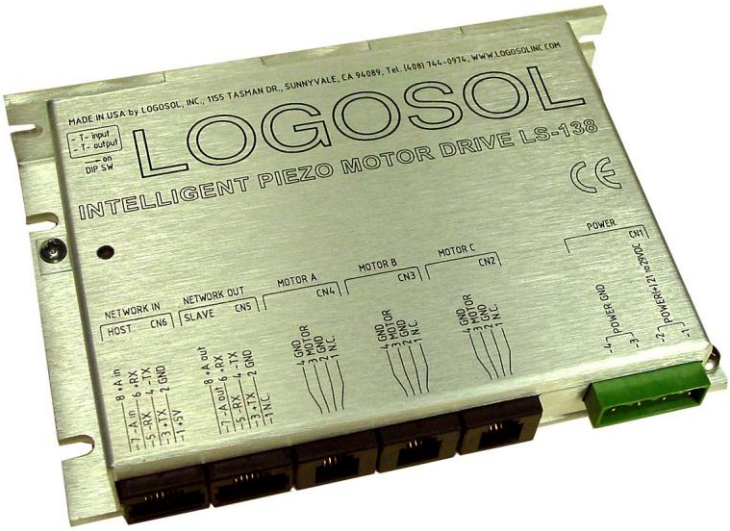
Logosol Intelligent Piezo Motor Drive LS-138

Doc. # 712137002 / Rev. 1.0, 04/04/2002

Features

- **Motors supported:**
  - **Standard Picomotors™**
  - **Tiny Picomotors™**
- **Three channels per module**
- **Position and velocity modes**
- **Trapezoidal profile generator**
- **Velocity range 1 Hz to 2 KHz**
- **32-bit position counter**
- **Motor output short protection**
- **Missing motor detection**
- **Single 24VDC power supply**
- **Member of Logosol's distributed control network (LDCN)**
- **Communication speed 19.2 to 115.Kbps**
- **Command rate up to 1000/sec**
- **Small footprint (5" x 3.3" x 0.85")**



Description

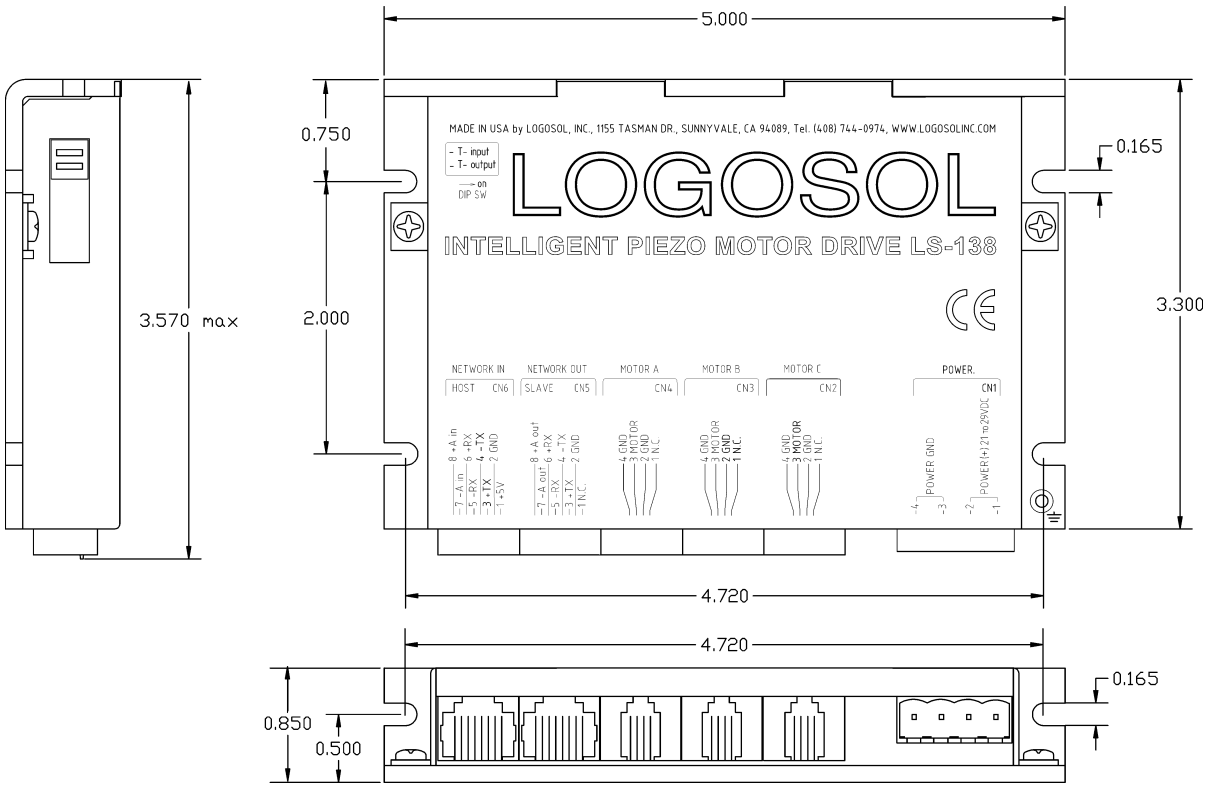
LS-138 is an intelligent Piezo Motor Drive designed for applications requiring a compact, high-resolution positioner. The drive controls Standard or Tiny Picomotors™ manufactured by NewFocus®, Inc.

The drive is a member of Logosol's distributed control network (LDCN). Up to 31 LDCN devices can be controlled over a multi-drop full duplex RS-485 network. Standard RJ-45 connectors and commercially available cables are used to connect modules into a network.

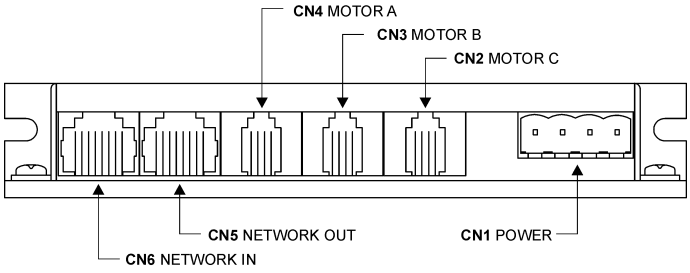
TECHNICAL SPECIFICATIONS rated at 25°C ambient, POWER (+) = 24VDC

