

Example	Command	Description
	PS	Pause execution until Continue (C) is entered
	MN	Set to mode normal (Preset Moves)
	A 5	Set acceleration to 5 revs/sec ²
	V 5	Set velocity to 5 revs/sec
	D 2000	Set distance to 2,000 revs
	G	Executes the move (Go)
	"MOVE_DONE	After the motor finishes the move, the IFX will send the message MOVE_DONE out the RS-232C port.
	C	Continue Execution

Q1 Set-Up		Enter Velocity Profiling Mode			VALID Software Version A
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES	
<a>Q1	N/A	N/A	None	Immediate Never Saved	
EXECUTION TIME			SEE ALSO Q0, RM		

Description The Q1 command enters the indexer in velocity profiling mode. Subsequent RM commands will cause an immediate change in motor velocity. Use Q0 to exit this mode.

Example	Command	Description
	Q1	Enter Velocity streaming mode
	RM0032	Accelerate to 0.25 revs/sec
	RM0064	Accelerate to 0.5 revs/sec
	RM00C9	Accelerate to 1 revs/sec
	RM0192	Accelerate to 2 revs/sec
	RM00C9	Decelerate to 1 revs/sec
	RM0064	Decelerate to 0.5 revs/sec
	RM0032	Decelerate to 0.25 revs/sec
	RM0000	Decelerate to 0 revs/sec
	Q0	Exit velocity streaming mode

This example assumes operation in the 400 step/rev mode.

QØ Set-Up	Exit Velocity Profiling Mode			VALID Software Version A
SYNTAX <a>QØ	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO Q1, RM		

Description The QØ command exits the velocity profiling mode. The motor will stop when QØ is issued.

Entering this command will cause the IFX to enter Mode Normal (MN).

Example See Q1 command

R Status	Request Indexer Status			VALID Software Version A
SYNTAX aR	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO RA, RB, RC		
RESPONSE TO aR IS *x				

Description The Request Indexer Status (R) command can be used to indicate the general status of the indexer. Possible responses are:

<u>Response Character</u>	<u>Definition</u>
*R	Ready
*S	Ready, Attention Needed
*B	Busy
*C	Busy, Attention Needed

When the indexer is not prepared to accept another command, the following conditions will cause a response indicating that the indexer is busy (*B):

- * Performing a preset move
- * Accelerating/decelerating during a continuous move
- * A time delay is in progress. (T command)
- * In RM mode
- * Paused
- * Waiting on a Trigger
- * In Jog mode
- * Going Home
- * In Power-on sequence mode
- * Running a sequence
- * Executing a loop

The following conditions will cause a response indicating that an error exists (*S or *C):

- * A feedback error condition exists.
- * Go home failed
- * Limit has been encountered
- * Sequence execution was unsuccessful

When the response indicates that attention is required, more details on the error condition are available by using the RA, RB, or RC commands.

It is not recommended that this command be used in tight polling loops which could result in microprocessor over load. Time delays can alleviate this problem.

This command is not intended to be used to determine if a move is complete. Rather it should be used after the move is complete to determine if there might be other errors or faults.

Use a buffered status request (CR or LF) command or a programmable output to indicate move completion.

Example

<u>Command</u>	<u>Response</u>
R	*R (Indexer ready to accept a command, and no error conditions exist.)

RA Status		Limit Switch Status Report			* VALID Software Version A
SYNTAX aRA	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved	
EXECUTION TIME			SEE ALSO R, RB		
RESPONSE TO aRA IS *x					

Description

The Limit Switch Status Report (RA) command responds with the status of the end of travel limits during the last move as well as the present condition. This is done by responding with one of 12 characters representing the conditions listed below.

Response Character	Last Move		Current	
	Terminated by		Limit Status	
	CW Limit	CCW Limit	CW Limit	CCW Limit
*G	NO	NO	OFF	OFF
*A	YES	NO	OFF	OFF
*B	NO	YES	OFF	OFF
*D	NO	NO	ON	OFF
*E	YES	NO	ON	OFF
*F	NO	YES	ON	OFF
*H	NO	NO	OFF	ON
*I	YES	NO	OFF	ON
*J	NO	YES	OFF	ON
*L	NO	NO	ON	ON
*M	YES	NO	ON	ON
*N	NO	YES	ON	ON

The RA command is useful when the motor will not move in either or both directions. The report back will indicate whether or not the last move was terminated by one or both end-of-travel limits.

If you are hitting a limit switch, the Ready Status (R) will return a *S.

This command is not intended to be used to determine if a move is complete. Rather it should be used after the move is complete to determine if there might be other errors or faults.

Example

Command	Response
1RA	*@ (By issuing a 1RA command, the indexer with the address of 1 responded with *@, indicating that the last move was not terminated by a limit and that no limits are currently active.)

RB Status	Loop, Pause, Shutdown, Trigger Status Request			VALID Software Version A
SYNTAX aRB	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO R, RA, TR, PS, L, ST		
RESPONSE TO aRB IS *x				

Description

This command receives a response from *@ to *O, as defined below. The four conditions for which status is indicated are as follows:

- Loop Active** A loop is in progress.
- Pause Active** Buffered commands are not being executed due to a Pause (P) command (waiting for a C command).
- Shutdown Active** The motor is shutdown by the ST1 command.
- Trigger Active** At least one trigger is active.

