ACCU ELECTRIC MOTORS INC

USA: (888) 932-9183 CANADA: (905) 829-2505

- Over 100 years cumulative experience
- 24 hour rush turnaround / technical support service
- Established in 1993

The leading independent repairer of servo motors and drives in North America.

Visit us on the web:

www.servo-repair.com

www.servorepair.ca

www.ferrocontrol.com
www.sandvikrepair.com
www.accuelectric.com

Scroll down to view your document!

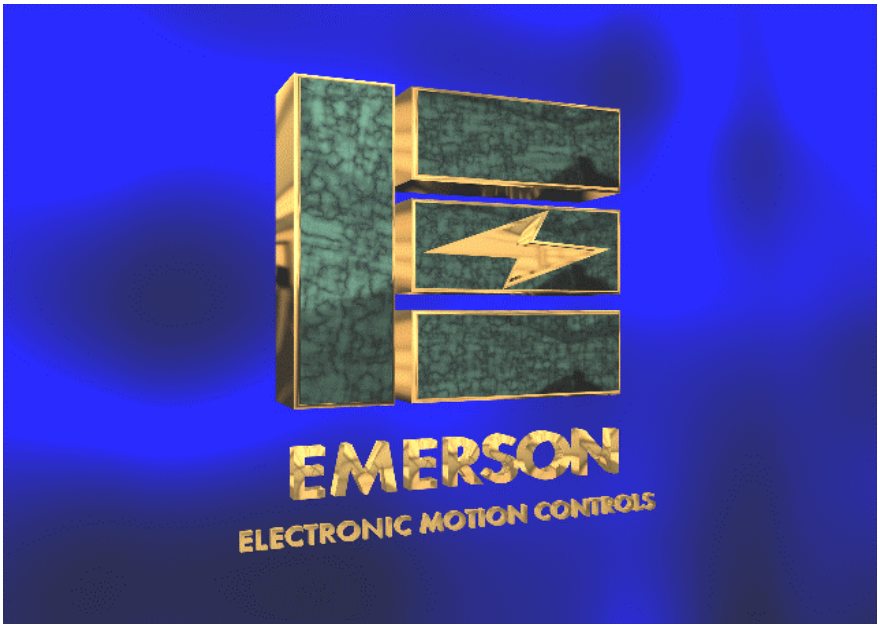
For 24/7 repair services :

USA: 1 (888) 932 - 9183
Canada: 1 (905) 829 -2505

Emergency After hours: 1 (416) 624 0386

Servicing USA and Canada

PCM-22 Rotary Knife Controller Operators Manual



P/N 400274-01
Revision: A2
Date: 1/15/96

Customer Services:

EMERSON EMC offers a wide range of services to support our customers' needs. Listed below are some examples of these services.

Service Support (612)-474-8833

Emerson Electronic Motion Control's products are backed by a team of professionals who will service your installation wherever it may be. Our customer service center in Minneapolis Minnesota, is ready to help you solve those occasional problems over the telephone.

Our customer service center is available 24 hours a day for emergency service to help speed any problem solving. Also, all hardware replacement parts, should they ever be needed, are available through our customer service organization.

Need on-site help? Emerson EMC provides on-site service, in most cases, the next day. Just call Emerson EMC's customer service center when on-site service or maintenance is required.

Training Services (612)-474-1116

Emerson EMC maintains a highly trained staff of instructors to familiarize customers with Emerson EMC products and their applications. A number of courses are offered, many of which can be taught in your plant upon request.

Application Engineering

An experienced staff of factory application engineers provide complete customer support for tough or complex applications. Our engineers offer you a broad base of experience and knowledge of electronic motion control applications.

EMERSON BBS (612) 474-8835.

PCX Software updates can be obtained from the Emerson BBS. 300 - 9600 Baud, N, 8, 1.

Table Of Contents

Overview

This manual provides setup and programming information for the PCM-22 Application Module. It is important that you become familiar with the FX Drives manual (P/N 400282-00) which provides the background information needed to setup and configure the FX amplifier using PCX 6.X software.

The PCM-22 has the drive setup parameters stored in the PCM module which allows it to be transferred to another FX amplifier (of the same size) without losing your setup parameters. Firmware updates are done on the PCM module eliminating the need to replace EPROM's on the FX amplifier.

Features

Not all of the features available with the PCM-22 will be used in every application. For example, there are 15 programming functions available with the PCM-22, however, a typical rotary knife program will use only a few of them. The features provided by the PCM-22 Rotary Knife Controller included:

- Programmed with easy-to-use PCX software.
- Allows an FX Drive to accurately position and match speed based on data received from a master axis.
- Calculates cycle profile based on user defined parameters while maintaining position and velocity.
- Parameters can be changed "on-the-fly".
- Two internal counters for each cycle: Product cycle counter and product batch counter.
- Stores up to 16 cycle motion profiles in memory to be called up for execution.
- 93 user assignable I/O functions.
- Expands I/O line capacity of FX Drive with eight optically isolated input lines and four optically isolated outputs.
- Stores parameters in non-volatile memory to allow the unit to be moved to another FX Drive of the same size without losing data.

Firmware Revision

The firmware revision on a PCM-22 module necessary for all of the programming features in this manual is A4 or higher. You can find the revision number of your module by looking at the serial number sticker located on the side of the module.

Theory Of Operation

The PCM-22 allows an FX Drive to monitor a product's position (with respect to its motor position) and produce uniformly spaced cuts or perforations on the product. The object of Rotary Knife Control is to cause the follower axis (defined as the rotary knife) to produce one or more cycles of motion for every cycle of motion produced by the master axis (defined by an encoder).

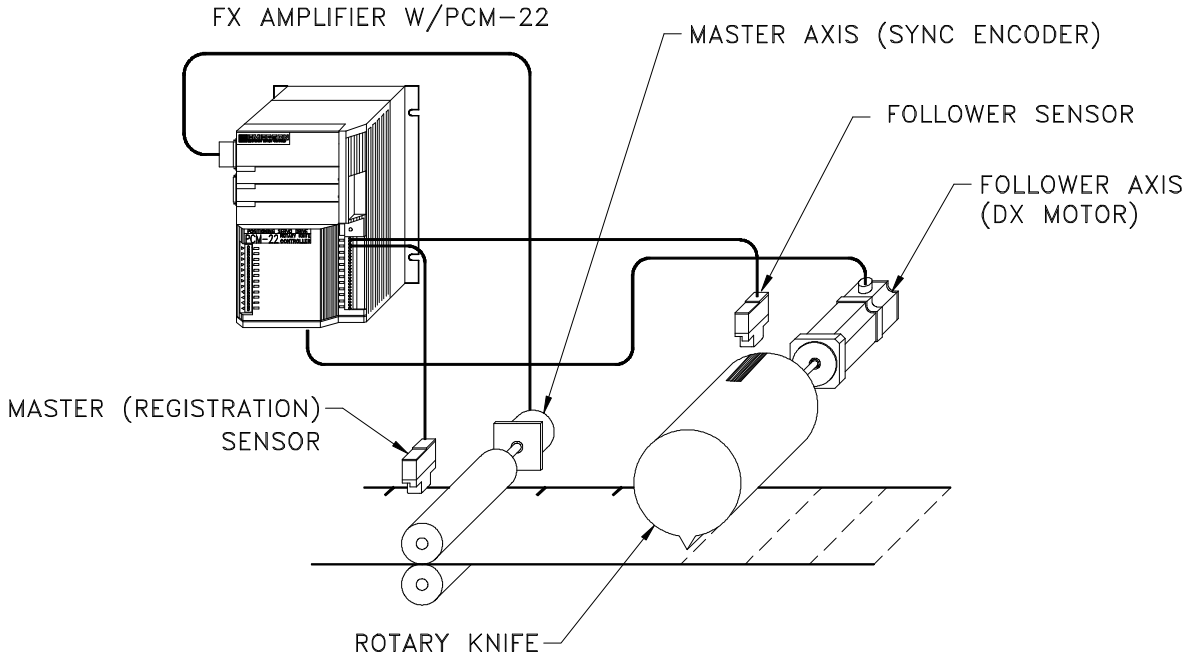


Figure 1 PCM-22 Rotary Knife System Example

Referring to Figure 1, in order to maintain proper synchronization between the two axes, the FX Drive monitors the start points of both the master and the follower axes. The follower axis will slightly speed up or slow down as each registration mark passes a sensor. If both the follower and the master axis are in sync, no speed correction is made.

The PCM-22 can "learn" new parameters based on sensor information which allows it to automatically compensate for new product lengths on the fly. For example, if the distance between registration marks on the master axis increases from 8.25" to 8.45", the PCM-22 will adjust the follower axis parameters to compensate for the increase.

PCM-22 Terminology

The terms used in PCX to define the system parameters and how the knife (perforator or crimper) operates are described below.

- **Working Segment Width:** Referring to the motion profile below, the cutting portion of the rotary knife profile is termed the "Working Segment". During the working segment the blade is in contact with the material that is being cut.
- **Working Ratio:** The Working Ratio defines how fast the blade moves relative to the product when the cut is being applied. Usually this ratio is close to 1:1. The "working segment" is a length during which the Working Ratio is applied. Figure 2 shows a typical motion profile for a PCM-22 Rotary Knife System.

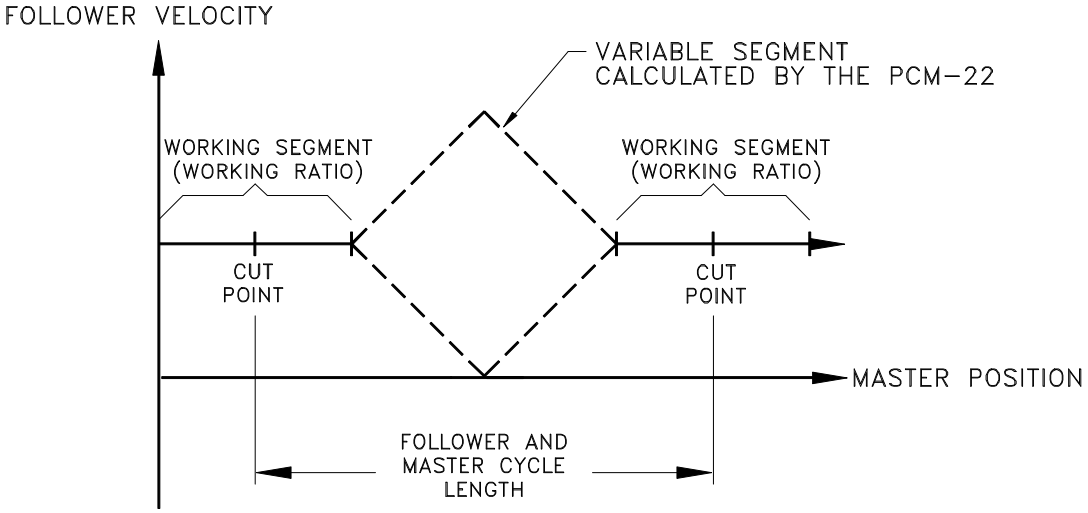


Figure 2 PCM-22 Motion Profile Example

- **Variable Segment:** The PCM-22 will automatically calculate the motion profile needed for the remaining portion of the cycle in order to be properly positioned for the next cut. This portion is called the “Variable Segment”.
- **Master Length:** Referring to Figure 3, the Master Length, in user units is the distance that the master axis (encoder or upstream drive) moves during each master cycle.
- **Follower Length:** Referring to Figure 3, the Follower Length, in user units, is the distance the follower axis (DX motor) moves during each follower cycle.
- **Working Segment Offset:** The Working Segment Offset is the distance that the follower sensor or zero point lags behind the center of the working segment (cut point).
- **Master Zero Point:** The master zero point is defined by the master sensor and is usually the leading edge of a product or a registration mark.

The master length, follower length, their respective phase relationships and offset positions and the working segment, can all be changed during

operation using inputs. The working ratio can also be adjusted using PCX software or with serial commands.

When any of these parameters are changed, the PCM-22 calculates and completes the profiles necessary to accommodate the new parameters.

Any changes made before the center of the working segment will take place during the next cycle. During the working segment the working ratio will always be maintained.

If parameters are entered that either cannot be accommodated within one cycle or cannot be accommodated at all, the **cycle limit reached** output (if assigned) will be activated until the limit has been solved.

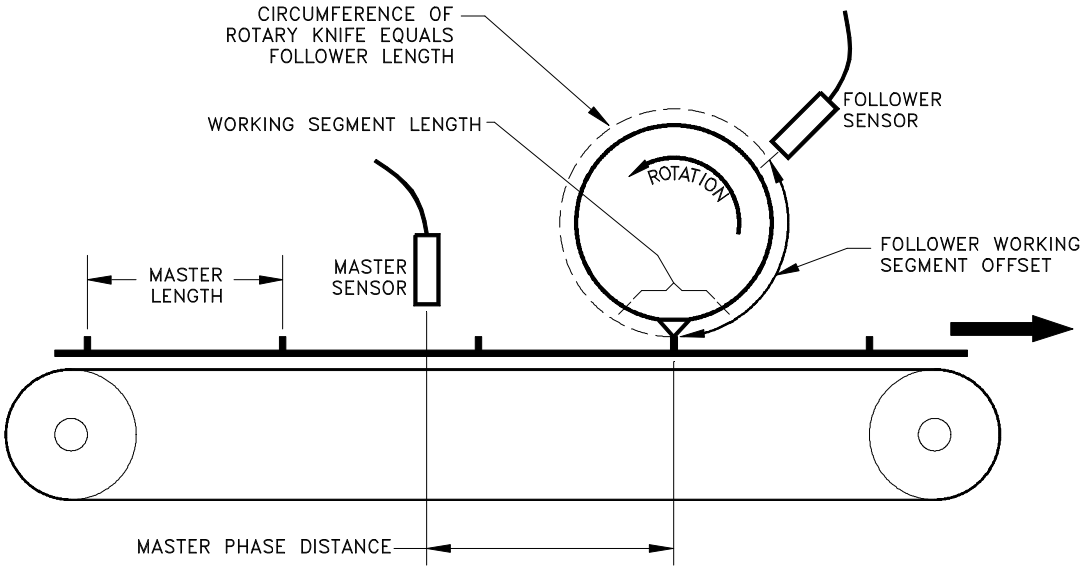


Figure 3 Rotary Knife System

- **Master Phase Distance:** The master phase distance parameter (located in the Master Cycles screen) defines the distance from the sensor to the center of the follower working segment (cut point).