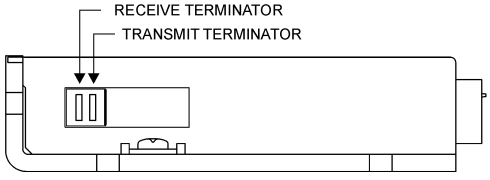
POWER SUPPLY VOLTAGE	21 to 29 VDC
POWER SUPPLY CURRENT	Max average 0.8 Amp DC Peak 1.25A / 200us per CW step Peak 2.5A / 100us per CCW step
OUTPUT FREQUENCY RANGE	1 Hz to 2KHz
MAXIMUM OUTPUT FREQUENCY W/O FORCED COOLING	1.5 KHz @ 100% duty cycle 2 KHz @ 50 % duty cycle (ON time max 120sec)
NUMBER OF CHANNEL PER MODULE	Three
NUMBER OF ACTIVE CHANNELS AT ONCE	One
COMMUNICATION PROTOCOL	Logosol network protocol (LDCN)
MAXIMUM NUMBER OF MODULES PER NETWORK	31
COMMUNICATION INTERFACE	RS-485
SERIAL BAUD RATE	19.2 to 115.2 Kbps
LED (TWO INTENSITY LEVELS)	Power 'OK' – low intensity Ready – high intensity
PROTECTION Output short (Motor Output to GND) Overtemperature	 Shutdown if motor output is shorted Shutdown at 70°C
FIRE SAFETY Internal fuse	 3A Quick blow
THERMAL REQUIREMENTS Storage temperature range Operating temperature range	 –30 to +85°C 10 to 45°C
MECHANICAL Size Weight	 L=5", H=0.85", D=3.3" 0.55lb. (250gr.)
MATING CONNECTORS Power supply Picomotor™ Communication	 4 pin Phoenix MSTB 2.5/4-ST-5.08 4 pin RJ-22 8 pin RJ-45

DIMENSIONAL DRAWING



CONNECTORS AND CONTROLS LAYOUT

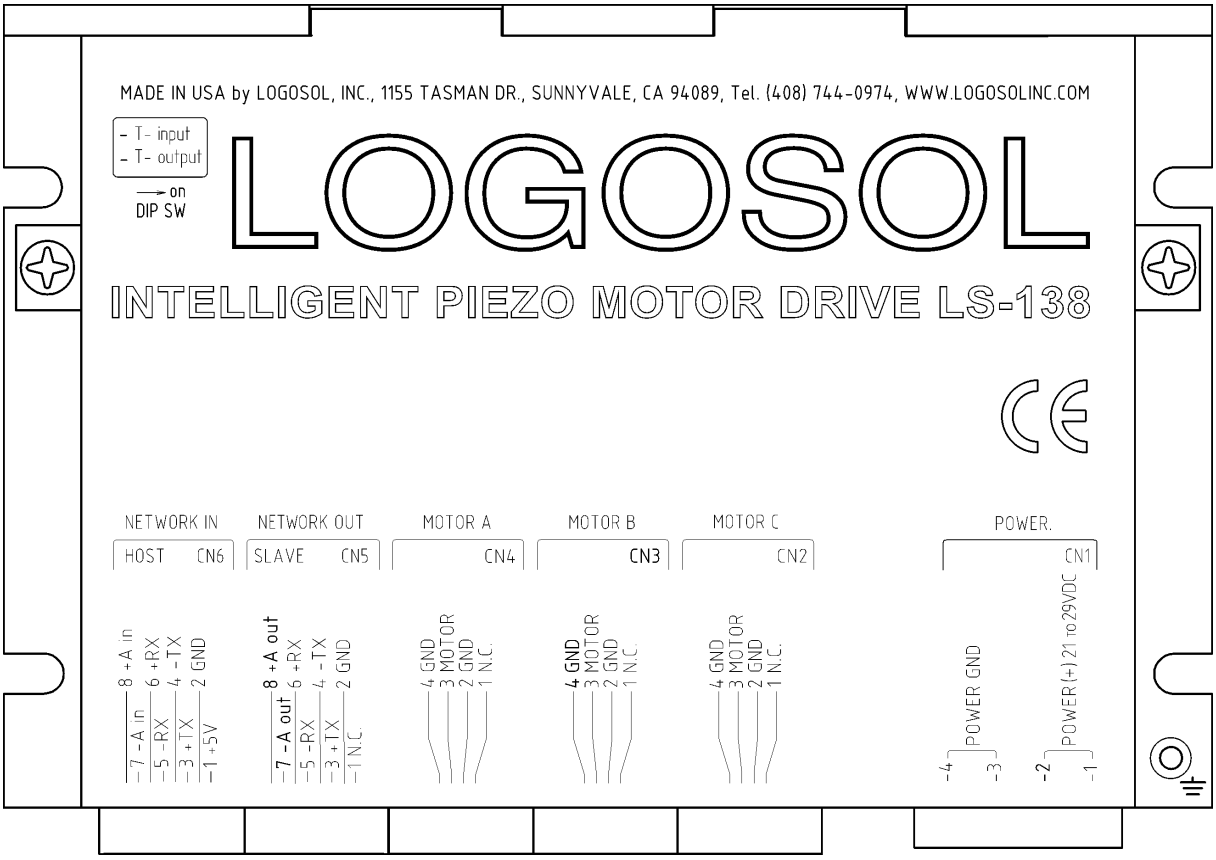




ORDERING GUIDE

PART NUMBER	MODEL	DESCRIPTION
912137004	LS-138	Three channels Intelligent Piezo motor Drive
230601028	LS-137/138-CN	Mating connector kit for LS-137/138

THREE CHANNELS INTELLIGENT PIEZO MOTOR DRIVE LS-138



DIP SWITCH

SW	SIGNAL	DESCRIPTION
1	T-input	Receive line terminator
2	T-output	Transmit line terminator

CONNECTORS PINOUT

CN1 – POWER CONNECTOR

PIN	SIGNAL	DESCRIPTION
1	POWER (+)	+21 to +29VDC power supply, positive terminal
2	POWER (+)	+21 to +29VDC power supply, positive terminal
3	POWER GND*	Power supply ground
4	POWER GND*	Power supply ground

CN2, CN3 and CN4 – MOTOR C to MOTOR A

PIN	SIGNAL	DESCRIPTION
1	N.C.	Not connected
2	GND*	Power ground
3	MOTOR	Motor output
4	GND*	Motor ground

CN5 – NETWORK OUT (SLAVE)

