



## *I/O Indexer Communication Protocol*

*Version 02 Pre-Release*

*Published February 6, 2013*

### **1.0 Overview**

The I/O COM systems comprise individual connectors for a 12 bit I/O parallel communication link, and an RS-232 serial link for programming.

The I/O Indexer Communication Protocol allows the user to select a Program Number and Program Line through the I/O Communication link. When triggered, the Sunstream program manager will branch to the program number and line specified by the I/O input.

Upon power up, the Sunstream servo controller will initiate communication with a Handheld Operator Interface or Windows GUI - if either device is connected to the controller (Refer to Interface Manuals). If no connection is present, the software will continue to execute.

To avoid any unexpected motion at power up, it is recommended that all ten (10) inputs on the parallel communication link are actively pulled low through internal logic. A pull-down resistor for each input is recommended.

A delay of 2ms is recommended as a buffer between executing read and write logic.

### **2.0 Hardware**

- Connector: Binder Series 723 with 14 pins: INPUTS 0-9, OUTPUTS 0-1, PWR, and GND. The PWR and GND pins are isolated from other power and ground planes in the controller.
- All communication pins have 2.5kV peak galvanic isolation provided by optically coupled isolators. Refer to 4.3 for further details.
- Both the power cable and the communication cable are fully shielded, and the shield for each is terminated (shorted) to the valve chassis. **The power cable shield, shield leads, and earth ground wire (EARTH) should all be terminated to earth ground at the panel. It is *not recommended* to tie shield of the communication cable to earth ground (at the panel) as doing so may create ground loops.**

### **3.0 Programming**

The Sunstream servo controller is programmable through the serial programming communication link. Unless otherwise specified by control through the HMI or GUI, the Sunstream servo will boot up to first call Program Line 00 of Program 1.

**Line 0 and Line 1 of Program 1 should be written as show in Table 3.0. The servo program index will loop between Line 0 an Line 1 until a trigger event is recorded (see Sect. 4.2).**

When a trigger event is recorded, the program index will jump to the **‘instruction group’** specified by Inputs 0-5 (specifying Program Line) and Inputs 6-7 (specifying Program Number).

Refer to the table in this section for an example of an ‘instruction group’. Lines 2-6 comprise an instruction group: the program index will branch to Line 2 with the corresponding binary inputs. After the instructions at lines 2-5 are executed, the program index will branch back to Line 0 with the Branch instruction at Line 6. Here, the program index will again loop between Line 0 and Line 1 until another trigger event is recorded.

**Table 3.0**

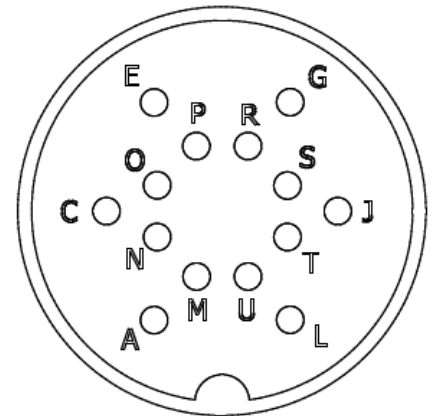
Line	Instruction	Value	Program #	Line	Input	Output
0	Branch, Specified by Input					
1	Branch		Program 1	00		
2	Move to Position	28000				
3	Set Maximum Acceleration	1.5 g's				
4	Set Maximum Velocity	10.0 in/s				
5	Move with Soft Touch	32000				
6	Branch		Program 1	00		

To learn more about programming the Sunstream servo controller, refer to the GUI Manual.

## 4.0 Pin Definitions

**Table 4.0**

Logic Des.	Function	Pin Des.	Cable Wire
Input 0	Bit 0 of prg line	O	Green/Blk
Input 1	Bit 1 of prg line	P	Yellow/Blk
Input 2	Bit 2 of prg line	R	Yellow
Input 3	Bit 3 of prg line	S	Orange
Input 4	Bit 4 of prg line	N	Green
Input 5	Bit 5 of prg line	M	Brown
Input 6	Bit 0 of prg #	U	Brown/Blk
Input 7	Bit 1 of prg #	T	Orange/Blk
Input 8	Trigger	A	Red
Input 9	Enable	L	Red/Blk
Output 0	Status	E	Blue
Output 1	Diagnostic	G	White
N/A	PWR	C	Blue/Blk
N/A	GND	J	White/Blk



## 4.1 Outputs

**Output bits are set and cleared by the Sunstream controller.**

**Output Bit 0:** The Status bit represents the operational status of the servo. When this bit is high (logic 1), the servo is *Busy* executing an instruction group. Once the servo has completed all instructions within an instruction group, the Status bit will be pulled low (logic 0) to indicate that it's *Ready* to accept a new input.