Response Character	Loop Active	Pause Active	Shutdown Active	Trigger Active
*@	NO	NO	NO	NO
*A	YES	NO	NO	NO
*B	NO	YES	NO	NO
*C	YES	YES	NO	NO
*D	NO	NO	YES	NO
*E	YES	NO	YES	NO
*F	NO	YES	YES	NO
*G	YES	YES	YES	NO
*H	NO	NO	NO	YES
*I	YES	NO	NO	YES
*J	NO	YES	NO	YES
*K	YES	YES	NO	YES
*L	NO	NO	YES	YES
*M	YES	NO	YES	YES
*N	NO	YES	YES	YES
*O	YES	YES	YES	YES

This command is not intended to be used to determine if a move is complete. Rather it should be used after the move is complete to determine if there might be other errors or faults.

Example

Command

Response

1RB

*A (After issuing a 1RB command, the response came back as *A. This means that the indexer is currently executing a loop.)

RC Status	Closed Loop Status			VALID Software Version A
SYNTAX aRC	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO R, RA, RB, FS commands, GH		
RESPONSE TO aRC IS *x				

Description

The Closed Loop (RC) command responds with a character indicating whether a Stall was detected on the last move, and whether the last Go Home attempt was successful.

- The stall detect function can be enabled by the FSH1 command
- The Go Home function can be executed by using the GH command.

Response Character	Stall Detected?	Go Home Successful?
*g	NO	YES
*A	YES	YES
*B	NO	NO
*c	YES	NO

This command is not intended to be used to determine if a move is complete. Rather it should be used after the move is complete to determine if there might be other errors or faults.

Example

Command

Response

1RC

*C (This means that while attempting the last move, the indexer detected a stall.)

RM Motion	Rate Multiplier in Velocity Streaming Mode			VALID Software Version A
SYNTAX <a>RMn	UNITS rev/sec	RANGE n = 0000- FFFF	DEFAULT N/A	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO Q1, QØ, D, H		

Description

The **RM** command sets an immediate velocity where n represents a 4 digit hexadecimal value. The value for n is determined with the following formula:

$$(\text{Revs/Sec}) \times (\text{Resolution Constant}) = \text{Decimal \# for velocity value}$$

In the above formula, Rev/Sec is the desired speed in revs per second, and the Resolution Constant is the value taken from the table below. The resolution setting defined with the **RM** command will determine which resolution constant will be used in the formula. The resulting decimal number must be converted to hexadecimal to obtain the value for n.

Motor Resolution	Resolution Constant
200	100.6918
400	201.3836
1000	332.7949
2000	665.5898
5000	639.9902
10000	597.3242
12800	655.3500
25000	1493.3105

The velocity change is essentially instantaneous; there is no acceleration/deceleration ramp between velocities. A limit switch-closure will stop movement while in velocity profiling mode, but does not cause the IFX to exit velocity streaming mode. To recover from encountering a limit in **RM** mode, the **QØ** command must be issued and the direction must be changed. **RM** (profiling) mode is uni-directional. The direction will be the last direction set either from a move or from a **D** or **H** command. Bi-directional moves using this mode can be made by returning to velocity zero, switching off **RM** mode, changing the direction, and re-enabling **RM** mode. Exiting **RM** mode via the **QØ** command will cause the IFX to enter the Mode Normal (**MN**).

Example	Command	Description
	Q1	Enter Velocity streaming mode
	RM0032	Accelerate to 0.25 revs/sec
	RM0064	Accelerate to 0.5 revs/sec
	RM00C9	Accelerate to 1 revs/sec
	RM0192	Accelerate to 2 revs/sec
	RM00C9	Decelerate to 1 revs/sec
	RM0064	Decelerate to 0.5 revs/sec
	RM0032	Decelerate to 0.25 revs/sec
	RM0000	Decelerate to 0 revs/sec
	Q0	Exit velocity streaming mode

This example assumes operation in the 400 step/rev mode.

RS Status		Status of Sequence Execution			VALID Software Version A
SYNTAX aRS	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved	
EXECUTION TIME			SEE ALSO XR, XP		
RESPONSE TO aRS IS *x					

Description

The RS command indicates the status of the latest sequence execution. Possible responses are as follows:

Response	Sequence Started	Sequence Ended
*g	NO	NO
*A	YES	NO
*B	NO	YES

Whenever a sequence is started, the sequence start bit is set and the sequence end bit is cleared (this occurs only if the sequence is VALID and is actually run). Whenever a sequence is ended, the start bit is cleared and the end bit is set. Any abrupt move termination (e.g., limit activation), or a K or S command, clears both bits.

Example

Command	Response
1RS	*A (Sequence in progress)

RV Status	Revision Level			VALID Software Version A
SYNTAX aRV	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO None		
RESPONSE TO aRV IS *nn-nnnn-nn<xn>				

Description

The Revision (RV) command responds with the software part number and its revision level. The response is in the form shown below:

```
*92-nnnn-nn<xn>[cr]
(part number, revision level)
```

The part number identifies which product the software is written for, as well as any special features that the software may include. The revision level identifies when the software was written. You may want to record this information in your own records for future use. This type of information is useful when you consult Parker Compumotor's Applications Department.

Example

<u>Command</u>	<u>Response</u>
1RV	*92-7818-01A

The product is identified by 92-007818-01, and the revision level is identified by A.

S Motion	Stop			VALID Software Version A
SYNTAX <a>S	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO K, Q0, A, SSH		

Description

This command decelerates the motor to a stop using the last defined Acceleration <a> command. This command clears the command buffer (at the end of a move, if one is in progress). the Sequence Definition (XD) command is aborted and a time delay is terminated. If SSH1 is set the indexer will stop the current move but it will not clear the command buffer.

The Stop (S) command does not stop the motor in Velocity Streaming or Rate Multiplier (RM) mode. If you are in the RM mode, issue an Exit Velocity Profiling Mode (QØ) command to stop the motor.

