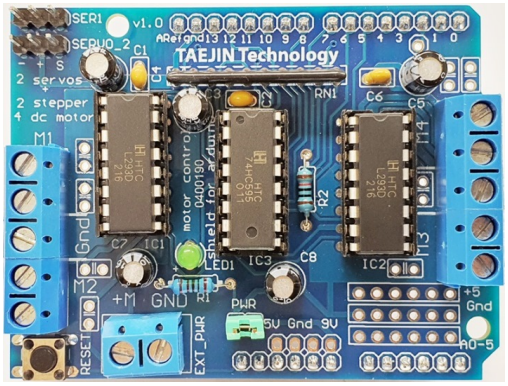
EN: This Datasheet/Document is presented by the manufacturer.

Please visit our website for pricing and availability at www.hestore.hu.

L293D Motor Shield

FEATURES

- 2 connections for 5V 'hobby' servos connected to the Arduino's high-resolution dedicated timer - no jitter
- 4 H-Bridges: L293D chipset provides 0.6A per bridge (1.2A peak) with thermal shutdown protection, internal kickback protection diodes. Can run motors on 4.5VDC to 25VDC.
- Up to 4 bi-directional DC motors with individual 8-bit speed selection (so, about 0.5% resolution)
- Up to 2 stepper motors (unipolar or bipolar) with single coil, double coil or interleaved stepping.
- Pull down resistors keep motors disabled during power-up
- Big terminal block connectors to easily hook up wires (18-26AWG) and power
- Arduino reset button brought up top
- 2-pin terminal block and jumper to connect external power, for separate logic/motor supplies
- Tested compatible with Arduino Mega 1280 & 2560, Diecimila, Duemilanove, and UNO
- Download the easy-to-use Arduino software library, check out the ready-to-go examples



ORDERING INFORMATION

Part Number	MMS0293-01
Device	L293D Motor Shield Module
Package	Assembled Module

APPLICATIONS

- Robotics
- Mechatronics
- Simple to medium-complexity projects

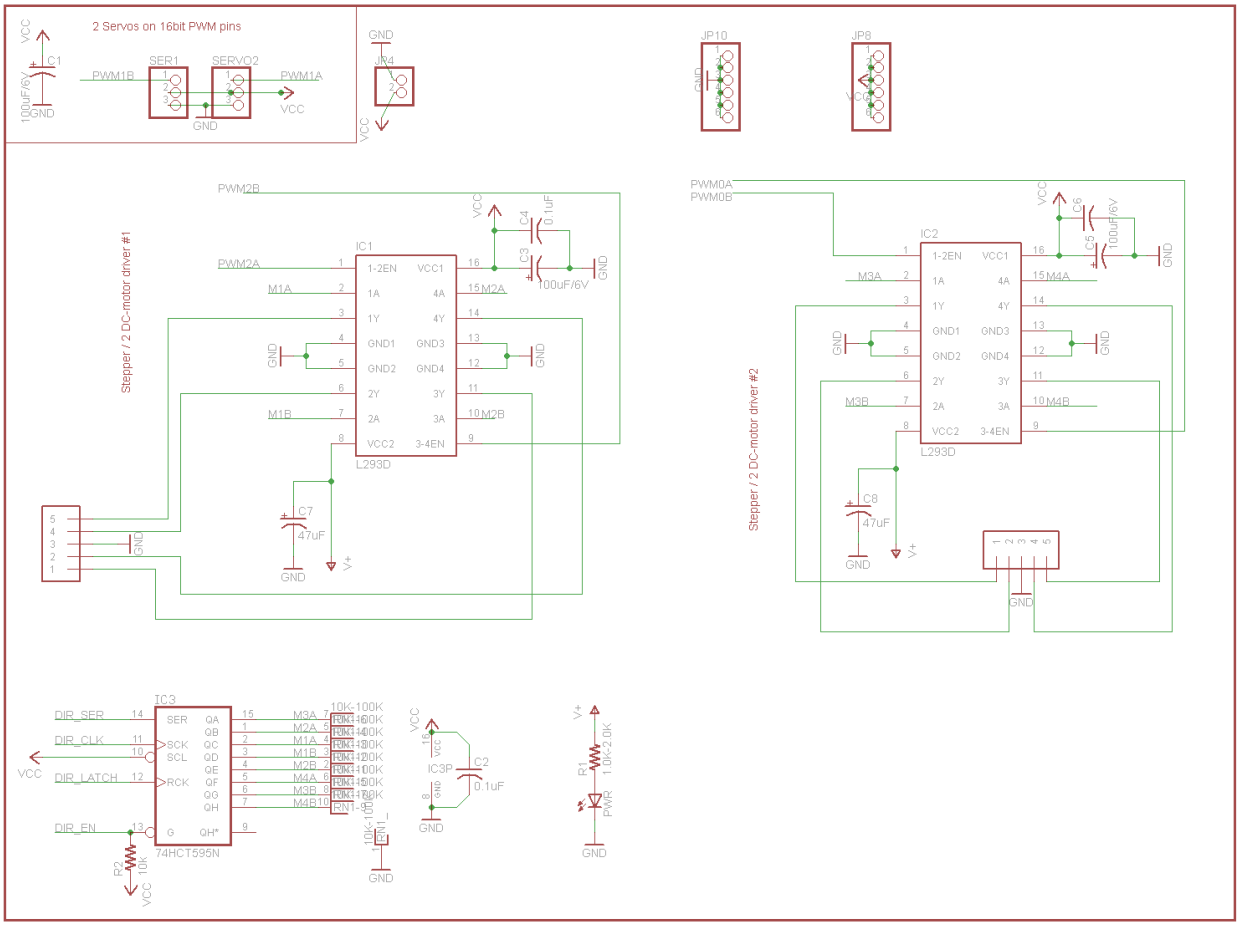
DESCRIPTION

The L293D Motor Shield is a great motor controller for Arduino. It control servos, steppers and DC motors with an Arduino.

PARTS LIST

Name	Description	P/N	Qty.	Distributor
PCB	Printed Circuit Board	Motor Shield PCB	1	TAEJIN
IC1, IC2	L293D, Quadruple Dual H-Bridge	L293DN	2	TAEJIN
IC3	74HC595, Serial to Parallel Output Latch	74HC595N	1	TAEJIN
IC1', IC2', IC3'	16 Pin Sockets	DIP 16 pin Sockets	3	Generic
LED1	LED for Power Indicator	3mm LED, Green	1	Generic
R1	1.0K Resistor for LED1	1.0K 1/4W 5% Resistor	1	Generic
R2	10K Pulldown Resistor	10K 1/4W 5% Resistor	1	Generic
RN1	10 Pin Bussed 10K Resistor Network	10K 5% Resistor Network	1	Generic
C2, C4, C6	0.1uF Capacitor	0.1µF/50V Ceramic Capacitor	3	Generic
C1, C3, C5, C7, C8	100µF Capacitor	100µF/16V Electrolytic Capacitor	5	Generic
X1	5-Position Terminal Block (Or 1 3-Position and 2-Position)	5-Position 0.2Inch Terminals	2	Generic
X2	2-Position Terminal Block	2-Position 0.2Inch Terminals	1	Generic
RESET	Tactile Switch	6mm Tactile Switch	1	Generic
PWR	Jumper/Shunt	0.1 Inch Jumper	1	Generic
	36 Pin Male Header (1x34)	1x36 Male Header	1	Generic

