

# PC68

## Intelligent Motion Controller for PC/104 or RS-232

### PC68 FEATURES

- **Controller capabilities**
  - Four axes of Stepper control
  - Expandable to 8 axes with the expansion module
  - Control architecture options - Stackable
- **Communications**
  - PC/104 bus
  - Stand-alone with high speed RS-232 port
  - Interrupt or Polling communication
  - 4 I/O registers for control & status
- **Sophisticated Control Functionality**
  - 16 bit DAC analog resolution
  - Independent and coordinated motion of all axes at the same time
  - Slip & Stall detection with encoder feedback
  - Crystal controlled step pulse from 0 to 1,044,000 steps per second
  - Configurable PID filter with feedforward coefficients
  - Circular Interpolation
  - Constant velocity linear interpolation (all axes)
  - Electronic Gearing
- **32 bit processor for extensive co-processing**
  - Does not burden the host with overhead
  - Custom, Parabolic, "S"-curve & Linear trajectory profiles
  - Patented technology to minimize torque ripple and velocity modulation
  - Internal Watchdog timer for safety
- **Control signals**
  - Single high density shielded SCSI connector
  - Up to 40 dedicated I/O
  - Expandable up to 24 "user definable" I/O
  - Expandable up to an additional 144 opto-isolated I/O with the Px6IO
  - Output is +/-10V or Step & Direction
  - Independent home and plus / minus over-travel inputs
- **OMS-EZ™ Software programming**
  - High level programming expertise not required
  - Over 150 commands, "universal" to all OMS controllers
  - Commands are ASCII characters
  - Automatic conversion to "user" defined units i.e. inches / revolutions
  - Software for WIN95/98, NT.
  - Software supplied at no additional cost
  - Compatible with most operating systems
- **Flash Memory**
  - Field upgradable firmware within 32 bit operating systems
  - Non-volatile program storage
- **Flexible and Expandable**
  - Conforms 100% to all PC/104 specifications
  - Expansion modules for application flexibility
  - Customizable solutions available for your requirements
- **Factory Direct Technical Support**
  - Person to person toll-free tech support: call 800-707-8111
  - Application notes and Documentation on the Web
  - Example programs and application code provided
  - All OMS controls are 100% burned-in, tested and quality inspected



## DESCRIPTION

The PC68 is both a PC/104 controller that supports the PC/104 ISA bus standard as well as a stand-alone controller via a high speed RS-232 communication port. The OMS PC/104 product family begins with the PC68, a four axis controller and the number of axes can be expanded to 8 with the expansion module. The PC68 supports up to 24 general purpose TTL I/O bits that can be expanded to include up to 144 optically isolated I/Os with the Px6IO expansion module. The architecture of the PC68 includes an additional board to board bus that is separate from the PC/104 J1 and J2 bus connector where the microcontroller of the PC68 is the master. This provides the ability to add additional features, i.e. more axes, I/O and even custom requirements, that may be required without having to pay for more than you need.

The step pulse is a TTL level 50% duty cycle square wave that supports velocities of 0 through 1,044,000 pulses per second. The encoder feedback functionality supports quadrature encoders up to 4 MHz at a 4 times resolution and is used as feedback for the stepper axes or as independent position feedback. The encoder feedback can provide slip or stall detection. Every axis includes dedicated +/- overtravel inputs, a home input, and an auxiliary output. The PC68 is available in several different models at several different prices so that you don't have to pay for a lot of functionality that you don't need.

## PROGRAMMING

PC68 motion controllers are easily programmed with double character ASCII commands through an extensive command structure. The commands are combined into character strings to create sophisticated motion profiles and are passed to the PC68 data I/O register. A separate 'FIFO' command queue for each axis is used to store the parsed commands by the PC68 until they are executed allowing the host to send a complex command sequence and attend to other tasks while the PC68 manages the motion process. These command queues store 200 command and parameter words and include a command loop counter which allows multiple executions of any command string.

The following commands are available in the PC68 family of motion controllers. All commands are sent to the controller as ASCII character strings. Some commands expect one or more numerical operands to follow. These commands are identified with a '#' after the command. The '#' indicates a signed integer input parameter or a signed fixed point number of the format ##.# when user units are enabled. With user units defined, distances, velocity and acceleration parameters may be input in inches, revolutions, etc.