In Figure 3 the master phase distance may be greater than the master length. If the cut point is on the registration mark, the exact length from the master sensor to the master cut point may be entered as master phase distance.

In this example the rotation of the rotary knife is counterclockwise and the product movement is from left to right.

This parameter is critical to system operation and must be entered properly. The remainder of the master phase distance divided by the master length is the actual phase offset.

If you want the beginning of the master length at a position other than at the beginning of the master length, you must factor in the distance from the center of the working segment to this parameter. Lengthen the sensor distance to move the center of the working segment before the start point of the product; shorten the distance to move the center of the working segment after the start point of the product.

Installing Your PCM Module

PCM modules attach to the front of any FX Amplifier with two locking arms. All electrical connections (except I/O) are made via the 48 pin connector.

All PCM modules are equipped with 8 input lines and 4 output lines, doubling the I/O capability of the standard FX drive. All inputs and outputs are optically isolated for +10.5 to +30 vdc operation. Each input and output line has 2 screw terminals associated with it to provide for either current sinking or current sourcing operation.

The first 8 pairs of terminals (numbered 13 - 20) are inputs and the last 4 pairs of terminals (numbered 21 - 24) are outputs. The outputs are capable of sinking or sourcing 200 mA. You must limit the output current to less than or equal to 200 mA per line.

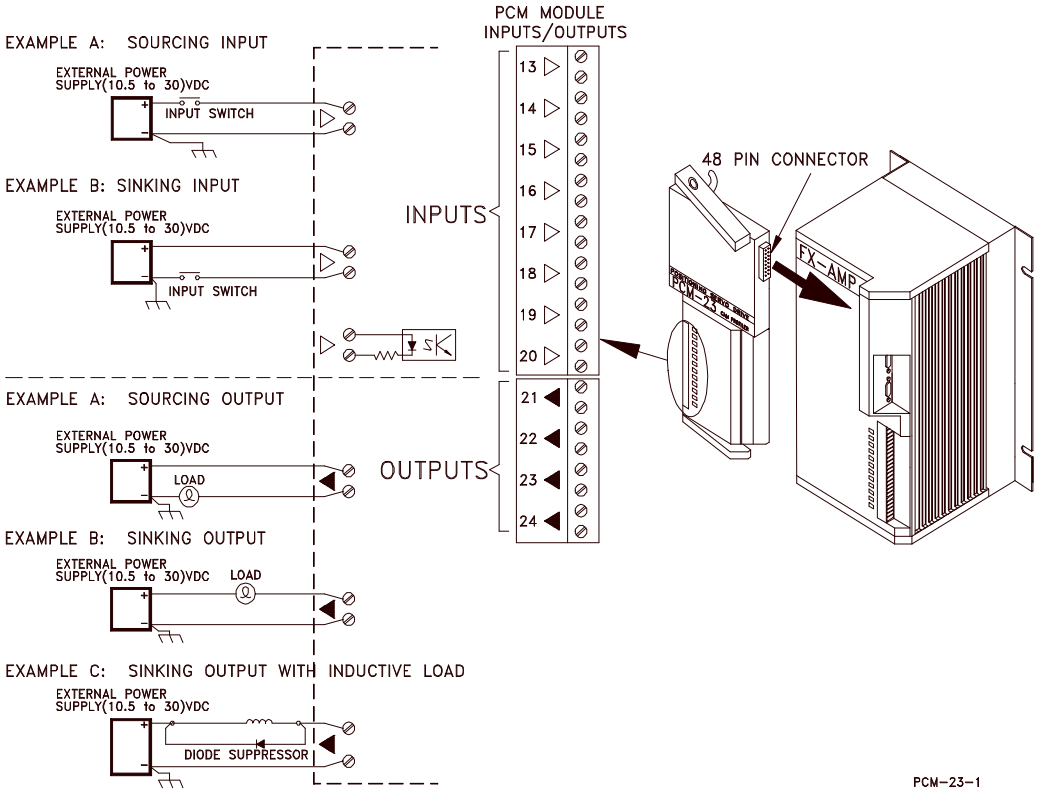


Figure 4 Installing Your PCM Module

Input Wiring

Inputs require an external voltage source for operation. This voltage source must be in the range of 10.5 to 30 VDC. The negative (-) lead of the external power supply must be connected to the enclosure or safety ground.

Output Wiring

Outputs are similar to inputs in that they can be connected to either sinking or sourcing type loads. Outputs require an external isolated supply voltage, usually the same supply as the inputs. Each output can sink or source 0.2 amps DC. The negative (-) lead of the external power supply must be connected to enclosure or safety ground.

If inductive loads such as DC relay coils are connected to the outputs, a suppression diode must be installed in parallel with the load coil with its cathode towards the positive end of the external supply.

Sensor Setup

In a typical rotary knife system the rotary knife will be designated as the follower axis and the product or material conveyor as the master axis. A sensor on the knife (or follower axis) can be used if the exact position of the blade changes due to mechanical configuration of the system.

Master and follower axes sensors must use FX Drive high speed inputs 1 and 2 only. Input functions number 45 and 46 (Zero Master Cycle and Zero Follower Cycle respectively) must be setup to operate with the appropriate sensors (see Input Functions) on input lines 1 and 2.

Follower Sensor

In most applications a follower sensor should be used to prevent the working segment and other parameters from drifting away from it's desired position due to accumulated length error. When entering data in the PCX software, certain values such as follower length can be rounded off which will cause the follower's position to drift over time. Maintaining follower position is critical in most applications.

The follower zero point must be established after initial power-up by executing a home cycle that will bring the knife into it's zero position or by applying a signal to the zero follower cycle input when the rotary knife is in it's proper zero position. The center of the working segment is also established at this time.

Master Sensor

In most applications a master axis sensor is critical to maintaining product registration. Like the follower axis, the master's position could also drift over time due to accumulated length error. The master axis sensor can be eliminated in applications where the number of external encoder steps in the Master Cycle Length is repeatable. This would mean that there is no product stretch or shrinkage or slip over time.

Whether the master axis is defined by a sensor or not, it must be "zeroed" at least once. The zero point can be established anytime by placing the master in its proper zero position and then applying a signal to the Zero Master Cycle input. No follower cycle motion will begin until the master is zeroed.

Programming The PCM-22

The first step in programming your FX Drive with a PCM-22 is to configure the drive using the 5 menu options in the Drive Configuration screen. The Drive Parameters, Limits, Inputs and Outputs screens are explained in the PCX section of the FX manual (P/N 400282-00).

The Master Axis screen is where you identify the **Signal Source** of the master axis and setup synchronization parameters.

The next step is to define the motion parameters in the Define Motion screen. The first three menu selections, Jog, Home and Index, are explained in the PCX section of the FX manual (P/N 400282-00).

The next two selections, Master Cycles and Follower Cycles, define the relationship between the master and follower axes.

The last step is to Create a program (or programs) that use the "Y" command (Execute Follower Cycle) with other programming functions as logical steps within a program.

The menu hierarchy charts on the following pages show, in the darker boxes, the menus that are added or changed in PCX 6.X software when a PCM-22 module is employed with an FX Drive.

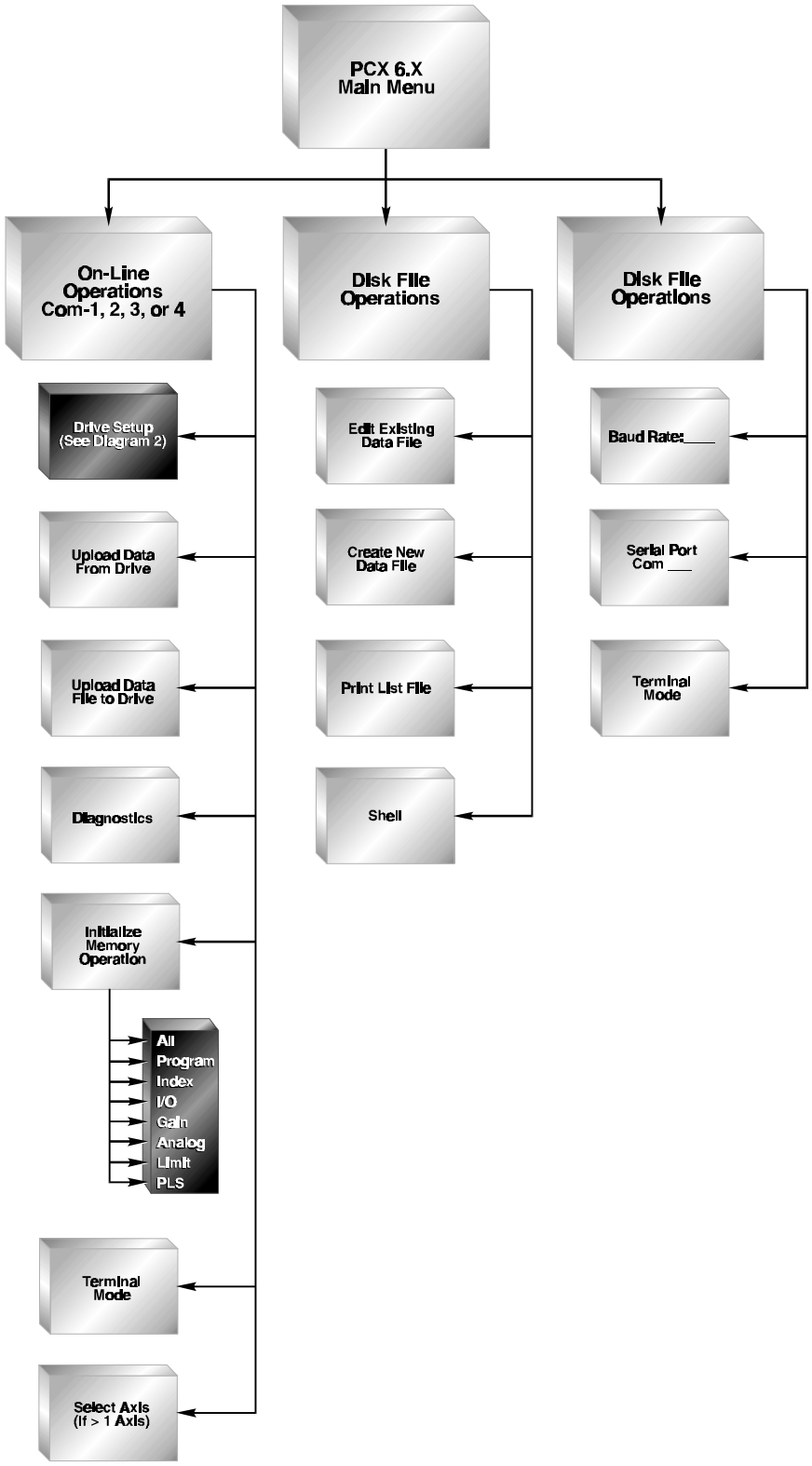


Figure 5 PCX Hierarchy Diagram #1

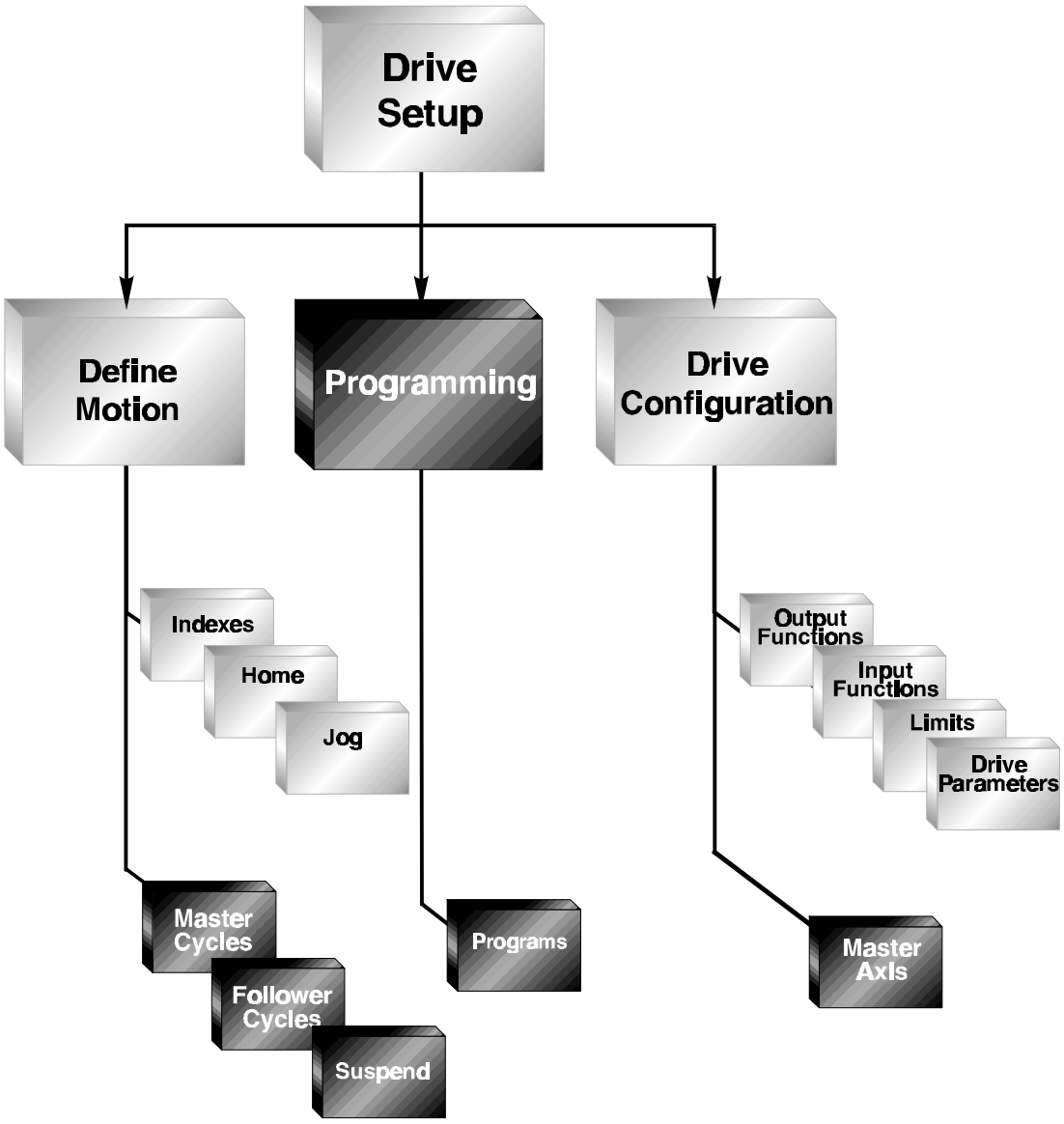


Figure 6 PCX Hierarchy Diagram #2

The PCX features necessary to run a Cycle Profile are listed below in the order in which they should be addressed. Each feature listed must be set up before you can use them in a Cycle Profile application.