Example

<u>Command</u>	<u>Description</u>
MC	Sets move in continuous mode
A1	Sets acceleration to 1 revs/sec ²
V1Ø	Sets velocity to 10 revs/sec
G	Executes the move (Go)
A5	Sets Acceleration to 5 revs/sec ²
S	Stops motor (motor comes to a stop at a deceleration rate of 5 revs/sec ²)

The S command is not buffered since it is an immediate command. As soon as the indexer receives the S command, it stops motion.

SS Status		Software Switch Function Status			VALID Software Version A
SYNTAX aSS	UNITS N/A	RANGE N/A	DEFAULT N/A	ATTRIBUTES Buffered Never Saved	
EXECUTION TIME			SEE ALSO SS commands		
RESPONSE TO aSS IS *nnnnnnnn					

Description

This command reports the status of the SS commands. From left to right, the eight-character response corresponds to SSA through SSH.

SSA Set-Up	RS-232C Echo Control			VALID Software Version A
SYNTAX <a>SSAn	UNITS N/A	RANGE n = 0, 1	DEFAULT 0	ATTRIBUTES Buffered Automatically Saved
EXECUTION TIME			SEE ALSO None	

Description

This command turns the RS-232C echo (transmission of characters received from the remote device by the IFX) on and off.

SSA0 = Echo on
SSA1 = Echo off

In the Echo On (SSA0) mode, characters that are received by the indexer are echoed automatically. In the Echo Off (SSA1) mode, characters are not echoed from the IFX. This command is useful if your computer cannot handle echoes. In a daisy chain, you must have the echo turned on (SSA0) to allow indexers further down the chain to receive commands.

Status commands do not echo the command sent, but transmit the requested status report.

Example

<u>Command</u>	<u>Description</u>
SSA1	Turns echo off (Characters sent to the indexer are not echoed back to the host.)

SSG Set-Up	Clear/Save the Command Buffer on Limit			VALID Software Version A
SYNTAX <a>SSGn	UNITS N/A	RANGE n = 0, 1	DEFAULT 0	ATTRIBUTES Buffered Automatically Saved
EXECUTION TIME			SEE ALSO LD	

Description

SSG0 = Clears command buffer on limit
SSG1 = Saves command buffer on limit

In most cases, it is desirable that upon activating an end-of-travel limit input all motion should cease until the problem causing the over-travel is rectified. This will be assured if all commands pending execution in the command buffer are cleared when hitting a limit. This is the case if `SSG0` is specified. If `SSG1` is specified and a limit is activated, the current move is aborted, but the remaining commands in the buffer continue to be executed.

Example

<u>Command</u>	<u>Description</u>
<code>SSG1</code>	Save buffer on limit
<code>A10</code>	Set acceleration to 10 revs/sec ²
<code>V5</code>	Set velocity to 5 revs/sec
<code>D25000</code>	Set distance to 25,000 steps
<code>G</code>	Executes the move (Go)
<code>O11</code>	Turn on outputs 1 and 2

If a limit switch is encountered while executing the move, outputs 1 and 2 will still go on.

SSH Set-Up		Clear/Save the Command Buffer on Stop			VALID Software Version A
SYNTAX <a>SSHn	UNITS N/A	RANGE n = 0, 1	DEFAULT 0	ATTRIBUTES Buffered Automatically Saved	
EXECUTION TIME			SEE ALSO LD, S		

Description	<code>SSH0</code> =	Clears command buffer on stop
	<code>SSH1</code> =	Saves command buffer on stop

In Normal Operation (`SSH0`) the Stop (S) command or a dedicated stop input will cause any commands in the command buffer to be cleared. If you select the Save Command Buffer On Stop (`SSH1`) command, a remote stop input or Stop (S) command will only stop execution of a move in progress. It will not stop execution of any commands that remain in the buffer.

Example

<u>Command</u>	<u>Description</u>
<code>SSH0</code>	Clears buffer on stop
<code>A10</code>	Sets acceleration to 10 revs/sec
<code>V5</code>	Sets velocity to 5 revs/sec
<code>D2000</code>	Sets distance to 2,000 units
<code>L50</code>	Loops 50 times
<code>G</code>	Executes the move (Go)
<code>T.5</code>	Pauses the motor 500 msec
<code>N</code>	Ends Loop
<code>S</code>	Stops motion

When you issue the **S** command, the indexer will clear the buffer and stop the move.

ST Programming	Shutdown			VALID Software Version A
SYNTAX <a>STn	UNITS N/A	RANGE r = 0, 1	DEFAULT 0	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO STM		

Description

The Shutdown (ST1) command rapidly decreases the motor current to zero. The system ignores move commands that you issue after the ST1 command. Torque on the motor is not maintained after you issue the ST1 command. *NOTE: Jumper LK4 must be removed on the SD drive card in order for ST1 to be recognized.*

The ST0 command rapidly increases the motor current to normal. Once you restore the current, you can execute moves.

This command is useful for reducing motor heating, and allows you to manually position the load. The motor position counter is set to the current position when you enter an ST0 command.

If you re-enable the drive using the ST0 command, you must wait 500 msec before using other commands.

For the ST command to work, the shutdown output from the indexer must be wired to the shutdown input of the drive. Also, there must be no jumpers installed on LK4.

Example

<u>Command</u>	<u>Description</u>
ST1	Shuts off current to the motor

STM Set-Up	Defaults			VALID Software Version A
SYNTAX <a>STMn	UNITS N/A	RANGE n = 0 - 3	DEFAULT 0	ATTRIBUTES Buffered Automatically Saved
EXECUTION TIME		SEE ALSO ST, Z		
RESPONSE TO aSTM IS *n				

Description

The Defaults (STM) command allows the user to automatically define a group of system parameters. In order for these parameters to take effect, a Reset (Z) command must be executed after the STM command. Then, upon power-up, a Go (G) command will execute motion with the parameter set by the STM command.