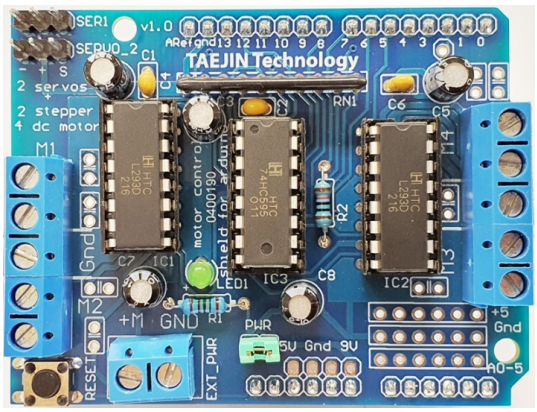
SCHEMATICS



< Connection with Arduino Board >

L293D Motor Shield

LAYOUT



I/O PORT CONFIGURATION AND DESCRIPTION

Designator	Port Name	Port Description
+M	EXT_PWR	External Power In. (6V to 12V)
GND	GND	Ground for External Power In.
RESET	RESET	Reset Switch for Arduino Board
PWR	Power Jumper	Disconnect the jumper if external power used.
M1	M1	DC Motor #1
M2	M2	DC Motor #2
M3	M3	DC Motor #3
M4	M4	DC Motor #4
Gnd	GND	Ground for Motors
SER1	SERVO1	Servo Motor #1
SERVO2	SERVO2	Servo Motor #2
JP1	D0-D7	Digital I/O. Connect to Arduino board.
JP2	D8-D13, gnd, ARef	Digital I/O, GND, and AREF. Connect to Arduino board.
JP3	Vin, Gnd, 5V, 3V, RST	VIN, GND, 5V, 3V, RST. Connect to Arduino board.
JP4	A0-A5	Analog In. Connect to Arduino board.

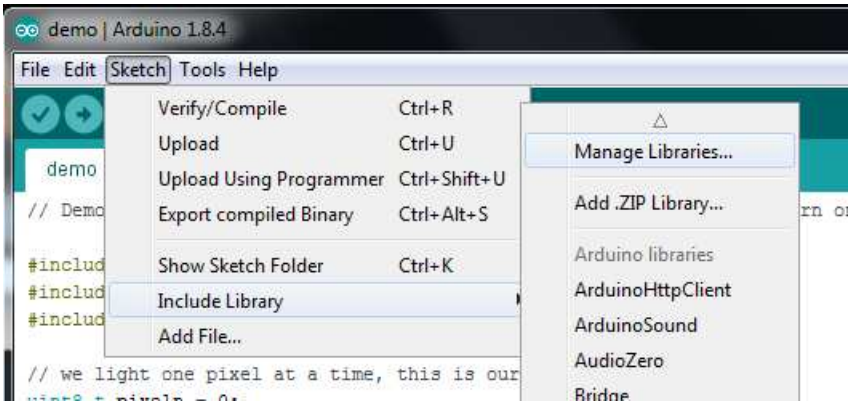
MOTOR CONNECTIONS

Types of Motors	Maximum Connection	Used Ports
Servo Motors	2	SER1, SERVO2
Stepper Motors	2	M1, M2, M3, M4
DC Motors	4	M1, M2, M3, M4

INSTALL THE ARDUINO LIBRARY

Before you can use the Motor shield, you must install the AF_Motor Arduino library - this will instruct the Arduino how to talk to the Adafruit Motor shield, and it isn't optional.

Open up the Arduino library manager:



adafruit_products_library_manager_menu.png
Search for Adafruit Motor library and install it. Make sure it is the library for the V1 motor shield.

POWER USAGE

POWERING YOUR DC MOTORS, VOLTAGE AND CURRENT REQUIREMENTS

Motors need a lot of energy, especially cheap motors since they're less efficient. The first important thing to figure out what voltage the motor is going to use. If you're lucky your motor came with some sort of specifications. Some small hobby motors are only intended to run at 1.5V, but it’s just as common to have 6-12V motors. The motor controllers on this shield are designed to run from 4.5V to 25V. Most 1.5-3V motors will not work.

Current requirements

The second thing to figure out is how much current your motor will need. The motor driver chips that come with the kit are designed to provide up to 600 mA per motor, with 1.2A peak current. Note that once you head towards 1A you'll probably want to put a heatsink on the motor driver, otherwise you will get thermal failure, possibly burning out the chip.

Need more power?

Buy another set of L293D drivers and solder them right on top of the ones on the board (piggyback). Voila, double the current capability. You can solder 2 more chips on top before it probably isnt going to get you much benefit

You can't run motors off of a 9V battery so don't even waste your time/batteries.

Use a big Lead Acid or NiMH battery pack. It’s also very much suggested that you set up two power supplies (split supply) one for the Arduino and one for the motors. 99% of 'weird motor problems' are due to noise on the power line from sharing power supplies and/or not having a powerful enough supply.

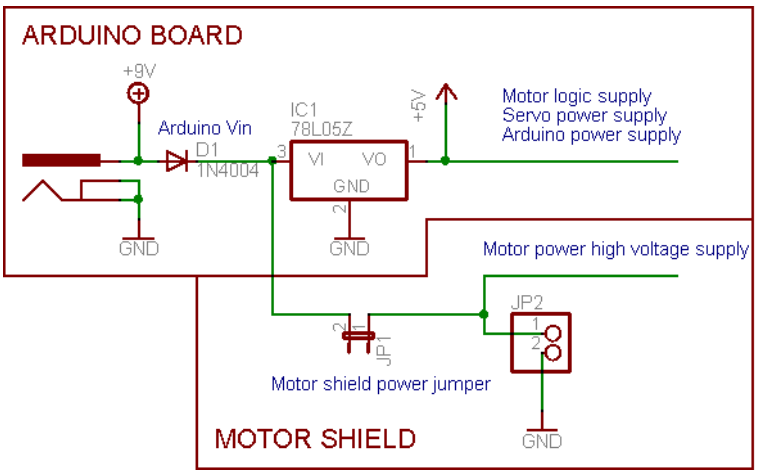
HOQ TO SET UP THE ARDUINO + SHIELD FOR POWERING MOTORS

Servos are powered off of the same regulated 5V that the Arduino uses. This is OK for the small hobby servos suggested. If you want something beefier, cut the trace going to + on the servo connectors and wire up your own 5-6V supply.

The DC motors are powered off of a 'high voltage supply' and NOT the regulated 5V. Don't connect the motor power supply to the 5V line. This is a very bad idea unless you are sure you know what you're doing.