Individual setup information for each feature is covered on the following pages.

1. **Drive Parameters:** The first step in setting up the PCM-22 is to define the parameters in the Drive Parameters screen. A description of the parameters in this screen can be found in the PCX section of the FX Drives manual (P/N 400282-00). The Drive Parameters screen can be found by selecting Drive Setup, Drive Configuration then Drive Parameters.

2. **Master Axis (Signal Source):** The next step is to identify the **Signal Source** of the master axis. In addition to running Cycle Profiles the PCM-22 also has the ability to run synchronized indexes. In this screen, only the Signal Source, Signal Polarity, Signal Interpretation and Signal For Sync Output From parameters are used by the PCM-22 when running Cycle Profiles. The Master Axis screen is found by selecting Drive Setup, Drive Configuration then Master Axis (see page 11).
3. **Input and Output lines:** The next step is to assign any input or output functions to the I/O lines on the FX amplifier and the PCM-22 module. There are several input and output function used specifically for Cycle Profiles. The input and output function screens can be found by selecting Drive Setup, Drive Configuration then either Inputs or outputs (see page 25 for inputs and page 34 for outputs). For a list of basic I/O functions see input and output functions in section 1 of the PCX 6.X manual P/N 400282-00).
4. **Master Cycle:** The next step is to define the parameters of the Master Cycle screen. (see page 16).
5. **Follower Cycle:** The next step is to define the parameters of the Follower Cycle screen (see page 19)
6. **Cycle Profile Programs:** The last step is to create a program using the “Y” programming function, Execute Cycle Profile (see page 38).

Master Axis Description

This screen is where you identify the **Signal Source** of the master axis. In addition to running Cycle Profiles the PCM-22 also has the ability to run synchronized indexes, however, in this screen only the **Signal Source**, **Signal Polarity**, **Signal Interpretation** and **Signal For Sync Output From** parameters are used by the PCM-22 when running Cycle Profiles.

An external master axis becomes the time base for motion control of the follower axis (your FX Drive equipped with a PCM-22). The basis of operation is determined by the relationship of the external master axis encoder or drive to the follower axis motor.

The master axis is typically an Emerson SCS series encoder, an upstream drive or it can be a customer supplied encoder of any line density yielding the appropriate steps per revolution.

The master axis encoder must not exceed 210KHz (see Master Maximum Velocity on page 14.

The base number system used for the FX Drive is binary and 12 bits. The smallest resolution is one part in 4096. Since this number may be difficult to work with because of units, the drive electronics allow for a conversion to any number from 200 to 25,000, with the default being 4000.

Master Axis Setup

To setup a relationship between the master axis and the follower axis, select the Master Axis Option from the Drive Configuration menu.

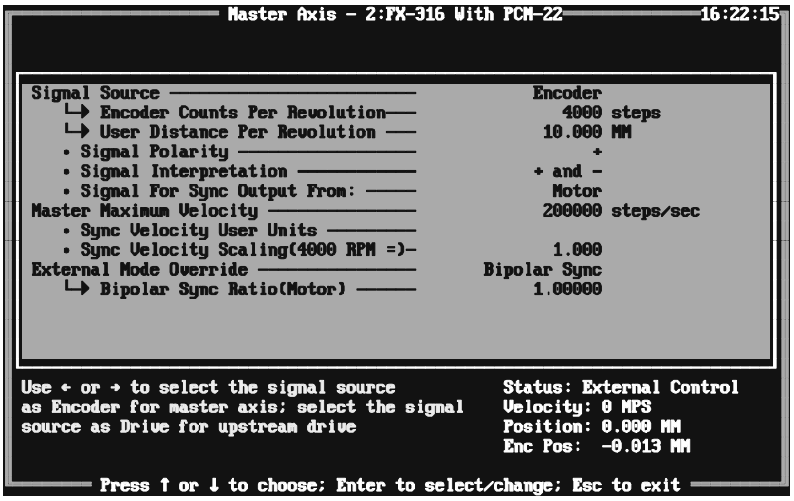


Figure 7 Master Axis Screen.

Signal Source

Select the origin of the signals (drive or encoder) used for master axis positional information. When a follower axis (PCM-22) is receiving its synchronization source from an FX drive lead axis, the count source becomes the binary 4096 counts per turn.

If your master axis is an **Encoder**, there will be 5 parameters that you need to setup that pertain to the signal source (see master axis screen above). If you select **Drive** for your signal source, the **Encoder Steps Per Revolution** parameter will not appear.

Encoder Steps Pre Revolution

*This parameter is not available if the Signal Source is set to **Drive**.*

This parameter is the number of encoder setps per revolution of the synchronization encoder. The default value is 4000 steps per revolution. Minimum value is 0 and maximum value is 65535.

User Distance Pre Revolution

The default value for this parameter is 4000 "user units" per revolution (see definition of user units in the PCX section of the FX Drive manual P/N 400282-00). Minimum value is 1 and maximum value is 65535.

Signal Polarity

This feature defines the direction of the synchronization encoder that corresponds to a positive master position change. Clockwise is indicated with a (+); counterclockwise is indicated with a (-). CW and CCW motion of the master axis is defined while facing the encoder shaft.

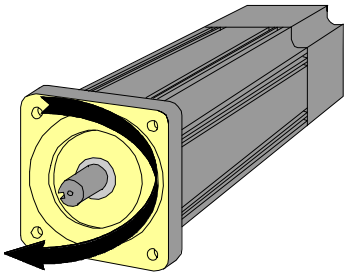


Figure 8 CW Motor Rotation

Signal Interpretation

The signal interpretation feature allows you to define how the follower reacts to clockwise and counterclockwise motion of the synchronization encoder.

The recommended signal interpretation mode when using a PCM-22 module is mode #4, COMP +. Signal interpretation modes apply only to ratio synchronization.

Mode #1 (+ and -): When the master axis moves either CW or CCW, the follower axis will move in its commanded direction. If the master axis changes direction the follower axis will continue in the original commanded direction. The follower axis will not reverse direction.

Mode #2 (+): The follower will only react to synchronization pulses when the master axis runs in the CW direction. CCW master axis pulses are ignored.

Mode #3 (-): The follower will only react to synchronization pulses when the master axis runs in the CCW direction. CW master axis pulses are ignored.

Mode #4 (COMP +): The follower will only react to synchronization pulses when the master axis runs in the CW direction. The drive counts the pulses received in the CCW direction and ignores that exact number of CW pulses before follower motion in the CW direction occurs.

This feature compensates for master axis motion in the opposite (CCW) direction. For example, the master stops, then inadvertently backs up due to conveyer slack, etc.

Mode #5 (COMP -): The follower axis will only react to synchronization pulses when the master axis runs in the CCW direction. The drive counts the pulses received in the CW direction and ignores that exact number of CCW pulses before follower motion in the CCW direction occurs. This feature compensates for master axis motion in the opposite (CW) direction.

Signal For Sync Output From

If you select **motor**, your FX drive will output a sync signal to the next FX drive based on the performance of its own motor. If you select **upstream drive**, your FX drive will output a signal that comes from the motor of the preceding amplifier.

Encoder pulses are passed to all amplifiers in the synchronization chain. How you answer this question has no effect on the integrity of the encoder signal. The **signal source** of the next FX drive and PCM-22 in the synchronization chin must be set to **drive** in order to operate with this signal.

Master Maximum Velocity

The Master Maximum Velocity, Sync Velocity User Units, and External Mode Override parameters are not used by the PCM-22 when running Cycle Profiles.

However, they would be used if you were running synchronized indexes or jogs.

The master maximum velocity is the maximum frequency that the master axis signal source is expected to produce when running at its' full speed. To calculate the master maximum velocity, use the following formula:

$$Master\ Max Velocity = \frac{(MV)(MS)}{60\,Sec \diagup Min}$$

MV = Master Axis Maximum Velocity (RPM's)
MS = Master Steps/Rev

If encoder is master: MS = (Encoder Line Density)*(4)

If drive is master: MS = 4096

For example: The master axis is a 1000 line encoder and rotates at a maximum speed of 3000 rpm, and, when quadratured, produces 4000 steps per revolution. Then:

$$\frac{(3000\,RPM)(4000\,Steps \diagup \mathrm{Re}\,v)}{60\,Seconds} = 200{,}000\,Steps\ Per\ Second$$

Maximum frequency into PCM-22 module cannot exceed 210 Khz or steps/second.

This value is the master encoder velocity at which synchronized time base and real time base are equal. This parameter is used to calculate actual follower velocity while running in synchronized time base.

Sync Velocity User Units

This parameter sets the units to be associated with all sync velocities. User units can be any three letter combination, such as IPS (inches per second), RPM (revolutions per minute), FPM (feet per minute), etc.

Sync Velocity Scaling (Max RPM Equals)

This parameter sets the sync velocity entry that will produce maximum velocity of the drive when the master axis signal source is at maximum velocity. When an index is running in sync time base, the velocity is specified in user units.

The default value is 1.000. A setting of 0.500 in an index velocity means the drive will accel to half of maximum velocity.

External Mode Override

External mode override works in conjunction with input function #38 to override the current mode of operation. When input function #38 is assigned and active, the drive will exit its' current operating mode and default to the mode selected with this parameter. There are three modes of operation, **analog velocity**, **analog torque** and **bi-polar sync**.

Analog Velocity/Torque

When set to analog velocity or torque mode, the drive will respond to a conventional 10VDC signal. In either of the two analog modes a 10VDC signal is equated to either (CW) or (CCW) maximum programmed velocity or maximum full peak torque rating.

If you enable analog torque mode and apply a voltage between 0 and 10 VDC to the command connector the FX drive will attempt to produce torque equal to:

$$Max\ Torque\left(\frac{Applied\ Voltage}{10\ VDC}\right) = Actual\ Torque$$

If there is no physical resistance to the torque at the motor shaft, the motor will very quickly accelerate to maximum speed.

Bi-polar Sync

When set to bi-polar sync, this parameter allows for direct movement of the FX drive motor ratioed to the sync encoder. This mode moves the motor shaft in direct response to encoder or drive signals. This means immediate velocity without ramping in your FX drive. If the master axis accelerates very quickly, your FX drive (follower axis) will try to follow just as quickly. Too fast an acceleration by the master axis could result in "F" (Following Error) faults for the follower axis.

Encoder Counts Per Motor Rev.

This parameter is the relationship of the **follower axis** position to the **master axis** position. For example, if you set this value to 3, for every 1 count of the master axis there will be 3 counts of the follower axis. Thus, the bipolar sync ratio would be 3:1.

Master Cycles

A master cycle is defined as the distance between successive master zero points. The PCM-22 allows the drive to position and maintain a phase relationship of its motor to an independent axis. The FX Drive, with the PCM-22 acting as the follower axis, will produce one or more cycles of motion for every cycle produced by the master axis.

The system example in Figure 2 shows a flat product (such as roll feed paper or plastic) that moves through a set of pinch rollers which feeds back positional information to the PCM-22 via a sync encoder.

The master registration sensor references the positional relationship of the master to the follower axis using registration marks on the material. The follower sensor provides registration information for the follower axis using a registration mark on the rotary knife.

In the above system example, the master cycle length is 5 inches and the master phase distance is 25 inches. In this example the cut will be directly on the registration mark and there would be one master cycle completed for every follower cycle. However, if you wanted the cut to be 1 inch to the left the registration mark you would enter 24 inches as the master phase distance.

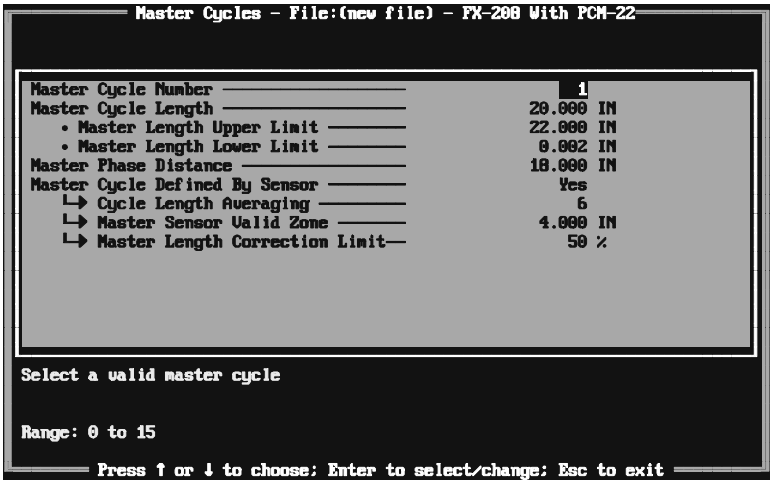


Figure 9 Master Cycles Screen

Master Cycle Number

This parameter identifies the master cycle currently being defined. Up to 16 different master cycles can be defined (0 to 15). Any master cycles can be used with any follower cycle.