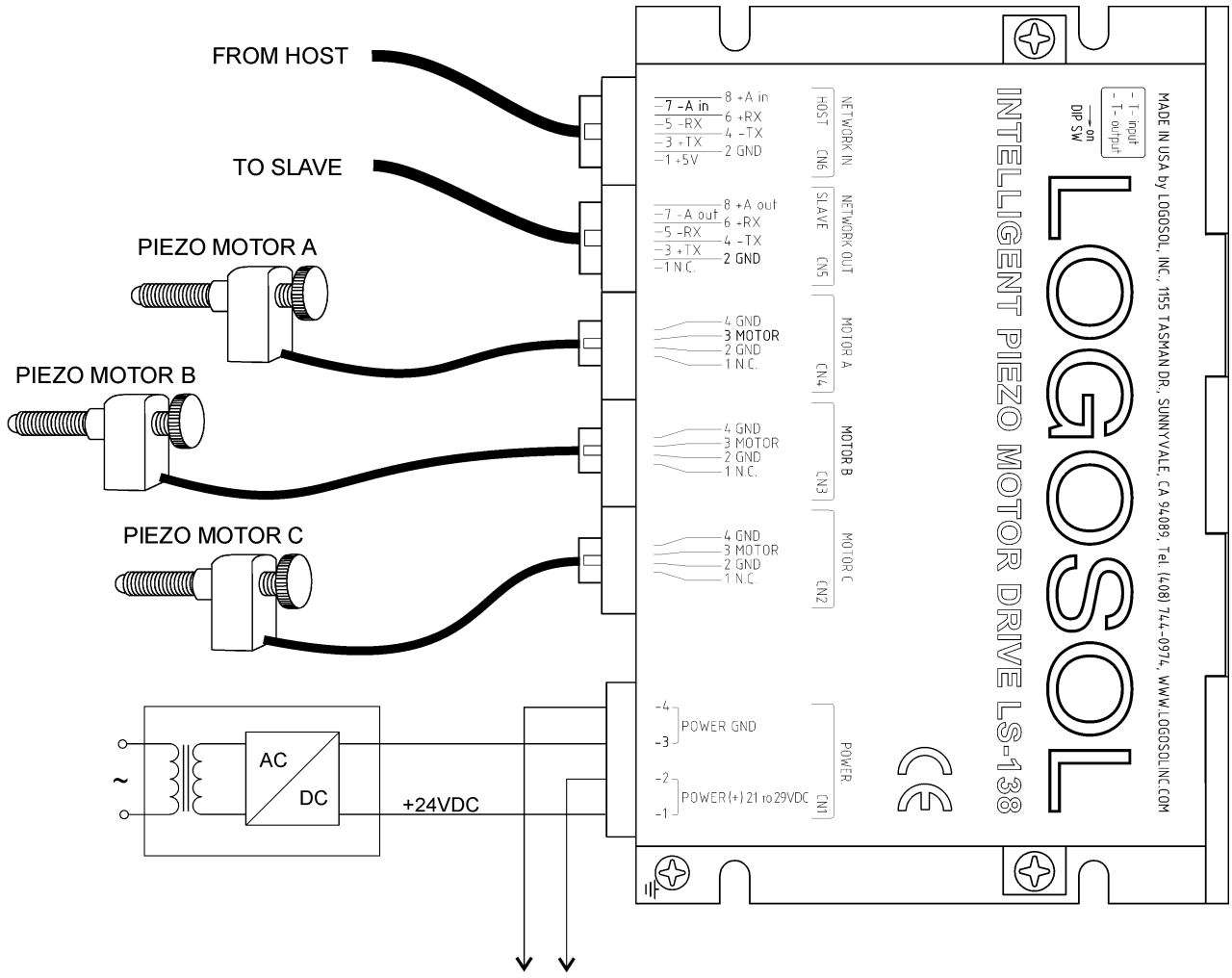
PIN	SIGNAL	DESCRIPTION
1	N.C.	Not connected
2	GND*	Interface ground
3	+TX	(+) Transmit data
4	–TX	(–) Transmit data
5	–RX	(–) Receive data
6	+RX	(+) Receive data
7	–A out	(–) Address output
8	+A out	(+) Address output

CN6 – NETWORK IN (HOST)

PIN	SIGNAL	DESCRIPTION
1	+5V	RS-232 adapter power supply (200 mA Max)
2	GND*	Interface ground
3	+TX	(+) Transmit data
4	–TX	(–) Transmit data
5	–RX	(–) Receive data
6	+RX	(+) Receive data
7	–A in	(–) Address input
8	+A in	(+) Address input

* *POWER GND, MOTOR GND and GND are electrically connected. Drive's case is isolated from the drive's circuitry and may be grounded externally.*

SAMPLE APPLICATION



LOGOSOL LS-138 QUICK START GUIDE

Hardware Setup

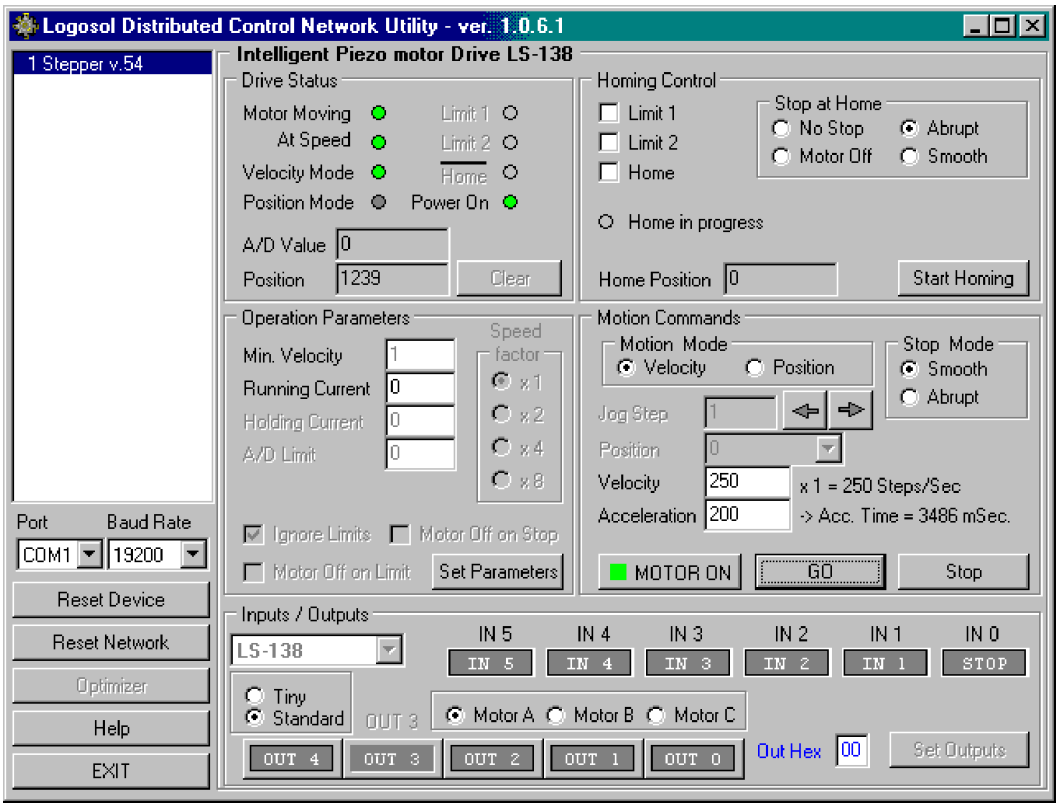
1. Connect power supply 24 VDC to LS-138.
2. Connect your Piezo motor(s).
3. Connect RS-232 adapter and RJ-45 network cable between LS-138 and your host computer.
4. Set DIP switches T-input and T-output to ON. If more than one device is connected, only the last one must have T-input and T-output set to ON. All of the rest must have T-input and T-output set to OFF.