There are two places you can get your motor 'high voltage supply' from. One is the DC jack on the Arduino board and the other is the 2-terminal block on the shield that is labeled EXT_PWR. The DC Jack on the Arduino has a protection diode so you won't be able to mess things up too bad if you plug in the wrong kind of power. However the EXT_PWR terminals on the shield do not have a protection diode (for a fairly good reason). Be utterly careful not to plug it in backwards or you will destroy the motor shield and/or your Arduino.

Here's how it works:



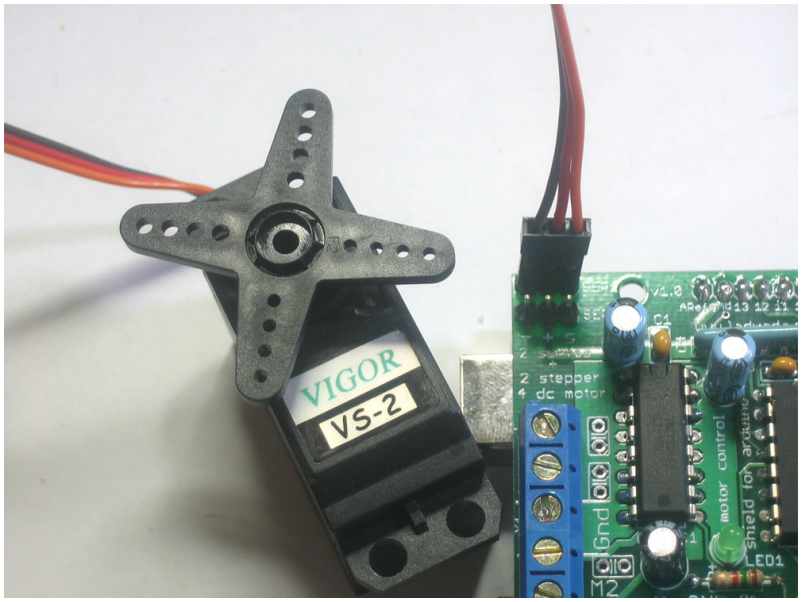
If you would like to have a single DC power supply for the Arduino and motors, simply plug it into the DC jack on the Arduino or the 2-pin PWR_EXT block on the shield. Place the power jumper on the motor shield.
If you have a Diecimila Arduino, set the Arduino power source jumper to EXT.
Note that you may have problems with Arduino resets if the battery supply is not able to provide constant power, and it is not a suggested way of powering your motor project.

If you would like to have the Arduino powered off of USB and the motors powered off of a DC power supply, plug in the USB cable. Then connect the motor supply to the PWR_EXT block on the shield. Do not place the jumper on the shield. This is a suggested method of powering your motor project
(If you have a Diecimila Arduino, don't forget to set the Arduino power jumper to USB. If you have a Diecimila, you can alternately do the following: plug the DC power supply into the Arduino, and place the jumper on the motor shield.)

If you would like to have 2 separate DC power supplies for the Arduino and motors. Plug in the supply for the Arduino into the DC jack, and connect the motor supply to the PWR_EXT block. Make sure the jumper is removed from the motor shield.
If you have a Diecimila Arduino, set the Arduino jumper to EXT. This is a suggested method of powering your motor project.

Either way, if you want to use the DC motor/Stepper system the motor shield LED should be lit indicating good motor power.

USING RC SERVOS



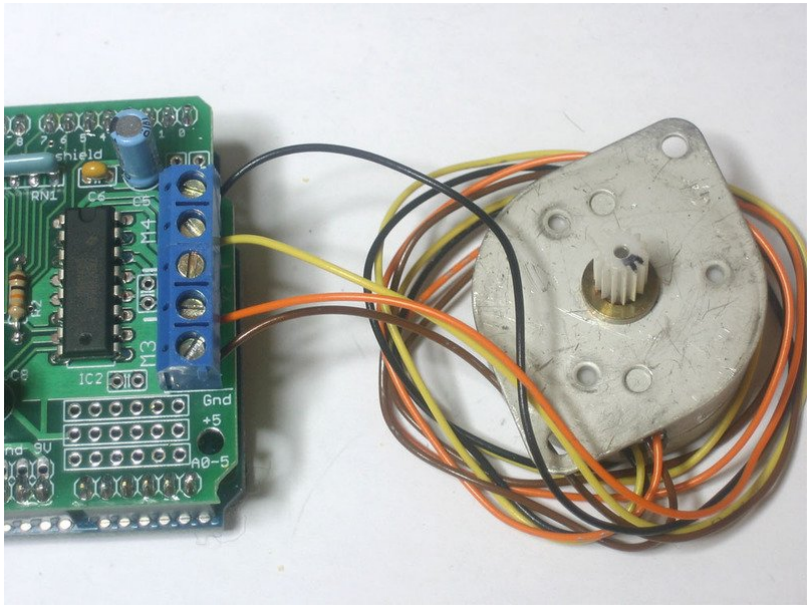
Hobby servos are the easiest way to get going with motor control. They have a 3-pin 0.1" female header connection with +5V, ground and signal inputs. The motor shield simply brings out the 16bit PWM output lines to two 3-pin headers so that its easy to plug in and go. They can take a lot of power so a 9V battery wont last more than a few minutes.

The nice thing about using the onboard PWM is that its very precise and goes about its business in the background. You can use the built in Servo library

Using the servos is easy, please read the official Arduino documentation for how to use them and see the example Servo sketches in the IDE.

Power for the Servos comes from the Arduino's on-board 5V regulator, powered directly from the USB or DC power jack on the Arduino. If you need an external supply, cut the trace right below the servo pins (on v1.2 boards) and connect a 5V or 6V DC supply directly. Using an external supply is for advanced users as you can accidentally destroy the servos by connecting a power supply incorrectly.

USING STEPPER MOTORS



Stepper motors are great for (semi-)precise control, perfect for many robot and CNC projects. This motor shield supports up to 2 stepper motors. The library works identically for bi-polar and uni-polar motors

For unipolar motors: to connect up the stepper, first figure out which pins connected to which coil, and which pins are the center taps. If it's a 5-wire motor then there will be 1 that is the center tap for both coils. There's plenty of tutorials online on how to reverse engineer the coils pinout. The center taps should both be connected together to the GND terminal on the motor shield output block. then coil 1 should connect to one motor port (say M1 or M3) and coil 2 should connect to the other motor port (M2 or M4).

For bipolar motors: it's just like unipolar motors except theres no 5th wire to connect to ground. The code is exactly the same.