Master Cycle Length

This parameter defines the length of the master cycle in user units. This is the distance that a master axis (encoder or upstream drive) moves during each master cycle.

Master length upper limit

This parameter sets the value of the longest master cycle length that will be allowed. This limit is imposed on length changes made serially, via inputs or sensors.

Master length lower limit

This parameter sets the value of the shortest master length that will be allowed. This limit is imposed on length changes made serially, via inputs or sensors.

Master phase distance

Sets the distance (in user units) that the master sensor lags behind the master cut point (center of the working segment). This value may be greater than the master cycle length. See Figure 2-3.

Master Cycle Defined by Sensor

This parameter determines how the master axis cycle zero position is defined. If yes is entered, the master axis cycle will be defined by a sensor on the zero master axis cycle input line. The sensor will then be used to determine the beginning of each master axis cycle. The encoder (or upstream drive) will provide positional information of the master cycle length.

If no is entered, the master axis cycle is defined only by the master cycle length. The zero master axis cycle input must be manually set once to define the zero position of the master axis cycle. Once the zero position has been set the master axis length is determined by encoder counts only.

Cycle Length Averaging

This parameter set the number of prior measured master length which are averaged to determine the operational master length.

$$\frac{\left((\mathrm{CLA}-1)\,x\,\mathrm{CAL}\right)+\mathrm{ML}}{\mathrm{CAL}}=\mathrm{NL}$$

Where: **CLA** is the Cycle Length Averaging.
CAL is the Current Average Length.
ML is the measured length.
And **NL** is the New Average Length.

For example, if (8) is entered, the new length will be:

$$\frac{\left((8-1)\,x\,\mathrm{CAL}\right)+\mathrm{ML}}{8}=\mathrm{NL}$$

As demonstrated by the equations above, the lower the average the faster length errors are corrected for, However, setting this value too low will adversely affect the PCM-22/drives ability to quickly correct for phase errors.

Length variations and/or errors are slowly averaged into the operational length.

Master Sensor Valid Zone

This parameter defines the area surrounding the zero position in which a Zero Master Cycle Sensor input signal will be considered valid. For example; if you enter ±10mm here, any input signal which appears on the Zero Master Cycle Sensor input which appears before -10mm or after +10mm of the last master sensor zero point will be ignored.

This is useful in applications where registration marks are printed in the same feed path as other printing (such as advertising, logos, instructions, etc.). The drive will ignore all master cycle sensor inputs and outputs except those which appear within the valid zone.

Master Length correction limit

The master length correction limit is used to reduce the amount of length correction performed during each master cycle. For example: When an input is received from the zero master cycle sensor input and error is found to exist between the current cycle length and the length determined by the new sensor input. The PCM-22 takes this percentage of that error which it uses to correct the length error.

This parameter does not filter the length changes due to sensor inputs. It only limits the rate at which length errors are corrected for. When operating at high line speeds rapid corrections for length errors may be undesirable especially with high inertial loads. This parameter may be used to slow the rate of correction of the length errors and allow smother operation.

If you entered 50% here, and the error was 1.0 inch between consecutive master cycles, the PCM-22 would use 50% (or .50 inches) of the error to correct the master length.

Follower Cycles

The next step is to define the follower cycle. A follower cycle is the distance between successive follower zero points. The PCM-22 allows the drive to position and maintain a phase relationship of its motor to an independent axis. The FX Drive, with the PCM-22 acting as the follower axis, will produce one cycle of motion for every cycle produced by the master axis.

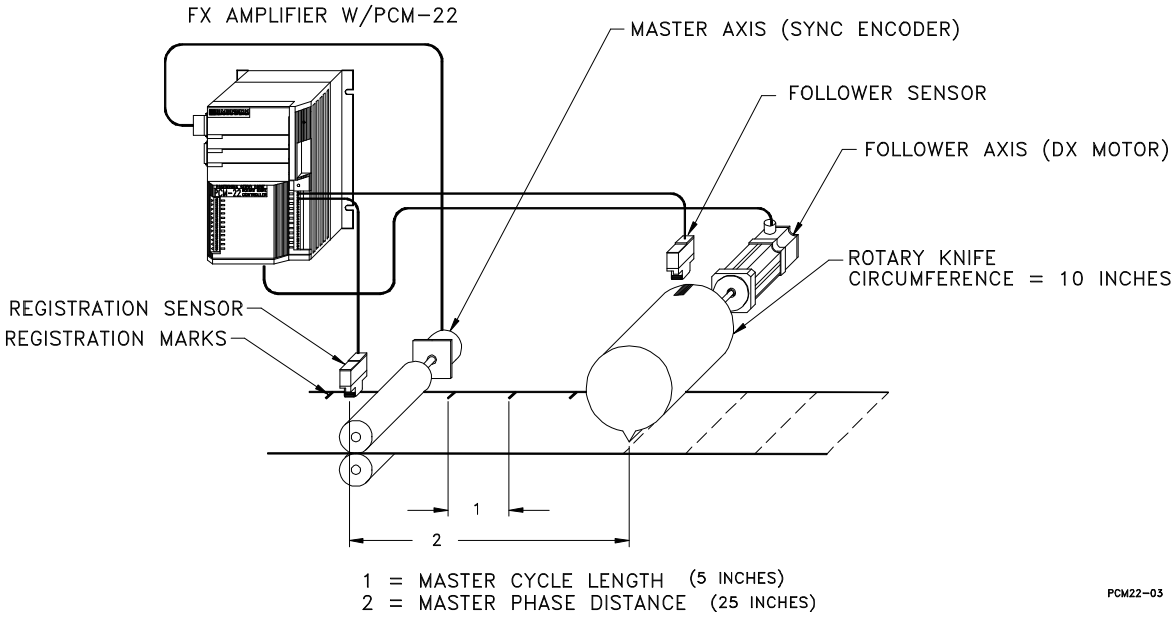


Figure 10 System Example

The system example above shows a flat product (such as roll feed paper or plastic) that moves through a set of pinch rollers which feeds back positional information to the PCM-22 via an encoder. The follower sensor provides positional information for the follower axis using a registration mark on the rotary knife.

Typically, the follower sensor would be used as a home sensor for the follower axis. The registration sensor references the positional relationship of the master to the follower axis using registration marks on the product. In this system the motor provides feedback information of the follower axis's position.

The **follower cycle length** is 10 inches, the master cycle length is 5 inches and the master phase distance is 25 inches. In this example the cut will be directly on the registration mark and there would be one master cycle completed for every follower cycle. Typically, there will be one follower cycle for every master cycle.

Follower Cycles Screen

The Follower Cycles screen is found by selecting Drive Setup, Define Motion then Follower Cycles. In this screen additional parameters appear when you reach the bottom of the screen using the down arrow or enter keys.

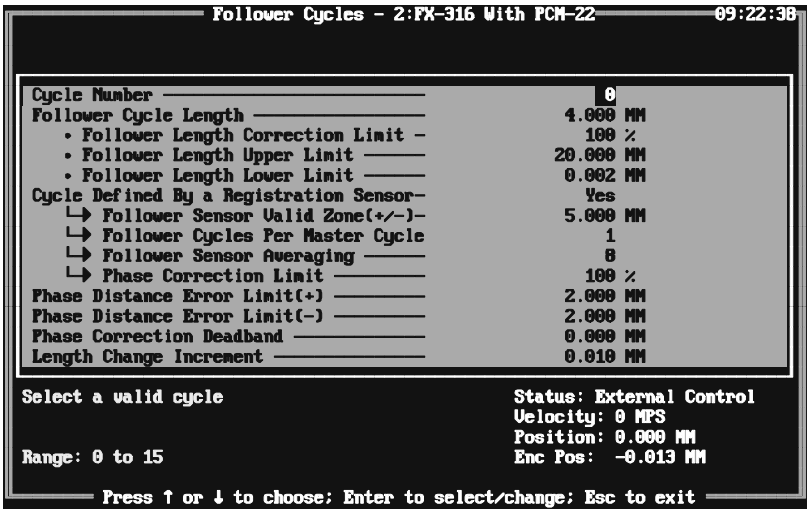


Figure 11 Follower Cycles Screen

Cycle Number

Up to 16 (0 through 15) different Follower Cycles can be defined. Each Follower Cycle requires a Master Cycle to be attached. The same Master Cycle can be used for all 16 Follower Cycles.

Follower Cycle Length

Follower Cycle Length is the distance in user units that the follower motor moves in order to complete one cycle of motion. This parameter is programmed in user units of the amplifier i.e., inch, millimeters, etc.

Follower Length correction limit

The follower length correction limit is used to reduce the amount of length correction performed during each follower cycle. For example: When an input is received from the zero follower cycle sensor input and error is found to exist between the current cycle length and the length determined by the new sensor input. The PCM-22 takes this percentage of that error which it uses to correct the length error.

This parameter does not filter the length changes due to sensor inputs. It only limits the rate at which length errors are corrected for. When operating at high line speeds rapid corrections for length errors may be undesirable especially with high inertial loads. This parameter may be used to slow the rate of correction of the length errors and allow smother operation.

If you entered 50% here, and the error was 1.0 inch between consecutive follower cycles, the PCM-22 would use 50% (or .50 inches) of the error to correct the follower length.

Follower length upper limit

This parameter sets the value of the longest follower length that will be allowed. This limit is imposed on length changes made serially, via inputs or induced via sensors.

Follower length lower limit

This parameter sets the value of the shortest follower length that will be allowed. This limit is imposed on length changes made serially, via inputs or induced via sensors.

Cycle Defined by Registration Sensor

If "Yes" is entered here, the follower zero position will constantly be updated by an external sensor using the zero follower cycle input (input #46). If "NO" is entered here, the follower zero position must be set once at the beginning of the cycle.

Follower Sensor Valid Zone

This function defines a window either plus or minus of the defined zero position that a Zero Follower Cycle Sensor input signal will be considered valid. For example; if the user enters ±10mm here, any input signal which appears on the Zero Follower Cycle Sensor input which appears before -10mm or after +10mm will be ignored.

This is useful in applications where registration marks are printed in the same feed path as other printing (such as advertising, logos, instructions, etc.). The amplifier will ignore all inputs except those which appear within the valid zone.

Follower cycles per master cycle

This parameter sets a ratio between the master and follower axis. The value entered here sets the number of follower cycles that will occur during each master cycle. For example, in Figure 3 (on page 5) there will be one master cycle per registration mark and one knife cut or follower cycle per master cycle.

Follower Sensor Averaging

This parameter determines how much the new measured length affects the current operational follower length. For example, if (1) is entered the new length will be:

$$\frac{\text{Old} + \text{ML}}{2} = \text{NL}$$

If (8) is entered here the new length would be:

$$\frac{(7\text{xOld}) + \text{ML}}{8} = \text{NL}$$

Where: **Old** is equal to the original operational follower length.

ML is equal to the current measured follower length.

And **NL** is the new follower length calculated by the PCM-22.

Therefore, the lower the average the faster length errors are corrected for. Setting this value lower will adversely affect the drives ability to quickly correct for phase errors.

Phase Correction Limit

This parameter is used to advance or retard the rate at which corrections are made for errors between the master and follower sensor phase positions.

For example: If a new master sensor indicates an error in the follower position in relation to the master position the PCM-22 will try to correct for the error within one cycle.

The parameter is used to scale both the master and follower errors individually and the sum of the two scaled values is the total phase positioning error for the system.

To properly set this parameter, decrease or increase the percentage value until the master and follower phase errors are within the desired range.

It is normal to have small amounts of error. However, large errors should not occur during cycle operation. Large errors at high speed could cause some undesirable acceleration rates. Lowering this percentage value would reduce the size of the phase corrections to the system.

Phase Distance Error Limit (+)

This parameter designates a maximum positive deviation from the phase distance or working offset. The you can assign a phase distance error limit output (function # 44) to be used as an indicator. Exceeding this limit will not stop the follower axis, however, you may employ this output with outside logic to generate an appropriate command.

Phase Distance Error Limit (-)

This parameter is identical to phase distance error limit (+), except that it detects a negative deviation from the phase distance or working offset.

Phase Correction Deadband

This parameter defines an area of the follower cycle in degrees. Any phase error or working offset error that occurs within this phase correction deadband area will be ignored.

Length Change Increment

This parameter selects the amount of distance incremented and/or decremented when the Increment Follower and Master Length and/or Decrement Follower and Master Length inputs are toggled (inputs 64,65, 66 and 67 respectively).

Master Cycle Number

Designates which Master Cycle this Follower Cycle will be synchronized to.

Batch Count

This parameter is used in conjunction with the Batch Count Reached output (output #43). The batch count reached output becomes active when the batch counter reaches this number.

The batch count is incremented automatically when the cycle counter reaches the programmed cycle count. When the batch counter reaches the programmed batch count output #43 is activated and the batch counter is cleared. The output will be de-activated automatically on the next cycle count. The batch counter may also be cleared by input #75, Reset Batch Counter.

Cycle Count

This parameter is used in conjunction with the Cycle Count Reached output (output #42). The cycle count reached output becomes active when the cycle counter reaches this number.

The cycle counter is incremented each time a cycle is completed if input #76 (cycle count hold) in not active. When the cycle counter reaches the programmed cycle count, the batch counter is incremented and the cycle counter is cleared. The cycle counter and cycle count reached output may also be cleared by input #74 (reset cycle counter). The cycle count reached output is cleared when the next cycle is completed.

Working Segment Width

This parameter sets the length of the segment for which the working ratio is applied. The position of this segment is centered around the cut position.

Working Ratio

This parameter is the follower to master ratio during the working segment.