Software Installation

1. Installation and using Logosol Distributed Control Network Utility

A. Installation

1. Insert the Logosol Distributed Control Network Utility installation disk into the floppy drive.
2. Select Run from the Windows 95/98/NT Start menu.
3. Type a:\dcnsetup and then click OK (a: represents the drive letter).
4. The installation wizard will guide you through the setup process.



B. Initial Connection to the Host

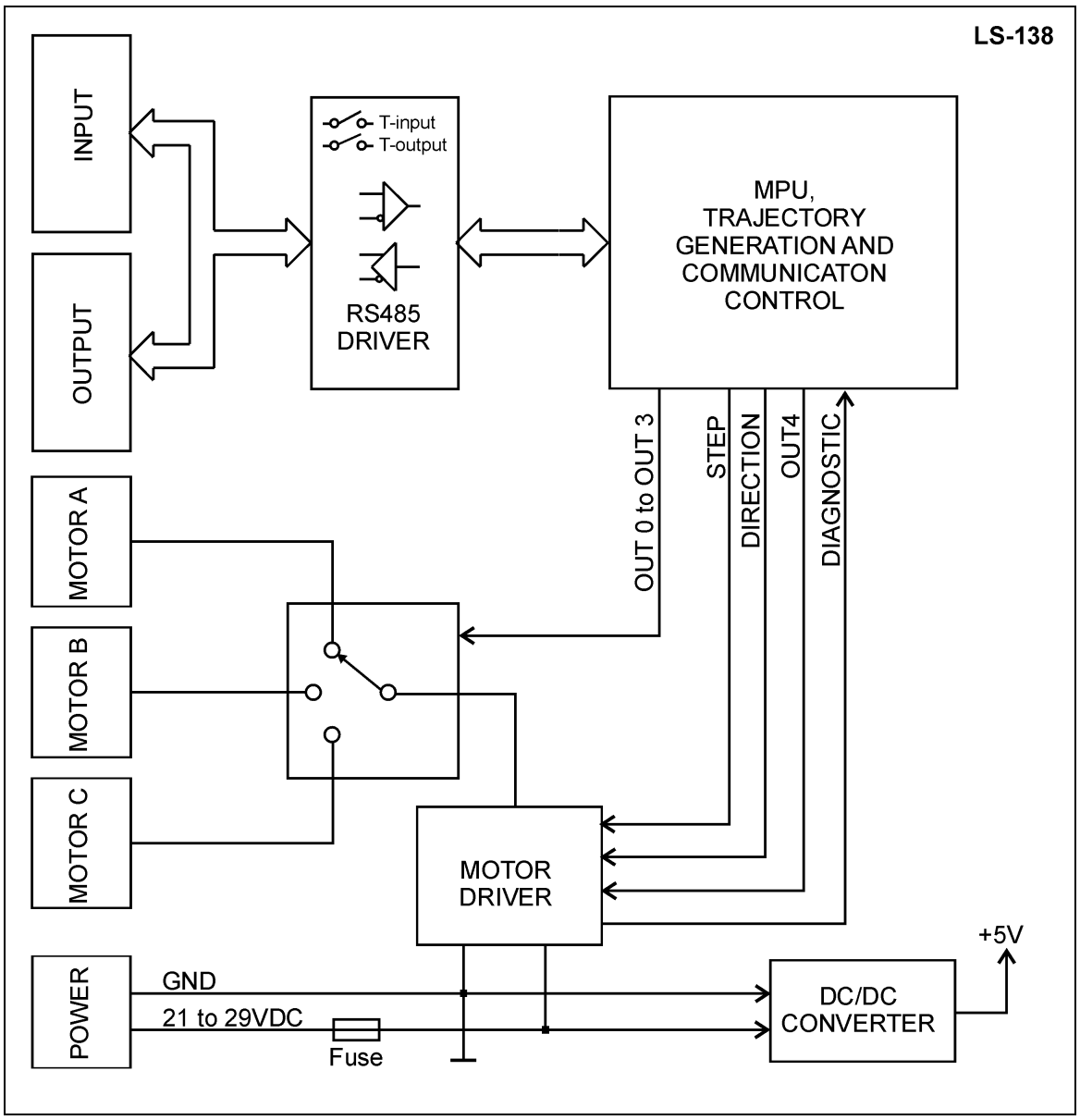
1. Turn on the power supply.
2. Run the Logosol Distributed Control Network Utility.
3. Choose the proper COM port and select the baud rate (default value is 19200).
4. Click “MOTOR ON” button.
5. Click “GO” button. The motor should rotate slowly in positive direction. Click “Stop” to interrupt the motion. More information about using LDCN utility is available in LDCN Help.

LS-138 ARCHITECTURE

Overview

The LS-138 Intelligent Piezo Motor Drive is fully integrated device including step and direction motion controller and three channel motor selector. It supports RS232 (with LS-801 adapter) or multi-drop RS485 serial interface using the same communications protocol as the other members of Logosol Distributed Motion Control Network. LS-138 supports velocity and trapezoidal position mode, using a 32-bit position counter.

A network of up to 31 controllers can be connected directly to a single standard serial port (RS-232 adapter may be required).



Functional Diagram

Identification

After Power-up or *Hard reset command* and before first *Stop motor command* with bit 0 set, input bits IN0 to IN5 from I/O byte are used to identify the device type.
For LS-138 the identification number is **0x01**. The identification sequence should occur after initializing the network and reading the device type and version:

1. Read the states of input bits IN0 to IN5.
2. Set OUT4 to 1.
3. Read the states of input bits IN0 to IN5.

If the input states are inverted (see the table below), the device does have an identification number and this number is the value in step 1.

OUT4	IN5	IN4	IN3	IN2	IN1	IN0
0	0	0	0	0	0	1
1	1	1	1	1	1	0

Note: The identification number is valid until first set to clear transition of OUT4 or before the first *Stop motor command* with bit 0 set.