Synchronized moves may be made by entering the AA or AM command. This command performs a context switch which allows entering commands of the format MRx#,y#,z#,t#,u#,v#,r#,s#;. Numbers are entered for each axis which is to be commanded to move. An axis may be skipped by entering the comma with no parameter. The command may

be prematurely terminated with a ";", i.e. a move requiring only the X and Y axes would use the command MRx#,y#; followed by the GO command. Each axis programmed to move will start together upon executing the GO command. The PC68 can be switched back to the unsynchronized mode by entering the desired axis command such as AX.

## COMMAND SUMMARY

This section contains a sampling of some of the commands available on the PC68.

### AXIS SPECIFIC COMMANDS:

#### **AA     **AXIS ALL****

Commands following the AA are directed to all axis queues for synchronized mode.

#### **AM     **AXIS MULTITASKING****

Commands following the AM will go into the specified axis queues for independent synchronized mode.

#### **AX     **AXIS X****

Commands following AX are directed to the X axis queue.

#### **AY     **AXIS Y****

The AY command directs all the following commands to the Y axis.

#### **AZ     **AXIS Z****

The AZ command directs all the following commands to the Z axis.

#### **AT     **AXIS T****

The AT command directs all the following commands to the T axis.

#### **AU     **AXIS U****

The AU command directs all the following commands to the U axis.

#### **AV     **AXIS V****

The AV command directs all the following commands to the V axis.

#### **AR     **AXIS R****

The AR command directs all the following commands to the R axis.

#### **AS     **AXIS S****

The AS command directs all the following commands to the S axis.

### SYSTEM CONTROL COMMANDS

These commands allow control of various system parameters and operating modes to allow the user to optimize the response of the system for his/her application needs.

#### **EN     **ECHO ON****

The EN command enables echoing from the PC68.

**EF ECHO OFF**

The EF command disables echoing from the PC68.

**HH HOME HIGH**

The HH command sets the sense of the home switch on the current axis to active high.

**HL HOME LOW**

The HL command sets the sense of the home switch on the current axis to active low.

**LH LIMIT HIGH**

The LH command sets the logic sense of the overtravel limit inputs to active high.

**LL LIMIT LOW**

The LL command sets the logic sense of the overtravel limit inputs to active low.

**LF LIMITS OFF**

The LF command turns off the limit switches for the addressed axis. This allows the stage to move beyond the limit switch and should be used with caution.

**LN LIMITS ON**

The LN command restores the operation of the limit switches for the addressed axis.

**SL SOFT LIMIT**

The SL command changes the operation of the limit inputs causing the output pulse train to ramp down instead of terminating immediately. The output queue is not flushed except for the current move.

**SF SOFT LIMIT OFF**

The SF command restores the normal operation of the limit switches.

**CN COSINE ON**

The CN command enables cosine velocity ramps i.e. half sinusoid acceleration profiles for all axes.

**PN# PARABOLIC ON**

The PN command enables parabolic ramps. The parameter selects the point of truncation.

**PF PARABOLIC OFF**

The PF command restores linear acceleration and deceleration ramps.

**BI BIPOLAR**

The BI command sets the analog torque outputs to bipolar.

**UN UNIPOLAR**

The UN command sets the analog torque outputs to unipolar.

**SB# SET BAUD**

The SB command sets the baud rate to the parameter supplied.

**MOVE SPECIFICATION COMMANDS**

These commands allow specification of move parameters. They allow move parameters to be tailored to the user's system requirements.

**AC# ACCELERATION**

The AC command sets the acceleration/deceleration value. This value is used to establish the rate of acceleration and deceleration when a move command is invoked.

**VL# VELOCITY**

The VL command sets the maximum velocity value of the axis being programmed. The value is used to establish the maximum velocity when one of the move execution commands is invoked.

**VB# VELOCITY BASE**

The VB command allows the velocity ramp to start at the specified velocity. This allows faster acceleration and the ability to pass through resonance quickly in some applications.

**LP# LOAD POSITION**

The LP command will immediately load the position supplied as a parameter into the absolute position register of the axis.

**MA# MOVE ABSOLUTE**

The MA command will set up the axis to move to the absolute position supplied as a parameter.

**MR# MOVE RELATIVE**

The MR command will set up the axis to move relative from the current position at the time the move is executed.

**MT#,# MOVE TO**

The MT command uses linear interpolation to perform a straight line move to the specified absolute position. Up to eight axes may be moved together in the AA or AM modes.

**ML#,# MOVE LINEAR**

The ML command uses linear interpolation to perform a straight line relative move to the new location. Up to eight axes may be moved together in the AA or AM modes.

**RM# REMAINDER (MODULO)**

The RM command will divide the position counter by the parameter supplied and replace both the position counter and the encoder position register with the resulting remainder. This command is useful in continuously rotating axis applications.