Running a stepper is a little more intricate than running a DC motor but its still very easy

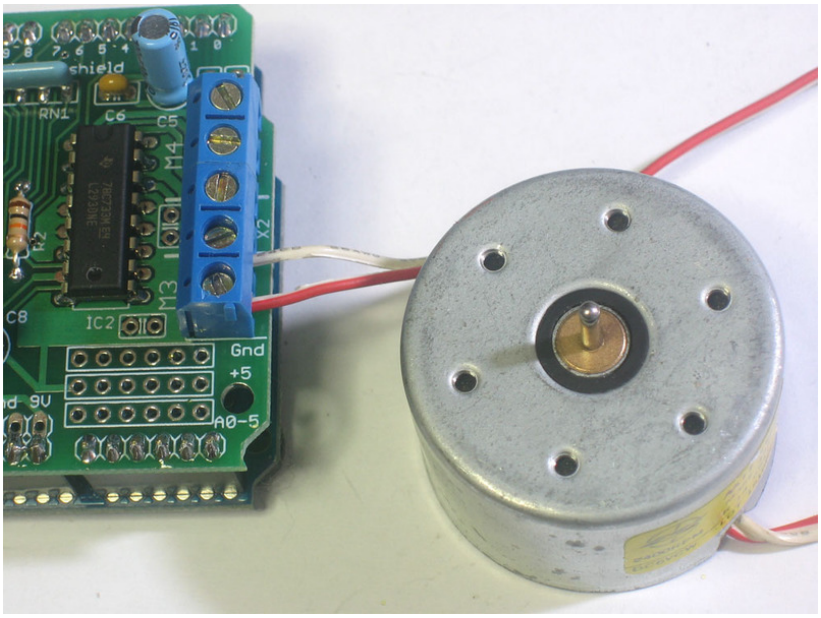
1. Make sure you #include <AFMotor.h>
2. Create the stepper motor object with AF_Stepper(steps, stepper#) to setup the motor H-bridge and latches. Steps indicates how many steps per revolution the motor has. a 7.5degree/step motor has 360/7.5 = 48 steps. Stepper# is which port it is connected to. If you're using M1 and M2, its port 1. If you're using M3 and M4 it's port 2
3. Set the speed of the motor using setSpeed(rpm) where rpm is how many revolutions per minute you want the stepper to turn.
4. Then every time you want the motor to move, call the step(#steps, direction, steptype) procedure.#steps is how many steps you'd like it to take. direction is either FORWARD or BACKWARD and the step type is SINGLE, DOUBLE. INTERLEAVE or MICROSTEP.
   "Single" means single-coil activation, "double" means 2 coils are activated at once (for higher torque) and "interleave" means that it alternates between single and double to get twice the resolution (but of course its half the speed). "Microstepping" is a method where the coils are PWM'd to create smooth motion

between steps. There's tons of information about the pros and cons of these different stepping methods in the resources page.
You can use whichever stepping method you want, changing it "on the fly" to as you may want minimum power, more torque, or more precision.

5. By default, the motor will 'hold' the position after its done stepping. If you want to release all the coils, so that it can spin freely, call release().
6. The stepping commands are 'blocking' and will return once the steps have finished.

Because the stepping commands 'block' - you have to instruct the Stepper motors each time you want them to move. If you want to have more of a 'background task' stepper control, check out AccelStepper library (install similarly to how you did with AFMotor) which has some examples for controlling two steppers simultaneously with varying acceleration.

USING DC MOTORS



DC motors are used for all sort of robotic projects.
The motor shield can drive up to 4 DC motors bi-directionally. That means they can be driven forwards and backwards. The speed can also be varied at 0.5% increments using the high-quality built in PWM. This means the speed is very smooth and won't vary.

Note that the H-bridge chip is not meant for driving loads over 0.6A or that peak over 1.2A so this is for small motors. Check the datasheet for information about the motor to verify it's OK.

To connect a motor, simply solder two wires to the terminals and then connect them to either the M1, M2, M3, or M4. Then follow these steps in your sketch

1. Make sure you #include <AFMotor.h>
2. Create the AF_DCMotor object with AF_DCMotor(motor#, frequency), to setup the motor H-bridge and latches. The constructor takes two arguments.The first is which port the motor is connected to, 1, 2, 3 or 4. frequency is how fast the speed controlling signal is.For motors 1 and 2 you can choose MOTOR12_64KHZ, MOTOR12_8KHZ, MOTOR12_2KHZ, orMOTOR12_1KHZ. A high speed like 64KHz won't be audible but a low speed like 1KHz will use less power. Motors 3 & 4 are only possible to run at 1KHz and will ignore any setting given.
3. Then you can set the speed of the motor using setSpeed(speed) where the speed ranges from 0 (stopped) to 255 (full speed). You can set the speed whenever you want.
4. To run the motor, call run(direction) where direction is FORWARD, BACKWARD or RELEASE. Of course, the Arduino doesn't actually know if the motor is 'forward' or 'backward', so if you want to change which way it thinks is forward, simply swap the two wires from the motor to the shield.

REVISION NOTICE

The description in this datasheet is subject to change without any notice to describe its electrical characteristics properly.

FEATURES

- Wide Supply-Voltage Range: 4.5V to 36V
- Separate Input-Logic Supply
- Internal ESD Protection (HBM 2kV)
- High-Noise-Immunity Inputs
- Output Current 600mA Per Channel
- Peak Output Current 1.2A Per Channel
- Output Clamp Diodes for Inductive Transient Suppression

APPLICATION

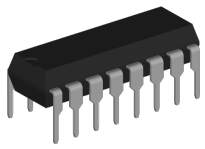
- Stepper Motor Drivers
- DC Motor Drivers
- Latching Relay Drivers

DESCRIPTION

The L293D device is quadruple high-current half-H drivers. The L293D is designed to provide bidirectional drive currents of up to 600mA at voltages from 4.5V to 36V. The device is designed to driver inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications.

Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by EN1/2 and drivers 3 and 4 enabled by EN3/4.

The L293D is characterized for operation from 0°C to 70°C.



DIP-16

ORDERING INFORMATION

Device	Package
L293DN	DIP-16

ABSOLUTE MAXIMUM RATINGS (Note 1)

CHARACTERISTIC	SYMBOL	MIN	MAX	UNIT
Logic Supply Voltage (Note 2)	V_{CC1}	−0.3	36	V
Power Supply Voltage	V_{CC2}	−0.3	36	V
Input Voltage	V_{IN}	−0.3	7.0	V
Enable Input Voltage	V_{EN}	−0.3	7.0	V
Output Voltage	V_{OUT}	−3.0	V_{CC2} + 3.0	V
Peak Output Current (non-repetitive, t ≤ 100 µs)	$I_{OUT.PEAK}$	−1.2	1.2	A
Continuous Output Current	$I_{OUT.CONT}$	−600	600	mA
Maximum Junction Temperature Range	T_J	-	150	°C
Storage Temperature Range	T_{STG}	−65	150	°C

Note 1. Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. These are stress rating only which do not imply functional operation of the device at these or any other conditions beyond those indicated under Recommended Operating conditions. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.

Note 2. All voltage values are with respect to the network ground terminal.