Motor selector

LS-138 has five internal control signals OUT0 to OUT4 used to control the Motor connector and type selector. OUT0 to OUT2 selects the motor connector. OUT3 is reserved and should be cleared. OUT4 selects the **drive signal** for **Standard Picomotor™** or **Tiny Picomotor™**. The current motor type and connector selection can be changed after sending *Stop motor command* with bit 0 cleared. If the selected motor connector is not supported, bit 3 in Status byte will be cleared. Here is the sequence to select another channel and / or to change the motor type:

1. Stop motor command with bit 0 cleared. (stop motion and disable Motor driver)
2. I/O control command with desired motor type and connector. (see the table below)

OUT4	OUT3	OUT2	OUT1	OUT0	Status byte bit 3	MOTOR SELECTED
0	0	0	0	0	1	MOTOR A Standard
0	0	0	0	1	1	MOTOR B Standard
0	0	0	1	0	1	MOTOR C Standard
1	0	0	0	0	1	MOTOR A Tiny
1	0	0	0	1	1	MOTOR B Tiny
1	0	0	1	0	1	MOTOR C Tiny
X	X	X	1	1	0	N.A.
X	X	1	X	X	0	N.A.
X	1	X	X	X	0	N.A.

3. Stop motor command with bit 0 set will enable the selected motor if **Status byte bit 3 = 1**.

CAUTION:

A Tiny Picomotor ™ can be damaged if it is driven with the wrong type of drive signal for an extended period of time, so it is important to ensure that the device is configured to generate the correct drive signals.

Diagnostic

LS-138 is protected against motor output short and overtemperature. In addition a missing motor can be detected while moving in negative direction. Status byte bit 2 (Motor On) and IN0 (STOP), IN1, and IN2 form the Input byte are used for diagnostic.

Motor On	IN2	IN1	IN0 (STOP)	Diagnostic
X	0	0	0	OK
0	0	0	1	NO MOTOR (single step negative direction with OUT4= 0)
0	1	0	1	MOTOR SHORT
X	X	1	0	OVERTEMPERATURE
0	X	1	1	OVERTEMPERATURE latched

Serial Command Interface

The serial communication with the LDCN Nodes adheres to a full-duplex (4 wire) 8 bit asynchronous protocol with one start bit, followed by 8 data bits (lsb first), followed by a single stop bit. The communication protocol also supports a full-duplex multi-drop RS-485 interface, which allows multiple LDCN Nodes to be controlled over a single RS-485 port. In this case, the host sends commands over its RS-485 transmit line and receives all status data back over the shared RS-485 receive line. The command protocol is a strict master/slave protocol in which the host master sends a command packet over the command line to a specific LDCN Node. The Node sends back a status packet. Typically, the host does not send another command until a status packet has been received to insure that it does not overwrite any previous command data still in use.

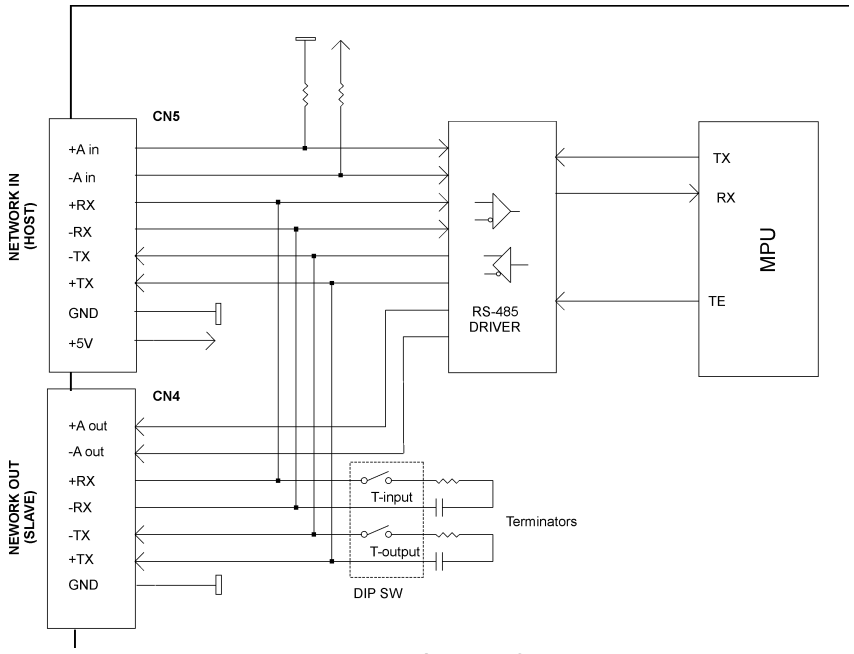
Each command packet consists of following:

Header byte (0xAA)
Address byte - individual or group (0x00 - 0xFF)
Command byte
0 - 15 data bytes
Checksum byte

The command byte is divided into upper and lower nibbles: the lower nibble is the command value; the upper nibble is the number of additional data bytes, which will follow the command byte. The checksum byte is 8-bit sum of the address byte, the command byte and the data bytes. The number of data bytes depends on the particular command chosen. After a command is issued, the corresponding controller will send back a status packet consisting of:

Status byte
0-16 optional bytes of status data
Checksum byte

The Status Byte contains basic status data about the LDCN Node, including a checksum error flag for the command just received. The number and the meaning of Optional Status Data Bytes are programmable by the user and may include any, none or all of data available from the module. The checksum byte is the 8-bit sum of the status byte and the additional optional status data bytes. All 16-bit and 32-bit data is sending with the least significant byte first.



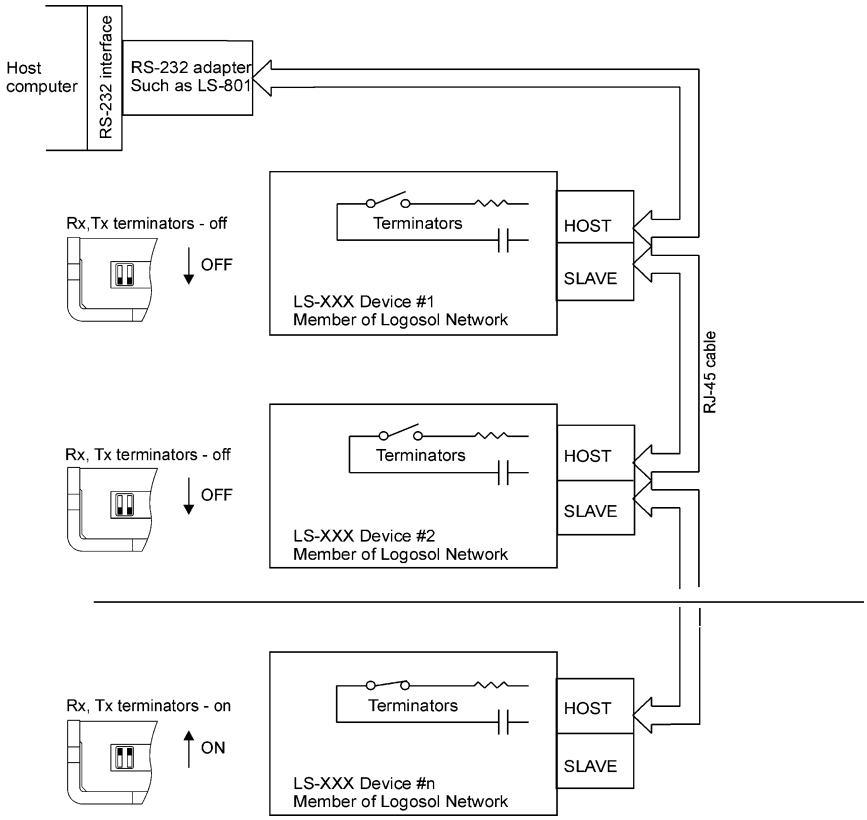
LDCN Node Interface

Terminators

DIP switches T-input and T-output, are used to turn ON/OFF receive and transmit lines terminators. In a network of Logosol devices only the last one (at the remote end) is supposed to have its line terminators ON. All other devices located between this device and the host must have their terminators set to OFF.