**MOVE EXECUTION COMMANDS**

These commands allow execution of the moves which have been previously specified.

**GO Go**

The GO command will initiate the move which has been previously programmed with such commands as MA, MR, MT, and ML.

**GD GO AND RESET DONE FLAG**

The GD command resets the done flags on the active axes then proceeds with the move identical to the GO command.

**JG# JOG**

The JG command is a velocity command and will jog the axis at the velocity supplied as a parameter. The velocity may be changed without stopping by entering another JG command.

**JF# JOG FRACTIONAL VELOCITIES**

The JF command will jog the current axis at fractional rates for applications requiring very slow velocities.

**FL FLUSH QUEUE**

The FL command will immediately flush the command queue. If the stage is moving it continues to move at the current velocity.

**MOVE TERMINATION COMMANDS**

The following commands allow termination of move sequences in process.

**ST STOP**

The ST command flushes the queue for the currently addressed axis only and causes the axis to decelerate to a stop at the rate previously specified in an AC command.

**SA STOP ALL**

The SA command flushes all queues and causes all axes to decelerate to a stop at the rate previously specified in an AC command.

**SD STOP AND RESET DONE**

The SD command will stop all axes and clear any done flags.

**KL KILL**

The KL command will flush the command queue and terminate pulse generation of all axes immediately.

**LOOP CONTROL COMMANDS**

These commands allow move sequences to be repeated within loops. Loops can be nested up to four levels deep on each axis.

**LS# LOOP START**

The LS command sets the loop counter for the axis being programmed. The parameter specifies the number of times the loop will be executed. Loops may be nested up to 4 levels deep.

**LE LOOP END**

The LE command terminates the most recent LS command.

*The following commands can be used to synchronize multiple PC68 boards or synchronize them to external events.*

**WS# WHILE SYNC TRUE**

The WS command will execute the commands between the WS and WD commands as a loop, while the general purpose input line is true, i.e. low. The test is at the bottom of the loop and thus will always be executed at least once.

**WD WHILE END**

The WD command serves as the loop terminator for the WS command.

**WH WHILE**

The WH command will execute all commands until the terminating WG command as a loop, until terminated by a CW command. This allows indefinite loops to be terminated by the host computer.

**WG WHILE FLAG END**

The WG command serves as the terminator for the WH command.

**CW CLEAR WHILE**

The CW command terminates the WH command sequence upon execution of the next WG instruction. This loop is always executed at least once.

**HOME AND INITIALIZATION CONTROL COMMANDS**

These commands allow the coordination of the physical stage home position with the PC68 position register.

**HM# HOME**

The HM command will find home and initialize the position counter to the position supplied as a parameter.

**HR# HOME REVERSE**

The HR command will find home in the reverse direction and initialize the position counter to the position supplied as a parameter.

**KM HOME AND KILL**

The KM command will find home and stop generating pulses immediately, i.e. no deceleration ramp will be generated. The position counter is not affected.

**KR HOME REVERSE AND KILL**

The KR command will find home in reverse and stop generating pulses immediately, i.e. no deceleration ramp will be generated. The position counter is not affected.

**MOVE SYNCHRONIZATION COMMANDS**

These commands allow the synchronization of moves with external events or multiple axis sequences.

**ID INTERRUPT DONE**

The ID command will return the done flag to the host and interrupt the host if the interrupt has been enabled.

**II INTERRUPT INDEPENDENT**

The II command allows each axis to interrupt the host when it completes its move independent of the status of the other axes.

**IN# INTERRUPT NEARLY DONE**

The IN command will interrupt the host when the move is nearly complete. The parameter specifies the number of counts left in the move when the interrupt request is generated.

**IC INTERRUPT CLEAR**

The IC command will clear the done and error flags.

**CA CLEAR AXIS DONE FLAG**

The CA command operates like the IC command, except it clears the done flag of the addressed axis only.

**WA WAIT FOR AXES**

The WA command, only valid in the AA mode, allows a command to wait until all moves on all axes are finished before it executes.

**WQ WAIT FOR QUEUE TO EMPTY**

The WQ command is a special command that stops the board from processing any new command until the queue for the current axis mode is empty.

**SW# SYNC WAIT**

The SW command can be used to synchronize to external events by commanding the PC68 to wait for the input line to go false.

#### **WT# WAIT TIME**

The WT command will wait for the specified number of milliseconds before proceeding with the next command.

### **SYSTEM STATUS REQUEST COMMANDS**

These commands allow the host to request the status of various move parameters, including the status of limit and home switches.

