PQD140 Remote Interface

Protocol: RS-232/EIA-232
Firmware version: 1.00
For board revision: PCB15083: All
Standard on all units.
Revision 1.01
Created: March 19, 2010
Revised: April 5, 2010
Applies to products: PQD140 (firmware 1.00 and higher)
By: Joel Styer
MicroImage Video Systems

Overview

The PQD140 includes a standard RS232 serial interface for remote control of the unit. This document describes the operation and interface of that port. This allows the PQD140 to be controlled by a computer, industrial controller or custom control device.

Physical Interface

The connections are made via a 3.5mm rear panel connector (an internal header connector is also available on the PC board). MicroImage Video Systems can provide standard cables to match standard serial ports. Customized cables are also available. The remote port specifications are:

Connection:	3.5mm Phone Jack (in addition to a 2mm Header Connector on the PC Board)
Physical Interface:	EIA232D (RS232)
Baud rate:	9600
Data bits:	8
Parity:	None
Stop bits:	1
Hardware Handshaking:	None
Protocol:	MicroImage Control Format Version 4 (MCL4-PQD140)

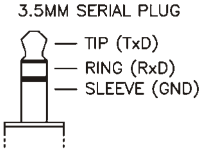
Connections

The PQD140 uses a small 3.5mm stereo phone jack for the serial connection. The connections are shown below.

BL-PQD140 board level units use a small 3 pin Hirose 2mm male connector for the serial connection. Some board level BL-PQD140 units may also have the 3.5mm phone jack as well. The Hirose 2mm pin connections are Pin 1 = Ground, Pin 2 = TxD and Pin 3 = RxD.

Data Transmission

This unit can receive a constant data stream from the host. Data is pipelined in such a way within the unit that the result may not be visible for up to 70 milliseconds after the last data byte in a command sequence has been received. Normally, this is not a problem since it still appears to happen immediately to the user. Response to queries may not start until up to 70mS after the request has finished. Queries are queued but may not always be transmitted in the order requests are made.



Command Structure

The command structure is composed of simple ASCII characters and is fairly easy to implement. It can also be tested with a standard ASCII terminal or emulation program. The command sequence consists of a Start character, Command ID, Data byte(s) and a Stop character. The start and stop bytes are very important and should not be forgotten. Data is not processed internally until the stop character ( ] - ASCII $5D hex) is received.

The command structure: [A####]

The PQD140 also implements an extended command structure for additional addressing capabilities. This command structure is as follows:

Extended command structure: [Zaaaa:dddd]

As with the simplified commands, it starts and ends with the brackets and has a single letter for the command. In the case of the PQD140, the command “Z” uses an extended address scheme that also requires an address and a colon to separate the address from the data. The address and data are in hexadecimal and each can be up to 16 bits/4 hex characters. Preceding zeros are not required but may be used for clarity or formatting if needed. Thus the command [Z4:86] is identical to [Z0004:0086].

The basic structure for queries is just a slight bit different with addition of a question mark:

Query structure: [?A]

The start character is an open bracket ([- ASCII $5B hex). Transmitting a “[“ character in the middle of any command sequence will cancel (reset) the previous sequence and start a new one. Transmitting a sequence of [ ] (open bracket, close bracket) will cancel any pending command.

Note: An extended query structure has been defined but not implented in this product. It will be similar to the extended command listed above with the addition of the question mark (?) Added to the command to signal that it is a query command.

The command ID can be any valid character from the list in the “commands” section of this document. It must follow the start character. In the PQD140 , only letters and some punctuation are used to initiate commands. In the PQD140, letters (A-Z) may be uppercase or lowercase for the command. If an invalid ID character is sent, the unit will respond with a question mark (~, ASCII $7E hex) as an acknowledgment. This will be sent regardless of the setting of the acknowledge level set by the [A] command. The acknowledge level may add additional characters as well.

The data sequence can consist of 0 to 4 numeric digits and has a different meaning for each command. Data can be in decimal or hexadecimal format and the hex letters (A-F) may be in uppercase (A-F) or lowercase (a-f).

Leading zeros may be used ahead of the number. If a command requires the number 1, this may be entered as 1, 01, 001 or 0001 - all will be interpreted correctly. As an example, to set the unit to L/R split, you could enter:

[M1]
[M01]
[M001]
[M0001]

and all will produce the correct result. If more than four numbers are entered (i.e. M98760001]) the last four entered are used, (0001 in the recent example).

