P2 Profiler Specifications

DESCRIPTION

The P2 Profiler is a standalone **server** housed in the standard QBX housing that incorporates a mini internal PLC controller. The application is for users that do not have a central PLC controller, however it can be connected to a PLC if desired. The **server** will execute output changes based on time and/or input triggers. The **server** can contain up to 96 events which are executed sequentially. With the execution of an event the output will go to the programmed event output for the programmed time. Once the time has expired it will advance to the next event, executing each event in order until all events have been executed at which time it will return to event one and begin again. The **server** can be programmed to accept a trigger on any event. If a trigger is programmed to start an event, the event will not execute until the trigger is actuated. Each event also has the option to ramp to the programmed output. Ramp events can be stacked to create the desired output air pressure profile. A specific sequence of events is an *event profile* or *profile*.

SPECIFICATIONS

ELECTRICAL POWER
SUPPLY VOLTAGE: 15-24 VDC
SUPPLY CURRENT: 400 mA
ELECTRICAL CONNECTOR: M12, 8-pin, A-Coded

DIGITAL
COMMAND RESOLUTION: 16 bits
FEEDBACK RESOLUTION: 16 bits

NETWORK INTERFACE
Interface: Ethernet 10Base-T or 100Base-TX (Autosensing)
Protocols: TCP/IP, UDP/IP, DCHP
Connector: M12, 4 pin, D-Coded Ethernet connectors.
Port number: 10001

ENVIRONMENTAL
Operation: 32-158°F (0-70°C)

OPERATIONAL DETAILS

Number of Events per Profile: 1 to 96 events

Event time range: 25 to 65,000ms

Event time resolution: 1ms ± 0.25

P2 Profiler Specifications

Pressure Resolution: ±0.2% F.S.

Hardware Trigger: +3 to +Vsupply constitutes a valid trigger. Three Triggers available.

Hardware Reset: +3 to +Vsupply constitutes a valid Reset.

Outputs: ON (approximately 0VDC) and OFF (approximately +Vsupply). Two Outputs available.

Figure 1 Power and network connectors

Input/Output Connector			Network Connector	
Pin	**Description**		**Pin**	**Description**
1	24Volt Power		1	+Tx
2	Output 1		2	+Rx
3	Output 2		3	-Tx
4	Reset Input		4	-Rx
5	Ground			
6	Trigger Three			
7	Trigger two			
8	Trigger One			

Table 1: Power and I/O Connector Pin Out

Networking: The P2 will connect to a standard TCP/IP network. By default the P2 is set to pull an IP address from a DHCP server.   If the P2 is connected directly to a laptop or PC where a DHCP network is not available, the P2 will default to a 169 address. To communicate with the P2 the in this case, the laptop or PC must be on the same network/subnet as the P2.   The P2 communicates on port number 10001.   Telnet may be used to test the connection to the P2.

P2 Profiler Specifications

Manual Automatic modes: The P2 had two modes of operation. In the automatic mode the P2 executes events sequentially from the stored event profile. In the manual mode in which events are not automatically executed and waits for a command to execute each event in the updated event profile. Some commands in the P2 will not function in automatic mode.

Reset: +3 to +24VDC. A momentary voltage transition from 0 VDC to at least +3 VDC applied on the hardware reset input will cause the **server** to go to event 1 and begin execution. The hardware reset line must not be left floating. Holding a constant trigger voltage on the reset pin will cause the **server** to remain stopped on event 1. A reset will cause the system to go to the pre-programmed fault. If a software "reset" command is sent, a "run" command must be sent to continue operation. The software reset command works the same as the hardware reset except it can only be used in manual mode.

Ramped Event: The **server** will ramp steps from the current pressure to the new pressure. The Ramp rate is determined by the event time and pressure. The shorter the time and the larger the pressure change, the steeper the ramp. Ramp events can be listed one after the other to create the desired air pressure profile.

Trigger: +3 to +24VDC. An input voltage of at least +3VDC applied to the trigger pin constitutes a satisfied trigger. One of three momentary input triggers can be applied to any event. If a trigger is programmed on an event, the **server** will stop at the beginning of the event until the trigger is satisfied. Once the trigger is satisfied, the event is executed. If a trigger is in the activate state at the beginning of an event, the trigger would be satisfied and the event will execute. This would be useful if it is desired to have the event execution to continue unless the input is not present. The trigger inputs are available on the I/O connector as shown in *Table 2: Power and I/O Connector Pin Out.* NOTE: The trigger works differently for the test event. See the test event for the details on its operation.