Addressing

Rather than using hard-wired or switch-selected address of each LDCN Node, the host dynamically sets the address of each node with the aid of the daisy-chained “A in” and “A out” lines. This allows additional LDCN Nodes to be added to an RS-485 network with no hardware changes. On Power-up, “A in” of the first LDCN Node is pulled low, its communication is enabled and the default address is 0x00. When the *Set Address* command is issued to give this node a new unique address, it will lower its “A out” line. Connecting “A out” to the “A in” of the next node on the network will enable its communication at default address of 0x00. Repeating this procedure allows a variable number of nodes present to be given unique addresses.



Multiple Controller Configuration

Group Addresses

In addition to the individual address, each node has a secondary group address. Several LDCN nodes may share a common group address. This address is useful for sending commands, which must be performed simultaneously by a number of nodes (e.g. *Set Baud Rate*, etc.). When a node receives a command sent to its group address, it will execute the command but not send back a status packet. This prevents data collisions on the shared response line. When programming group addresses, however, the host can specify that one member of the group is the “group leader”. The group leader will send back a status packet just like it would for a command sent to its individual address. The group address is programmed at the same time as unique individual address using the *Set Address* command.

Changing Communications Rates

The default baud rate after power-up is 19.2 Kbps. Baud rates up to 115.2 Kbps may be used. After communication has been established with all the controllers on a single network, the baud rate may be changed to a higher value with the *Set Baud Rate* command.

THEORY OF OPERATION

Velocity Profile Mode

Velocity profile mode is used to smoothly accelerate from one velocity to another. Commanded velocities are specified as integer values **S** between 1 and 250. Minimum and maximum velocities for the different speed modes appear in the table below:

Speed factor	*Step Rate Multiplier (K) (Minimum Velocity) in steps/sec.*	*Max. Step Rate in steps/sec. (Velocity = 250)*
1x	1	250
2x	2	500
4x	4	1,000
8x	8	2,000

Step rate multipliers and Maximum Velocities for Different Speed factors.

The actual velocity **V** in steps per second can be obtained using the formula:

V = S * K,
where

S is commanded velocity (range 1 - 250) and
K is the step rate multiplier for current speed factor (see the table)

The acceleration or deceleration is achieved by incrementing (or decrementing) the current integer velocity value by one until the goal velocity is reached. The actual time for acceleration from one velocity to another can be obtained using the formula:

$\mathbf{T_{acc} = | (64 – 0.25 * Acc) * (S1 – S0) |}$ in **ms**,
where

Acc is the acceleration value (range 1 - 255),
S0 is current velocity (range 1 - 250),
S1 is target velocity (range 1 - 250), and
$\mathbf{T_{acc}}$ is the time to accelerate from velocity **S0** to **S1** with acceleration **Acc**

Examples:

1. Accelerating to velocity 125 with minimum profile velocity = 25 and acceleration 100.
   Acc = 100, S0 = 25, S1 = 125.
   | (64 – 0.25 * 100) * (125 – 25) | = | 39 * 100 | = | 3900 | = 3900 ms (3.9 s)

2. Decelerating from velocity 125 to stop with minimum profile velocity = 25 and acceleration 100.
   Acc = 100, S0 = 125, S1 = 25.
   | (64 – 0.25 * 100) * (25 – 125) | = | 39 * (- 100) | = | -3900 | = 3900 ms (3.9 s)

Note: To change the direction of motion, a stop command must first be issued before a velocity in the opposite direction is commanded.

Position (Trapezoidal) Profile Mode

Trapezoidal profile mode is used to move to a goal position by first accelerating up to a running velocity, slewing at the running velocity, and finally decelerating to a stop at the commanded goal position.

The desired target position (number of steps) should be multiplied by 25 before it is sent to the device. For example, if the current position is 0, in order to make 100 steps, the commanded position sent to the device should be 2500.

Power-up and Reset Conditions

On Power-up or reset, the following state is established:

Motor position is reset to zero;
Velocity and acceleration values are set to zero;
All parameters are set to zero;
All outputs are cleared;
The Motor driver is disabled;
The default status data is the status byte only;
The individual address is set to 0x00 and the group address to 0xFF (group leader not set);
Communications are disabled pending a low value of "A in";
The baud rate is set to 19.2 KBPS;

COMMAND SPECIFICATION

List of Commands

Command	CMD Code	# Data bytes	Description	While Moving?
Reset position	0x0	0	Sets position counter to zero	No
Set address	0x1	2	Sets the individual and group addresses	Yes
Define status	0x2	1	Defines which data should be sent in every status packet	Yes
Read status	0x3	1	Causes particular status data to be returned just once	Yes
Load trajectory	0x4	1-7	Loads motion trajectory parameters	See Note
Start motion	0x5	0	Executes the previously loaded trajectory	See Note
Set parameters	0x6	5	Sets the motion parameters and operating limits	Yes
Stop motor	0x7	1	Stops the motor in one of three manners	Yes
I/O control	0x8	1	Sets the output values	Yes
Reserved	0x9			
Set baud rate	0xA	1	Sets the baud rate	Yes
Reserved	0xB			
Reserved	0xC			
Reserved	0xD			
Nop	0xE	0	Simply causes the defined status data to be returned	Yes
Hard reset	0xF	0	Resets the controller to its power-up state.	Yes

Note: The speed and acceleration can be changed only in velocity mode. The motion direction cannot be changed.

LS-138 Command Description

Reset Position

Command value: *0x0*
Number of data bytes: *0*
Command byte: ***0x00***

Description:

Resets the 32-bit encoder counter to 0. Do not issue this command while executing a position (trapezoidal) profile motion.