**Output Bit 1:** This is a diagnostic bit. If the servo is operating properly, this bit will be pulled to a high logic state. When pulled low, there has been a fault in servo operation.

## 4.2 Inputs

**Input bits are set and cleared by a Host, and read by the Sunstream controller.**

**Input Bits 0-5:** These bits concatenate to form an input value related to Program Line.

**Table 4.1**

Program Line	Bit 5 (MSB)	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0 (LSB)
0	0	0	0	0	0	0
....	....	....	....	....	....	....
32	1	0	0	0	0	0
....	....	....	....	....	....	....
63	1	1	1	1	1	1

**Input Bits 6-7:** Program Number Selection Logic Table –

**Table 4.2**

Bit 7	Bit 6	Instruction
0	0	None Selected
0	1	Program 1
1	0	Program 2
1	1	Program 3

**Input Bit 8:** The Trigger Bit signals the Sunstream controller to scan Input bits 0-5 in order to calculate a new Program Line, and Input Bits 6-7 to calculate a new Program Number. If the servo status is *Busy* while the Trigger is set to logic 1, the Sunstream controller will not execute a new instruction group. If the servo status is *Ready* when the Trigger bit is pushed high, a new instruction group will be executed.

**Input Bit 9:** Enable Servo. When pulled low, the Enable bit restricts the Servo controller from initiating an instruction, and places the spool in flow mode. Regardless of the state of the Status bit, the servo

cannot be triggered to move to a new point if the Enable bit is pulled low. This bit should be set to logic 0 upon applying +24V power to the servo controller. Pull this bit high (logic 1) to enable normal operation.

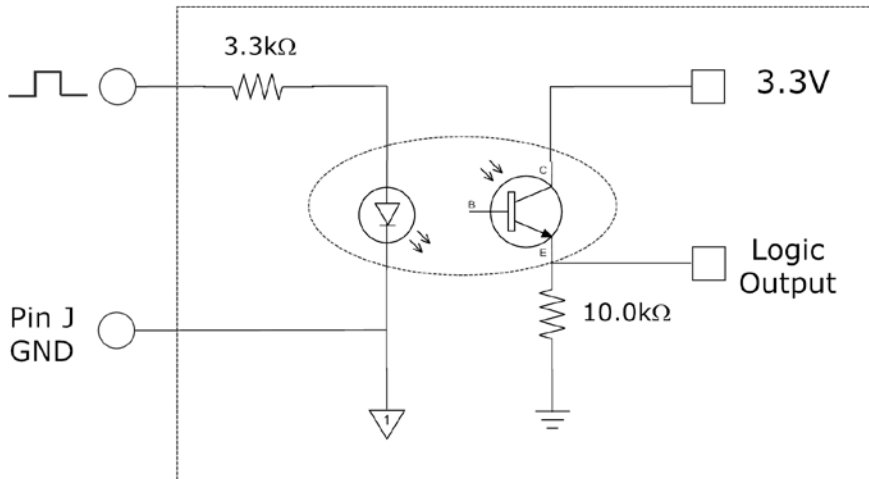
By pulling the Enable Bit low during operation, the Sunstream servo will transition back to the “stand-by” flow mode. In this mode, the spool will be actively centered, but the position loop will be disengaged. To transition back into an operational mode, pull the Enable Bit high, and transmit a new instruction group.