Test Event: The test event is used to compare the output pressure to the previous set point command. The test event has special rules that only apply to the test event as noted below:
1. The value in the event time field is the hold time before performing the test.
2. The test is performed in the last 10ms of the event time.
3. The event retains the pressure set by the event before the test event.
4. The event field is the amount of allowable error between the set point and the measured pressure. Caution: Zero should not be used as it will nearly always fail due to analog to digital translation errors.
5. If a trigger is selected for the test event, execution of the test event stops at the end of the test event, whereas for all other events it stops at the beginning of the event. When the trigger is satisfied the server advances to the next event.
6. If no triggers are selected, and the test fails, the server will stop at the end of the test event, but if the test passes it will continue to the next event. If the server stops on a test failure, the server will require a high on the reset input pin to reset the server to event 1. When the reset is removed the server will execute event 1. The server will also continue to the next event if a software *"Run Command"* is issued.
7. If output bit(s) are selected and the test fails, the selected output(s) will activate. No outputs will activate if the test passes.

Programming: Uploading of the customer profile would be via the programming connector on the **server** via a laptop. It is not necessary for the programming laptop to remain connected once programming of the profile is complete. A graphic interface can be downloaded from the Proportion-Air web site to make programming easy and to manually control the **server**.   See the manual for the programming graphic interface for information on its use. The commands listed in the "Programming Details" can be used to program the server using a PLC.

Memory: All events are stored in nonvolatile memory to retain the customer programming if a power loss occurs.   If power is lost the system restarts at event one.

INDICATORS

Two indicators are located on the side of the P2 can. The right indicator is the current network status of the P2.   A right LED would indicate a the current network link status of the P2. A green LED condition would indicate the P2 is linked and accessible on the network.   The left indicator is the major/minor failure indication. A red LED would indication a failure of the internal memory of the P2. Minor errors will blink the major/minor red LED.   If cycling the power on the P2 does not clear a major/minor error contract Proportion-Air for assistance.

PROGRAMMING DETAILS

Customer programming of the events is done via a customer generated spreadsheet using common software such as Excel. Each line on the spreadsheet would represent an event.   Excel has the capability of plotting /time of graphical form so the customer can view the /time curve before uploading it.

For an example, assume that the application is to move an air cylinder rapidly to an object but not touch it. At this point, we wait until the trigger indicates the object is in place. Then slowly move forward until contact is made. Once the cylinder is in contact with the object, accelerate rapidly to a predetermined location and slow until the object is in place. We will wait for a trigger to indicate the object is in place and return to the starting position. The air profile may look like *Figure 2: Example Air Profile*.

An example of an event profile in spread sheet format is shown in *Figure 3*.

P2 Profiler Specifications
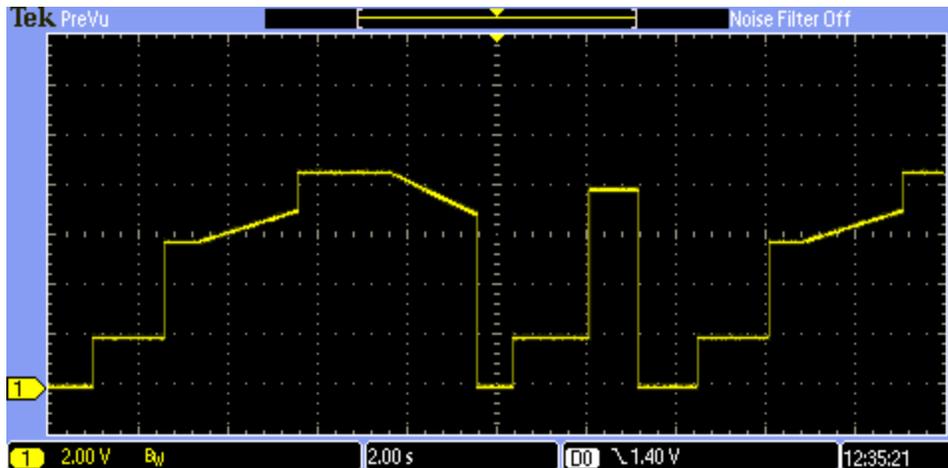
Figure 2: Example Air Profile.