Working Offset

This parameter sets the distance between the follower sensor or zero point and the center of the working segment (cut point). When the cycle is executed, the follower is assumed to be at the sensor position (or zero point).

If the follower zero point falls within the programmed working segment, the cycle will not execute because the working segment ratio could not possibly be maintained. For this reason the working segment offset must be greater than 1/2 the working segment length at startup.

Upper Working Segment Limit

This parameter limits the maximum length of the working segment.

Lower Working Segment Limit

This parameter limits the minimum length of the working segment.

Upper Working Ratio Limit

This parameter limits the maximum working ratio.

Lower Working Ratio Limit

This parameter limits the maximum working ratio.

Offset/Phase Fine Increment

This parameter is used to select the amount of increment/decrement distance which occurs when the follower offset or master phase increment/decrement input are toggled while the fine phase adjustment input is active.

Offset/Phase Coarse Increment

This parameter is used to select the amount of increment/decrement distance which occurs when the follower offset or master phase increment/decrement input are toggled while the fine phase adjustment input is inactive.

Inputs/Outputs

This section list all the input and output functions which are available with a PCM-22. Refer to the FX Drives manual (P/N 400282-00) for complete instructions on assigning the input and output functions listed below. The FX Drives manual also provides information and instructions on input polarity, input response time, input filtering and high speed inputs.

Input Functions

Inputs require an external voltage source for operation. This voltage source must be in the range of 10.5 to 30 VDC.

0 Index Initiate

Executes the selected index. If index select function #14 is not assigned, index 00 will initiate. (See **Indexes** in the PCX section of the FX manual P/N 400282-00).

1 Home(0) Initiate

Executes home cycle #0. (See **Home Cycles** in the PCX section of the FX manual P/N 400282-00).

2 Clear To Initiate

Clear to initiate is intended to prevent unwanted motion during certain operations of a machine cycle. This function, when assigned to an input line, prevents repeated and undesired attempts at index initiate, program initiate or home initiate.

The default polarity is (-) or normally off and the default signal sensitivity is **edge sensitive**. The signal sensitivity can be changed to be **level sensitive** using the FI serial command. For edge sensitivity send FI=0, for level sensitivity send FI=1.

Operation of this input with normally off (-) polarity and edge sensitivity (FI=0) is as follows:

- On power up index, home and program initiate functions are blocked.
- Sending a clear to initiate input signal will allow one each index, home and program initiate. To allow another index, home and program initiate after the first one, you must send another clear to initiate input signal to the drive.
- If the clear to initiate input pulse high duration extends past the initiate of a home, index or program motion, the following home, index or program initiation will be blocked until another clear to initiate signal is sent.

Operation of this input when set to level sensitivity (FI=1) is as follows:

- When clear to initiate input is high (On), unlimited initiates are allowed. when low (Off) initiates are inhibited.

3 Hold

This function stops index motion or home motion as long as the input is active. Index or home motion resumes when this input is released. Decel time is set with the stop/hold decel time parameter in limits dialog box. Acceleration time used to resume motion is the accel time in the index you are interrupting.

4 Jog

Starts and stops a jog in the direction specified by the jog direction input function #7. Defaults to the programmed jog fast velocity.

5 Jog CW

Starts and stops a clockwise jog. If input function #8 is not assigned the jog speed defaults to the jog fast velocity.

6 Jog CCW

Starts and stops a counterclockwise jog. If function #8 (slow jog) is not assigned the jog speed defaults to the jog fast velocity.

7 Jog Direction

Sets the direction for the jog input function #4 (active = CCW).

8 Slow Jog

Causes the jog slow velocity to be used when Jog, Jog CW, or Jog CCW are executed.

9 CW Travel Limit

Forces clockwise velocity to zero and inhibits any further clockwise motion. The drive will display an **"L"** on the diagnostics display. When released the "**L"** or travel limit is reset automatically.

10 CCW Travel Limit

Forces counterclockwise velocity to zero and inhibits any further counterclockwise motion. The drive will display an **"L"** on the diagnostics display. When released, the **"L"** or travel limit is reset automatically.

11 Stop Motion

Stops motion according to the stop/hold decel time in the limits screen. A stop input will terminate indexes, homes or programs (returning the drive to external control). Motion resumes only if given another motion command. The stop input is active during all modes of operation.

12 Inhibit (drive)

Prevents all motion by disabling the drive's bridge transistors and engaging the holding brake. The drive is re-enabled when input is removed.

13 Brake Override

Releases the fail-safe brake when Inhibit is active. This allows the motor to be rotated by hand.

14 Index Select

Allows you to choose an index using input lines and sets the starting input line number for a given number of index select lines. For example: If your system requires four index select lines you could assign any four consecutive input lines. If you assigned the index select function to input line #5, then entered a 4 as the number of index select lines, the index select function would now be assigned to input lines #5, 6, 7 and 8.

The format of the index select Lines is binary. That is, the first line assigned has the value of 1, the second a value of 2, the third a value of 4, the fourth a value of 8, and so on. The index number selected is the sum of the values of the lines activated.

In the previous paragraph, with four lines assigned beginning with line 5, if no lines are active, index 0 is selected and it will be initiated when you activate Index Initiate. If you activate lines 5 and 7 and the Index Initiate line simultaneously, you will initiate Index 5 (line 5 has a value of 1 and line 7 has a value of 4). If you activate all four lines and the Initiate Index line simultaneously, you are selecting Index Number 15 (1 + 2 + 4 + 8 = 15). Index numbers available in the basic drive are indexes 0 through 31.

15 Home Sensor (0)

Defines which input line is used for the home sensor of home #0.

16 Feed Sensor (0)

Provides a sensor input for feed sensor indexes.

17 Clear End Of Index Output

Clears the end of index output. If this input is not assigned, the end of index output will automatically clear when the next motion occurs. If this input is assigned, the output will not be cleared until this input is active.

18 Clear End Of Home

Clears the end of home output. If not assigned, the end of home will automatically clear when the next motion occurs. If this input is assigned, the output will not be cleared until this input is active.

19 Clear End Of Index Count Output

Clears the end of index count output. If not assigned, the end of index count will clear when the next motion occurs. If this input is assigned, the output will only be cleared by a clear end of index count input.

20 Clear Sensor Limit Distance Hit

Clears the sensor limit distance hit output. The sensor limit distance is programmed by PCXWin or serial commands. This input is used in feed to sensor and registration indexes.

21 Reset Fault

Duplicates the function of the reset switch on the front of the digital drive. It will clear non-fatal drive faults.

22 Zero Position

Sets the current position of the motor to zero. All absolute positions will be referenced to this set zero position.

23 Zero Display

This function zeroes all position query commands (FP and CP), Absolute position is unaffected.

24 Remember Position

Stores the current position of the drive in a **return** absolute index. (default = index 0). This index number may be changed in the suspend function dialog box.

25 Return To Position

Returns to the position defined by remember position input. This input uses the **return index** which must be set up as an absolute index.

26 Clear End Of Sequence

Clears the **end of sequence output**. If not assigned, the end of sequence output will automatically clear when the next motion occurs. If this input is assigned, the output can only be cleared by activating a clear end of sequence input.

27 Clear Programmable Output

Clears a programmable output. Each assigned Programmable Output requires a separate Clear Programmable Output line which corresponds to the PGOs (output function #12) in the same order. If four lines are selected for PGOs, this input function requires four lines to clear the PGOs.

28 Wait/Jump Inputs

Used with the **wait for input** program function to stop program execution until the input line assigned with External Input is activated, and with the Jump program command to change program flow based on line conditions. Sets the starting Input Line number for multiple Wait/Jump Input Lines. If your system requires four (4) wait for external input lines, you could assign any four consecutive input lines.

29 Program Initiate

Executes the selected program. Edge sensitive function. Must be taken from low level to high level to be recognized.

30 Program Select Lines

This input function allows you to select a program with input lines and sets the starting input line number for a number of **program select lines**. For example: If your system requires four program select lines, you could assign any four consecutive Input Lines. If you assign the program select function to Input Line #5, then entered a 4 as the

number of Program Select Lines, the Program Select function would now be assigned to Input Lines #5, 6, 7 and 8. You could now select programs 0 through 15.

The default format of the program select lines is binary. That is, the first line that you assign has a value of 1, the second a value of 2, the third a value of 4, the fourth a value of 8 and so on. The program number selected is the sum of the values of the lines activated.

Selecting program 5 in this example would require activating lines 5 (value 1) and 7 (value 4) while activating the line assigned to input function 29, program Initiate. If no lines are activated, this is the same as selecting program #0.

31 Suspend Program

Stops the execution of a program until the input line assigned with resume is activated.

32 Resume Program

Resumes the execution of a suspended program.

33 Abort Suspend

Clears the **in suspend output** and returns the drive from a suspend cycle to normal operation without the need to "resume" and completion of the suspended program. (Also see suspend/resume functions in program section of the PCM module manual).

34 Clear End Program

Clears the **end of program output**. If not assigned, the end of program output will automatically clear when further motion is initiated. If this input is assigned, an input is required to clear the output.

35 Clear End Program Count

Clears the **end of program count output**. If not assigned, the end of program count output will automatically clear when the next motion occurs. If this input is assigned, an input is required to clear the output.

36 Clear All Prog Outputs

When the line assigned to this function is activated, this function will clear (set to **off**) all programmable output lines. (See Input function #27 to clear individual programmable output lines.)

37 Feed Sensor (1)

Defines which input line will be used for the feed sensor indexes.

38 External Mode Override

An active input allows direct and immediate movement of the FX drive motor using the bi-polar sync setup in the master axis dialog box.

39 Home (1) Initiate

Executes home cycle #1.

40 Home (1) Sensor

Defines which input line will be used for Home Sensor #1.

41 Index Direction

When you assign this input to an input line the **use index direction input** check box in the index dialog box will become active. If you activate the input line assigned to this function at the same time as an index that has the **use index direction input** box checked, the index initiated will run in the opposite direction to that in the index dialog box.

42 Torque Jog

This does not apply to serial jog velocity.

This function enables torque jog when you simultaneously activate the input line assigned to this function and a jog, jog CW or a jog CCW. The commanded torque used for torque jog is equal to the speed set for fast jog divided by the maximum motor speed, times the drive peak torque rating. For example: If fast jog velocity is set at 1000 RPM on an FX-340 and torque jog was enabled, the torque produced would be equal to (1000 RPM/3000 RPM) * (80 lb.-in.), or 26.67 lb.-in.

If there is no resistance on the motor shaft when Torque Jog is enabled, the motor will quickly go to its maximum speed.

43 Time Base Override

Any motion initiated when this input is active will use the time base selected in the drive parameters dialog box. (See Drive Parameters in the PCX section of the FX Drives manual P/N 400282-00).

44 Clear Torque Limit Output

Clears a torque limit hit output after a torque limit has been exceeded. torque limit hit output is set up via the output functions dialog box.

45 Zero Master Cycle

This input establishes master zero point. If the Zero Master Cycle sensor is used to sense each Master Cycle, this input must be assigned to the high speed input lines 1 or 2. If the Zero Master Cycle input is used to manually establish the first zero position with the master stationary, it can be assigned to any input line. This input must be set each time a new follower cycle is initiated.

46 Zero Follower Cycle

This input establishes follower zero degree point. If the Zero Follower Cycle sensor is used to sense and redefine each Follower Cycle, this input must be assigned to high speed input lines 1 or 2. If the Zero Follower Cycle input is used to manually establish the first zero position with the follower stationary, it can be set to any input line. The Follower Cycle Zero must be set each time a new Follower Cycle is initiated. This can be done by toggling this input line or executing a Home.

49 Cycle Drop Out

Commands the follower to decelerate at the follower deceleration rate and stop in the cycle drop out stop position.

64 Master Length Increment

This input will increment the master length a set value each time an input signal is received.

65 Master Length Decrement

This input will decrement the master length a set value each time an input signal is received.

66 Follower Length Increment

This input will increment the follower length a set value each time an input signal is received.

67 Follower Length Decrement

This input will decrement the follower length a set value each time an input signal is received.

68 Working Offset Increment

This input is used to increment the working offset value which is set in the follower cycles screen.

69 Working Offset Decrement

This input is used to decrement the working offset value which is set in the follower cycles screen.

70 Working Segment Increase

This input is used to increment the working segment value set in the cycle profiles screen.

71 Working Segment Decrease

This input is used to decrement the working segment value set in the cycle profiles screen.

72 Master Phase Increment

This input is used to increment the master phase distance which is set in the master cycles screen.

73 Master Phase Decrement

This input is used to decrement the master phase distance which is set in the master cycles screen.

74 Reset Cycle Counter

This input is used in conjunction with the cycle count parameter (in the follower cycles screen) to reset the cycle counter after the cycle count value has been reached.

75 Reset Batch Counter

This input is used in conjunction with the batch count parameter (in the follower cycles screen) to reset the batch counter after the batch count value has been reached.

76 Cycle Counter Hold

This input is used to temporarily stop the cycle counter from counting.

77 Phase Fine Inc Select

This input is used to toggle between fine and coarse increment settings. When this input is active all adjustments made to the master phase distance and the follower cycle length via inputs or serial commands will be made using the fine adjustment value. When not active, the coarse value will be used.

Output Functions

Outputs require an external isolated supply voltage, usually the same supply as the inputs. Each output can sink or source 0.2 amps DC.

0 Ready

If the microprocessor and its memory are operating correctly, this output is on continuously after power up. It turns Off when either a watchdog timer fault or non-volatile memory fault occurs. To activate this output after you assign it to an output line, You must cycle AC power to the FX drive amplifier (and PCM module if applicable).

1 External Mode

This output is **on** when the FX drive is ready to be operated from external I/O. This output is **off** when the FX drive is in serial control, such as when PCXWin home or Index dialog box are in control.

2 In Motion

This output is **on** whenever the commanded velocity is not zero.

3 End Of Index

This output comes **on** after the completion of an index. If the index count is greater than 1 this output will come **on** after each index execution. It is turned Off automatically with the next motion or when the **clear end of index output** input is used.