RECOMMENDED OPERATING CONDITIONS

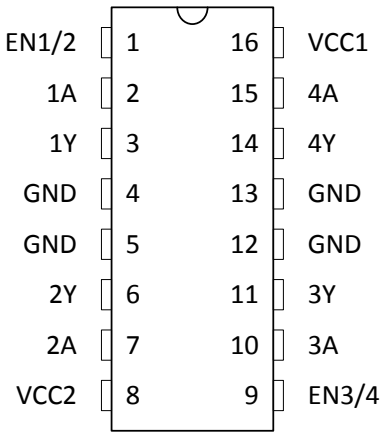
CHARACTERISTIC	SYMBOL	MIN	MAX	UNIT
Logic Supply Voltage	V_{CC1}	4.5	7.0	V
Power Supply Voltage	V_{CC2}	V_{CC1}	36	V
High Level Input Voltage (V_{CC1} ≤ 7.0 V)	V_{IH}	2.3	V_{CC1}	V
High Level Input Voltage (V_{CC1} > 7.0 V)	V_{IH}	2.3	7.0	V
Low Level Input Voltage	V_{IL}	−0.3 (Note 3)	1.5	V
Operating Free-Air Temperature Range	T_A	0	70	°C

Note 3. The algebraic convention, in which the least positive (most negative) designated minimum, is used in this datasheet for logic voltage levels.

ORDERING INFORMATION

Package	Order No.	Description	Supplied As	Status
DIP-16	L293DN	Quadruple Half-H Drivers	Tube	Active

PIN CONFIGURATION

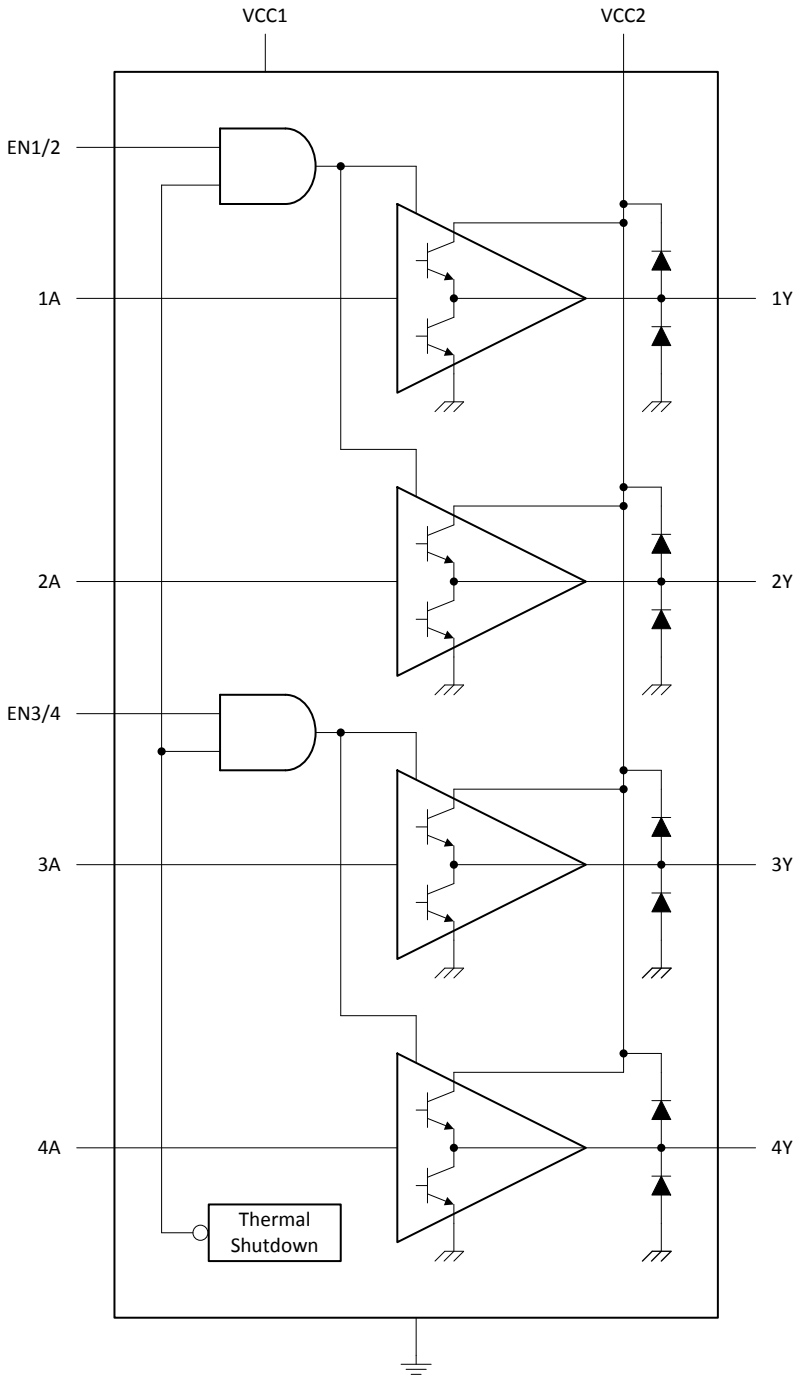


DIP-16

PIN DESCRIPTION

Pin No.	Pin Name	Pin Function
1	EN1/2	Enable Driver Channels 1 and 2. Active High Input.
2	1A	Driver 1 Input. Non-Inverting.
3	1Y	Driver 1 Output.
4, 5, 12, 13	GND	Device Ground and Heat Sink pin. Connect to PCB ground plane with multiple solid vias.
6	2Y	Driver 2 Output.
7	2A	Driver 1 Input. Non-Inverting.
8	VCC2	Power VCC for Drivers 4.5V to 36V.
9	EN3/4	Enable Driver Channels 3 and 4. Active High Input.
10	3A	Driver 3 Input. Non-Inverting.
11	3Y	Driver 3 Output.
14	4Y	Driver 4 Output.
15	4A	Driver 4 input. Non-Inverting.
16	VCC1	5V Supply for Internal Logic Translation.

FUNCTIONAL BLOCK DIAGRAM



DC ELECTRICAL CHARACTERISTICS

T_A = 25°C, unless otherwise noted.