Set Address

Command value: *0x1*
Number of data bytes: *2*
Command byte: ***0x21***
Data bytes:

1. *Individual address: 0x01-0x7F (initial address 0x00)*
2. *Group Address: 0x80-0xFF (initial value 0xFF)*

Description:
Sets the individual address and group address. Group addresses are always interpreted as being between 0x80 and 0xFF. If a Drive is to be a group leader, clear bit 7 of the desired group address in the second data byte. The Step device will automatically set bit 7 internally after flagging the Drive as a group leader (If bit 7 of the second data byte is set, the module will be a group member by default). The first time this command is issued after power-up or reset, it will also enable communications for the next Drive in the network chain by lowering it's "A out" signal.

Define Status

Command value: *0x2*
Number of data bytes: *1*
Command byte: ***0x12***
Data bytes:
1. Status items: (default: 0x00)

Bit	*0:*	*send position (4 bytes)* ***see Note***
	1:	*Reserved. Set to 0*
	2:	*Reserved. Set to 0*
	3:	*send input byte (1 byte)*
	4:	*Reserved. Set to 0*
	5:	*send device ID and version number (2 bytes)*
		(motor controller device ID = 3, version number = 50-59)
	6:	*send I/O state byte*
	7:	*Reserved. Set to 0*

Description:
Defines what additional data will be sent in the status packet along with the status byte. Setting bits in the command's data byte will cause the corresponding additional data bytes to be sent after the status byte. The status data will always be sent in the order listed. For example if bits 0 and 3 are set, the status packet will consist of the status byte followed by four bytes of position data, followed by the input byte, followed by the checksum. The status packet returned in response to this command will include the additional data bytes specified. On power-up or reset, the default status packet will include only the status byte and the checksum byte.
Note: The device reports the current position of the motor multiplied by 25. The control system (host) should divide this value by 25 on order to obtain the actual number of steps.

Read Status

Command value: *0x3*
Number of data bytes: *1*
Command byte: ***0x13***
Data bytes:
1. Status items:

Bit 0: *send position (4 bytes)* ***see Note***
1: *Reserved. Set to 0*
2: *Reserved. Set to 0*
3: *send input byte (1 byte)*
4: *Reserved. Set to 0*
5: *send device ID and version number (2 bytes)*
(motor controller device ID = 3, version number = 50 - 59)
6: *send I/O state byte*
7: *Reserved. Set to 0*

Description:
This is a non-permanent version of the *Define Status* command. The status packet returned in response to this command will incorporate the data bytes specified, but subsequent status packets will include only the data bytes previously specified with the *Define Status* command.
Note: The device reports the current position of the motor multiplied by 25. The control system (host) should divide this value by 25 on order to obtain the actual number of steps.

Load Trajectory

Command value: *0x4*
Number of data bytes: *n = 1-7*
Command byte: ***0xn4***
Data bytes:
1.Control byte:

Bit 0: *load position data (n = n + 4 bytes)* ***see Note***
1: *load velocity data (n = n + 1 byte)*
2: *load acceleration data (n = n + 1 bytes)*
3: *Reserved. Set to 0*
4: *direction - 0 = positive, 1 = negative*
5,6 *Reserved. Set to 0*
7 *start motion now*

Description:
All motion parameters are set with this command. Setting one of the first three bits in the control byte will require additional data bytes to be sent (as indicated) in the order listed. The position data (range* +/- 0x7FFFFFFF) is only used as the goal position in position profile mode. The value sent to the device should be 25 times the desired target position. The velocity data (range 1 to 250) is used as the goal velocity in velocity profile mode or as the maximum velocity in trapezoidal profile mode. The acceleration data (range 1 to 255) is used in both trapezoidal and velocity profile mode.
Bit 4 indicates the velocity direction and is ignored in trapezoidal profile mode.
Note: The velocity should be greater than minimum profile velocity (see *Set Parameters* command).

* *While the position may range from -0x7FFFFFFF to +0x7FFFFFFF, the goal position should not differ from the current position by more then 0x7FFFFFFF.*

Start Motion

Command value: *0x5*
Number of data bytes: *0*
Command byte: ***0x05***

Description:
Causes the trajectory information loaded with the most recent Load Trajectory command to execute. This is useful for loading several step devices with trajectory information and then starting them simultaneously with a group command.

Set Parameters

Command value: *0x6*
Number of data bytes: *5*
Command byte: ***0x56***
Data bytes:
1.Control byte:
Bit 1,0: speed factor 00 = 8x, 01 = 4x, 10 = 2x, 11 = 1x
2: Reserved. Set to 1
3-7: Reserved. Set to 0
2. Minimum profile velocity (1 – 250)
3. Reserved. Set to 0
4. Reserved. Set to 0
5. Reserved. Set to 0

Description:
Sets control parameters and limits governing the behavior of the motor. This command must be issued before any motion can be executed.

Motor ON / Stop

Command value: *0x7*
Number of data bytes: *1*
Command byte: ***0x17***
Data bytes:
1.Stop control byte:
Bit 0: turn motor on/off
1: Reserved. Set to 0
2: stop abruptly
3: stop smoothly
4-7: Reserved – Set all to 0

Description:
Stops the motor in the specified manner. If bit 0 of the Stop Control Byte is set, the Motor driver will be turned on. If bit 0 is cleared Motor driver will be turned off, regardless of the state of the other bits. If bit 2 is set, the motor will stop moving abruptly. Setting bit 3 enters a more graceful stop mode - the motor will decelerate to a stop. Only one of bits 2 or 3 should be set at one time.

Set outputs

Command value: *0x8*
Number of data bytes: *n=1*
Command byte: ***0x18***
Data bytes:
1. Set outputs control byte
Bit 0-4: output values OUT0-OUT4
5-7: *Reserved – Set all to 0*
All bits cleared after *Hard reset command* or Power-up.

Description:
Selects the motor type and connector to control. Motor connector and type can be changed only when the motor driver is off – see *Motor ON / Stop* command.

Set Baud Rate

Command value: *0xA*
Number of data bytes: *1*
Command byte: ***0x1A***
Data bytes:
1. Baud rate divisor, *BRD*

sample values:

9600	BRD = 0x81
19200	BRD = 0x3F
57600	BRD = 0x14
115200	BRD = 0x0A

Description:
Sets the communication baud rate. All step drives on the network must have their baud rates changed at the same time; therefore this command should only be issued to a group including all of the controllers on the network. A status packet returned from this command would be at the new baud rate, so typically (unless the host's baud rate can be accurately synchronized) there should be no group leader when this command is issued.

No Operation

Command value: *0xE*
Number of data bytes: 0
Command byte: *0x0E*

Description:
Does nothing except cause a status packet with the currently defined status data to be returned.

Hard Reset

Command value: *0xF*
Number of data bytes: *0*
Command byte: ***0x0F***

Description:
Resets the control module to its power-up state. No status will be returned. Typically, this command is issued to all the modules on the network, although if the baud rate is set at the default, it is possible to reset and re-initialize the addresses of a contiguous sub-chain of modules.
Note: *Hard Reset* command sent at address 0xFF will be executed by all step Drives, regardless of their own group address.