Spreadsheet Rules:
1. Fields can not be left blank.
2. Values outside the acceptable range will be rejected and cause the profile not to load.

Spreadsheet Fields
Time field accepts a value from 20ms to 65000ms.

Percent of calibrated value field accepts a value from 0psi to calibrated value.

Trigger field accepts a value of 0 for no trigger, 1 for trigger one, 2 for trigger two or 3 for trigger three.
Ramp field accepts a 1 to ramp this event or 0 for no ramp.

Output field accepts a 1 to turn the output ON or a 0 to turn the output OFF. A selected output is activated at the beginning of the event.

ValveON/OFF field excepts a 0 to allow the valves to make updates or 1 to disable updates. Because the valves must be enabled to make updates, if the new event does not have the same value in the **Percent of calibrated value field** as the previous event, the Valve disable field is ignored.

TestValue field accepts a 1 if the current percent of full scale value is to be compared to the set percent of full scale value or a 0 if no test is to be performed.

Profile Example:
The spreadsheet in *Figure 3* creates an interesting profile with just 7 steps as explained below:

Event 1: When trigger 1 is received to start the event which will ramp the from 0 to 16.67 percent of full scale over a 3 seconds (3000ms) period.

Event 2: Hold the 16.67 percent of full scale for another 3 seconds.

Event 3: Turn off the valves and wait 7 second (7000ms) then compare the current value to the 16.67 percent of full scale . If the error is more than ±0.05percent of full scale activate output 1 and stop. If the error is less than ±0.05percent of full scale than continue to the next event.

Event 4: Turn the valves back on and maintain 16.67 percent of full scale for 3 seconds. Also deactivate output 1 and activate output 2.

Event 5: At the beginning of the event, the outputs are deactivated and the valves are deactivated. The event waits 7 second (7000ms) and in the final 10ms of the event compares the current value to the 16.67 percent of full scale set . If the error is more than ±0.05percent of full scale output 1 is activated and execution is stopped. If the error is less than ±0.05percent of full scale no outputs are activated and execution continues to the next event.

Event 6: Set the value to 0, activates output 2 and stops event execution.

An input on the reset pin would set the event counter to 1 and it will be ready to preform the next test.
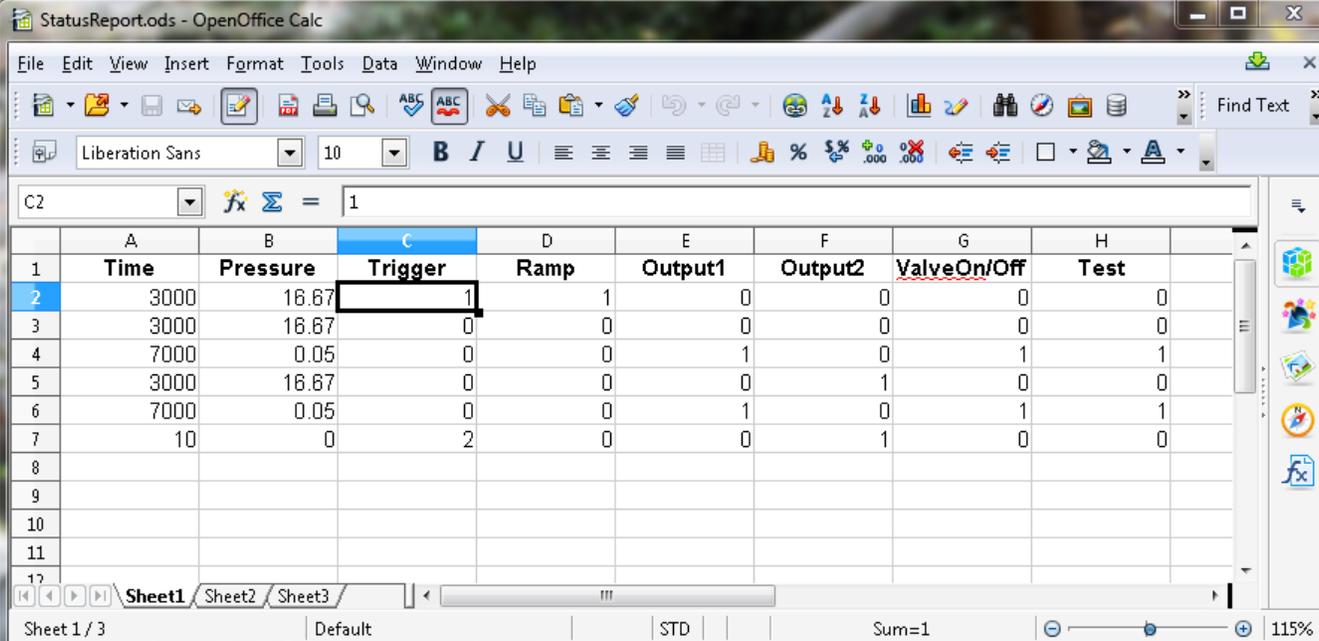
Time	Pressure	Trigger	Ramp	Output1	Output2	ValveOn/Off	Test
3000	16.67	1	1	0	0	0	0
3000	16.67	0	0	0	0	0	0
7000	0.05	0	0	1	0	1	1
3000	16.67	0	0	0	1	0	0
7000	0.05	0	0	1	0	1	1
10	0	2	0	0	1	0	0