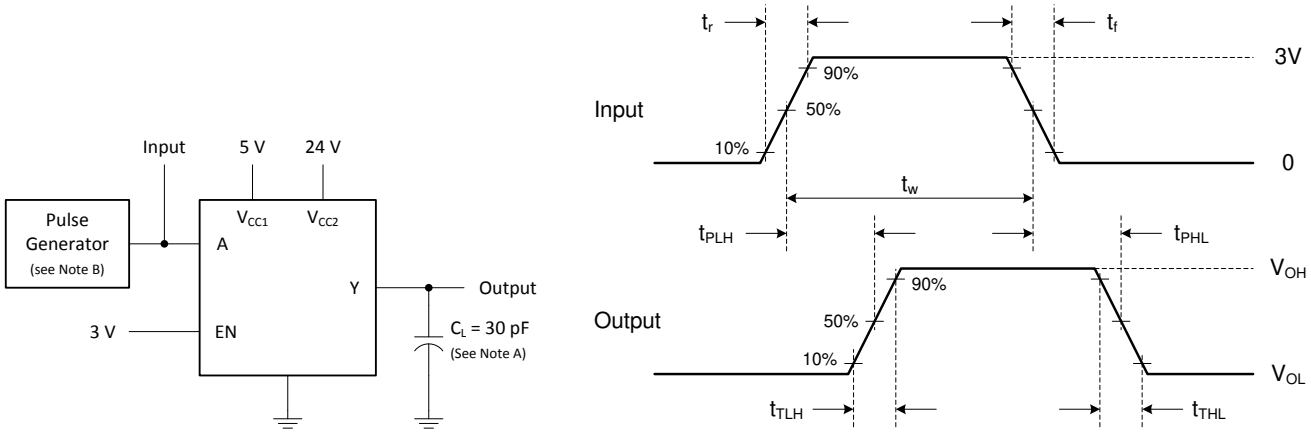
PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNIT
High Level Output Voltage	V_{OH}	I_{OH} = −0.6 A	V_{CC2} − 1.8	V_{CC2} −1.4	-	V
Low Level Output Voltage	V_{OL}	I_{OL} = 0.6 A	-	1.2	1.8	V
High Level Output Clamp Voltage	V_{OKH}	I_{OK} = −0.6 A	-	V_{CC2} + 1.3	-	V
Low Level Output Clamp Voltage	V_{OKL}	I_{OK} = 0.6 A	-	1.3	-	V
Input Voltage High Level	V_{IH}		2.3	-	-	V
Input Voltage Low Level	V_{IL}		-	-	1.5	V
Enable Voltage High Level	V_{ENH}		2.3	-	-	V
Enable Voltage Low Level	V_{ENL}		-	-	1.5	V
Input Current High Level	I_{IH}	V_{IN} = 7.0 V	-	0.2	100	µA
Input Current Low Level	I_{IL}	V_{IN} = 0 V	-	−3.0	−10	µA
Enable Current High Level	I_{ENH}	V_{IN} = 7.0 V	-	0.2	10	µA
Enable Current Low Level	I_{ENL}	V_{IN} = 0 V	-	−2.0	−100	µA
Logic Supply Current	I_{CC1}	I_{OUT} = 0 A, All Outputs at High Level	-	13	26	mA
		I_{OUT} = 0 A, All Outputs at Low Level	-	35	60	
		I_{OUT} = 0 A, All Outputs at High Impedance	-	8.0	24	
Power Supply Current	I_{CC2}	I_{OUT} = 0 A, All Outputs at High Level	-	14	24	mA
		I_{OUT} = 0 A, All Outputs at Low Level	-	2.0	6.0	
		I_{OUT} = 0 A, All Outputs at High Impedance	-	2.0	4.0	

AC ELECTRICAL CHARACTERISTICS

T_A = 25°C, V_{CC1} = 5.0 V, V_{CC2} = 24 V, unless otherwise noted.

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNIT
Turn-On Delay from Input to Output	t_{PLH}	C_L = 30 pF	-	800	-	ns
Turn-Off Delay from Input to Output	t_{PHL}	C_L = 30 pF	-	400	-	ns
Transition Time, Output Low to High	t_{TLH}	C_L = 30 pF	-	300	-	ns
Transition Time, Output High to Low	t_{THL}	C_L = 30 pF	-	300	-	ns

PARAMETER MEASUREMENT INFORMATION



TEST CIRCUIT VOLTAGE WAVEFORMS

Note A. C_L includes probe and jig capacitance.
Note B. The pulse generator has the following characteristics: $t_r \leq 10$ ns, $t_f \leq 10$ ns, $t_w = 10$ µs, PRR = 5 kHz, $Z_O = 50$ Ω.

Test Circuit and Voltage Waveforms

FUNCTION TABLE

Input (A)	Input (EN)	Output (Y)
H	H	H
L	H	L
X	L	Z

APPLICATION INFOMATION

OVERVIEW

The L293D is quadruple high-current half-H driver. This device is designed to drive a wide array of inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as other high current and high voltage loads. All inputs are TTL compatible and tolerant up to 7.0 V.

Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by EN1/2 and drivers 3 and 4 enabled by EN3/4. When an enable input is high, the associated drivers are enabled, and their outputs are active and inphase with their inputs. When the enable input is low, those drivers are disabled, and their outputs are off and in the high-impedance state. With the proper data inputs, each pair of drivers forms a full-H (or bridge) reversible drive suitable for solenoid or motor applications.

On the L293D, output clamp diodes are integrated to reduce system complexity and overall system size. A VCC1 terminal, separate from VCC2, is provided for the logic inputs to minimize device power dissipation. The L293D is characterized for operation from 0 °C to 70 °C.

The L293D has TTL compatible inputs and high voltage outputs for inductive load switching. Current outputs can get up to 1.2A using L293D.

POWER SUPPLY RECOMMENDATIONS

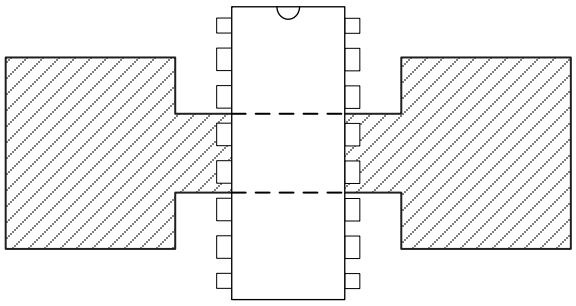
VCC1 is 5.0 V ± 0.5 V and VCC2 can be same supply as VCC1 or a higher voltage supply with peak voltage up to 36 V. Bypass capacitors of 0.1 μF or greater should be used at VCC1 and VCC2 pins. There are no power-up or power-down supply sequence order requirements.

Properly heatsinking the L293D when driving high-current is critical to design. The θ_{JA} of the L293D can be reduced by soldering the GND to a suitable copper area of the PCB or to an external heat sink.

The external heatsink or copper area must be connected to electrical ground.

LAYOUT GUIDELINES

Place the device near the load to keep output traces to reduce EMI. Use solid vias to transfer heat from ground pins to ground plane of the PCB.



Layout Example of PCB Copper Area (Used as Heat Sink)

REVISION NOTICE

The description in this datasheet is subject to change without any notice to describe its electrical characteristics properly.

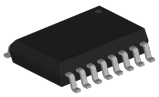
8-Bit Shift Registers With Latched 3-State Outputs 74HC595

FEATURES

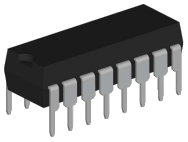
- Wide Operating Voltage Range of 2.0V to 6.0V
- 8-Bit Serial-Input, Serial or Parallel-Out Shift
- Outputs Directly Interface to CMOS, NMOS, and TTL
- Low Input Current: 1.0μA
- High Noise Immunity Characteristic of CMOS Devices

APPLICATIONS

- Network Switches
- Power Infrastructure
- LED Displays
- Servers



SOP-16 / TSSOP-16



DIP-16

DESCRIPTION

The 74HC595 devices contain and 8-bit, serial-in, parallel-out shift register that feeds an 8-bit D-type latch with parallel 3-state outputs. Separate clocks are provided for both the shift register and latch. The shift register has a direct overriding clear input, serial input, and serial outputs for cascading. This device also has an asynchronous reset for the shift register.