### 4.3 Electrical

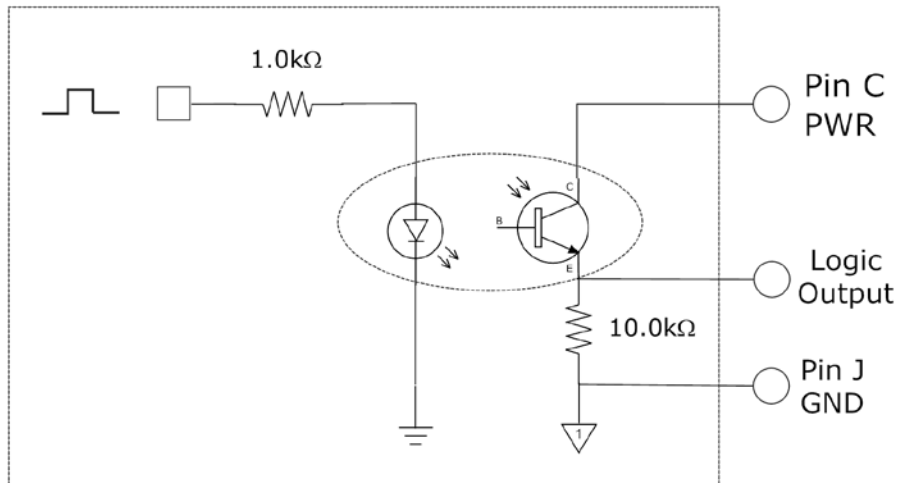
**Pin C:** GND, Isolated.

**Pin J:** PWR, Isolated.

**Input Pins:** All input pins are tolerant of 5-30V logic inputs.



**Output Pins:** The output voltage level is equivalent to Pin C PWR.



## 5.0 Operational Protocol

This section describes how a user controller – such as a PLC – herein referred to as the “Host”, may be employed to command the Sunstream pneumatic servo.

**All Input bits are set and cleared by the Host, and read by the Sunstream controller.**

**All Output bits are set and cleared by the Sunstream controller, and read by the Host.**

Before applying +24V power to the Sunstream servo controller, ensure that all Input bits 0 - 9 are pulled low by the Host. Failure to ensure that these bits are cleared to logic 0 may result in unpredictable servo behavior.

Once power is applied, Output Bit 0, the Status Bit, will be pulled high by the Sunstream controller's internal hardware logic. After completing memory initialization, the Sunstream controller polls the Enable Bit awaiting detection of a high state. The Host is required to pull the Enable Bit (Input Bit 9) high in order to activate the servo loop. When logic 1 is detected, the controller will progress through an initialization process until the servo loop is modulating in a “stand-by” flow mode. The Sunstream controller subsequently pulls the Status Bit (Output Bit 0) to logic 0, indicating that it is *Ready* to execute the first Instruction.

**Unless otherwise specified by control through the HMI or GUI, the Sunstream servo will boot up to first call Program Line 00 of Program 1. Line 0 and Line 1 of Program 1 should be written as show in Table 3.0. The servo program index will loop between Line 0 and Line 1 while scanning the Trigger Bit (Input 8) for a Logic 1.**

When a trigger event is recorded the program index will branch to the location in the program code specified by Input Bits 0-7. The Sunstream controller will acknowledge that the trigger bit has been recognized but pulling the Status Bit to logic 1, indicating that the

After the Status Bit has been pulled to logic 1, the Host should clear the Trigger Bit by pulling it to logic 0.

Each time the program index branches back to the ‘Branch, Specified by Input’ Instruction line, the Sunstream controller will pull the Status Bit to logic 0.

### 5.1 Summary of I/O Handshaking

When the servo is Ready to receive a new ‘instruction group’, the Status Bit (Output 0) will be pulled to a low logic state.

To command the servo to execute a new instruction group, first write the desired Program Number to input bits 6-7 and the desired Program Line to input bits 0-5. Next pull the Trigger Bit (Input 8) to a logic high state.

When the servo has properly read the new program line and number, the servo will acknowledge the Trigger Bit by pulling the Status Bit to a high logic state.

When a rising edge on the Status Bit has been detected, the host should pull the Trigger Bit low.

A delay of 2ms is recommended as a buffer between executing read and write logic.

## 5.2 Operational Example

A simple example will be provided for a joining application, such as a servo resistance or ultrasonic welding. In this example, two parts of varying heights will be pressed together at high force while a welding operation occurs. In operation, the servo must first rapidly move to a “pre-contact” position before decelerating to softly engage the workpieces. After contact, the servo autonomously synchronizes from a servo position loop to a servo force loop, and ramps to 150 lbf. The servo holds the force at 150 lbf. while a measurement of height is executed. After the measurement is completed, the servo ramps the force from 150 lbf. to 250 lbf.

The instructions, as programmed through the Windows GUI interface, are illustrated in Table 5.0.

A detailed, step-by-step analysis of this operation follows. Refer to the GUI Manual for further details on the Instructions.