Figure *3: Sample Spread Sheet*

Loading the Server with Events

To program the device can be a PC or laptop that has a network port to connect to the server, office

software loaded that can create a spreadsheet such as Excel and the Proportion-Air provided graphic software would read the profile. Once the profile is created on the spreadsheet, it would be saved as a comma delimited file (CSV) which is a standard option with most spreadsheet software. See the appendix in the documentation for the Proportion-Air graphical software for a detail explanation on how to create the CSV file.   A Proportion-Air provided graphic software will read the CVS file and upload it into to **server** via the programming port on the **server**.   The programming device can be removed once programming is finished.   Once the upload to the **server** is complete, the graphical software will display the event map as seen by the **server** and provide a method for the user to interact with the **server** to manually control, test and fine tune the **server's** program.

 The profile can than be retrieved from the **server** using the provided software for storage in the rare case that the **server** should need to be replaced and then reloaded.

PLC COMMANDS

The was not the intent of the P2 **server** to be connected to a PLC however it is capable of having a PLC connection which allows it to be programmed and controlled by a PLC if so desired. The following are a list of   commands. A complete list can be seen in *Table 2: Command Table*. It is not the intent of this **server** to be PLC controlled and other Proportion-Air devices may be better suited for PLC control.

Add Event Command (aevc): The add event command loads the incoming event into the event table. As many events as desired can be loaded by this command up to the maximum allowable events.   The format is aevc:vv,wwww,XXXxx,yyyy where vv is the event number, wwww is the time in milliseconds, XXXxx is the . assumes there are always two digits after the decimal place so that an input of 4567 yields the of 45.67.   If a 54.3 is desired the input would be 5430. yyyy is the option field with each bit representing an option as show in table 1. The option field is transmitted as a hexadecimal number.   When loading an event manually, it is important to not skip or duplicate the event numbers however events can be loaded in any order.   When the **server** receives an event of 0 (aevc:0:0:0:0), the event table is written to the **server'**s memory.   It is best to use the Proportion-Air graphical interface to load events as it prevents many errors. This command only works in manual control mode.

option bits:
Low byte

Bit:	7	6	5	4	3	2	1	0
Option:	Test Results	Valve enabled / disabled	Output1	Output2	Ramp	Trigger	Trigger	Trigger

High byte

Bit:	7	6	5	4	3	2	1	0
Error:								

P2 Profiler Specifications

Table 1 Option bit structure

Automatic Mode Command (amc): The automatic mode command causes the **server** to automatically execute the events.   When the automatic mode command is issued, the event currently being pointed to will be executed.

Automatic Event Command (avr): This responds is returned when the server automatically advances to next event in the event list. If a trigger is programmed into the event, it will wait for the trigger. If a next event command is issued on an event that requires a trigger that has yet to be satisfied, the event progresses to the next event without running the triggered event. When the current event is at the last event in memory and a next event command is issued it issues the first event in memory. This command only works in manual control mode.