The **STM0** command is used for full and half step motor/drives; **STM1** and **STM2** are used for mini-step motor/drives; **STM3** is used for micro-step motor/drives.

SD Drives can operate only in STM0 mode.

STM3 mode will not be applicable to rack mount drive systems.

*Note that issuing the **STM** command (preceded by a device address) without the default number (0 - 3) lets you know which default is selected for that specific device. For instance, if you issue **4STM**, you will receive a response indicating which default (0-3) device number 4 is operating under.*

n = 0

<u>Parameter</u>	<u>Value</u>	<u>Units</u>
Velocity	.2	RPS
Accel	100	RPS/SEC
Distance	400	STEPS (= 1 revolution)
Mtr Res	400 (MR01)	STEPS/REV
Pos Mode	MPI	INCREMENTAL MODE
CG	8	N/A
Shutdown	0	Volts (Drive enabled)
Go Home	OSB1	N/A
Active Edge	CW	(Of Home Limit Input)
Active Level	Low	(Of Home Limit Input)

n = 1

<u>Parameter</u>	<u>Value</u>	<u>Units</u>
Direction Delay	30	Milliseconds (per direction change)
Velocity	.2	RPS
Accel	100	RPS/SEC
Distance	1000	STEPS (=1 revolution)
Mtr Res	1000(MR03)	STEP/REV
Pos Mode	MPI	INCREMENTAL MODE
CG	8	N/A
Shutdown	0	Volts (Drive enabled)
Go Home	OSB1	N/A
Active Edge	CW	(Of Home Limit Input)
Active Level	Low	(Of Home Limit Input)

n = 2

<u>Parameter</u>	<u>Value</u>	<u>Units</u>
Velocity	.2	RPS
Accel	100	RPS/SEC
Distance	2000	STEPS (=1 revolution)
Mtr Res	2000(MR21)	STEP/REV
Pos Mode	MPI	INCREMENTAL MODE
CG	8	N/A
Shutdown	0	Volts (Drive enabled)
Go Home	OSB1	N/A
Active Edge	CW	(Of Home Limit Input)
Active Level	Low	(Of Home Limit Input)

$n = 3$		
<u>Parameter</u>	<u>Value</u>	<u>Units</u>
Velocity	.2	RPS
Accel	100	RPS/SEC
Distance	25,000	STEPS (=1 revolution)
Mtr Res	25,000(MR10)	STEP/REV
Pos Mode	MPI	N/A
CG	8	N/A
Shutdown	5	Volts (Drive enabled)
Go Home	OSB1	N/A
Active Edge	CW	(Of Home Limit Input)
Active Level	Low	(Of Home Limit Input)

Example

<u>Command</u>	<u>Description</u>
STMØ	Sets power-up mode to $n = \emptyset$
Z	Resets the Drive to execute default parameters

SV		Servoing Parameter			VALID
Programming					Software Version A
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES	
<a>SVn	N/A	$n = 0 - 3$	None	Buffered Savable in Sequence	
EXECUTION TIME			SEE ALSO FS, ST		

Description

The Servoing Parameter (sv) command provides four different ways of simultaneously changing state of the motor shutdown and position maintenance functions. The four commands are as follows:

- SVØ** This command causes the position maintenance function to be turned off, but does not turn off motor power. It is identical in function to the FSCØ command.
- SV1** This command causes the position maintenance function to be turned off and the motor to be shut down simultaneously.
- SV2** This command causes the position maintenance function to be turned on and turns the motor power back on if it was turned off due to SV1 or SVØ command. The encoder position will be read and this newest position will be maintained.
- SV3** This command causes the position maintenance function to be turned on and turns the motor power on if it was turned off. The indexer will servo back to the rest position held before the position maintenance function was disabled.

Example

<u>Command</u>	<u>Description</u>
SV1	Simultaneously turns off Position Maintenance function and shuts down the motor.

T Programming	Time Delay			VALID Software Version A
SYNTAX <a>Tn	UNITS seconds	RANGE n = 0.01 - 99999.99	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME			SEE ALSO None	

Description The Time (T) command causes the indexer to wait the number of seconds that you specify before it executes the next command in the buffer. This command is useful whenever you need to delay the motor's actions or when you wish to move the motor in continuous velocity for preset time..

Example	<u>Command</u>	<u>Description</u>
	MN	Sets mode to normal
	A 5	Sets acceleration to 5 revs/sec ²
	V 5	Sets velocity to 5 revs/sec
	D25000	Sets distance to 25,000 steps
	T10	Pauses motor movement 10 seconds
	G	Executes the move (Go)
	T 5	Pauses the motor for 5 seconds after the move
	G	Executes the move (Go)

TR Programming	Wait For Trigger			VALID Software Version A
SYNTAX <a>TRann	UNITS N/A	RANGE n = 0, 1, or X	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME			SEE ALSO TS	

Description This command allows you to specify a trigger configuration to be matched before continuing execution of the move, where *nnn* corresponds to triggers 1, 2, and 3 respectively. The possible values for *n* are as follows:

n = 1	Wait for the trigger input to be high (opened)
n = 0	Wait for the trigger input to be low (grounded)
n = X	Ignore the trigger input

Example	<u>Command</u>	<u>Description</u>
	TR10X	Wait for input 1 to be opened and input 2 to be grounded before going on to the next command. Input 3 will be ignored.
	A10	Sets acceleration to 10 revs/sec ²
	V 5	Sets velocity to 5 revs/sec
	D25000	Sets distance to 25,000 steps
	G	Executes the move (Go)

TS Status	Trigger Input Status			VALID Software Version A
SYNTAX aTS	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO TR		
RESPONSE TO aTS IS *nnn				

Description

This command retrieves the state of the trigger inputs. The response is in the form nnn, where nnn reports the status of triggers 1, 2, and 3 respectively. The possible values for n are as follows:

n = 1 Input is high (opened)
n = 0 Input is low (closed)

The **TS** command is useful for checking the status of the trigger inputs when it appears the execution is being halted by a **TR** command. To make sure that your trigger pattern is met, you can check with the **TS** command.

