter word /f n or f n
          / newline otherwise same line
          f = number of digits to appear after decimal point

          k = 0 output as integer
          total printing positions = 12-f
          k=4096 output as a fraction / mixed number
          if scaling factor is m, total printing
          positions = f+m+3

```

Errors

```

0020      attempt to divide by zero
0021      input number format error
0022      input overflow [number too large]
0024      more than 10 digits in input fraction

```

ELLIOTT 900 SERIES SIMULATOR

0025 15 instruction with illegal address
0026 overflow during division

QDAATAN: routine to calculate $[1/\pi]$ arctan $[x/y]$ and
 $1/[2\pi]$ true bearing
 QDAATAN+136,137 x
 QDAATAN+138,139 y
 QDAATAN+142,143 arctan
 QDAATAN+146,147 true bearing
 if $x=y=0$ punches 0000.001 continuously

QDASIN: routine to calculate $0.5 \sin \pi x / 0.5 \cos \pi x$
 argument in QDASIN+98,99 argument m.s. in
 accumulator
 QDASIN+102,103 sin
 QDASIN+104,105 cos

QDASQRT: routine to calculate sqrt x
 QDASQRT+44,45 x
 QDASQRT+46,47 sqrt x
 if $x < 0$ punches 0000.010 continuously

Summary of floating point routines (FLTPAK).

QF: Floating Point Interpreter

QF+1: work with packed numbers (two words per real)
QF+2: work with unpacked numbers (three words per real)

Entry followed by series of words interpreted as instructions
to a 900 series like machine with floating point arithmetic.

FPA = pseudo Floating Point Accumulator

Special instructions

0, 7, 8, 9, 10, 11 work on single word items

0 0 exit QF
0 X Load pseudo B register with contents of location X
3 X Exchange: exchange contents of FPA and location X
6 0 Reset presumed formats for i/o
6 1 Assume packed format
6 2 Assume unpacked format
6 3 float (integer). Address of integer in QF+3

ELLIOTT 900 SERIES SIMULATOR

```
6 4 entier (FPA). Address for result in QF+4
6 5 float (fraction). Address of fraction in QF+5
6 6 Convert FPA to fraction. Address for result in QF+6
6 7 Further I/O instructions refer to integers
6 8 Further I/O instructions refer to real numbers
6 8191 set output format, specified by following word
    /F N or F N:
    / sameline otherwise newline
    F = 0, freepoint(N)          N <= 8
    F = 1, aligned(m, n)        N=16m+n  m+n <= 15
    F = 2, scaled(N)            n <= 8
    F = 4, digits (N)           n <= 6
14 arithmetic shifts and re-standardizes FPA
15 2048 read number from paper tape
15 2052 read number from teletype
15 6144 punch number to paper tape
15 6148 punch number to teletype
```

Error code:

```
QF!:      impermissible instruction
ROF!:     floating point overflow
RTI!:     integer overflow
RTF!:     fraction overflow
LN!:      argument <= 0
SQ!:      argument < 0
EX!:      argument > 2
EE2:      FPA not standardized on output
EE3:      integer overflow on input
EE4:      contextual error on input
EE5:      attempt to input string
EE6:      parity error in input
EE7:      floating point overflow on input
```

Summary of SHELLSORT.

Shellsort provides a means to sort files of data records held in store.

```
11 SHELLSORT
8  SHELLSORT+1
0  SORTAB      address of sort table
```

Sort Table