Current Status Command (csc): The current status command returns the current state of the **server** in the form of: csr:[NumberOfEventsInTheServer],[ServerStatus]. NumberOfEventsInTheServer is the current number of events in the system.   The ServerStatus is a Hex value that returns the current status of the server. The bit values of the status byte are shown in the *Table 2 Status Byte* below:

7	6	5	4	3	2	1	0
Minor error	Major error			Reset active	Pause active	Manual mode	Trigger hold

Table 2 Status Byte

Get Current Event Command (gec): The get current event command will return the parameters of the current event in the form "rer:e,tttt,ppppp,yyyy" where "e" is the event number, "tttt" is the time, "ppppp" is the from 0 to 100percent of calculated   ,yyyy is the option field with each bit represents an option as show in table 1. The option field is transmitted as a hexadecimal number where the bit match Table 1 .   If event number is included in the event request the current event is returned.   This command only works in manual mode.

Initializes Memory Command (imc): The initializes memory command remove all events from memory. This command only works in manual mode.

Jump to Event Command (jec): The jump to event command moves the event counter to the event number included in the command. For example, a jump to event command of jsc:43 would jump to event 43. The event is not executed until a run command is executed. If the event that is jumped to has a trigger attached to the event, the **server** waits for the trigger to execute the event. An event with a trigger only requires the trigger and does not need to be followed by a run command, but, it should be noted that if the event trigger is already satisfied, the event will run without waiting for a trigger or the run command. This command only works in manual control mode.

Last Event Command (lec): The last event command moves the event pointer to the event that was just executed. It sets the to the value that was present at the beginning of the event before it was executed. It than executes the event again. Note: it does not ramp to the previous . If a trigger is programmed

into the event it will wait for the trigger. If a last event command is issued on an event that requires a trigger that has yet to be satisfied the event progresses to the last event without running the triggered event. This command only works in manual control mode.

Manual Mode Command (mmc): The manual mode command stops the automatic executions of the events at the end of the current event.

Model Number Command (mnc): The model number command will return the model number of the **server**.

Next Event Command (nec): This command executes the next event in the event list. If a trigger is programmed into the event, it will wait for the trigger. If a next event command is issued on an event that requires a trigger that has yet to be satisfied, the event progresses to the next event without running the triggered event. When the current event is at the last event in memory and a next event command is issued it issues the first event in memory. This command only works in manual control mode.

Pause Command (pec): The pause command can be used to pause any event. The event is paused at the state the event is in when the command is received. This can be in the middle of a ramp. Be aware that if the pause command is executed in manual mode and then an automatic mode command is issued, the pause is still in effect until the run command is executed. This command works in both automatic and manual modes.

Resets Event Command (rec): The reset command moves the event pointer to event one. The event is not executed until a run command is executed.   If the first event has a trigger attached to the event, the **server** waits for the trigger and not the run command to execute the event. This command will first respond with the Reset Event Response (rer) followed by the event one details using the same format as the Get Event Command (gec). This command only works in manual control mode.

Read Fault Command (rfpc): The read fault command returns the current fault ouput that has been programmed into the **server**.

Read System Ouput Command (rpc): The command returns the current system value . This command is used when the server is connected to a host to gather sample data from the server.

Run Command (ruc): The command is used after a pause or reset command to execute the next command. In manual mode it will run the current event. In automatic mode, it will cause the **server** to continue the execution of events.

Set Fault Command (sfpc): The set fault command set the   the **server** will go to when the reset input on the server is activated. The **server** will remain at the fault until the reset input is deactivated. The fault range is from 0 to calibration in the form of xxx.xx

Status Command (stc): The status command will return the serial number, board revision number, and firmware revision number information of the **server.**

P2 Profiler Specifications

Software Trigger Command (tec): The trigger command allows any one of 3 triggers to be triggered by a software command. The software trigger command has the highest priority of all triggers and works in manual or automatic modes.   If a trigger command is issued that does not match the expected event trigger, the command is disregarded and the **server** will continue to wait for the accepted event trigger.

Test Output Command (tfc):   When **test output command "tfc"** is received and the outputs are taken to ground, it will remain there until a **test normal command "tnc"** is received or the P2 is put into automatically mode.   This command only works in manual control mode.

Test Normal Command (tnc): When the **test normal command "tnc"** is received, the outputs are returned to normal.

Wrong Mode Response (wmr):   A command was sent to the **server** that is not allowed in the current mode for example: Sending a command that only works in manual mode while the **server** is in automatic mode.