Example

<u>Command</u>	<u>Response</u>
1TS	*101

Trigger bits 1 and 3 are high (opened) and Trigger bit 2 is low (closed)

U Programming	Pause and Wait for Continue			VALID Software Version A
SYNTAX <a>U	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO PS, C		

Description This command causes the indexer to complete the move in progress, then wait until it receives a Continue (C) to resume processing. Since the buffer is saved, the indexer continues to execute the program (at the point where it was interrupted). The indexer continues processing when it receives the C command. This command is typically used to stop a machine while it is unattended.

Example	<u>Command</u>	<u>Description</u>
	MN	Sets move to Normal mode
	A 5	Sets acceleration to 5 revs/sec ²
	V 5	Sets velocity to 5 revs/sec
	L Ø	Loops indefinitely
	D 25 600	Sets distance to 25,600 steps
	G	Executes the move (G)
	T 10	Waits 10 seconds after the move
	N	Ends loop
	U	Halts execution until the indexer receives the Continue command.

This command string pauses at the point where the U command is entered. A Continue (C) command causes execution to resume at the point where it was paused. In this example, the loop stops at the end of a move, and resumes when the indexer receives the C command. In reaction to the T10 command in the loop, there may be a 10-second delay before motion resumes after the C command is executed, depending on when the Pause and Wait for Continue (U) command is completed.

V Motion	Velocity			VALID Software Version A
SYNTAX <a>Vn	UNITS revs/sec	RANGE n = 0.01 - MAX (see table)	DEFAULT 0	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME			SEE ALSO A, D, G, GH, VS	

Description

The Velocity (**v**) command defines the maximum speed at which the motor will run when given the Go (**G**) command. The actual speed of the motor or output frequency of the indexer will vary, depending on the resolution setting (see table below).

Resolution Setting	Maximum Velocity*
200 steps/sec	325.42 rps
400 steps/sec	162.71 rps
1000 steps/sec	98.461 rps
2000 steps/sec	49.230 rps
5000 steps/sec	51.200 rps
10,000 steps/sec	54.857 rps
12,800 steps/sec	50.000 rps
25,000 steps/sec	21.943 rps

* These maximum velocity settings are higher than most stepper/microstepper drives can attain. This table is intended only to show the maximum velocity output of the IFX, NOT the maximum obtainable speed of the motor/drive combination being used with the IFX.

When the Go Home (**GH**) command is executed, the velocity is changed accordingly; subsequent moves should include a new **v** command unless the Go Home velocity is applicable.

Once you define the velocity, that velocity will be valid until you define another velocity, cycle AC power or issue a **z** command.

*NOTE: If the value specified for the **v** command is not valid, the IFX ignores that value and defaults to the value specified in the last **v** command.*

Example

<u>Command</u>	<u>Description</u>
MC	Sets move to continuous
A 5	Sets acceleration to 5 revs/sec ²
V 5	Sets velocity to 5 revs/sec
G	Go (Begin motion)

In preset mode, Mode Normal (**MN**) the maximum velocity may also be limited when the resulting move profile is triangular. In Mode Continuous (**MC**), a Go (**G**) command is completed—the indexer moves on to the next command in the buffer—once the specified velocity is reached.

VS Motion	Set Start/Stop Velocity			VALID Software Version A
SYNTAX <a>VSn	UNITS revs/sec	RANGE n = 0.01 - MAX (refer to table) or V command	DEFAULT 2	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME			SEE ALSO V, MR	

Description

See the velocity (v) command maximum velocity table for the VALID range for n. This is the minimum velocity employed by any MN or MC moves utilizing motor resolutions of 200 or 400 steps per revolution. The motor will jump immediately to the VS value before ramping to a v value. It is typically used to prevent a full-step motor from stalling at resonance point. If the VS value is greater than the v value for a move, the v value will be used regardless of the VS command. (It will move up to the v value immediately). This command should be entered before a velocity command is entered because the remote velocity input is altered by this command.

You must be careful with the Velocity (VS) you enter. If the start velocity is too high, the motor will stall.

Example

<u>Command</u>	<u>Description</u>
MR00	Sets motor resolution select to 200 steps/rev
MC	Sets mode continuous
VS1	Start/Stop velocity set to 1 rev/sec
V10	Sets velocity to 10 revs/sec
A2	Sets acceleration to 2 revs/sec ²
G	Executes the move (Go)

The motor will immediately reach 1 rps, then accelerate at 2 revs/sec² to 10 rps.

W1 Status	Signed Binary Position Report			VALID Software Version A
SYNTAX aW1	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO W3, PR		
RESPONSE TO aW1 IS *xxxxx				

Description Report back gives immediate binary representation of position relative to start of the current move. The format of the response is a four character response (nnnn) that is interpreted as a 32-bit binary number. The number must then be interpreted by the computer to give a numerical position in steps. The format is in 2's complement notation. Moves in the negative direction (CCW) will report back negative numbers (bit 31 is set to 1).