```
Word 0:      number of records
Word 1:      number of words in each record
Word 2:      address of first word of first record
```

ELLIOTT 900 SERIES SIMULATOR

Word 3: sort list
Word 4: word of first key - first word is word 1
Word 5: mask of first key
etc, etc for up to 9 keys
+0 to terminate

Sort List

Word 0: +1 ascending order, -1 descending order
Word 1: highest priority key
Word 2: next highest priority key
etc
+0 to terminate

Graph Plotting (QPLOT) Summary.

QPLOT+3 x2 next x position
QPLOT+4 y2 next y position
QPLOT+5 x1 last x position
QPLOT+6 y1 last y position

Move / Draw to x2, y2

4 x2
5 QPLOT+3
4 y2
5 QPLOT+4
11 QPLOT
8 QPLOT+1
n (parameter word)

n = +0 - no change in pen status
n = +1 - raise pen before moving
n = +2 - lower pen before moving, i.e., draw
x1, y1 updated on exit with x2, y2

Plot character

4 CH (in internal code, e.g., \ A)
11 QPLOT
8 QPLOT+2
n (parameter word)

n = +0 - no change in scaling or positioning
n < 0 - draw character centred on x1, y1 (finish pen up at centre)
n > 0 - draw character along line x=x1 (x1 updated to point to next position)

ELLIOTT 900 SERIES SIMULATOR

Errors

Message	Cause	Effect of re-entry
PLOT L!	Impermissible parameter	Pen Down substituted
PLOT C1	Impermissible character Code	space is substituted

Character tables.

Characters are represented in QPLOT by vectors encoded in a table. The table is at location of QPLOT+324: each entry n in this table contains the address of an array of turning points for the n-th character in the 903 SIR internal code, or -1 for characters for which there is no graphic. The turning points are coded as pairs of 9 bit x, y coordinates packed into a single word. The x coordinate may vary from -7 to +7 and the y coordinate from -9 to +9. If the pen is to be raised for the next move +32 should be added to the y coordinate. If the turning point is the last in the sequence for the character, +16 should be added to the y coordinate. DEMO12 in the 903SIR folder contains an example of adding entries for the characters (, / and).

QCARDIN Summary.

QCARDIN is used to read from an Elliott 4100 card reader.

In addition to the QCARDIN subroutine, an instance of the Elliott SSYS2 - SIR Systems Tape 2 modular interrupt routines is required to handle line printer interrupts. (This tape was issued on an installation-by-installation basis, configured for the specific devices at that site). An example can be found in DEMO13 in the 903SIR folder that handles both card reader and line printer.

Before reading the first card, QCARDIN+2 should be set to -1.

If QCARDIN is to operate with single buffering, the address of an 80-word buffer should be placed in QCARDIN+3 and -1 in QCARDIN+4. If double buffering is to be used, as recommended, QCARDIN+3 should hold the address of the first 80-word buffer and QCARDIN+4 the address of the second 80 word buffer.

ELLIOTT 900 SERIES SIMULATOR

There was no standard Elliott card code (indeed there was no single industry punched card code in the 1960's) so the programmer is required to provide a translation from card codes to SIR internal code. QCARDIN+5 should contain the address of a 64-word translation table. It should contain entries giving the card code for each character in the SIR internal code. (Unique dummy entries should be given for any SIR characters that do not have a corresponding card code). Each entry should have the card image in the most significant bits and the SIR character code in the least significant 6 bits. Note the card image is stored inverted - the most significant bit (bit 18) corresponds to the bottom row of the card column. The entries should be put in ascending numerical order, treated as unsigned 18 bit numbers. QCARDIN uses a binary search to decode each input column to its SIR equivalent. Card codes with no equivalent are translated to space (0).

Input can either be card-by-card, or character-by-character.

To input a card, enter

```
11  QCARDIN
8   QCARDIN+1
```

A card will be read and the decoded image stored in one of the buffers specified, which are used alternatively. On exit from QCARDIN, the accumulator will contain the address of the decoded buffer unless an error has been detected in which case the accumulator will be negative and the card reader status code will be in the 8 least significant bits. The data in the decoded buffer will be in the same format as the translation table.

To input a character from the card reader, enter

```
11  QCINCH
8   QCINCH+1
```

On exit from the subroutine, the character will be in the accumulator in SIR internal code. QCINCH+3 will hold the address of the word containing the column image, in the same format as the translation table. If the internal code is given as 0, the either the whole column image will be 0, representing a blank column (space) or the most significant 12 bits will hold an image with no equivalent in the translation table.

If it is required to skip the remainder of a card then set QCINCH+2 to >80, to skip n columns add +n to QCINCH+2. (Note if the addition of n gives a result greater than 80, the program will skip to the FIRST column of a new card).

ELLIOTT 900 SERIES SIMULATOR

If an error occurs in QCINCH it will be reported using QERROR, with the card reader status word as

*CRD <status in octal> <address of entry>.

QLPOUT Summary.

QLPOUT is used to organize line printer output to an Elliott 4100 scientific line printer.

As with QCARDIN, QLPOUT requires the SSYS2 routines to manage interrupts.

Before printing the first line, the location QLPOUT+4 should be set to +1 (or a blank line output).

To output a character in SIR internal code, use