If the command is entered without the data argument (i.e. [M]), the data will be interpreted as 0000 (Zero). In this case, it would set the system to the Input A mode, the same as if the command [M0] would have been entered. If it is an extended command, it will be interpreted as 0:0 - zero for the address and zero for the data. If only one is missing, that one will be interpreted as zero.

All characters in the data sequence are hexadecimal formatted. Some commands have less than nine functions so the data can be hexadecimal or decimal with the same result.

The stop byte consists of a closing bracket ( ] - ASCII $5D hex). Upon reception of the stop character, the command is sent to a buffer which holds the full command until the process can parsed and the command executed. This may take up to 70mS to complete after the reception of the stop byte. A second command can be started while the first one is executing but the stop byte for the second command should not be sent until the first function has completed.

Data Reception

Data is received in the unit by an interrupt routine but is not parsed until the entire string has been sent. The data is double buffered and the parsing occurs within a separate task. The acknowledgment codes (if used) are also generated by a separate task. Due to this multitasking within the unit, the programmed action may not occur for up to 70mS after the stop byte ( ]) is received. If the acknowledge command is set to send an acknowledgment, it will be sent after the command operation has completed.

Commands

A **Acknowledge Level** **[A##] where # = 0 through 99 as defined below**
[uppercase A]

Command Examples:

[A0] = Quiet mode. Only errors will be acknowledged with a "?" (Power on default).

[A1] = Simple Acknowledge - will return a semicolon (";" ASCII $3B hex) only, except when there is a command error, then it will only return a question mark (?).

[A2] = Command Acknowledge - will return the sent command character and the semicolon except when there is a command error, then it will only return question mark (?).

[A3] = Command and Data Acknowledge - will return the command character, four data characters and the semicolon terminator. Note that regardless of the number of data characters sent, four will always be returned. The leading characters will be all zeros if not used. The exception is when there is a command error, then only the question mark (?) Will be returned.

[A99] = Debug Mode - Will return all the data as in [A3] above plus add a carriage return (CR) and line feed (LF) (ASCII $0D and $0A hex respectfully). The CRLF WILL be appended to the command error acknowledge (?). Debug mode should NOT be used in system. It is designed only for testing and the format could change in the future.

B **Border** **[B#]** **where # = 0 to 4**
[uppercase B]

Command Examples:

[B0] = No Border (Disabled)
[B1] = Black Border
[B2] = Dark Gray Border
[B3] = Light Gray Border
[B4] = White Border

At power up, the last setting used is restored from EEPROM.
The indicator on the front panel will illuminate when this command is enabled, off when disabled.

Note that the border is only visible in the multi display modes like Dual and Quad. See the operation manual for additional information.

E **Exchange** **[E#]** **RESERVED**
[uppercase E]

M **Mode** **[M#]** **RESERVED**
[uppercase M]

X **System Settings** **[X#]** **RESERVED**
[uppercase X]

Z Display Programming [Zaaaa:dddd]

[uppercase Z]

Command Examples:

[Z1:26] = Shifts the A image up 32 lines (26 h =38 dec. minus default value of 6.
[Z47:3C] = Sets the start of the C image window 3/4th of the way down screen.
[Z69:78] = Sets the stop (end) of the D window at the bottom of the screen.
[Z20:28] = Shifts the B channel image left by 8 pixels (minus default of 20h).
[Z0:1C] = Shifts the A channel image right by 4 pixels.
[Z18:90] = Set the brightness of channel A brighter.
[Z5A:0] = Turns the chroma (color) down to 0 (monochrome) for channel C.
[ZF0F0] = Resets all channel adjustments (those from Z commands).
[ZF0F1] = Resets entire unit back to factory defaults.

NOTE: All values for address and data are in hexadecimal.
NOTE: The delay (position shift) commands have default values of 20 (horizontal) and 6 (vertical) for a centered picture. It is important to account for these offsets when shifting the start point of the image - i.e. a value of 21 will shift the image one pixel left of default horizontal center position (within the display window) and a value of 7 will shift the image one pixel up from the default vertical center.

The Z command is an extension of the standard command set and includes an address so many different functions and values can be set via the Z command. The Z command is used to set the display values for the quad controller. These include the scaling, cropping and shifting of the source image along with the position and size of the four available display windows.