ORDERING INFORMATION

Device	Package
74HC595D	SOP-16
74HC595TD	TSSOP-16
74HC595N	DIP-16

ABSOLUTE MAXIMUM RATINGS (Note 1)

CHARACTERISTIC	SYMBOL	MIN.	MAX.	UNIT
DC Supply Voltage (Referenced to GND)	V_{CC}	−0.5	7.0	V
DC Input Voltage (Referenced to GND)	V_{IN}	−0.5	V_{CC} + 0.5	V
DC Output Voltage (Referenced to GND)	V_{OUT}	−0.5	V_{CC} + 0.5	V
DC Input Current	I_{IN}	-	±20	mA
DC Output Current	I_{OUT}	-	±35	mA
DC Supply Current	I_{CC}	-	±75	mA
Maximum Junction Temperature	T_J	-	150	°C
Storage Temperature	T_{STG}	−65	150	°C

Note1. Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions* is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

RECOMMENDED OPERATING CONDITIONS (Note 2)

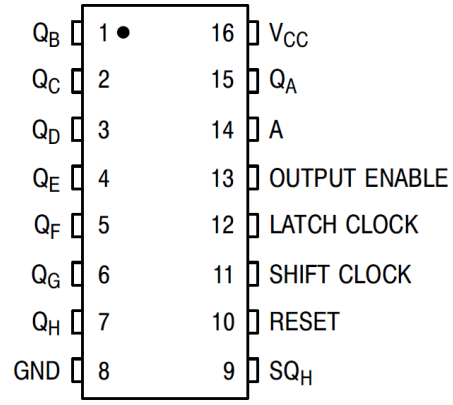
CHARACTERISTIC	SYMBOL	MIN.	MAX.	UNIT
Supply Voltage	V_{CC}	2.0	6.0	V
DC Input Voltage	V_{IN}	0	V_{CC}	V
DC Output Voltage	V_{OUT}	0	V_{CC}	V
Operating Free-Air Temperature Range	T_A	−55	125	°C

Note 2. The device is not guaranteed to function outside its operating ratings.

ORDERING INFORMATION

Package	Order No.	Description	Package Marking	Status
SOP-16	74HC595D	8-Bit Shift Resisters	74HC595	Active
TSSOP-16	74HC595TD	8-Bit Shift Resisters	74HC595	Active
DIP-16	74HC595N	8-Bit Shift Resisters	74HC595	Active

PIN CONFIGURATION

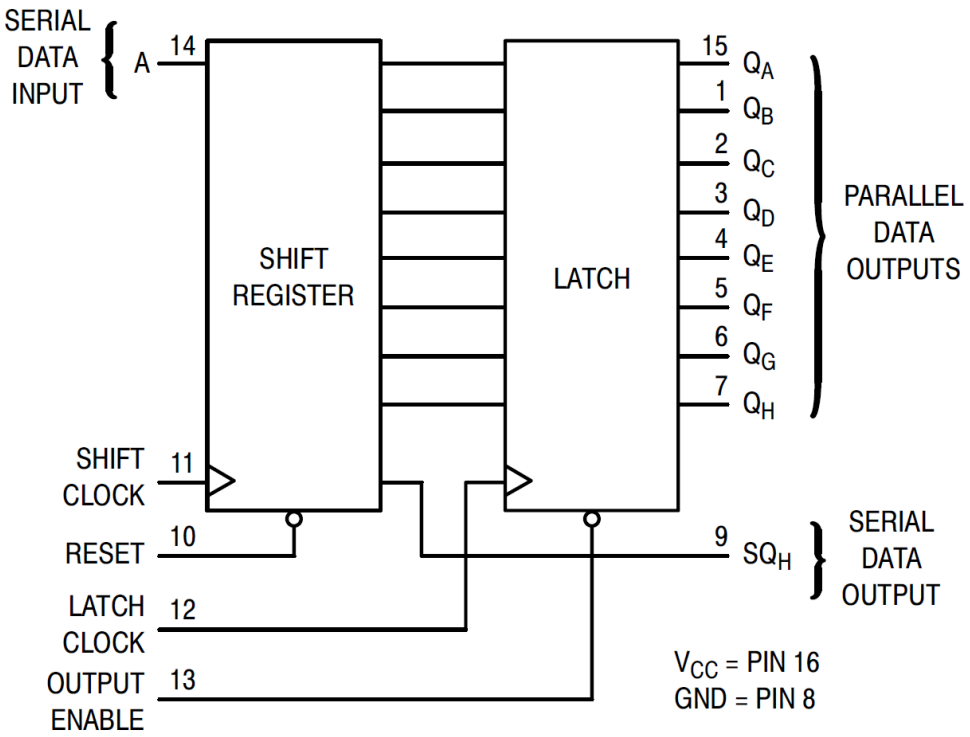


SOP-16 / TSSOP-16 / DIP-16

PIN DESCRIPTION

Pin No.			Pin Name	Pin Function
SOP-16	TSSOP-16	DIP-16		
1	1	1	Q_B	Parallel Data Q_B Output
2	2	2	Q_C	Parallel Data Q_C Output
3	3	3	Q_D	Parallel Data Q_D Output
4	4	4	Q_E	Parallel Data Q_E Output
5	5	5	Q_F	Parallel Data Q_F Output
6	6	6	Q_G	Parallel Data Q_G Output
7	7	7	Q_H	Parallel Data Q_H Output
8	8	8	GND	Ground
9	9	9	SQH	Serial Data Output
10	10	10	RESET	Shift Register Reset Input
11	11	11	SHIFT CLOCK	Shift Register Clock Input.
12	12	12	LATCH CLOCK	Parallel Latch Clock Input
13	13	13	OUTPUT ENABLE	Output Enable
14	14	14	A	Serial Data Input
15	15	15	Q_A	Parallel Data Q_A Output
16	16	16	VCC	Power Supply

BLOCK DIAGRAM



DC ELECTRICAL CHARACTERISTICS

Voltages referenced to ground.

