

File & Terminal I/O Statements

ACCEPT (channel, variable, Label)	Receive character from channel
CLOSE channel	Close channel
DELETE (channel)	Delete record from ISAM file
DISPLAY (channel, item, ...)	Send output to device or file
FIND (channel, record, key_spec)	Find record
FORMS (channel, control_code)	Process ASCII control codes
GET (channel, data_area, record)	Receive data from channel
GETS (channel, data_area, Label)	Get sequential binary data
PUT (channel, data_area, record)	Write fixed-length data
PUTS (channel, data_area, Label)	Write sequential fixed-length data
OPEN (channel, mode:submode, file)	Open channel
READ (channel, data_area, key_spec)	Read specific record
READS (channel, data_area, Label)	Read next sequential record
STORE (channel, data_area)	Store record to ISAM file
UNLOCK channel, RFA:match_rfa	Release record lock on channel
WRITE (channel, data_area, record)	Write record to file
WRITES (channel, data_area, Label)	Write next sequential record

Modes & Submodes

	TT:	I	O	U	I:I	U:I	I:R	O:R	U:R
ACCEPT	✓	✓							
DELETE						✓			
DISPLAY	✓		✓						
FIND		✓	✓	✓	✓	✓	✓	✓	✓
FORMS	✓		✓						
GET		✓	✓	✓					
GETS	✓	✓	✓						
PUT			✓	✓					
PUTS	✓		✓						
READ		✓	✓	✓	✓	✓	✓	✓	✓
READS	✓	✓	✓		✓	✓	✓	✓	✓
STORE						✓			
WRITE			✓	✓		✓	✓	✓	✓
WRITES	✓		✓				✓	✓	✓

Use TT: to I/O to the terminal

I/O Mode

I	Input	Read existing file
O	Output	Write new file
U	Update	Read/Write existing file

I/O Submode

R	Relative	Relative access to file
I	Indexed	Access ISAM file

File & Terminal I/O Statements

ACCEPT (channel, variable, Label)	Receive character from channel
CLOSE channel	Close channel
DELETE (channel)	Delete record from ISAM file
DISPLAY (channel, item, ...)	Send output to device or file
FIND (channel, record, key_spec)	Find record
FORMS (channel, control_code)	Process ASCII control codes
GET (channel, data_area, record)	Receive data from channel
GETS (channel, data_area, Label)	Get sequential binary data
PUT (channel, data_area, record)	Write fixed-length data
PUTS (channel, data_area, Label)	Write sequential fixed-length data
OPEN (channel, mode:submode, file)	Open channel
READ (channel, data_area, key_spec)	Read specific record
READS (channel, data_area, Label)	Read next sequential record
STORE (channel, data_area)	Store record to ISAM file
UNLOCK channel, RFA:match_rfa	Release record lock on channel
WRITE (channel, data_area, record)	Write record to file
WRITES (channel, data_area, Label)	Write next sequential record

Modes & Submodes

	TT:	I	O	U	I:I	U:I	I:R	O:R	U:R
ACCEPT	✓	✓							
DELETE						✓			
DISPLAY	✓		✓						
FIND		✓	✓	✓	✓	✓	✓	✓	✓
FORMS	✓		✓						
GET		✓	✓	✓					
GETS	✓	✓	✓						
PUT			✓	✓					
PUTS	✓		✓						
READ		✓	✓	✓	✓	✓	✓	✓	✓
READS	✓	✓	✓		✓	✓	✓	✓	✓
STORE						✓			
WRITE			✓	✓		✓	✓	✓	✓
WRITES	✓		✓				✓	✓	✓

Use TT: to I/O to the terminal

I/O Mode

I	Input	Read existing file
O	Output	Write new file
U	Update	Read/Write existing file

I/O Submode

R	Relative	Relative access to file
I	Indexed	Access ISAM file

Display Items

DISPLAY(*channel*, *item*, ...)

\$SRC_POS (<i>row</i> , <i>col</i>)	Reset absolute position of cursor
\$SCR_CLR (<i>function</i>)	Clear specific portion of screen
SCREEN	Entire screen
EOL	Current position to end of line
EOS	Current position to end of screen
LINE	Current line
BOL	Beginning of line to current position
BOS	Beginning of screen to current position
\$SCR_MOV (<i>row</i> , <i>col</i>)	Move cursor relative to current position
\$SCR_ATT (<i>function</i> , ...)	Modify screen attributes
CLEAR	All attributes off
BOLD	Bolding on
UNDER	Underlining on
BLINK	Blinking on
REVERSE	Reverse video on
GON	Graphics on
GOFF	Graphics off
SAVE	Save attributes (Unix & OpenVMS only)
RESTORE	Restore attributes from last SAVE (Unix & OpenVMS only)

Runtime Option Flags

xcall **FLAGS**(*flags*, *control*)

10	Don't echo terminator character
9	Allow terminal input for READS without explicit terminal characters
8	Ignore interrupt characters
7	Suppress program termination message
6	No record attributes (OpenVMS)
5	Don't echo input
4	Replace most recent characters with paint characters on delete
3	Check for duplicate filenames
2	Process case-sensitive terminal input
1	Apply European formatting conventions



Resource Center: synergex.com/welcomeRC

Documentation: synergex.com/docs

Email: support@synergex.com

Call: 800.366.3472

916.635.7300

Display Items

DISPLAY(*channel*, *item*, ...)

\$SRC_POS (<i>row</i> , <i>col</i>)	Reset absolute position of cursor
\$SCR_CLR (<i>function</i>)	Clear specific portion of screen
SCREEN	Entire screen
EOL	Current position to end of line
EOS	Current position to end of screen
LINE	Current line
BOL	Beginning of line to current position
BOS	Beginning of screen to current position
\$SCR_MOV (<i>row</i> , <i>col</i>)	Move cursor relative to current position
\$SCR_ATT (<i>function</i> , ...)	Modify screen attributes
CLEAR	All attributes off
BOLD	Bolding on
UNDER	Underlining on
BLINK	Blinking on
REVERSE	Reverse video on
GON	Graphics on
GOFF	Graphics off
SAVE	Save attributes (Unix & OpenVMS only)
RESTORE	Restore attributes from last SAVE (Unix & OpenVMS only)

Runtime Option Flags

xcall **FLAGS**(*flags*, *control*)

10	Don't echo terminator character
9	Allow terminal input for READS without explicit terminal characters
8	Ignore interrupt characters
7	Suppress program termination message
6	No record attributes (OpenVMS)
5	Don't echo input
4	Replace most recent characters with paint characters on delete
3	Check for duplicate filenames
2	Process case-sensitive terminal input
1	Apply European formatting conventions



Resource Center: synergex.com/welcomeRC

Documentation: synergex.com/docs

Email: support@synergex.com

Call: 800.366.3472

916.635.7300