PQD140 Extended Command Set

Address A	Address B	Address C	Address D	Command	Range	Default	Resolution
0	20	40	60	Source Delay H	0-3FFh	1Ah	1 pixel
1	21	41	61	Source Delay Y	0-1FFh	6h	2 lines
2	22	42	62	Source Active X	0-2D0h	2D0h	1 pixel
3	23	43	63	Source Active Y	0-F0h	F0h	2 lines
4	24	44	64	Source Scale X	0-FFFFh	FFFFh	
5	25	45	65	Source Scale Y	0-FFFFh	FFFFh	
6	26	46	66	Display Start X (Left)	0-B4h	0	4 pixels
7	27	47	67	Display Start Y	0-78h(N), 0-90h(P)	0	4 lines
8	28	48	68	Display End X	0-B4h*	B4h	4 pixels
9	29	49	69	Display End Y	0-78h(N)*, 0-90h(P)*	78h(N), 90h(P)	4 lines
A	2A	4A	6A	Channel Pop-Up & Enable	0-3h	1h	N/A
B	2B	4B	6B	*RESERVED*			
C	2C	4C	6C	*RESERVED*			
D	2D	4D	6D	*RESERVED*			
E	2E	4E	6E	*RESERVED*			
F	2F	4F	6F	*RESERVED*			
10	30	50	70	*RESERVED*			
11	31	51	71	*RESERVED*			
12	32	52	72	*Background Color (FUTURE)*			
13	33	53	73	*Mirror/Flip (FUTURE)*			
14	34	54	74	*Freeze/Blank (FUTURE)*			
15	35	55	75	*Input Channel Select (FUTURE)*			
16	36	56	76	*RESERVED*			
17	37	57	77	*Decoder Status (FUTURE)*			
18	38	58	78	Brightness Adjust	0-FFh	80h	0.2 IRE
19	39	59	79	Contrast Adjust	0-FFh	80h	0.80%
1A	3A	5A	7A	Chroma Adjust	0-FFh	80h	0.80%
1B	3B	5B	7B	Hue Adjust	0-FFh	80h	1.4 deg
1C	3C	5C	7C	*RESERVED*	0-FFh	80h	0.80%
1D	3D	5D	7D	*RESERVED*	0-FFh	80h	0.80%
1E	3E	5E	7E	*RESERVED*	0-FFh	82h	0.40%
1F	3F	5F	7F	*RESERVED*	0-FFh	82h	0.40%

NOTE: “h” indicates a hexadecimal value.	Rev 1.0
* = In order to display the border at the right and bottom of the screen, you may need to subtract 1 from each of these values.	

At the present time, there are sixteen basic settings associated with the Z command. Each can be set individually for each of the four channels. The commands are as follows:

DELAY H - Sets the horizontal position of the source image within the display window. Higher values move the image to the left.
DELAY V - Sets the vertical position of the source image within the display window. Higher values move the image up.
ACTIVE H - Sets the number of horizontal active pixels from the source image. There must be enough to fill the display window or noise will be visible in the gap.
ACTIVE V - Sets the number of vertical active pixels from the source image. There must be enough to fill the display window or noise will be visible in the gap.
SCALE H - This sets the horizontal scaling value of the source image. A value of $FFFF is full size and not scaled. A value of $7FFF is half size, $3FFF is one-fourth size, etc.
SCALE V - This sets the vertical scaling value of the source image. A value of $FFFF is full size and not scaled. A value of $7FFF is half size, $3FFF is one-fourth size, etc.
START H - This is the horizontal position (left) of where the display window starts (in increments of four pixels).
START V - This is the vertical position (top) of where the display window starts (in increments of four pixels).
STOP H - This is the horizontal position (right) of where the display window stops (in increments of four pixels).
STOP V - This is the vertical position (bottom) of where the display window stops (in increments of four pixels).
POP-UP/EN - This selects the display priority when images are stacked and enables each channel.
BRIGHTNESS - Changes the brightness of the selected channel.
CONTRAST - Changes the contrast of the selected channel.
CHROMA- Changes the chroma level (color) of the selected channel.
HUE - Changes the hue (tint/ sc phase) of the selected channel.

Calculating values for the above can be difficult the first time. First, layout how you want the display to look, noting the locations for each of the four windows. Set unused windows to a stop and start point of zero. An NTSC screen is 720 pixels wide (H) by 480 pixels high (V). The position of each window is set in blocks of four pixels. To display one window full screen, you would set the H START and V START values to 0, then set the H STOP to $B4 (hexadecimal value of 720 divided by 4) and the V STOP VALUE to $78 (hexadecimal value of 480 divided by 4).