#### **WY WHO ARE YOU**

The WY command returns the model and firmware revision of the board or system being addressed.

#### **RP RETURN POSITION**

The RP command requests the current position.

#### **RQ RETURN QUEUE STATUS**

The RQ command returns the number of entries available in the command queue.

#### **RA RETURN AXIS INTERRUPT STATUS**

The RA command returns the state of the limit and home switches, and the done and direction flags for the currently addressed axis. The done flag is reset.

#### **RI RETURN INTERRUPT STATUS**

The RI command returns the state of the limit and home switches, and the done and direction flags for all axes. The done flags are reset.

#### **QA QUERY AXIS**

The QA command returns the status of the single addressed axis like the RA command, except the status register and flags are not affected.

#### **QI QUERY INTERRUPT STATUS**

The QI command returns the status of all axes like the RI command, except the status register and flags are not affected.

#### **RC REQUEST ACCELERATION**

The RC command will return the current programmed acceleration or deceleration of the current axis.

#### **RV REQUEST VELOCITY**

The RV command will return the current velocity at which the axis is moving.

#### **RU REPORT POSITION IN USER UNITS**

The RU command returns the current position in user units.

### **USER UNIT COMMANDS**

The following commands allow specification of move parameters in user defined units. The OMS controls will

automatically convert all move parameters to these units once they have been initialized.

#### **UU# USER UNITS**

The UU command converts all move velocities, distances, etc. to user specified units by multiplying by the parameter given in this command.

#### **UF USER UNITS OFF**

The UF command turns off user units and causes the PC68 board to use its default units.

### **USER I/O COMMANDS**

The following commands allow manipulation and testing of the user definable I/O.

#### **AN AUXILIARY ON**

The AN command sets the auxiliary output to the high level. The open collector driver is off allowing the output to be pulled high by a pull-up resistor. It may be used to change power level on driver modules so equipped or as a user specified output.

#### **AF AUXILIARY OFF**

The AF command sets the auxiliary output to the low level. The open collector driver is on causing the line to be near ground. It may be used to change power level on driver modules so equipped or as a user specified output.

#### **PA# POWER AUTOMATIC**

The PA command will perform an AN command at the beginning of each move and an AF command after the move. See AN and AF commands.

#### **SE# SETTLING TIME**

The SE command allows specification of a settling time, in milliseconds, to be used before the power is reduced, when using the PA mode.

#### **BL# BIT LOW**

The BL command sets the selected general purpose output on, i.e. logic low.

#### **BH# BIT HIGH**

The BH command sets the selected general purpose output off, i.e. logic high.

#### **BX BIT REQUEST IN HEX**

The BX command returns the state of the general purpose I/O bits in hex format.

#### **RB RETURN OUTPUT BITS**

The RB command returns the direction of the general purpose I/O lines as they are currently defined in hex format.

**CONSTANT VELOCITY CONTOURING COMMANDS**

The contouring command set allows the building of a command sequence which can later be executed at constant velocity for machine tool and other similar applications.

**AF## AUXILIARY OFF**

The AF command turns off any combination of auxiliary ports, when encountered in the command stream, allowing control of other peripherals such as a laser beam for machining.

**AN## AUXILIARY ON**

The AN commands turns on any combination of auxiliary output ports when encountered in the contouring command stream.

**CD#,#; CONTOUR DEFINE**

The CD command allows entry of a contour definition which will start at the position specified. Any combination of axes may be used in the contour mode.

**CE CONTOUR END**

The CE command ends the definition of the contour sequence, i.e. terminate the CD mode or ramp to a stop and exit when the contour is executed.

**CK CONTOUR END AND KILL**

The CK command ends the definition of the contour sequence and stops output generation immediately when executed.

**CR#,#,# CIRCULAR INTERPOLATION**

The CR command causes the axes defined by the CD command to move in a circular pattern from the entry position. The parameters specify the center of the circle and distance to travel in radians. The CR command is only valid with contours of 2 axes.

**CV# CONTOUR VELOCITY**

The CV command allows the specification of the contouring velocity.

**CX CONTOUR EXECUTE**

The CX command causes the PC68 controller to execute the previously defined contour sequence.

**MT## MOVE TO**

The MT command causes the axes defined by the CD command to move to the specified absolute position using linear interpolation at constant velocity.

**RQ REQUEST QUEUE STATUS**

The RQ command returns the number of entries available in the contouring queue.

**PID FILTER CONTROL COMMANDS**

The following commands set the PID filter parameters

**KP# PROPORTIONAL GAIN COEFFICIENT**

The KP command sets the proportional gain coefficient on servo axes.