```
11 QLPOUT
8   QLPOUT+1
```

with the internal code value in the accumulator.

Characters are stored in a buffer until a newline character is received, when the line is output preceded by one paper throw. If no newline is received by the time the 122nd character is stored, the line will be output automatically.

The paper is normally thrown one line space for each line printed. This may be overridden with

```
11 QLPOUT
8   QLPOUT+2
```

with a number +n in the accumulator. This will cause n lines to be thrown and the contents of the buffer are then output. N should be in the range 0<=n<=15. (Note: the simulator does not support n=0).

Note that QLPOUT requires the last character output by a program is a newline or a vertical format character, otherwise characters may be left in the buffer and never printed.

Note that the SIR internal code 63 prints as ' (close quote) rather than ←.

Errors are output to the teleprinter using QERROR in the form:

*LPT <octal number> <address of entry>

ELLIOTT 900 SERIES SIMULATOR

The octal number is the line printer status word (see line printer hardware description earlier).

Note that the simulator signals a non-recoverable error if an attempt is made to continue by entry at 9 following an error.

ELLIOTT 900 SERIES SIMULATOR

Summary of Magnetic Tape Routines.

The basic magnetic tape routines are distributed as a SIR source tape called "MTBLOCK".

Find and Read a Block n.

```
A := +n

11  QMT
8   QMFIND
0   FT          (file table)
8   ERROR       (error handling)
```

Go Back n Blocks and Read.

```
A := -n

11  QMT
8   QMREAD
0   FT          (file table)
8   ERROR       (error handling)
```

Read Next Block in Sequence.

```
A := 0

11  QMT
8   QMREAD
0   FT          (file table)
8   ERROR       (error handling)
```

The tape is advanced or retreated (by backspace or rewind) to locate the specified block and the content of that block copied to the buffer specified by the file table.

If block length in file table is X and block length on tape is W and $W > X$ and error exit is made and extra words are not read into the buffer. If $W < X$, the block will be read into the first W words and the next $W-X$ will be undefined. In all cases when the block is successfully read word 1; of the buffer will hold +W or /0 W, the latter indicating a label block.

ELLIOTT 900 SERIES SIMULATOR

On normal exit, A will be positive, unless an end-of-tape marker has been detected in which case A will be negative.

If the error exit is taken, an error code will be in A as follows.

Bit 18	Specified block cannot be found
Bit 15	An instruction has been treated as 'do nothing' (generally due to switching the handler to manual while a read or write is in progress).
Bit 14	File not opened for reading.
Bit 10	Zero character, i.e., false end of block; repeated hardware error.
Bit 7	Write permit ring not in. Ignore if reading.
Bit 6	Long block. If bits 2, 3, 4 & 10 are zero the First X words are read correctly.
Bit 4	Parity failure: repeated hardware error.
Bit 3	Missed transfer: repeated hardware error.
Bit 2	Handler in manual (& bits 10-3 undefined).

On the computer the routine makes repeated attempts to overcome transient hardware errors. On the simulator, these cannot arise unless the programmer uses the basic magnetic tape instructions to write a deliberately corrupt tape (e.g., with bad parity).

Write a Block.

11	QMT	
8	QMWRITE	
0	FT	(file table)
8	ERROR	(error handling)

A block is written to tape from the buffer specified in the file table. Words 0; to 4; of the file table must be set up correctly. If a label block is to be written, word 2; must hold the length of the block with bit 18 set to 1. QMWRITE will overwrite words 0; and 1; of the buffer with the block header. A block must not consist entirely of zero words. The tape must be positioned correctly as the block is written immediately.

On normal exit, A(17-1) hold the number of the block written. If bit 18 is set the block has been written correctly but the end of tape marker has been detected.

ELLIOTT 900 SERIES SIMULATOR

On an error exit, the accumulator contains an error code as follows:

Bit 18	End of tape detected, block not written.
Bit 16	Handler not available, other bits undefined.
Bit 15	An instruction has been treated as 'do nothing' (generally due to switching the handler to manual while a read or write is in progress).
Bit 14	File not opened for reading.
Bit 13	File not opened for writing.
Bit 10	Zero character, i.e., false end of block; repeated hardware error.
Bit 7	Write permit if zero, writing not permitted.
Bit 6	Long block. If bits 2, 3, 4 & 10 are zero the First X words are read correctly.
Bit 4	Parity failure: repeated hardware error.
Bit 3	Missed transfer: repeated hardware error.
Bit 2	Handler in manual (& bits 10-3 undefined).

Open File.

Files can be opened for reading, updating or writing. In all cases the tape is first rewound and then a file protection check occurs as described below. If successful, the file is positioned ready for the next operation.

11	QMT	
8	QMOPEN	
0	FT	(file table)
8	ERROR	
mode parameter		

The possible opening modes are:

Open for Writing: /0 0

The header block on tape is compared with the header block in the file table. If the names are identical, or the tape is headed "SCRATCH", the file is opened for reading and writing and a new header is written, overwriting the old, with the contents of words 10; to 35; of the file table. If the names do not agree the error exit is taken.

Open for Updating: /8 0

The header block on tape is compared against the header in the file table, but NOT overwritten. If they agree the file is

ELLIOTT 900 SERIES SIMULATOR