Next decide if any scaling is needed. If so, the value needs to be calculated based on system needs. If you have a display window 1/2 the screen size in height (240 pixels) and you want to scale a full size image to fit it, the scaling factor would be 1/2 (a value of $7FFF or $FFFF divided by 2) and the number of active pixels would be 480 (full raster - the number of source pixels required BEFORE scaling)

If scaling is not needed, then the value of ACTIVE pixels is simply 4 times the value of the window STOP minus START. If the display window covers the top 1/4th of the screen, it’s start and stop value would be 0 and $1E ($78/4) respectively. The number of active pixels would be four times that or a value of $78 (120). In some cases, the value of active pixels can be higher, up to full screen size. This depends on the priority order of the images (see the pop-up section below)

DELAY is the value used to offset the start of the image within a window. When set to the default value, the image will always start at the left or top inside the window, regardless of the window placement on the screen. If the window is smaller than full screen, you can shift the start position inside the video source to center or otherwise reposition the image with the window. Delay is independent of START and STOP. Shifting the image out of the window may produce unexpected results and noise in the undefined areas. The image can be moved in increments of 1 pixel or 1 line.

ACTIVE is used to indicate the active number of pixels within a window. This works in conjunction with Delay and Scale. If the active number of pixels (after scaling and delay) is less than the window size, noise may appear in the undefined areas of the window. Active pixels are set in increments of 1 pixel or 1 line.

SCALE is a 16 bit coefficient used to define the amount of scaling applied to the image. A value of $0FFFF (hex) indicates no scaling. Divide 0FFFF (hex) by the scaling ratio you require. If the image should be 1/4th the screen width, then divide $0FFFFh by 4 to get $03FFFF (hex) and the proper scaling ratio.

START is the location where the display window view begins on the screen. The value is in groups of four pixels so the values will be 1/4th of the values for DELAY and ACTIVE.

STOP is the location where the display window view ends on the screen. The value is in groups of four pixels so the values will be 1/4th of the values for DELAY and ACTIVE.

An Example: suppose you had a window that was 1/4th of the screen width; 720/4 = 180 ($B4), and you required the center half of the source image to be displayed within that window; 720/2 = 360 pixels so ACTIVE value would be $168. Since you need the center of the image, the half not used is split equally on each side, thus 180 pixels are on each side. So the value for DELAY would be 180+26 (the default value) = 206 ($CE in hex). Since the image is scaled to half (50%), the scaling value would be 1/2 or $7FFF hex. If the display window was to be in the upper right corner of the screen, it would have a horizontal START value of $5A hex which is half of the total width value of $B4 hex (720 pixels divided by 4 or 180 decimal). The STOP value would be the end of the raster or $B4.

POP-UP/ENABLE is used to turn off or on an individual channel and to enable the pop-up which sets the designated channel to a higher priority than a channel that has pop-up turned off. Normally, channel D has priority over C, which has priority over B, and A has the lowest priority for display. So if images overlap, or to make a picture-in-picture type display, the pop-up function can be set to insure the correct channel has priority. There are three useful values:

0 = The channel is disabled.
1 = The channel is enabled but pop-up priority is not set (default setting).
2 = Not used.
3 = The channel is enabled and and pop-up priority is set.

BRIGHTNESS allows you to make the image brighter or darker than the default setting.

CONTRAST allows you to increase or decrease the image contrast to more or less definition.

CHROMA allows you to increase or decrease the color in the image. Setting chroma to zero will give you a monochrome (B&W) image.

HUE - also referred to as phase and tint, hue allows you to change the phase angle of the decoder to correct for incorrect colors (incorrect colors that were/are modulated at the wrong phase angle or a video signal with a damaged color burst or poor line timing). The default setting is typically the best.

Predefined Profiles

The PQD140 also has built-in predefined profiles. Each of these profiles will set the commands and functions listed under the "Z" command section. Currently, there are two ways to access these profiles. The first method (via command [Z9000:dddd] will set the display position parameters and the display levels (brightness, contrast, etc). The second access method [Z9001:dddd] will only change the display parameters and not affect the display levels. This makes it easier to change profiles and not lose or have to resend level adjustments for each channel. The four hex data characters after the colon select the desired profile. Profiles can not be created or edited by the user.