STATUS BYTE AND I/O BYTE DEFINITIONS

Status Byte

Bit	Name	Definition
0	Motor is moving	This bit is set when the motor is moving and cleared otherwise.
1	Cheksum error	Set if there was a checksum error in the command packet received.
2	Motor is on	Set if the motor power driver is enabled.
3	Motor selector status	Cleared if selected motor connector is out of range.
4	At commanded velocity	Set if the commanded velocity is reached.
5	Velocity profile mode	Set if the motor is moving in velocity mode.
6	Position (trapezoidal) profile mode	Set if the motor is moving in trapezoidal mode.
7	Reserved	

Input Byte

Bit	Name	Definition
0	IN0	The value of bit IN0 (STOP)
1	IN1	The value of bit IN1
2	IN2	The value of bit IN2
3	IN3	The value of bit IN3
4	IN4	The value of bit IN4
5	IN5	The value of bit IN5
6	Reserved	
7	Reserved	

I/O State Byte

Bit	Name	Definition
0	IN0	The value of bit IN0 (STOP)
1	IN1	The value of bit IN1
2	IN2	The value of bit IN2
3	OUT0	The value of Motor selector bit 0
4	OUT1	The value of Motor selector bit 1
5	OUT2	The value of Motor selector bit 2
6	OUT3	The value of output bit 3 (*Reserved)*
7	OUT4	The value of Motor selector bit 4 (0 = Standard, 1 = Tiny)

Note: IN0, IN1 and IN2 in Input byte and I/O state byte are the same inputs.

Examples of Managing Two LS-138 Piezo-Motor Drives

#1 – Resets all modules with group command.
#2 and #3 - Set the addresses of drives 1 and 2.
#4 and #5 - Set parameters of drives 1 and 2.
#6 and #7 – Select standard motor, channel A for drive 1 and tiny motor, channel A for drive 2.
#8 and #9 – Read the status byte and the I/O state byte. Check "Motor selector status" bit to ensure the selected channels are supported. Continue only if this bit is set.
#10 and #11 – Turn on the motor driver of drives 1 and 2. Initialization is complete at this point.
#12 and #13 - Load trajectories (velocities and accelerations) for drives 1 and 2 with "start motion now" bit set to 1. These commands will start the motors.
#14, #15, and 16 - Load new trajectories for drives 1 and 2 and start the motors simultaneously with one command sent to the drives' group address.

#	Hexadecimal code of command	Comments
1	AA FF 0F 0E	*Hard Reset*
2	AA 00 21 01 FF 21	*Set Address* 01h for drive 1. Group address=FFh.
3	AA 00 21 02 FF 22	*Set Address* 02h for drive 2. Group address=FFh.
4	AA 01 56 04 01 00 00 00 5C	*Set parameters.* Sets drive 1 parameters: speed factor=8, minimum velocity=1
5	AA 02 56 04 01 00 00 00 5D	*Set parameters.* Sets drive 2 parameters: speed factor=8, minimum velocity=1
6	AA 01 18 01 1A	*Set outputs* for drive 1 – OUT4=0 – standard motor, OUT3=0, OUT2=0, OUT1=0, OUT0=0 – channel A
7	AA 02 18 10 2A	*Set outputs* for drive 2 – OUT4=1 – tiny motor, OUT3=0, OUT2=0, OUT1=0, OUT0=0 – channel A
8	AA 01 13 40 54	*Read status* – reads the status byte and the I/O state byte
9	AA 02 13 40 55	*Read status* – reads the status byte and the I/O state byte
10	AA 01 17 05 1D	*Motor ON/Stop* – enables the motor driver
11	AA 02 17 05 1E	*Motor ON/Stop* – enables the motor driver
12	AA 01 34 86 7D FF 37	*Load Trajectory* of drive 1: start motion now, load velocity 125 (125 x 8 = 1KHz) and acceleration 255
13	AA 02 34 86 7D FF 38	*Load Trajectory* of drive 2: start motion now, load velocity 125 (125 x 8 = 1KHz) and acceleration 255
14	AA 01 34 16 7D FF C7	*Load Trajectory* of drive 1: reverse direction, load velocity 125 (125 x 8 = 1KHz) and acceleration 255
15	AA 02 34 16 7D FF C8	*Load Trajectory* of drive 2: reverse direction, load velocity 125 (125 x 8 = 1KHz) and acceleration 255
16	AA FF 05 04	*Start Motion* – executes previously loaded trajectories. The command is sent to the drives' group address FFh.

Device identification example

The device type (3) and the firmware version (50 to 59) do not identify LS-138. In order to identify the device, the following sequence of commands should be sent to the device immediately after Hard Reset, Set Address, and Read Status (read device type and version) commands. LS-138 will return 0x01 on step 1 and 0x3E on step 3.

1	AA 01 13 08 1C	Read the Input byte. LS-138 will return 0x01.
2	AA 01 18 10 29	Set OUT4 to 1.
3	AA 01 13 08 1C	Read the Input byte again. LS-138 will return 0x3E.
4	AA 01 18 00 19	Set OUT4 to 0. Compare the values of IN0 to IN5 obtained at steps 1 and 3. If all bits are inverted, then the value obtained at step 1 is the ID of the device.

Unsupported channel detecting example

Here is an example how to detect if a channel is supported. It is recommended to perform this procedure once after the initialization of all the controllers and after identifying LS-138.

1	AA 01 17 00 18	Motor ON/Stop with bit 0 cleared.
2	AA 01 18 03 1C	*Set outputs* for drive 1 – OUT4=0 – standard motor, OUT3=0, OUT2=0, OUT1=1, OUT0=1 (unsupported channel).
3	AA 01 0E 0F	No Operation - read the status byte. Motor selector status (bit 3) in Status byte will be 0.

Missing motor detecting example

Here is an example how to detect a missing motor. It is recommended to perform this procedure once after the initialization of all the controllers and after identifying LS-138.

1	AA 01 17 00 18	Motor ON/Stop with bit 0 cleared.
2	AA 01 18 00 19	Select channel A (standard motor mode) with Set Outputs command.
3	AA 01 17 05 18	Motor ON/Stop with bit 0 set – enable channel A.
4	AA 01 00 01	Reset position command.
5	AA 01 64 E6 FF FF FF 64 C8 74	Load trajectory command – go to position –1 (the actual position sent to the device is –25) with velocity 100 steps per second and acceleration 200.
6	AA 01 13 08 1C	Read Status command – read the status byte and the Input byte. Keep sending this command while the motor is moving (bit 0 in Status byte is 1). If Motor on (bit 2) in Status byte is 0, IN2 and IN1 are 0, and IN0 is 1, then the device has detected missing motor on the selected channel.