Internal error (ine): The internal error response indicates that there was an internal hardware fault with the **server,** and it cannot continue to operate. If removing power and reapplying power does not clear this error, contact Proportion-Air for assistance.

Bad Command (bcr): The bad command response indicates that the **server** received a command that was not a valid command. This could also indicate that command data was lost due to transmission cable issues.

Bad Data Command (bdr): The bad data response indicates that the data received was outside the acceptable limits for the **server.**

CMD	Response	Description
"aevc:x,x,x,x,x"	"aevr"	**Add Event Command** adds an event into the event table. If the **server** is in auto mode the event execution begins.
"amc"	"amr"	**Auto Mode Command**. Puts the **server** in Automatic Mode which will continue to execute the events in memory in order.
"csc"	"csr:ww,yy"	The **Current State Command** returns the current state of the server where ww equals the number of events, yy is the server status in hex. See *Table 2 Status Byte.*
"gec:yy"	"ger:x,x,x,x"	**Get Event Command** returns the event information from memory for event "yy". The return data format is [eventNumber],[Time],[percentOfFullScale],[OptionValue] .

P2 Profiler Specifications

CMD	Response	Description
"imc"	"imr"	**Initializes memory command** removes all events from memory. Only works in manual mode.
"jec:X"	"evr:x,x,x,x"	**Jump to Event Command** jumps X where X is the event number, executes the event and and returns the event data. Only works in manual mode. The return data format is [eventNumber],[Time],[percentOfFullScale],[OptionValue] .
"lec"	"evr:x,x,x,x"	**Last Event Command** moves to the previous event, executes the event and returns the event data. The return data format is [eventNumber],[Time],[percentOfFullScale],[OptionValue] .
"mmc"	"mmr:yy"	**Manual Mode Command**. Stop automatic execution of events at the completion of the current event. The yy is the total number of events in the **server'**s memory.
"mnc"	"mnr: QBXT-23-33"	**Model Number Command** returns the model number. In this example it is QBXT-23-33
"nec"	"evr:x,x,x,x"	**Next event Command** steps forward one event, executes the event and and returns the event data. Only works in manual mode. The return data format is [eventNumber],[Time],[percentOfFullScale],[OptionValue] .
"pec"	"per"	**Pauses Event command** will pause the current event where it is. Works in both automatic and manual modes.
"rec"	"rer"	**Reset Event Command** resets the event pointer to the first event. It does not run the event. To run the event execute the Run command. Only works in manual mode.
"rfpc"	"rfpc:xxxx"	The **Read Fault Command** returns the current ly programmed fault where xxxx is the fault .
"rpc"	"rpr: 36.77"	The **read command** in this example reads a of 36.77
"ruc"	"rur"	**Run command.** If in automatic mode it restarts the event running after a pause. If in manual mode it will run the event current. This is used after a reset to run the first event.
"sfpc:xxxx"	"sfpr:xxxx"	The **Set Fault Command** is the that system goes to when the reset input is activated where xxxx is the fault .
"stc"	"str: 13434, 2.44, 3	**Serial Number Command** returns the serial number, software version number and board revision number.

P2 Profiler Specifications

CMD	Response	Description
		In this example it is serial number 13434 software version 2.44 board Rev 3
"tec:x"	"ter:x"	**Software trigger command** of the current event where x is the trigger to use.
"tfc"	"tfr"	**Test Fault Command** closes the fault output for testing.
"tnc"	"tnr"	**Test Fault Normal Command** returns the fault output to normal.
N/A	"avr:x,x,x,x"	When the server automatically advances to the next event a response is returned to the host indicating the advancement has occurred. The return data format is [eventNumber],[Time],[percentOfFullScale],[OptionValue].
N/A	"rtr"	The **Reset True Response** is returned when the server went into a hardware reset mode.
N/A	"rfr"	The **Reset False Response** is returned when the server came out of a hardware reset mode.
Wrong mode	"wmr"	A command is being issued in the wrong mode.
Internal Error	"ine"	Internal unrecoverable error. Contact Proportion-Air for assistance.
Unknown command	"bcr"	Bad command received. The command received does not match any of the listed commands.
Unknown data	"bdr"	Bad data received. Data is not within acceptable values.

Table 2: Command Table