**KI# INTEGRAL GAIN COEFFICIENT**

The KI command sets the integral gain coefficient on servo axes.

**KD# DIFFERENTIAL GAIN COEFFICIENT**

The KD command sets the differential gain coefficient on servo axes.

**KV# VELOCITY FEEDFORWARD COEFFICIENT**

The KV command sets the velocity feedforward coefficient on servo axes.

**KA# ACCELERATION FEEDFORWARD COEFFICIENT**

The KA command sets the acceleration feedforward coefficient on servo axes.

**KO# OFFSET COEFFICIENT**

The KO command sets a DC offset to compensate for torque offset in the load.

**ENCODER COMMANDS**

The following are encoder support commands for use with the PC68.

*The following are position maintenance control commands:*

**ER## ENCODER RATIO**

The ER command allows specification of encoder ratio by entering encoder counts followed by motor counts, for position maintenance mode. (for digital output only)

**HV# HOLD VELOCITY**

The HV command specifies maximum position hold correction velocity. This is the peak velocity which will be used while making position corrections.

**HG# HOLD FILTER PARAMETERS**

The HG command specifies the gain for stepper applications with feedback.

**HD# HOLD DEADBAND**

The HD command specifies deadband counts for position hold. The PC68 will consider the control in position when the stage is within the specified parameter counts during position correction.

**HF HOLD OFF**

The HF command disables position hold, stall detection and tracking modes.

**HN HOLD ON**

The HN command enables position correction.

**IP INTERRUPT WHEN IN POSITION**

The IP command operates like the ID command, except the interrupt is deferred until the stage is within the specified deadband.

*The following commands control the slip or stall detection mode for stepper with encoder models of the PC68.*

**ES# ENCODER SLIP TOLERANCE**

The ES command parameter specifies tolerance before slip or stall is flagged in the status register.

**TN SLIP TOLERANCE KILL ON**

The TN command enables additional action taken by the encoder slip tolerance system when the position error exceeds that specified by the ES command. When this mode is on, the controller will flush the command queue, terminate the motion, set the slip flag and disable position maintenance for that particular axis.

**TF SLIP TOLERANCE KILL OFF**

The TF command disables the TN command.

**IS INTERRUPT ON SLIP**

The IS command will enable interrupts to the host when the position error during a move exceeds the parameter specified by an ES command. The interrupt will occur if the done interrupt has been enabled in the control register. A bit in the status register is also set to flag the source of the interrupt.

**RL RETURN SLIP STATUS**

The RL command returns the slip detection status of each axis. An S is returned if slip has occurred for that axis, or else an N is returned.

*The following command controls the tracking mode of the controls.*

**ET ENCODER TRACKING**

The ET command turns on the encoder tracking mode. The axis will track its encoder input, thus allowing one axis to follow the activity of another or a thumbwheel for manual positioning or the movement of another device that produces a signal compatible to the encoder inputs.

*The following commands control the home sequence when used with an encoder.*

**HE HOME ENCODER**

The HE command enables the encoder index mode, i.e. home is defined as the logical AND of the encoder index, the external home enable and the encoder quadrant.

**HS HOME SWITCH**

The HS command enables non-encoder (switch only) home mode.

*The following commands return status information about the encoder to the host.*

**EA ENCODER STATUS**

The EA command returns the encoder status of the currently addressed axis.

**RE REQUEST ENCODER POSITION**

The RE command returns the current encoder position of the currently addressed axis in encoder counts.

**VELOCITY STAIRCASE COMMANDS**

The following commands describe the velocity staircase mode. This mode is useful in applications requiring a change in velocity at a prescribed position without stopping.

**MP MOVE POSITIVE**

The MP command sets the direction logic to move in the positive direction.

**MM MOVE MINUS**

The MM command sets the direction logic to move in the negative direction.

**MV## MOVE VELOCITY**

The MV command causes the motor to run to the new absolute position at the specified velocity. A velocity staircase may be generated by queuing a sequence of MV commands.

**SP# STOP AT POSITION**

The SP command will cause the axis to stop at the specified position.

**FP# FORCE POSITION**

The FP command will flush the command queue and attempt to stop at the specified position.

**MACRO CONTROL COMMANDS****MD# MACRO DEFINITION**

The MD# command allows the user to combine several SRX commands into one of 20 macros.

**MX# MACRO EXECUTE**

The MS command allows the user to execute a specified macro.