4 End Of Home (0)

This output comes **on** after the completion of a home #Ø cycle. It is turned **off** automatically with the next motion or when the clear end of home input function is used.

5 End Of Index Count

This output comes **on** only after an index is executed the number of times specified by the index count. It is turned off automatically with the next motion or when the clear end of index count input is used.

6 Hardware Fault

This output comes **on** after the occurrence of a hardware drive fault. The specific hardware fault is indicated by a number between 0 -7 on the diagnostics display. It is turned **off** when all hardware faults are cleared.

7 Travel Limit Fault

This output comes **on** when either a hardware or software travel limit is hit and turned **off** when the fault is cleared. A **travel limit fault** will also cause an **"L"** to appear in the diagnostic display.

8 Position Error Fault

This output comes **on** whenever a positioning error fault occurs and is turned **off** when the fault is cleared. Position error (or following error) fault will cause an **"F"** to appear in the diagnostic display.

9 Home Completed

This output comes **on** after the completion of a home cycle and stays on continuously unless absolute position is lost.

10 Sensor Limit Distance Hit

This output comes **on** during a **feed sensor** index if the sensor limit distance is exceeded before a sensor signal is received. This output also comes on during a home when the home limit distance is exceeded. This output is turned **off** with the **clear sensor limit** input. The condition that turns on this output is not a fault and the index will continue.

11 End Of Sequence

This output comes on after the completion of a motion sequence before the last dwell time (if any). A motion sequence is completed when a home, index, and/or program or a combination thereof, is executed and no further motion commands are given. This output turns off automatically with the next motion or when the clear end of sequence is used.

12 Programmable Outputs

The output lines assigned to this function can be turned on or off with the set PGO programming function. The maximum number of PGO lines is 16; the maximum number of jump/wait Input lines is 8.

Similarly, the internal output lines support programmable outputs (PGOs). The PGOs can be overlapped between the external and internal output lines. For example, if output #12 (PGOs) is assigned to external line 9, and sixteen output lines are selected, then the PGOs would be assigned to external output lines 9,10,11, and 12, along with internal output lines 1 through 12.

13 End Of Program

This output is activated after each completion of all steps (motion and non-motion) in a program. It is automatically deactivated with the next motion or when the clear end of program input is used.

14 End Of Program Count

This output is activated only after a program is executed the number of times specified by the Program Count. It is automatically deactivated with the next motion or when the Clear End Of Program Count is used.

15 In Suspend

This output is activated when the suspend program function is active. Also see suspend/resume functions section of your PCM module operators manual.

16 End Of Home (1)

This output comes **on** after the completion of a home cycle of home function #1. It is turned **off** automatically with the next motion or when the clear end of home is used.

17 At Sync

This output is activated when the follower axis is in sync with the master axis (i.e. the acceleration portion of the index is complete). For example; In a fly cut-off system, the cut-off can not be made until the follower and master are in sync. This output would signal the cut-off mechanism that the follower axis is in sync with master axis.

18 Sync Fault/Fly Cut-off

This output indicates that the programmed cut length reached a count that should have initiated another flying cutoff but the FX drive was not in position to initiate the flying cutoff.

19 External Brake Output

This output is **on** when bridge is enabled and **off** when bridge is disabled (unless overridden with the brake override input function or serial command BK=1"). This output function is used to disengage an external brake when active.

20 Torque Limit Hit

This output is normally **off** and will turn **on** when the Torque Limit is reached as set in limits dialog box (see page **Error! Bookmark not defined.**). The only way to turn this output **off** is with the “clear torque lmt output” input function.

21 Master Cycle Defined

Output turns on once the Master Cycle is defined.

22 Follower Cycle Defined

Output turns on once the Follower Cycle is defined.

24 In Phase

Compliment of Phase Distance Error Limit output. When a home is executed, then the follower cycle is initiated, the In Phase output will not be active until the follower has completed its phase correction.

25 Master Sensor Missing

This output turns on if the Master Cycle sensor input fails to appear after 1½ times the Master Cycle length

26 Follower Sensor Missing

This output turns on if the Follower Cycle sensor input fails to appear after 1½ times the follower Cycle length.

41 Cycle Limit Reached

This output will become active when any of the operational and/or user imposed limits listed below are reached.

- Working segment ratio change could not be done in this cycle.
- Maximum velocity limitation caused extended master length.
- Operator tried to select working segment past limit.
- Operator tried to select working ratio past limit
- Operator tried to select master length past limit
- Operator tried to select follower length past limit

42 Cycle Count Reached

This output will become active when the cycle count number (set in the follower cycles screen) is reached.

43 Batch Count Reached

This output will become active when the batch count number (set in the follower cycles screen) is reached.

44 Phase Distance Error Limit

This output will activate any time the Phase Distance Error Limit is reached.

Executing Follower Cycles

The last step is to create a motion program (or programs) that uses the "Y" command which executes the follower cycle.

Typically the program is started using the Program Initiate input function (Function #29).

The follower axis must be zeroed or homed before a follower cycle is executed (the home is necessary to establish the center of the working segment). Typically, this home would be a feed to sensor type, where the home sensor is also the follower zero sensor. Therefore, your program would have a home command followed by an execute follower cycle command.

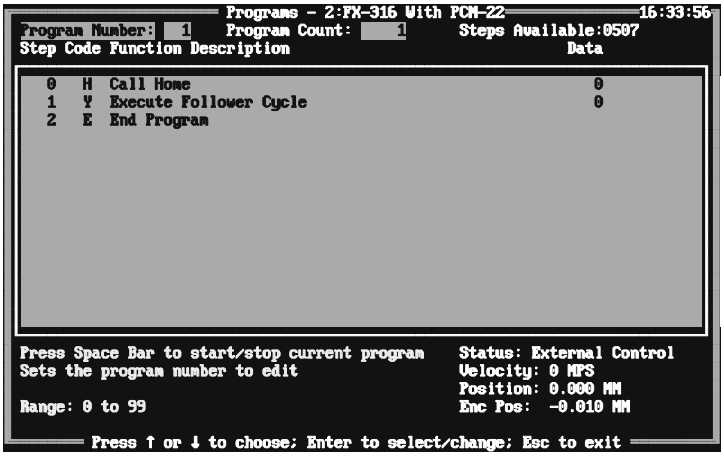


Figure 12 PCM-22 Program Using The "Y" Function

Creating A Motion Program

In addition to using motion programs to execute follower cycles using the “Y” programming function, the PCM-22 can also take advantage of 14 additional programming functions. However, once a follower cycle (Y) has been initiated, the PCM-22 can not execute additional steps within that program.

Indexes are described in the FX Drive Operator's Manual in the PCX section (P/N 400282-00).

Motion Programs are a series of indexes that have been previously set up that you combine with other programming steps to create a motion profile. Each motion program provides a series of movements in conjunction with other machine functions. The movements are used to perform a particular machine operation.

Multiple programs can be created using PCX software and stored in the PCM-22, each designed for a different machine function. The PCM-22 is capable of storing up to 256 indexes, 100 motion programs (∅ to 99), and a maximum of 1024 program steps in the non-volatile memory.

The number of available programs and average number of steps per program are directly related to each other. The memory is set up such that if you require 100 programs (maximum), each program can have an average of 10 program steps each. If the number of programs is reduced to a minimum, you could have as many as 255 steps in a single program.

A motion program is created by entering program functions in the order in which they are to be executed. A motion program is made up of function codes, some of which are listed across the bottom of the PCX program screen. To see the complete list of available codes when you are in the program screen lower half, press the <F1> key. A pop-up screen will display all of the function codes.

As you enter steps in a program, the function codes and function data (index numbers, program numbers, dwell times, etc.) are displayed on the program screen so you can easily follow the program sequence.

You may use any index or program which you have previously created to build your program. The example motion program shown in Figure 13 could be accomplished with one program; however, two programs have been used to show the use of the Call Program (P) function. In this example, program numbers 1 and 2 are used and index numbers 1, 2, 3, 4, and 5 are used.

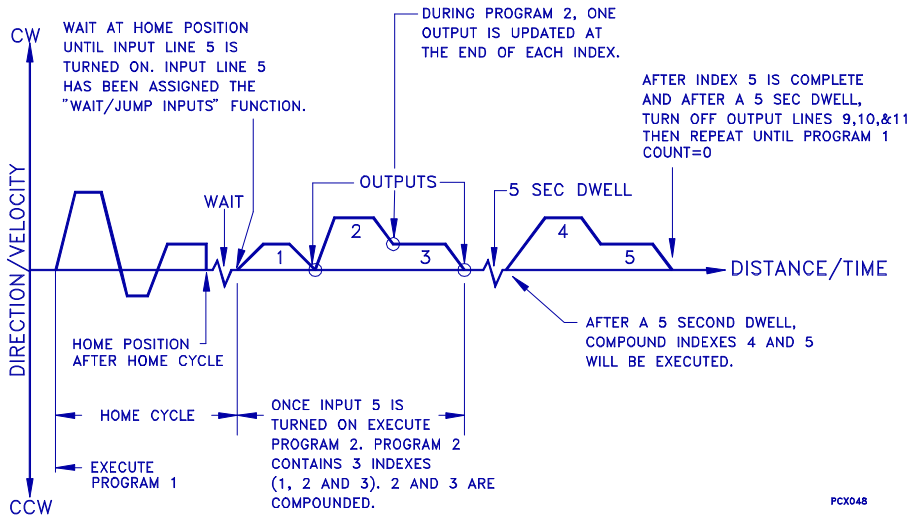


Figure 13 Motion Profile Of Example Program

show the program setup screens used to generate the motion program in . The program count determines how many times the program will be executed. In this example the program count for Program 1 is 10. This means everything within Program 1 will repeat 10 times including Program 2. If the program count is set equal to zero, the program will not execute. If the program count is set equal to 65535 or larger, the program will execute indefinitely.

The program function codes determine the actual moves to be executed. Each function will be performed in the sequence that is shown in the program screen. Once you enter the sequence, you can download the program to the FX drive by pressing the <Esc> key, or by moving the cursor up to the Program Number position using the arrow keys. The upper left corner of the screen will display a "BUSY" message during the download.

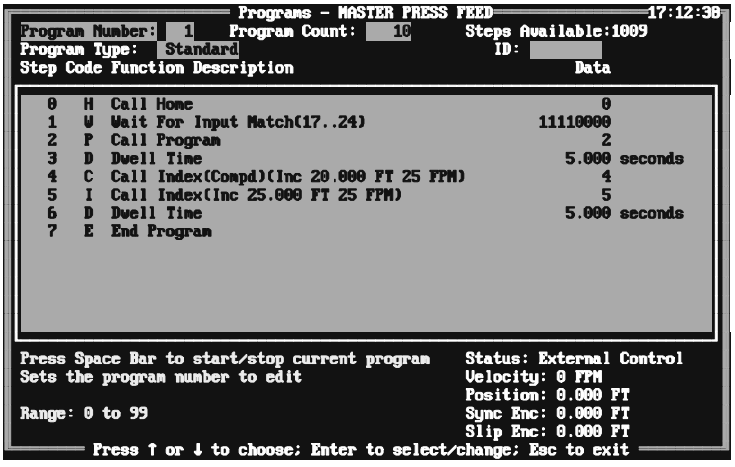


Figure 14 Program Example

In the example shown above, Program 1 is the main program and Program 2 is called as a subroutine of Program 1. This call can be seen in step 2 of Program 1.

Time Base (Indexes And Homes)

The "Time Base" feature is available from the Index and Home screens and can be set to Real Time, Synchronized or Analog.

The Time Base feature relates to the velocity scale factor pre-determined by drive parameters. If you select Time Base Synchronized, then you program index, jog and home velocities based on the sync scale factor. The PCX program screens will show the scale factors you choose.

To run a synchronized index you must call the index from a program. The program must toggle the time base from Real Time to Synchronized prior to running the index.

Programming Functions

Each program function has a designaed single or double letter function code that is used when creating a program. This section describes the functions.

C Compound Next Index

The Compound Index feature allows you to link two indexes together without stopping motion between the indexes.

A Compound Index is an index whose final velocity is not zero, but the velocity of the next index. Because a Compound Index ends by accelerating or decelerating to a velocity, not a dead stop, that compounded index cannot be used again as a regular index. However, identical Compound Index sequences can be repeated in a program. Three Compound Index examples are shown below:

Example 1: You can use the Compound Index feature to run special Indexes which have different velocities and distances.

In this example, the drive will accelerate at the Index #1 acceleration rate until it reaches the velocity of Index #1. Then, after this distance in Index #1, the drive decelerates at the deceleration rate of Index #1 to the velocity of Index #2, without coming to a stop.

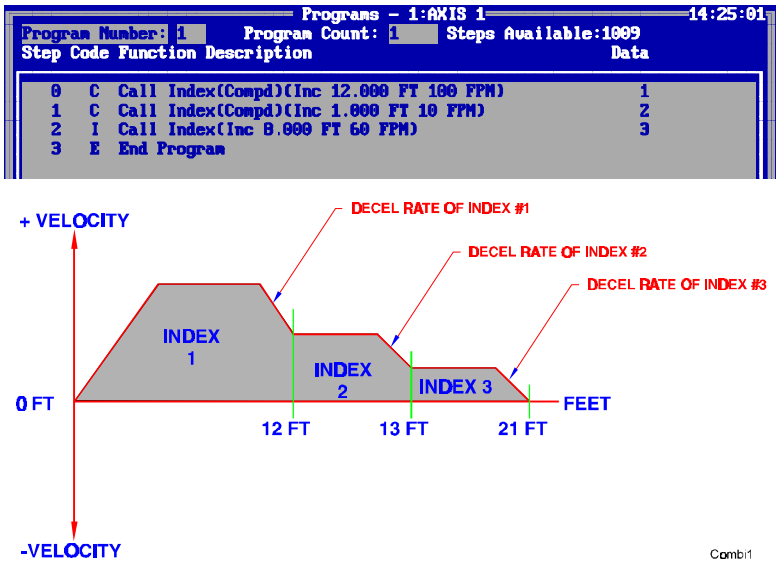
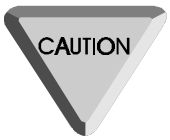


Figure 15 Compound Index Example

The drive will continue at that velocity until it approaches the programmed distance of Index 2, then decelerate at the Index #2 deceleration rate to the Index #3 programmed velocity. It will continue at that velocity until it approaches the programmed distance of Index #3, at which time it will decelerate at the Index #3 deceleration rate and stop.



