



HiTechnic FIRST Servo Controller Specification

For LEGO MINDSTORMS NXT

Dataport Systems, Inc – HiTechnic Division

HiTechnic FIRST Servo Controller Specification

Contents

1. Summary	3
2. Description	3
2.1. Function.....	3
3. Technical Description	4
3.1. Operation.....	4
4. Appendix 1	5
4.1. Electrical Connections.....	5

HiTechnic FIRST Servo Controller Specification

HiTechnic First Motor Controller for LEGO NXT

1. Summary

This specification defines a servo controller for use with the LEGO NXT Mindstorms system. The servo controller will provide six channels each of which can support an FTC servo. The servos are driven by standard PWM signals.



Power will be supplied from an external 9v – 15v battery via a screw terminal block.

The design and manufacture will be in compliance with worldwide standards for safety and chemical composition per the LEGO company's requirements.

2. Description

2.1. Function

The servo controller will interface with the NXT brick using the 6 wire sensor interface. All of the 6 wires will be used, as this device uses the legacy analog signal of 5v (not 9v) supplied through a 10.0k resistor to automatically generate daisy chain I2C addresses.

The servo controller will communicate with the NXT brick using the LEGO company's defined use of the I2C interface.

The servo controller will accept six single byte position values for each of the six servo channels.

The servo controller will accept a timing parameter to control the rate of servo movement.

The servo controller will accept an enable parameter to control output of the servo PWM signals.

The servo controller will return a status indicating busy.

The first servo controller in the daisy chain will use an I2C address of 02/03. Subsequent controllers will obtain addresses of 04/05, 06/07 and 08/09. Only four controllers may be daisy chained.

The servo controller will consume 5mA or less from the NXT "brick" 4.3v unregulated supply.

HiTechnic FIRST Servo Controller Specification

The servo controller will supply up to 4 amps total to all six servo channels when running from 9v – 15v.

The servo controller uses separate grounds for the NXT I2C interface and the battery supply to reduce ground loop problems. These grounds are tied together with (33 ohm) resistors in each module. Care should be taken when wiring the battery supply between modules to ensure that the terminals are tight and that an adequate gauge of wire is used for the likely servo current consumption.

3. Technical Description

3.1. Operation

The servo controller firmware supports the following memory model.

Address	Type	Contents
00 – 07H	chars	Sensor version number
08 – 0FH	chars	Manufacturer
10 – 17H	chars	Sensor type
18 – 3DH	bytes	Not used
3E, 3FH	chars	Reserved
40H	byte	Status
41H	byte	Step time
42H	byte	Servo 1 position
43H	byte	Servo 2 position
44H	byte	Servo 3 position
45H	byte	Servo 4 position
46H	byte	Servo 5 position
47H	byte	Servo 6 position
48H	byte	PWM enable

The *Sensor version number* field will report a revision number in the format “Vn.m” where *n* is the major version number and *m* is the revision level. Revision numbers will typically reflect the firmware level. The version number will be used to indicate the hardware level.

The *Manufacturer* field will contain “HiTechnic”.

The *Sensor type* field will contain “ServoCon”.

The *Status* field returns 1 if there are still servos changing position, else it will return 0.

The *Step time* field sets the step time for the servo channel which has the furthest to move. Other servo channels which are not at their designated positions yet will run at a slower rate to ensure they reach their destination positions at the same time. This value

HiTechnic FIRST Servo Controller Specification

may range from 1 – 15. If the field is set to 0, no timing is applied, ie; the servos move at their maximum rate.

The *Servo 1/2/3/4/5/6 position* fields will accept a value between 0 – 255 to set the PWM output from 0.75 – 2.25mS. Note that some servos may hit their internal mechanical limits at each end of this range causing them to consume excessive current.

The *PWM enable* field is used to control the PWM output. If it is set to zero (00H), it will enable the PWM outputs, causing the servos to be active, and start a ten second timeout running. If it is set to 255 (0FFH), it will disable the PWM outputs, causing the servos to be inactive. If no I2C communication is received during the ten second timeout period, the *PWM enable* field will automatically revert to 255 (0FFH), disabling the servos. If the *PWM enable* field is set to 170 (0AAH), it will enable the PWM outputs, causing the servos to be active, and disable the ten second timeout. The *PWM enable* field is set to 255 (0FFH) during power up and after the timeout has elapsed.

4. Appendix 1

4.1. Electrical Connections

Servo + Motor Controller Daisy Chain