**Table 5.0**

Line	Instruction	Value	Program #	Line	Input	Output
0	Branch, Specified by Input					
1	Branch		Program 1	0		
2	Set Maximum Acceleration	1.0 g's				
3	Set Maximum Velocity	4 in/s				
4	Move to Position	pos. 2000				
5	Branch		Program 1	0		
6	Set Maximum Acceleration	6 g's				
7	Set Maximum Velocity	60 in/s				
8	Move to Position	pos. 25000				
9	Branch		Program 1	0		
10	Set Maximum Acceleration	6 g's				
11	Set Maximum Velocity	60 in/s				
12	Move to Position	Pos. 28500				
13	Branch		Program 1	0		
14	Set Maximum Acceleration	0.5 g's				
15	Set Maximum Velocity	2 in/s				
16	Move with Soft Touch	pos. 32000				
17	Branch		Program 1	0		
18	Ramp to Force	+250 lbf.				
19	Branch		Program 1	0		
20	Ramp to Force	+150 lbf.				
21	Branch		Program 1	0		
22	Set Maximum Acceleration	6 g's				
23	Set Maximum Velocity	60 in/s				
24	Branch		Program 1	4		

**STEP 1: Startup**

- a) Pull all Input bits to logic 0.
- b) Apply +24V to Sunstream servo controller
- c) Sunstream controller will pull up the Status Bit (Output bit 0) to logic 1.
- d) Pull Enable Bit to logic 1.
- e) Sunstream controller reads Enable Bit, completes software initialization and enables servo loop.
- f) Sunstream controller pulls the Status bit (Output bit 0) low to indicate *Ready*.
- g) A delay of 2ms is recommended as a buffer between executing all read and write logic.

**STEP 2: Move servo to home position at low speed.**

- a) Read the Status Bit to verify Output 0 is pulled to low logic level.
- b) Write the desired program line and number to Inputs 0-7 as demonstrated in Table 5.2
- c) Pull the Trigger Bit to logic high.
- d) Read the Status Bit to verify that the Trigger was acknowledged and that Output 0 is pulled to logic high.
- e) Pull the Trigger Bit to logic low – after a delay of 2ms or greater.
- f) The Sunstream program index will branch to Line 2 of Program 1. The Sunstream controller will set the Maximum Acceleration at Line 2, set the Maximum Velocity at Line 3, move to position 2000 at Line 4, and complete the instruction group with a branch to Program 1, Line 0
- g) Once the program index has returned to Line 0, the Status Bit will pull to logic 0.

**Table 5.2**

Program #	Program Line	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	2	0	1	0	0	0	0	1	0

**STEP 3: Move servo to pre-contact position at high speed.**

- a) Repeat Steps a-e as shown in **STEP 2**, but referring to Table 5.3
- b) The Sunstream program index will branch to Line 10 of Program 1, and return to Line 0 upon completion of the instruction group.

**Table 5.3**

Program #	Program Line	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	10	0	1	0	0	1	0	1	0

**STEP 4: Make low speed, soft contact with workpiece with the “Move with Soft Touch” instruction.** Once a pre-set Force Limit has been reached (see GUI Manual), the servo will autonomously synchronize to a servo force loop, and hold the force specified by ‘Force Limit’.

- a) Repeat Steps a-e as shown in **STEP 2**, but referring to Table 5.4
- b) The Sunstream program index will branch to Line 14 of Program 1, and return to Line 0 upon completion of the instruction group.

**Table 5.4**

Program #	Program Line	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	14	0	1	0	0	1	1	1	0

**STEP 5: Ramp force to 150 lbf. at a pre-determined rate.**

- a) Repeat Steps a-e as shown in **STEP 2**, but referring to Table 5.5
- b) The Sunstream program index will branch to Line 20 of Program 1, and return to Line 0 upon completion of the instruction group.

**Table 5.5**

Program #	Program Line	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	20	0	1	0	1	0	1	0	0

**STEP 6: Ramp force to 250 lbf. at a pre-determined rate.**

- a) Repeat Steps a-e as shown in **STEP 2**, but referring to Table 5.6
- b) The Sunstream program index will branch to Line 18 of Program 1, and return to Line 0 upon completion of the instruction group.

**Table 5.6**

Program #	Program Line	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	18	0	1	0	1	0	0	1	0

**STEP 7: Move servo to home position at high speed.**

- a) Repeat Steps a-e as shown in **STEP 2**, but referring to Table 5.7
- b) The Sunstream program index will branch to Line 22 of Program 1, and return to Line 0 upon completion of the instruction group.

**Table 5.7**

Program #	Program Line	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	22	0	1	0	1	0	1	1	0