**PROGRAMMING EXAMPLES**

In a typical move requirement where it is desired to home the stage then move to a specified position, the following will demonstrate the programming:

Initialize the velocity and acceleration parameters to a low value suitable for homing. Set a PID filter proportional gain of 2 and a derivative gain of 6. Perform the home operation initializing the position counter to zero.

Initialize the velocity and acceleration parameters to perform a faster motion and move to an absolute position of 10,000 counts from home in the negative direction and set the done flag when finished.

The following would be input from the host computer:

```
AX
VL1000 AC10000
KP2 KD6 HN
HM0
VL5000 AC50000
MA10000 GO ID
```

In a move requiring a three axis coordinated move to a position in free space the following could be used:

AX KP2 KD6 HN  
 AY KP2 KD6 HN  
 AZ KP2 KD6 HN  
 AM  
 VL5000,5000,5000;  
 AC50000,50000,50000;  
 MA1000,2000,3000; GO ID

The controller would calculate the relative velocities required to perform a straight line move from the current position to the desired position.

The following demonstrates cutting a hole with a 10,000 count radius using constant velocity contouring and circular interpolation:

The contouring velocity is set to 1000 counts per second. A contour is defined beginning at coordinates 0,0 on the Z and T axes.

Auxiliary output on the X axis is turned on, which could turn on the cutting torch or laser starting the cut at the center of the circle.

A half circle is cut from the center to the outside of the hole positioning the cutting tool at the start of the hole.

The hole is then cut, the torch turned off, the stage stopped and the contour definition completed.

The stage is then positioned and the contour definition executed.

The following would be input from the host computer:

CV1000  
 CD,,0,0;  
 AN0;  
 CR0,5000,3.1415926  
 CR0,0,6.2831853  
 AF0;  
 MT-10,10000  
 CE  
 MT,,-1000,0; GO  
 CX

## **SPECIFICATIONS**

### **Velocity**

0 to 1,044,000 counts per second simultaneous on each axis

### **Acceleration**

0 to 8,000,000 counts per second per second

### **Position range**

67,000,000 counts ( $\pm 33,500,000$ )

### **Accuracy**

Position accuracy and repeatability  $\pm 0$  counts for point to point moves

Velocity accuracy  $\pm 0.01\%$  for step pulse output

### **Environmental**

Operating temperature range: 0 to 50 degrees centigrade

Storage temperature range: -20 to 85 degrees centigrade

Humidity: 0 to 90% non-condensing

### **Power**

+5VDC at 1 amp typical

+12VDC at 0.1 amp typical

-12VDC at 0.1 amp typical

### **Dimensions**

3.550X3.775X0.5 inches high

### **Communication**

PC/104: Meets all signal specifications for PC/104 ISA bus specifications (IEEE P996.1)

RS-232: Baud rates of 300 to 38.4K

### **Limit switch inputs**

TTL input levels with on board 2.2K pull up resistor, requires only external switch closure to ground or TTL level input signal. Input sense (low or high true) selectable by command input for each axis.

### **Home switch inputs**

TTL input levels with on board 2.2K pull up resistor, requires only external switch closure to ground or TTL level input signal. Input sense (low or high true) selectable by command input for each axis.

### **User definable I/O**

Up to 12 bits of user definable I/O on PC68 models with 4 axes or less. 8 bits are user configurable as inputs or outputs. One auxiliary output per axis and are fixed as outputs. Factory default is 4 inputs, 4 outputs and 1 auxiliary per axis. (For PC68 models with more than 4 axes, up to twice this number of I/O in the same as described above.)

TTL input levels with on board 2.2K pull up resistor, requires only external switch closure to ground or TTL level input signal.

The auxiliary outputs are TTL open collector outputs (7406, max 48mA). The other outputs are TTL totem pole outputs (74LS243, max 24mA).

### **Analog outputs**

$\pm 10V$  and 0 to +10V

### **Step pulse output**

Pulse width 50% duty cycle. Open collector TTL level signal (7406, max 48mA).

### **Direction output**

Open collector TTL level signal (7406, max 48mA).

### **Encoder Feedback**

Maximum 4 MHz after 4x quadrature detection

Differential TTL level signal MC26G32, max 150mA

### **PC/104 interrupt**

Interrupts are user selectable, 2 through 7. The factory default is level 5.

### **PC/104 bus I/O address**

The I/O address block utilizes 4 consecutive addresses and is user selectable. The factory default is 300-303 hex.