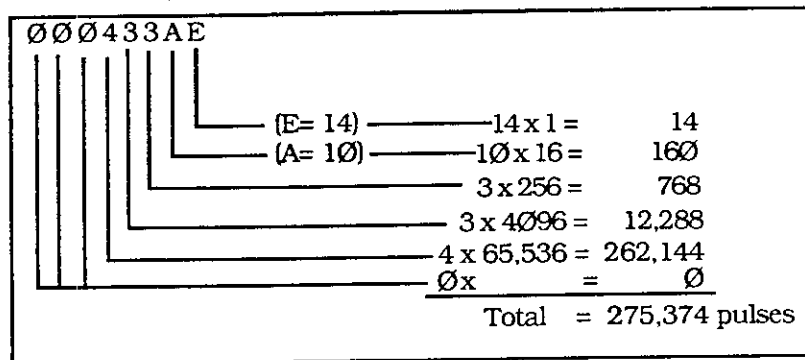
If you are using a terminal to communicate with the indexer, the response may not be a printable character. The response must be decoded using a computer.

This command is useful if you want to receive a position report while the motor is moving.

W3 Status	Hexadecimal Position Report			VALID Software Version A
SYNTAX aW3	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO W1, PR		
RESPONSE TO aW3 IS *xxxxxxxx				

Description This command will respond with an immediate hexadecimal character position report back in 2's complement format. The position response indicates the motor position relative to the current move. The format of the response is an asterisk, followed by an eight digit ASCII hexadecimal number.

Assume the response was *000433AE. The decimal value would be:



If the first digit of the response is an "F", then the response represents a "two's complement" negative number.

Follow the following steps to interpret a negative number (starting with "F"):

The binary approach:

1. Convert the hexadecimal response to binary form.
2. Complement the binary number
3. Add 1 to the binary result
4. Convert the binary result to decimal value with a minus sign placed ahead of the decimal value.

The computer approach:

Subtract the hexadecimal number from 168 (232) (4,294,967,296).

The easy way:

1. Chop off all the leading "F"'s, and convert to decimal
2. Convert and subtract the next largest power of 16.

Example:

If the indexer response to W3 is *FFFF9E58:

- | | | | |
|----------------------|-----------|---|---------|
| 1. Leave off the Fs: | 9E58 hex | = | 40,536 |
| 2. Subtract from 164 | 10000 hex | = | 65,536 |
| | Results | = | -25,000 |

Example

<u>Command</u>	<u>Response</u>
1W3	*FFFA19C

In the current move, you are at 24,163 steps from the initiation of the move.

XC Status	Sequence Checksum			VALID Software Version A
SYNTAX aXC	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO XD, XE		

Description This command computes the EEPROM checksum. After the unit has been programmed, the response can be used for system error checking. The response is three decimals (with a range of 000 - 255) followed by a [cr]. The number reported does not indicate the number of bytes programmed. This response is designed to be used for comparison. As long as the IFX is not re-programmed, the checksum response should always be the same.

Example

<u>Command</u>	<u>Response</u>
1XC	*149

XD Programming	Sequence Definition			VALID Software Version A
SYNTAX <a>XDn	UNITS Sequences	RANGE n = 1 - 7	DEFAULT none	ATTRIBUTES Buffered Never Saved
EXECUTION TIME		SEE ALSO XT, XE, XR, XRP		

Description This command begins sequence definition for a specific sequence. All the commands between the XD command and the Sequence Termination (XT) command will be defined as a sequence. The sequences will automatically be saved when the XT command is issued. If a sequence you are trying to define already exists, you must erase that sequence before defining it using the Erase Sequence (XE) command. Each sequence cannot be longer than 256 characters

Immediate commands cannot be entered into a sequence.

Example	Command	Description
	XE1	Erase sequence #1
	XD1	Define sequence #1
	MN	Sets mode normal
	A10	Sets acceleration to 10 revs/sec
	V5	Sets acceleration to 5 revs/sec
	D10000	Sets distance to 10,000 steps
	G	Executes the move (Go)
	XT	End defining sequence #1
	XR1	Execute sequence #1

The commands in sequence 1 are defined and executed.

XE Programming		Sequence Erase			VALID Software Version A
SYNTAX <a>XEn	UNITS Sequences	RANGE n = 1 - 7	DEFAULT None	ATTRIBUTES Buffered Never Saved	
EXECUTION TIME			SEE ALSO XD, XT, XR, XRP		

Description This command allows you to delete a sequence. The sequence that you specify (n) will be deleted when you issue the command.

As a good practice, you should delete a sequence before defining it.

Example	Command	Description
	XE1	Deletes Sequence 1
	XD1	Define Sequence 1

XP Set-Up	Set Power-up Sequence Mode			VALID Software Version A
SYNTAX <a>XPn	UNITS Sequences	RANGE n = 0 - 9	DEFAULT 0	ATTRIBUTES Buffered Automatically Saved
EXECUTION TIME		SEE ALSO XQ, XSP, XSR, XZ		

Description

This command will cause a single sequence or multiple sequences to be executed on power-up.

A value of 1-7 for *n* will result in the sequence whose value = *n* being executed on power up; control will then be passed to the RS232C interface.

A value of 8 for *n* will result in the sequence whose number appears on the sequence select inputs (**SEQ1 - SEQ3**) to be executed on power-up; control will then be passed to the RS232C interface.

A value of 9 for *n* will cause the sequence whose number appears on the Sequence Select inputs (**SEQ1 - SEQ3**) to be executed on power-up. When the first sequence is finished in XP9 mode, the IFX will scan the Sequence Select inputs again and execute the next sequence whose number appears on the inputs. This cycle will continue until a Stop (S) or Kill (K) command is issued, a limit is encountered, or the unit is powered down.