opened for reading and writing, positioned to read or overwrite the block following the header block. If the names do not agree the error exit is taken.

Open for Reading: 0 0

The header block on tape is compared against the header in the file table, but NOT overwritten. If they agree the file is opened for reading, positioned to read the block following the header block. If the names do not agree the error exit is taken.

Error codes:

Bit 18	Wrong name on header.
Bit 16	Handler not available, other bits undefined.
Bit 15	An instruction has been treated as 'do nothing' (generally due to switching the handler to manual while a read or write is in progress).
Bit 10	Zero character, i.e., false end of block; repeated hardware error.
Bit 4	Parity failure: repeated hardware error.
Bit 2	Handler in manual (& bits 10-3 undefined).

Close File.

11	QMT
8	QMCLOSE
0	FT
8	ERROR

mode parameter

/8	0	Write EOF block, do not close file.
/3	0	Close for reading and writing, rewind in manual.
/1	0	Close file for reading and writing, rewind.
/5	0	Close file for writing only, rewind.
3	0	Close file for reading, rewind in manual.
1	0	Close file for writing, rewind.
5	0	Rewind, do not close file.

Close for writing writes an EOF block. The tape should not have been moved since the last data block was written.

Error codes:

Bit 16	Handler not available, other bits undefined.
--------	--

ELLIOTT 900 SERIES SIMULATOR

Bit 15	An instruction has been treated as 'do nothing' (generally due to switching the handler to manual while a read or write is in progress).
Bit 13	Attempting to close a file for writing when only open for reading.
Bit 10	Zero character, i.e., false end of block; repeated hardware error.
Bit 4	Parity failure: repeated hardware error.
Bit 3	Missed transfer, repeated hardware error.

Check Status Word.

11	QMSTAT	
8	QMSTAT+1	
0	FT	(file table)

Status codes:

Bit 2	Handler in manual.
Bit 1	handler busy, rewinding.

Summary of Magnetic Tape Record Handling Routines.

Location 23; of the file table must be set either negative if variable length records are being used, or to a positive integer giving the number of words per fixed length record. This word may be set implicitly by reading down the header block or explicitly by the user.

Before the first record is read or written, location 5; of the file table must be set zero or negative to indicate that a new block should be read or written as appropriate.

Read Record.

11	QMR	
8	QMRIN	
0	FT	(file table)
0	RECORD	(buffer for record)
8	ERROR	(error handling)

Words 0; 1; 2; 5; & 23; of the file table must be set correctly, or not disturbed since the previous entry to QMROUT. The record buffer must be distinct from the (tape) block buffer and large enough to hold the largest record.

ELLIOTT 900 SERIES SIMULATOR

QMRIN uses QMREAD to read the next block along the tape and the first record is picked up from the block buffer and copied to the record buffer. Location 5; of the file table is then set to the address of the next record. Subsequent calls of QMRIN pick up the following records until all records in the block have been processed. Location 5; of the file table is then set negative to force a new block to be read on the next call of QMRIN.

On normal exit, A holds the number of words in the record. The record buffer holds the contents of the record excluding the record length, nor will this extra word be included in the number of words given in A.

In case of an error exit, if A contains zero a label block has been read (and can be inspected in the block buffer), otherwise A contains a QMREAD error code.

Fixed length records must not have 0 in the first word (any such records will be ignored and the next block read). If QMRIN reads a block from tape that does not contain a record structure and is not a label block, the error is not detected.

The action of the record routines is undefined if
2 <= record length <= (tape buffer length - 2).

Write Record.

11	QMR	
8	QMROUT	
0	FT	(file table)
0	RECORD	(record buffer)
8	ERROR	(error handling)

Words 0; 1; 2; 5; and 23; of the file table must be set up correctly, or not disturbed since the previous entry to QMROUT. The record buffer must be distinct from the (tape) block buffer. If fixed length records are to be used location 23; of the file table must hold the number of words in the record. If variable length records are to be written the length must be held in A on entry to QMROUT and 23; in the file table must be negative. The value itself and the record itself must not contain the extra word holding the record length (of a variable length record).

ELLIOTT 900 SERIES SIMULATOR

If an error occurs the error exit will be taken with a QMWRITE error code in A. If A = 0, the end of tape marker has been detected. If the error is recoverable, the call to QMWRITE can be repeated with the same record specified again.

Flush Last Record.

Before closing the file or writing anything else to tape other than a further record, the user's program should flush the final record, if any, as follows:

11	QMT	(N.B. QMT not QMR)
8	QMREND	
0	FT	(file table)
8	ERROR	(error handling).

Error handling is as for QMROUT.

Files in 9003SIR directory

ELLIOTT 900 SERIES SIMULATOR

ACD SIR Subroutines.

ACD issued their own library subroutines to their customers. Many of these either included or were derived from the 903 SIR subroutines listed above. In general the changes were to allow the use of 920 telecode as well as 900 telecode for i/o. The routines were distributed in a set of packages, summarized below. The descriptions are taken from the ACD Library Book No. 108, Copy No. 23, Amendment 5, "903/920 SIR SUBROUTINES".

CHIOL - Character I/O and Legible Tape Routines.

Buffered character input