**SIGNAL CONNECTOR (J5)**

| Pin# | Description    | Pin# | Description |
|------|----------------|------|-------------|
| 1    | Digital Ground | 35   | +5VDC       |
| 2    | I/O 1          | 36   | I/O 0       |
| 3    | I/O 3          | 37   | I/O 2       |
| 4    | I/O 5          | 38   | I/O 4       |
| 5    | I/O 7          | 39   | I/O 6       |
| 6    | Digital Ground | 40   | +5VDC       |
| 7    | X Index +      | 41   | Reserved    |
| 8    | X Index -      | 42   | X Step      |
| 9    | X Phase A +    | 43   | X Auxiliary |
| 10   | X Phase A -    | 44   | X Direction |
| 11   | X Phase B +    | 45   | X Limit +   |
| 12   | X Phase B -    | 46   | X Limit -   |
| 13   | Reserved       | 47   | X Home      |
| 14   | Y Index +      | 48   | Y Step      |
| 15   | Y Index -      | 49   | Y Auxiliary |
| 16   | Y Phase A +    | 50   | Y Direction |
| 17   | Y Phase A -    | 51   | Y Limit +   |
| 18   | Y Phase B +    | 52   | Y Limit -   |
| 19   | Y Phase B -    | 53   | Y Home      |
| 20   | Analog Ground  | 54   | +5VDC       |
| 21   | Z Index +      | 55   | Reserved    |
| 22   | Z Index -      | 56   | Z Step      |
| 23   | Z Phase A +    | 57   | Z Auxiliary |
| 24   | Z Phase A -    | 58   | Z Direction |
| 25   | Z Phase B +    | 59   | Z Limit +   |
| 26   | Z Phase B -    | 60   | Z Limit -   |
| 27   | Reserved       | 61   | Z Home      |
| 28   | T Index +      | 62   | T Step      |
| 29   | T Index -      | 63   | T Auxiliary |
| 30   | T Phase A +    | 64   | T Direction |
| 31   | T Phase A -    | 65   | T Limit +   |
| 32   | T Phase B +    | 66   | T Limit -   |
| 33   | T Phase B -    | 67   | T Home      |
| 34   | Digital Ground | 68   | +5VDC       |

**POWER SUPPLY CONNECTOR (J3)**

| Description | Pin | Pin | Description    |
|-------------|-----|-----|----------------|
| +5VDC       | 5   | 6   | Digital Ground |
| -12VDC      | 3   | 4   | N/C            |
| +12VDC      | 1   | 2   | Analog Ground  |

**RS-232 CONNECTOR (J4)**

| Description | Pin | Pin | Description |
|-------------|-----|-----|-------------|
| Ground      | 5   | 9   | N/C         |
| DTR         | 4   | 8   | CTS         |
| TxD         | 3   | 7   | N/C         |
| RxD         | 2   | 6   | N/C         |
| N/C         | 1   |     |             |

**ORDERING INFORMATION****OMS PC68 Intelligent Motion Controls**

| MODEL     | INTERFACE |        | SERVO AXES      | STEPPER AXES |          | USER I/O |
|-----------|-----------|--------|-----------------|--------------|----------|----------|
|           | PC/104    | RS-232 |                 | CONTROL      | FEEDBACK |          |
| PC68-21   |           | ✓      | Consult Factory | 2            |          | 10       |
| PC68-22   | ✓         | ✓      |                 | 2            |          | 10       |
| PC68-26   | ✓         | ✓      |                 |              | 2        | 10       |
| PC68-41   |           | ✓      |                 | 4            |          | 12       |
| PC68-42   | ✓         | ✓      |                 | 4            |          | 12       |
| PC68-1008 |           | ✓      |                 | 8            |          | 24       |
| PC68-2008 | ✓         | ✓      |                 | 8            |          | 24       |

**ACCESSORIES**

| MODEL    | DESCRIPTION   |
|----------|---|
| IO68-M   | I/O Breakout board for standard or expansion PC68 motion control boards |
| IO68-I   | I/O Breakout board for Px6IO daughter board                             |
| CBL68-3  | 3 ft cable w/mating connector 68 pin, (IO68/PC68)                       |
| CBL68-6  | 6 ft cable w/mating connector 68 pin, (IO68/PC68)                       |
| Px6O-224 | I/O daughter board for PC68 optically coupled 24 in and 24 out          |
| Px6O-48I | I/O daughter board for PC68 48 optically coupled inputs                 |
| Px6O-48O | I/O daughter board for PC68 48 optically coupled outputs                |
| ODSWSUPP | DLLs and drivers for Windows NT, 95 and 98                              |
| CDMAN    | Electronic Manual 1 per shipment provided, unless otherwise requested   |