The possible settings for this command are as follows:

- n = 0** No sequence is executed on power-up
- n = 1 - 7** Sequence 1 - 7 is executed on power-up
- n = 8** Sequence Select inputs are read (single run) on power-up
- n = 9** Sequence Select inputs are read (continuous run) on power-up

In the XP9 mode, you can use the **XQ1** command to stop the IFX from selecting the next sequence until all the sequence select inputs are first opened. Refer to the Hardware Reference section for the remote sequence select input combinations.

Example	Command	Description
	XP 1	Execute Sequence #1 on power-up
	XE 1	Erase sequence #1
	XD 1	Define sequence #1
	LD 3	Disable CW & CCW limits
	A 1 0	Sets acceleration to 10 revs/sec ²
	V 5	Sets velocity to 5 revs/sec
	D 2 5 0 0 0	Sets distance to 25,000 steps
	G	Executes the move (Go)
	XT	End of sequence #1
	Z	Reset the indexer

The motor will move 25,000 steps every time you power-up or reset using the Z command.

XQ Set-Up		Sequence Interrupted Run Mode			VALID Software Version A
SYNTAX <a>XQn	UNITS N/A	RANGE n = 0, 1	DEFAULT 0	ATTRIBUTES Buffered Savable in Sequence	
EXECUTION TIME			SEE ALSO XP		

Description	n = 1 n = 0	Interrupted run mode is set (on) Interrupted run mode is reset (off)
--------------------	----------------	---

This command can be used only when stand-alone power-up sequencing in XP9 mode. In XP9 mode, if XQ1 is executed, the IFX will not accept a sequence select input until all sequence select inputs are OFF (open). After all lines have simultaneously been brought to a high state (OFF), the indexer will then read the sequence select lines and execute the sequence whose number appears there. This paused mode will continue until an XQ0 command is executed. You may use S or K command to stop sequence execution.

Example	Command	Description
	XE 1	Erase sequence #1
	XD 1	Define sequence #1
	LD 3	Disable CW & CCW limits
	XQ 1	Sets interrupted mode on
	XT	End Sequence #1
	XP 9	Sets power-up sequences as sequence select inputs
	Z	Resets the indexer to start sequence scanning

If you execute Sequence 1 upon power up by setting SEQ1-SEQ3 inputs properly, the interrupted run mode will be set. Sequence select input lines all need to go high (open) before selecting any other sequences.

XR Programming	Run a Sequence			VALID Software Version A
SYNTAX <a>XRn	UNITS N/A	RANGE n = 1 - 7	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO XE, XD, XT, XRP		

Description

This command loads a pre-defined sequence into the command buffer (clears the buffer first) and executes these commands as a normal set of commands. This command automatically recalls the sequence from EEPROM.

Before executing the specified sequence, the IFX is put into the power-on default state with respect to the following conditions:

- The values for distance, velocity, and acceleration are zeroed.
- The move mode is set to normal; the position mode is set to relative (incremental).
- The direction is set to CW.

An XR command can be used within one sequence to start execution of another sequence; however, all commands in the first sequence following the XR will be ignored (in this respect an XR acts like a GOTO not a GOSUB). An XR command placed within a loop will be ignored.

Example

<u>Command</u>	<u>Description</u>
XE1	Erase sequence #1
XD1	Define sequence #1
A10	Sets acceleration to 10 revs/sec
V5	Sets acceleration to 5 revs/sec
D10000	Sets distance to 10,000 steps
G	Executes the move (Go)
XT	End defining sequence #1
XR1	Execute sequence #1

Sequence 1 is defined and executed using XD1 and XR1 commands respectively

XRP Programming	Sequence Run With Pause			VALID Software Version A
SYNTAX <a>XRPn	UNITS N/A	RANGE n = 1 - 7	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO XR, XD, XT, XE, C		

Description This command is identical to the Sequence Run (XR) command, except that it automatically generates a pause condition. You must clear this condition with the Continue (C) command before the indexer executes the command buffer. The pause condition is asserted only if the sequence is VALID. This allows you to execute a sequence without the delay of buffering that sequence.

Example	<u>Command</u>	<u>Description</u>
	XE 5	Erases Sequence #5
	XD 5	Defines Sequence #5
	A 1 0	Sets acceleration to 10 revs/sec ²
	V 5	Sets velocity to 5 revs/sec
	D 1 0 0 0 0	Sets distance to 10,000 steps
	G	Executes the move (Go)
	XT	Ends defining Sequence #5
	XRP 5	Runs Sequence #5 with a pause
	C	Indexer executes Sequence #5

Upon issuing XRP5, Sequence #5 is entered into the command buffer, but is not executed. You must issue a Continue (C) command to execute Sequence #5.

XSD Status	Sequence Status Definition			VALID Software Version A
SYNTAX aXSD	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO XD, XE, XT		
RESPONSE TO aXSD IS *n				

Description This command reports back the status of the previous sequence definition (XD...XT). The response is 0 - 2. The VALID values and descriptions of possible responses are shown below:

- n = 0 Download O.K.
- n = 1 A sequence already exists with the number you have specified.
- n = 2 Out of memory. The sequence buffer is full.

The XSD command is useful for verifying that the last sequence definition attempt was successful.

Example

<u>Command</u>	<u>Response</u>
1XSD	*1

When you issued the XSD command to device #1, the response was 1, indicating that you need to erase the existing sequence (with the XE command) before defining that specific sequence.

XSP Status	Sequence Status Power-up			VALID Software Version A
SYNTAX aXSP	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Buffered Never Saved
EXECUTION TIME		SEE ALSO XP, XSR, XQ, XZ		
RESPONSE TO aXSP IS *n				

Description The Sequence Status Power-up (XSP) determines which, if any, sequence will be executed on power-up. After setting a power-up sequence using the Sequence Power-up (XP) command, you can check to make sure that proper sequence will be executed on power-up using the XSP command. The command reports back the sequence that the system will execute during power-up. The range of sequences is 0 - 9.