Examples:

```
Z9000:dddd
Z9001:dddd
```

Standard profiles:

Currently defined profiles for version 1.0 include the following:

[Z900x:0]	Full frame channel A
[Z900x:1]	Full frame channel B
[Z900x:2]	Full frame channel C
[Z900x:3]	Full frame channel D
[Z900x:4]	Traditional quad (four quadrant) display
[Z900x:5]	Four equal size horizontal bands of each channel
[Z900x:6]	Test pattern - THIS WILL CHANGE IN THE FUTURE
[Z900x:7]	Test pattern - THIS WILL CHANGE IN THE FUTURE

WHERE:

x=0 for command Z9000 (set parameters)
x=1 for command Z9001 (set only display dimension parameters, not brightness, contrast, chroma and hue)

Restoring Defaults

Resetting Display Parameters and Restore System to Factory Defaults

[ZF0F0] Resets all settings listed within the Z command and sets the profile to #4 (quad). No data required after address, if the colon or data are sent, they are ignored.

[ZF0F1] Initializes the unit to factory defaults. Resets ALL PQD140 settings and sets the profile to #4 (quad). No data required after address, if the colon or data are sent, they are ignored.

Query Commands

The user or computer system can ask the unit for information (data query). Virtually all settings can be transmitted from the unit. Below are the commands for queries.

! **Test** **!** (hex $21, does not use brackets)
[exclamation point]

Command Example: ! (hex $21) (single character, does not use brackets)

Returned Data: ! (hex $21) (single character, returns same character)

When the unit receives the test character, it will return the same character to the sender within 70mS. This is a quick way to check communications with the unit and for a computer system to make sure the unit is booted and available.

**ID** **#** (does not use brackets)
[number symbol]

Command Example: # (hex $23, single character, does not use brackets)

Returned Data: when the unit receives the ID character, it will return a string of comma delimited data about the product. An example string would look similar to the following:

```
[mPQD140,v0100,l0000,d20100319.]
```

Other fields may also be present as well as needs dictate.

Explanation of fields:

m = Model Number field.
v = firmware code version
l = logic/DSP code version
d = date of code base

^ **Send All** **^** (hex $5E, single character, does not use brackets)
[caret]

Command Example:	^ (hex $5E, single character, does not use brackets)
Returned Data:	when the unit receives the Send All character (^), it will return a string of data that is the same as issuing most query commands: [A##][E#][H#][M#][X#] Please see the individual query commands for the exact formats and options associated with each.

This command is the same as issuing all of the following commands: [?A][?E][?H][?M][?X].

The Send All command is an easy way to query the unit for it's current settings, such as on power up so a remote system can synchronize with it.

Note: In the future, more commands are likely to be added to this sequence.
Note: Commands may not always be sent in the same order or sequence. Decode the commands based on the start character and the following command letter, never by the byte position in the sequence.

?A Query Acknowledge Level [?A]
[uppercase ?A]

Command Example:	[?A]
Returned Data:	[A##]

where # = 00 through FFh (see Acknowledge Level command for above for details)

?B Query Border [?B]
[uppercase ?E]

Command Example:	[?B]
Returned Data:	[B0] = No Border (Disabled) [B1] = Black Border [B2] = Dark Gray Border [B3] = Light Gray Border [B4] = White Border

At power up, the last setting used is restored from EEPROM.
This command will display changes made via the BORDER switch on the front panel or via the serial B command. The BORDER indicator on the front panel will also indicate the state of this setting - illuminated when enabled.

?E Query Exchange [?E] RESERVED
[uppercase ?E]

?M Query Mode [?M] RESERVED
[uppercase ?M]

?X Query System Settings [?X] RESERVED
[uppercase ?X]

Changes

In the future, there will likely be additions to this command structure. We will strive to keep all new functions backward compatible to minimize any problems. Do not send any commands that are not currently implemented, as this could cause future incompatibilities.

The latest version of this document will be on the MicroImage Video Systems web site at www.mivs.com. Look in the Technical (Support)/Manuals section or at the PQD140 page(s).

Assistance

If you have any questions regarding this document or product, please call:

MicroImage Video Systems
div of World Video Sales Co., Inc.
PO Box 331, Boyertown, PA 19512
Phone 610-754-6800, Fax 610-754-9766
Email engineering@mivs.com
Web www.mivs.com