When using multiple compound indexes within the same program, all compound index directions must be the same. The direction of movement is set by the first of the compound indexes and is not scanned again until the next index is initiated that is outside the compound ones.

Example 2: This example is much the same as #1, except in this compound index the drive will accelerate instead of decelerate at the end of each individual index.

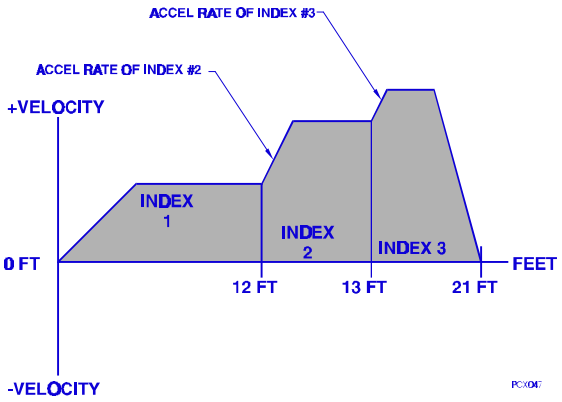


Figure 16 Example #2, Compound Index Motion Profile

After completing Index #1, the drive will accelerate at the acceleration rate of Index #2 until it reaches the programmed velocity of Index #2. After completing Index #2 the drive will accelerate to the programmed velocity of Index #3 and move at that velocity until it approaches the programmed distance, at which time it will decelerate at the Index #3 deceleration rate and stop.

Example 3: Use the compound index feature to turn an output on, then off without stopping motion. Each index is incremental, has a count of 1, and has the same velocity.

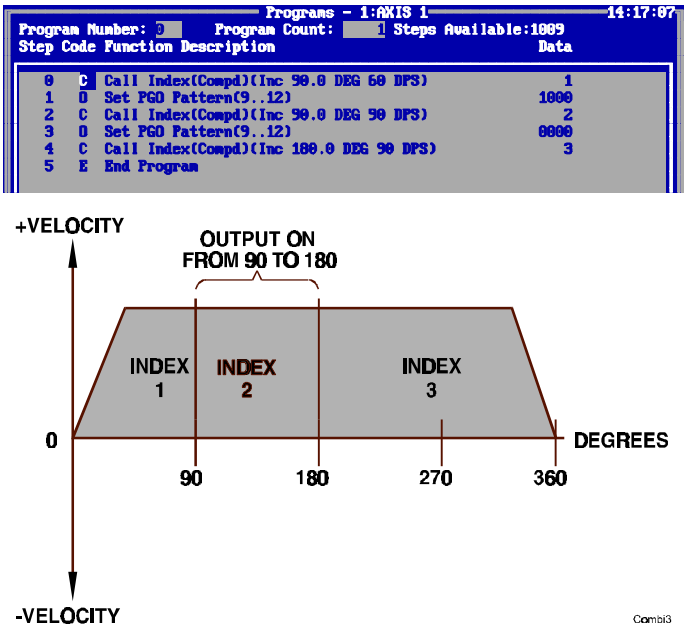


Figure 17 Compound Index Example #3

D Dwell Time

The Dwell Time function allows the program to pause for a number of seconds between program functions. Time may be directly entered or recalled from one of the 64 user registers. Minimum time is 1 ms; maximum time is 65.535 seconds.

E End Program

This function is used to designate the end of a program. It may be used more than once when using the Jump commands. The END function is also used to terminate programs prior to the last program step.

F Set Maximum Following Error

This function overrides the "Maximum Following Error" which is set up in the "Limits" screen. When the program has completed, Following Error is reset to the value programmed in the Limits screen. This function is often used with the "Q, Set Maximum Torque" function. A large value entered here prevents the drive from faulting out during a "Torque" move.

H Call Home

The Call Home function is used to initiate a previously programmed Home Cycle. There are two Home Cycles that can be called within a program.

I Call Index

The Call Index function is used to initiate the execution of an index. When this function is used, the index number must also be given. After you enter an index number in the data field, PCX will display that index's distance and speed.

O Set Outputs

Entering "O" in a program step will display a sub-menu in the middle of the program screen.

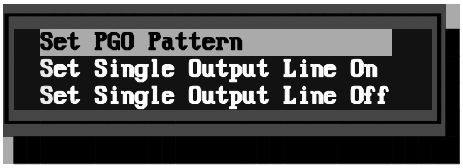


Figure 18 Programmable Output Options

Set PGO pattern.

Before this function can be used, you must first assign Output Function #12 "Programmable Outputs" to one or more internal and/or external output lines. This function allows you to change the status of any or all of the programmable outputs you assigned in the output function screen.

Once selected, the cursor will move to the data field where zeros that represent the programmable outputs are displayed. The leftmost zero represents the first programmable output line and the others follow in numerical sequence. If you enter a one, that output will be forced on.

If you enter a zero, that output will be forced off. These outputs will remain in this pattern until they are updated. If you enter an "X" for any line, that line state will not be changed.

Set single output line On

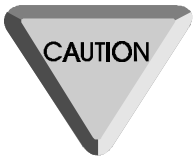
Turns any output, hardware or internal, on. Output remains in this state.

Set single output line Off

Turns any output, hardware or internal, off. Output remains in this state.

P Call Program

The Call Program function is used to initiate a program from within a program. You can use the "P" function to call additional programs as sub-routines of your original program. This is called "Nesting". PCX allows you to nest up to 30 programs under any one Program Call function before returning to the original program.



Use care in constructing programs that create a continuous loop condition by "Nesting" programs. When nesting programs, your program must return to the "Master" program somewhere in the loop to prevent programs from becoming stacked up in memory.

For example: You initiate Program #1, which uses the "P" function to call Program #2, which uses the "P" to call Program #3, which uses the "P" function to again call Program #2. This continuous loop condition could cause the memory to get stacked up, causing the drive to drop out of the program and return to external mode or register a "∅" fault on the diagnostic display.

To avoid this condition, make all of your program calls from your original program (Program #1 in this example). Then use the Jump function (J) right after the "P" function calling Program #3 to jump back to "P" function calling Program #2. This will allow the memory (RAM) to clear and the loop to continuously run program #2 and #3 from Program #1.

Q Set Maximum Torque Output

This function overrides the "Maximum Torque Output" which is set up in the "Limits" screen. When the program has completed, the original value is restored. If you use a low value for "Q" (less then 100%) you may need to set a high value for following error (F). This will avoid faulting the drive if it runs into a condition requiring more torque than it is allowed to deliver.

R Wait For Counter

An (R) code allows you to wait for the number of counts you enter before moving on to the next program step. Enter the desired number of counts in the data field. Counting is determined by the CW(+) and CCW(-) rotation of the sync encoder as viewed from the shaft end of the sync encoder. The count originates from the last Start External Counter (S) or Wait For Counter (R) code.

S Start Counter

An (S) code resets and starts the internal counter for counting the external encoder pulses. The SCS-2 with 2 (1000 line) channels produces 4000 counts/steps per revolution.

T Time Base

This code determines whether the Time Base for any programming function placed after the Time Base code in a program (including Dwells and Homes) is Synchronized (sync encoder), Real Time (normal) or Analog. It is important to remember that if you want a limited number of program steps done in Synchronized mode, you must restore the Time Base to Real Time after those program steps.

J, W Jump/Wait

The Jump and Wait For External Input Functions include unconditional jumps to another program step, jumps if an input line pattern is matched, or a jump/wait on a single input line being made active, whether that line has been assigned as an External Input Line or not.

The statements "Jump If Input Match" and "Wait For Input Match" are satisfied if the input line pattern of active lines (hardware and/or internal) matches the specified pattern exactly. In addition, a "don't care" character, "X", is available to mask off from the pattern any input lines whose condition should be ignored at this point.

Using Jump Statements in a Program

When the "J" command is entered as a program step, PCX displays the screen shown in Figure 32. The first choice, Jump Always, means exactly that, an unconditional jump to another statement number or label.

The second choice, Jump on Input Pattern (PGI), brings up another window with these match selections:

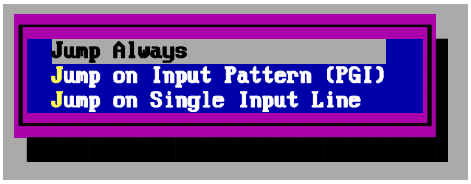


Figure 19 Jump Command Options

The first and third choices in the figure above are alike in that they are asking for an AND condition of the selected Input Lines before jumping to the destination program step. In other words, the Jump will be made if, for example, selected Lines 1 AND 3 AND 4 are ALL on or off, depending on your selection. A numeral 1 looks for that Input Line to be on or off; the letter X causes that Input Line condition to be ignored.

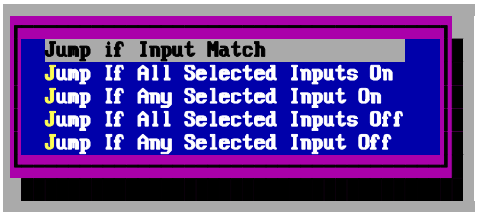


Figure 20 Jump "If" Options

The second and fourth choices are alike in that they are asking for an OR condition of the selected Input Lines before jumping to the destination program step. That is, the Jump will be made if Line 1 OR Line 3 OR Line 4 is on or off, depending on your selection. Again, a numeral 1 looks for that Input Line to be on or off; the letter X causes that Input Line condition to be ignored.

The fifth choice, Jump if Input Match, will then ask you for an Input Line pattern to match before Jumping, with numeral 1 representing a Line ON, numeral ∅ representing a Line OFF, and X representing a Line whose condition is ignored by this statement. This choice will Jump program execution to the specified program step if and only if the exact PATTERN is matched line for line. If any of these conditions are not met, program execution proceeds to the next step in the sequence.

Another feature of both jumps and waits is the ability to monitor a single input line (hardware or internal) that is independent (or part of) of the existing jump/wait pattern (i.e. Programmable Input (PGI) Pattern). The following four jump/wait options are available.

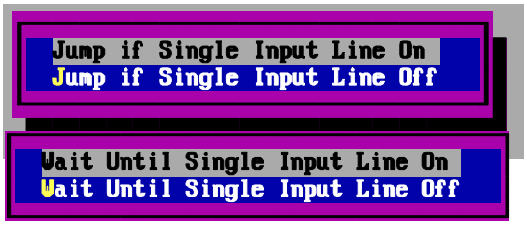


Figure 21 Jump/Wait Single Input

Wait for External Input Functions are very similar, except that program execution is halted at that spot in the program until the desired Input Line condition is met. If the condition is not met, program execution proceeds immediately to the next step.

Jump Commands

To select this option, enter "J" in the program step. Then select one of three Jump options shown in Figure 19.

The "Jump Always" command is an unconditional jump statement that causes program execution to move immediately to the specified program step. The "Jump on Input Pattern (PGI)" command causes program execution to skip to the designated program step if the specified input lines match the pattern you have selected. If the specified lines do not match that pattern the program continues to the next step in the program.

Before this command can be used in a program, Input Function #28 Wait/Jump Input" must be assigned to one or more hardware or internal input lines (see "Input Functions" section).

Once the Jump Command has been selected, the cursor will move to the center of the PCX screen. The zeroes that appear in this field correspond with the number of Input Lines selected with Input Function #28. The left-most zero corresponds to the first input line selected.

You may now choose which lines to monitor during this step by entering a (1) in the field corresponding to the input line you wish to monitor. Any field that has a zero or an "X" will cause that input line to be ignored during execution of this step.

The "Jump on Single Input Line" command causes program execution to jump to the designated program step if the specified input line matches the condition you set in the next screen.

This line can be any input line and is not restricted to those lines assigned to Input function #28, Wait/Jump Inputs. If the specified line does not match the condition (On or Off), the program continues to the next step.

Wait For Input Commands

Before this command can be used in a program, you must assign Input Function #28 "Wait/Jump Input" to one or more hardware or internal input lines (see Input Functions, of this Manual). To use Wait For External Input when editing or writing a program, enter W as a program step. When you have entered the W, the screen will offer the choices shown below.

Figure 22 Wait For Input Command Options

When the "Wait On Single Input Line" function is selected, the system will hold program execution at this step until the designated input line meets the On or Off condition you select in the next screen. This line can be any hardware or internal input line and is not restricted to those lines assigned to Input Function #28, Wait/Jump Inputs.

When the "Wait On Input Pattern (PGI)" function is selected, the system will hold program execution at this step until the input pattern you designate in the pop up menu is satisfied. These lines are restricted to those that you have assigned as "Wait/Jump" inputs.

Once the Wait Command has been selected and a choice made from the pop-up screen, the cursor will then move to the center of the PCX screen. Zeroes will appear in this field that correspond to the number of External Lines selected in the Input Function Screen. The left-most zero will correspond to the first line selected.

You now choose which lines to monitor during this step. You make this choice by entering a "1" in the field corresponding to the line whose state you wish to monitor. Any field that has a zero or an "X" will cause that line to be ignored during execution of this step.

Suspend/Resume Functions

The Suspend Function is used to temporally suspend the operation of a running program and is initiated with Input Function #31. The Resume Function causes the system to finish the program that was suspended. Resume is initiated with Input Function #32.

When a Suspend command is received, the FX Drive will stop motion using the Stop/Hold Decel ramp entered in the Limits screen. The motor position will then be stored in a Return To Position Index, and the interrupted program task will be stored in the memory of the drive. The drive will now accept and execute new motion commands such as Jog, Home, Index, or Program Initiate. This can continue until a Resume command is received.