Example Command Description
 1XSP *3 (Indicates that sequence number 3, if it exists, will be executed upon power-up or reset)

XSR Status	Sequence Status Run			VALID Software Version A
SYNTAX aXSR	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO XR, XRP		
RESPONSE TO aXSR IS *n				

Description This command allows you to check whether or not the last sequence you issued was executed successfully without hitting limits, Stop (S), or Kill (K). The VALID values and descriptions of each response are shown below.

- 0 = Running
- Non-zero = Not running:
 - 1 = In a loop
 - 2 = VALID sequence
 - 3 = Erased
 - 4 = Bad checksum

Example Command Response
 1XSR *0 (Sequence ran O.K.)

XSS Status	Sequence Status			VALID Software Version A
SYNTAX aXSSn	UNITS Sequences	RANGE n = 1 - 7	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO XD, XT, XE		
RESPONSE TO aXSS IS *n				

Description This command reports whether the sequence specified by n (representing one of the sequences 1 - 7) is empty, has bad checksum, or is OK.

0 = Empty
1 = Bad Checksum
3 = O.K.

This command is useful to see if the particular sequence exists and if that portion of memory has been corrupted.

Example

<u>Command</u>	<u>Response</u>
1XSS1	*0 (Sequence 1 of device 1 is not defined.)

XT Programming	Sequence Termination			VALID Software Version A
SYNTAX <a>XT	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Buffered Never Saved
EXECUTION TIME		SEE ALSO XD, XE, XR, XRP		

Description The XT command is a sequence terminator. This command flags the end of the sequence currently being defined. Sequence definition is not complete until this command is issued.

Properly defined sequences are saved into EEPROM automatically by issuing this command.

Example	Command	Description
	XE1	Erase sequence #1
	XD1	Define sequence #1
	MN	Sets to mode normal
	A1Ø	Sets acceleration to 10 revs/sec ²
	V5	Sets velocity to 5 revs/sec
	D25ØØØ	Sets distance to 25,000 steps
	G	Executes the move (Go)
	XT	End sequence definition and save the sequences to EEPROM

XU Status		Upload Sequence			VALID Software Version A
SYNTAX aXUn	UNITS Sequences	RANGE n = 1 - 7	DEFAULT None	ATTRIBUTES Buffered Never Saved	
EXECUTION TIME			SEE ALSO XD, XE, XT, F		
RESPONSE TO aXUn IS Contents of sequence n					

Description This command sends the contents of sequence *n* to the host computer via RS-232C interface, terminated by a [cr]. The command of that sequence will appear on the computer CRT. All command delimiters in the sequence will be sent out as spaces (2ØH). Any device identifiers that were included in the original sequence will also be eliminated (they are not stored in the sequence).

Note: When using a daisy-chain, this command must be used cautiously as the contents of the sequence will go to all controllers in the loop between the indexer that is uploading and the host. The F command may be useful in this context to turn off communication on units you are not uploading from.

Example	Command	Description
	2F	Turns off communication to unit #2
	3F	Turns off communication to unit #3
	1XU1	Uploads sequence #1 from unit #1

XZ Programming	Set Power-up Sequence to Zero			VALID Software Version A
SYNTAX <a>XZ	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO z, XSP, XSR, XP, XQ		

Description

This command sets the power-on sequence number to zero (thereby disabling sequence activation on power-on). The purpose of this command is to handle the situation in which the set power-up sequence has a checksum error, causing control to pass to the error routine (flashing LED); only immediate commands are active in this error state. In testing externally controlled sequences, this command can also be used to disable continuous-run sequencing (followed by a **Z** command to reset the indexer).

The main difference between **XZ** command and **XPØ** command is that **XZ** is executed immediately and **XPØ** is entered into a buffer. If you are already running a sequence, use this command.

Example

None

Y Programming	Stop Loop			VALID Software Version A
SYNTAX <a>Y	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO L, N		

Description

The Stop Loop (**Y**) command takes you out of a loop when the loop completes its current pass. This command does not halt processing of the commands in the loop until the indexer processes reach the last command of the current loop. At that time, the indexer executes the command that follows the End Loop (**N**) command. You cannot restart the command loop unless you enter the entire command structure, including the Loop (**L**) and End Loop (**N**) commands.

Example	Command	Description
	L	Loops indefinitely
	A10	Sets acceleration to 10 revs/sec ²
	V5	Sets velocity to 5 revs/sec
	D25000	Sets distance to 25,000 steps
	T2	Waits 2 seconds
	G	Executes the move (Go)
	N	Ends loop
	Y	Stops loop

The loop requires the motor to move 25,000 steps CW and then wait for 2 seconds. The loop terminates at the end of the loop cycle it is executing when it receives the Y command.

Z		Reset			VALID Software Version A
Programming					
SYNTAX <a>Z	UNITS N/A	RANGE N/A	DEFAULT None	ATTRIBUTES Immediate Never Saved	
EXECUTION TIME 1,000ms			SEE ALSO S, K		

Description

The Reset (Z) command is equivalent to cycling AC power to the indexer. This command returns all internal settings to their power-up values. It clears the command buffer. Like the Kill (K) command, the Z command immediately stops output pulses to the motor.

When you use the Reset command, the indexer is busy for 1,000 milliseconds and ignores all commands.

Any changes that you do not save before issuing this command will be lost.

This command sets all position counters to zero.

Example

Command	Description
1 Z	Resets indexer with address 1