```
11  CHIPL
8   CHIPE
```

Reads next character into accumulator and location CHIP (translated to ACD internal code).

CHIPF - set this to +0 before reading a new tape. New tapes must start with a newline sequence, which determines the telecode that will be used to decode the rest of the tape.

Character output

```
11  CHOPL
8   CHOPE
```

Output character in accumulator (presumed to be in ACD internal code). To print the character N times, subtract 128N from the character value). On return from the subroutine the accumulator will contain the bit pattern actually punched.

CHOPF - set this to +0 before punching the first character of a new tape (outputs blanks as a leader).

CHOPC - output code to use: +1 for 900 telecode, +0 for 920 telecode or &400000 for legible paper tape.

Error reporting

```
11  ERRORL
8   ERRERE
message
```

The message to be output should take the form of a sequence of alphanumeric groups. The first "." encountered will be interpreted as the end of message marker.

A version of CHIOL called CHIOSIR1 was issued containing the telecode i/o designed to be run when the ACD 1-Pass SIR

ELLIOTT 900 SERIES SIMULATOR

assembler was also in store to enable less store to be used by the user's code through sharing SIR assembler's internal i/o routines.

The names of the ACD packages are listed below. The interfaces are essentially the same as for the equivalent 903 library subroutines, with changes to how some errors are reported. These changes are noted in comments at head of each packages source file.

QSIO - Single Length I/O.

Versions of QIN1 and QOUT1 using the CHIOL routines.

QSMATH - Single Length Mathematical Functions.

Versions of QATAN, QEXP, QLN, QSIN, QSQRT with modified error reporting (legible tape output).

OPTIMATH - Optimised versions of QATAN, QSIN and QSQRT.

QDMATH - Double Length Arithmetic and Mathematical Functions.

Versions of QDLA, QDAATAN, QASIN, QASQRT (i.e., DBLPAK in 903SIR directory) using CHIOL for i/o and modified error reporting (legible tape output).

QFMATH - Floating Point Arithmetic and Mathematical Functions.

Versions of QF, QFINOUT and QFMATH (i.e., FLTPAK in 903SIR directory) using CHIOL for i/o and modified error reporting (legible tape output). Note that QFMATH contains a modified QFINOUT that correctly prints the integer zero on 920M processors.