Upon receiving a Resume command, the drive will continue the execution of the program which was suspended. If an index was in progress when the suspend occurred, any index distance remaining at the end of the Suspend ramp will be executed by the Resume function, regardless of the motor position at the time of Resume.

If you move the motor in any way during a Suspend function, and you want the motor position to be the same as if the Suspend had not occurred, you must run a "Return To Position Index" before you initiate a Resume function. You must have previously set up this index as an absolute index with the speed and ramps you wish to use for the Return to Suspended Position move, along with a repeat count of 1. The index you set up is the one you select when you enter "Return/Resume Index Number" in the Suspend/Resume screen.

Suspend, Resume, Return to Position, and Clear Suspend functions are activated through input functions in the I/O.

A Suspend will not be accepted while a Hold cycle is active, while a previous Suspend is active, or a decel ramp is in progress. The memory for the Suspend function will only hold one event. If you attempt a Suspend command after the drive has already been suspended and the Suspend memory not cleared, the new information will not be retained.

You can clear suspended memory by using the Abort Suspend Function (Input Function #33), or a hardware Stop command (Input Function #11).

Suspend Screen Parameters

When you select the Suspend option from the "Define Motion" screen, the Suspend data entry screen shown in Figure 40 is displayed.

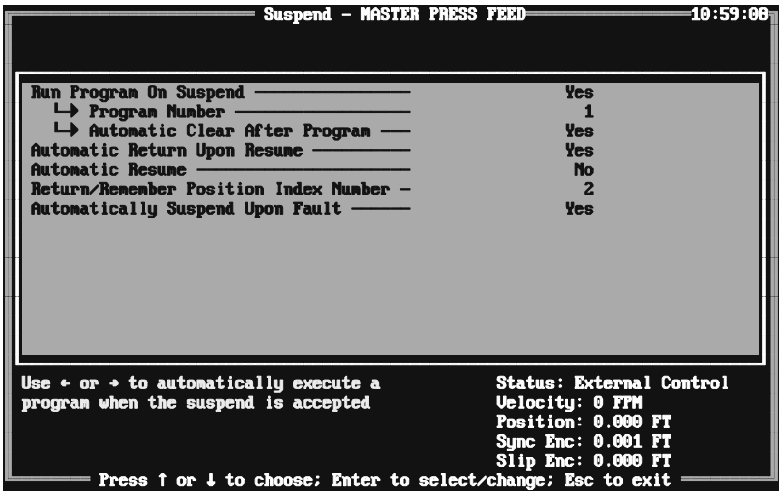


Figure 23 Suspend Setup Screen

Run Program On Suspend

This feature allows another program to be automatically initiated when a Suspend is accepted. After the FX Drive stops on Suspend, the program you specify in the program number parameter will be executed without the need for a Program Initiate.

Program Number

This parameter determines which program is to be initiated when a suspend is accepted and the "Run Program On Suspend" feature is used.

Automatically Clear After Program

This feature may be used to automatically clear a Suspend cycle. When used with the Run Program On Suspend feature, the program you selected will be executed before the Automatic Clear occurs. Once a Suspend cycle is cleared, the original program will not be resumed and the controller will be free to accept another Suspend input or motion command. When this feature is not used, a Resume input is needed to clear the Suspend cycle.

Automatically Resume

This feature will cause the drive to resume a suspended program without the need for a Resume input following the Run Program On Suspend program.

Automatically Return After Resume

This feature causes the "Return Index Number" index to be automatically initiated when a Resume input is accepted. When a Suspend input is accepted and the motor stops, the current position is stored in the index position specified by the "Return Index Number" value. To perform properly this index must be set up as an absolute index with a count of one. Then the motor will return to the position where the Suspend occurred, and continue with the original program.

Return/Remember Position Index Number

This parameter determines which index is to be initiated when a Resume is accepted and the "Automatically Return After Program" feature is used. When a Suspend input is accepted, the current position will be saved to this index. All other index information should be set up by the operator. This index must be set up as an absolute index.

Automatically Suspend Upon Fault

This function allows a fault to be handled without losing the current program setup or motor position. When this function is enabled, the following conditions apply:

1. A Suspend is automatically executed following a reset of a qualified fault. After the drive has been reset, operation is identical to a normal Suspend.
2. The Suspend-Upon-Fault capability applies only to the following faults:

   1 fault - RMS current or Continuous Torque Exceeded fault

   5 fault - Motor temperature fault

   6 fault - Bridge circuit breaker is open or low AC voltage

   7 fault - Amplifier bridge temperature faultF fault - Following error fault only.
3. The Suspend Upon Fault may only occur in a situation where Suspends are allowed, i.e. Programs. Unlike other Suspend conditions however, a Suspend Upon Fault may occur during the ramp down portion of an index which occurs in a program.
4. When a fault occurs during a Suspend function, the program which was running during the Suspend is terminated. However, the Suspend remains active following a fault reset.

All suspend functions and rules apply to fault initiated suspends.The "Out Of Index" output (output #45) is used to indicate that the motor position is within the faulted index distance when a Suspend occurs as a result of a fault. This output will become active if the motor moves past the end of an index during a Suspend Upon Fault, or is moved backwards before the faulted position where the Suspend was activated.

Encoder Operation

The SCS-2 sync encoder is a two channel amplified sine wave encoder. Each of the channels has 1000 lines. The PCM-14 inputs are A, A {complement}, B and B {compliment}. Quadrature encoding of these two signals produces a signal whose frequency is 4 times the line count. One revolution of the sync encoder produces the equivalent of 4000 counts/steps per revolution.

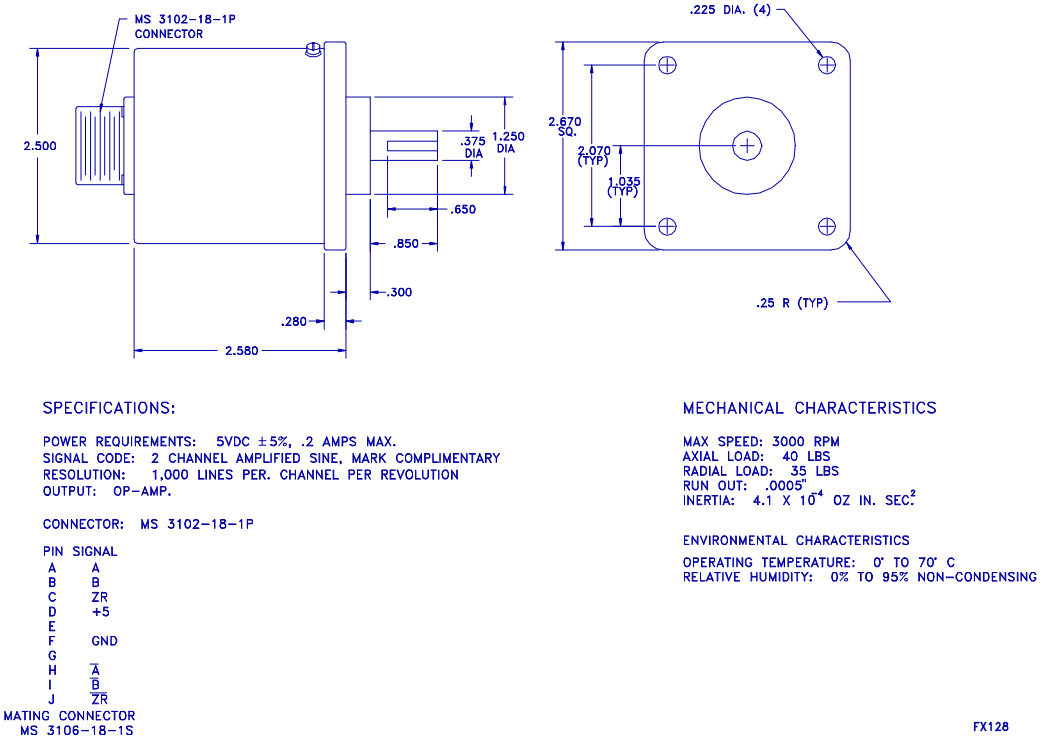


Figure 24 SCS-2 Encoder Electrical/Mechanical Information.

Multi Axis Synchronization

If an external encoder is used as the method of accomplishing synchronization, The connection of the external encoder to the first axis is accomplished with this sync cable. As many as 10 axis can be synchronized to a single encoder.

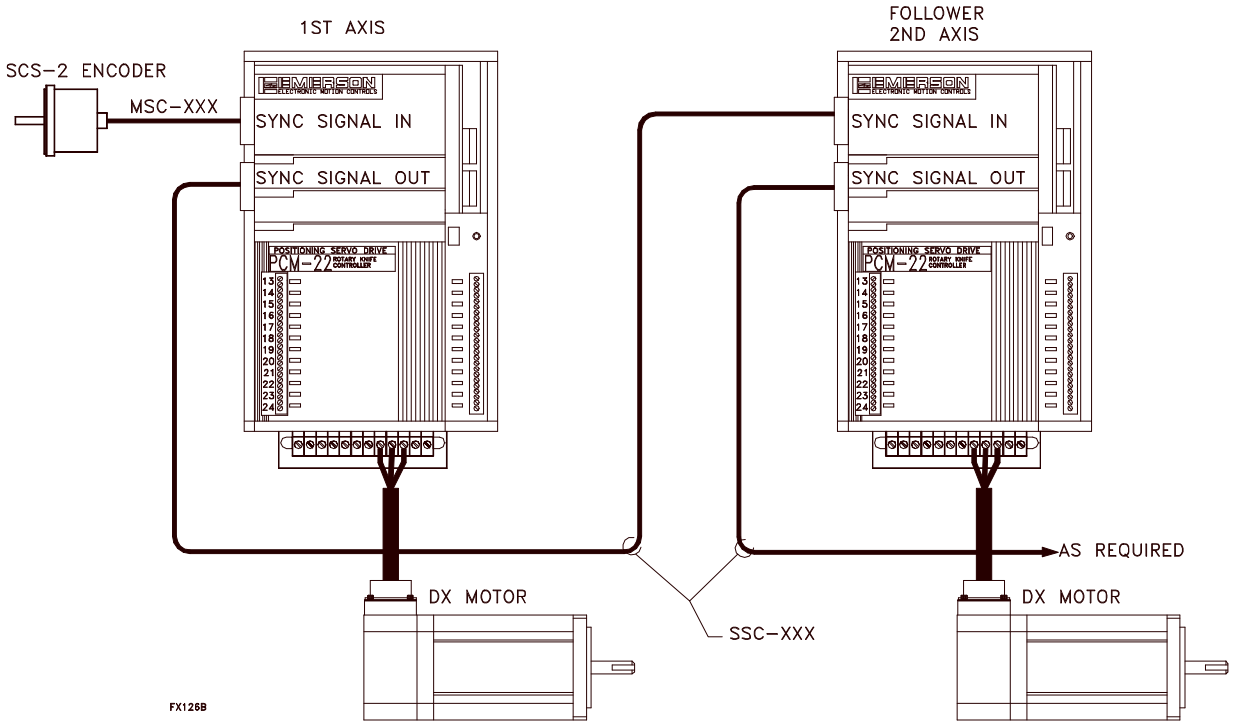


Figure 25 Second Axis Application Cable Connections.

Sync Cables

The MSC-XXX cable is available in three different lengths.

- MSC-015 = 15 FT.
- MSC-025 = 25 FT.
- MSC-050 = 50 FT.

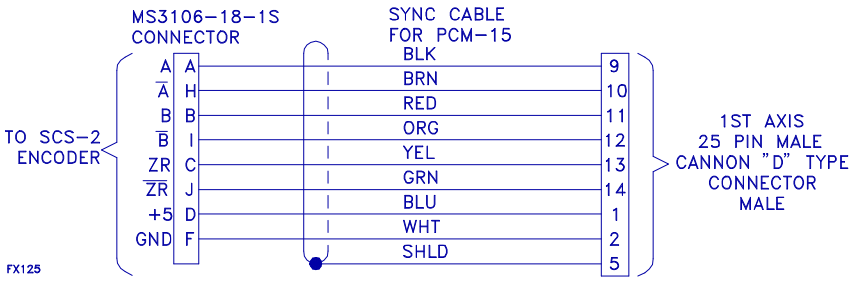


Figure 26 MSC Cable Wiring Diagram

For some applications it may be an advantage to **not** have an external encoder as the source of synchronization. Instead, another FX drive could be the source. This is the purpose of the second connector on the PCM-15. The first axis can provide the source of synchronization for the second axis.

The SSC-XXX cable is a symmetrical cable and its ends can be interchanged and is available in three different lengths.

- SSC-003 = 3 FT.
- SSC-006 = 6 FT.
- SSC-010 = 10 FT.

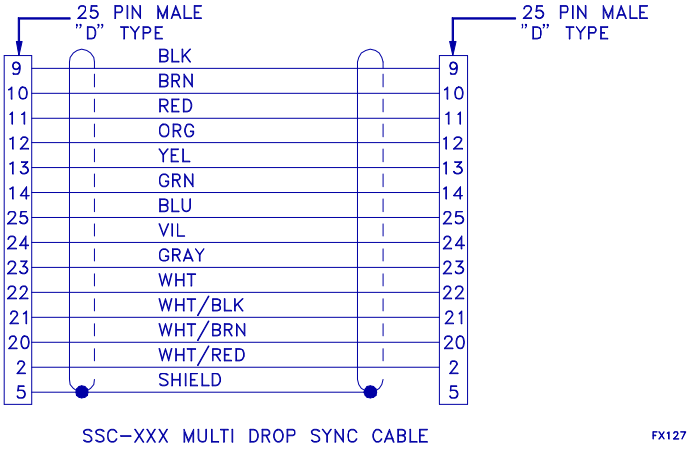


Figure 27 SSC Cable Wiring Diagram