3701-1000000

Revision E

**Oregon Micro Systems, Inc.™***A Pro-Dex Company*

1800 NW 169th Place • Suite C100

Beaverton, Oregon 97006

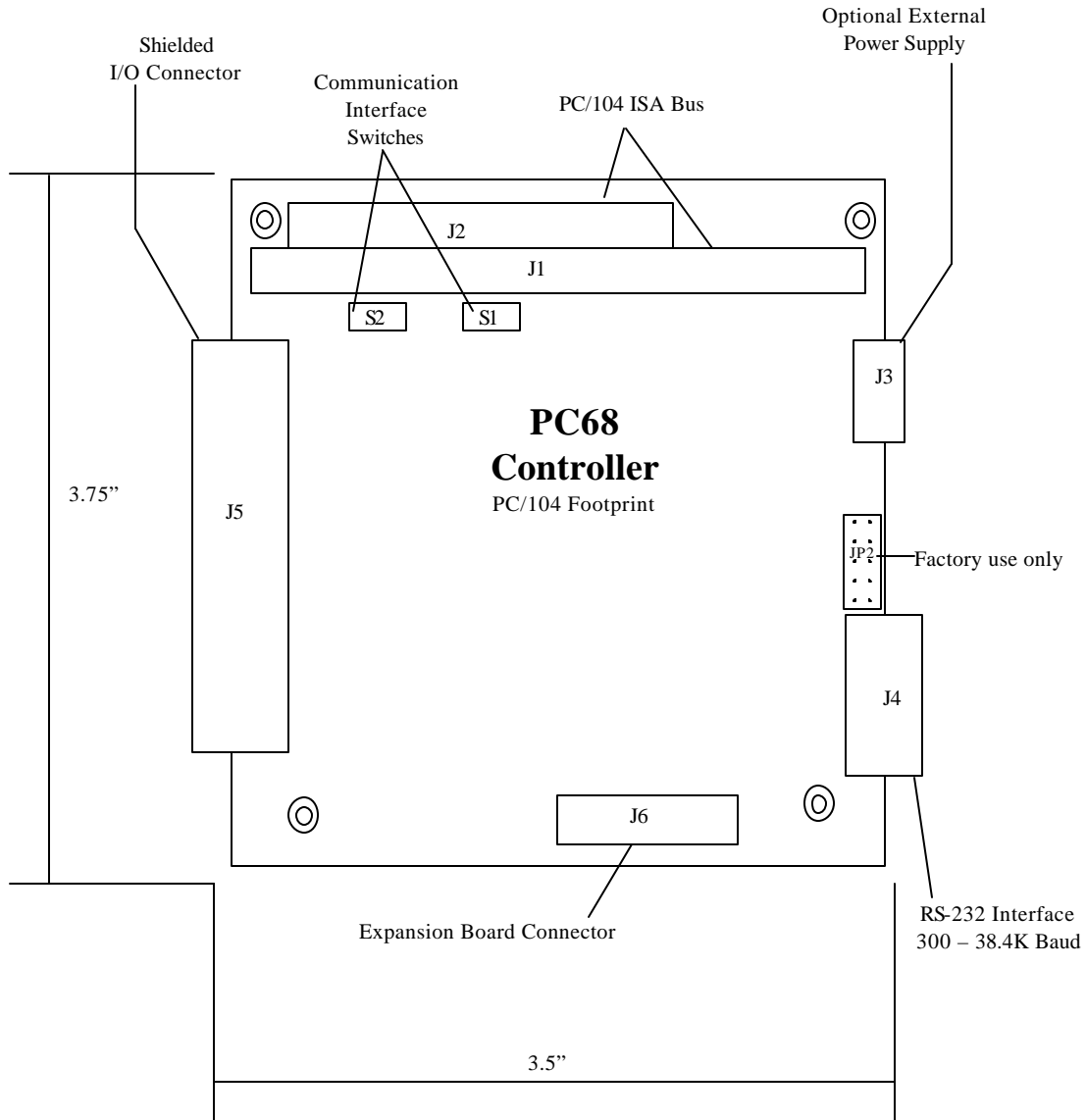
(503) 629-8081

Fax (503) 629-0688

Email: sales@OMSmotion.com

Web Site: www.OMSmotion.com

*Actual Size Depiction below*



**Oregon Micro Systems, Inc.**<sup>™</sup>

*A Pro-Dex Company*

1800 NW 169th Place • Suite C100

Beaverton, Oregon 97006

(503) 629-8081

Fax (503) 629-0688

Email: [sales@OMSmotion.com](mailto:sales@OMSmotion.com)

Web Site: [www.OMSmotion.com](http://www.OMSmotion.com)