SYMBOL	PARAMETER	TEST CONDITION		VCC	Limit			UNIT
					≤ 25°C	≤ 85°C	≤ 125°C	
V_{IH}	Minimum High-Level Input Voltage	V_{OUT} = 0.1V or V_{CC} − 0.1V $\|I_{OUT}\|$ ≤ 20 μA		2.0 V	1.50	1.50	1.50	V
				4.5 V	3.15	3.15	3.15	
				6.0 V	4.20	4.20	4.20	
V_{IL}	Maximum Low-Level Input Voltage	V_{OUT} = 0.1V or V_{CC} − 0.1V $\|I_{OUT}\|$ ≤ 20 μA		2.0 V	0.50	0.50	0.50	V
				4.5 V	1.35	1.35	1.35	
				6.0 V	1.80	1.80	1.80	
V_{OH}	Minimum High-Level Output Voltage, Q_A − Q_H	V_{IN} = V_{IH} or V_{IL}	$\|I_{OUT}\|$ ≤ 20 μA	2.0 V	1.9	1.9	1.9	V
				4.5 V	4.4	4.4	4.4	
				6.0 V	5.9	5.9	5.9	
			$\|I_{OUT}\|$ ≤ 6.0 mA	4.5 V	3.98	3.84	3.7	
			$\|I_{OUT}\|$ ≤ 7.8 mA	6.0 V	5.48	5.34	5.2	
V_{OL}	Maximum Low-Level Output Voltage, Q_A − Q_H	V_{IN} = V_{IH} or V_{IL}	$\|I_{OUT}\|$ ≤ 20 μA	2.0 V	0.1	0.1	0.1	V
				4.5 V	0.1	0.1	0.1	
				6.0 V	0.1	0.1	0.1	
			$\|I_{OUT}\|$ ≤ 6.0 mA	4.5 V	0.26	0.33	0.4	
			$\|I_{OUT}\|$ ≤ 7.8 mA	6.0 V	0.26	0.33	0.4	
V_{OH}	Minimum High-Level Output Voltage, SQ_H	V_{IN} = V_{IH} or V_{IL}	$\|I_{OUT}\|$ ≤ 20 μA	2.0 V	1.9	1.9	1.9	V
				4.5 V	4.4	4.4	4.4	
				6.0 V	5.9	5.9	5.9	
			$\|I_{OUT}\|$ ≤ 4.0 mA	4.5 V	3.98	3.84	3.7	
			$\|I_{OUT}\|$ ≤ 5.2 mA	6.0 V	5.48	5.34	5.2	
V_{OL}	Maximum Low-Level Output Voltage, SQ_H	V_{IN} = V_{IH} or V_{IL}	$\|I_{OUT}\|$ ≤ 20 μA	2.0 V	0.1	0.1	0.1	V
				4.5 V	0.1	0.1	0.1	
				6.0 V	0.1	0.1	0.1	
			$\|I_{OUT}\|$ ≤ 4.0 mA	4.5 V	0.26	0.33	0.4	
			$\|I_{OUT}\|$ ≤ 5.2 mA	6.0 V	0.26	0.33	0.4	
I_{IN}	Maximum Input Leakage Current	V_{IN} = V_{CC} or GND		6.0 V	±0.1	±1.0	±1.0	μA
I_{OZ}	Maximum Three-State Leakage Current, Q_A − Q_H	Output in High-Impedance State V_{IN} = V_{IH} or V_{IL} V_{OUT} = V_{CC} or GND		6.0 V	±0.5	±5.0	±10	μA
I_{CC}	Maximum Quiescent Supply Current (per Pacakge)	V_{IN} = V_{CC} or GND I_{OUT} = 0 μA		6.0 V	4.0	40	160	μA

AC ELECTRICAL CHARACTERISTICS

C_L = 50 pF, Input $t_r = t_f$ = 6.0 ns

SYMBOL	PARAMETER	VCC	Limit			UNIT
			≤ 25°C	≤ 85°C	≤ 125°C	
f_{max}	Maximum Clock Frequency (50% Duty Cycle) (Figures 1 and 7)	2.0 V	6.0	4.8	4.0	MHz
		4.5 V	30	24	20	
		6.0 V	35	28	24	
t_{PLH}, t_{PHL}	Maximum Propagation Delay, Shift Clock to SQ_H (Figures 1 and 7)	2.0 V	140	175	210	ns
		4.5 V	28	35	42	
		6.0 V	24	30	36	
t_{PHL}	Maximum Propagation Delay, Reset to SQ_H (Figures 2 and 7)	2.0 V	145	180	220	ns
		4.5 V	29	36	44	
		6.0 V	25	31	38	
t_{PLH}, t_{PHL}	Maximum Propagation Delay, Latch Clock to Q_A – Q_H (Figures 3 and 7)	2.0 V	140	175	210	ns
		4.5 V	28	35	42	
		6.0 V	24	30	36	
t_{PLZ}, t_{PHZ}	Maximum Propagation Delay, Output Enable to Q_A – Q_H (Figures 4 and 8)	2.0 V	150	190	225	ns
		4.5 V	30	38	45	
		6.0 V	26	33	38	
t_{PZL}, t_{PZH}	Maximum Propagation Delay, Output Enable to Q_A – Q_H (Figures 4 and 8)	2.0 V	135	170	205	ns
		4.5 V	27	34	41	
		6.0 V	23	29	35	
t_{TLH}, t_{THL}	Maximum Output Transition Time, Q_A − Q_H (Figures 3 and 7)	2.0 V	60	75	90	ns
		4.5 V	12	15	18	
		6.0 V	10	13	15	
t_{TLH}, t_{THL}	Maximum Output Transition Time, SQ_H (Figures 1 and 7)	2.0 V	75	95	110	ns
		4.5 V	15	19	22	
		6.0 V	13	16	19	
C_{IN}	Maximum Input Capacitance	–	10	10	10	pF
C_{OUT}	Maximum Three-State Output Capacitance (Output in High-Impedance State), Q_A − Q_H	–	15	15	15	pF
C_{PD}	Power Dissipation Capacitance (per Package) (Note 3)	5.0 V	300 @ 25°C			pF

Note 3. Used to determine the no-load dynamic power consumption: $P_D = C_{PD}V_{CC}^2f + I_{CC}V_{CC}$.

TIMING REQUIREMENTS

C_L = 50 pF, Input $t_r = t_f$ = 6.0 ns

SYMBOL	PARAMETER	VCC	Limit			UNIT
			≤ 25°C	≤ 85°C	≤ 125°C	
t_{su}	Minimum Setup Time, Serial Data Input A to Shift Clock (Figure 5)	2.0 V	50	65	75	MHz
		4.5 V	10	13	15	
		6.0 V	9	11	13	
t_{su}	Minimum Setup Time, Shift Clock to Latch Clock (Figure 6)	2.0 V	75	95	110	ns
		4.5 V	15	19	22	
		6.0 V	13	16	19	
t_h	Minimum Hold Time, Shift Clock to Serial Data Input A (Figure 5)	2.0 V	5	5	5	ns
		4.5 V	5	5	5	
		6.0 V	5	5	5	
t_{rec}	Minimum Recovery Time, Reset Inactive to Shift Clock (Figure 2)	2.0 V	50	65	75	ns
		4.5 V	10	13	15	
		6.0 V	9	11	13	
t_w	Minimum Pulse Width, Reset (Figure 2)	2.0 V	60	75	90	ns
		4.5 V	12	15	18	
		6.0 V	10	13	15	
t_w	Minimum Pulse Width, Shift Clock (Figure 1)	2.0 V	50	65	75	ns
		4.5 V	10	13	15	
		6.0 V	9	11	13	
t_w	Minimum Pulse Width, Latch Clock (Figure 6)	2.0 V	50	65	75	ns
		4.5 V	10	13	15	
		6.0 V	9	11	13	
t_r, t_f	Maximum Input Rise and Fall Times (Figure 1)	2.0 V	1000	1000	1000	ns
		4.5 V	500	500	500	
		6.0 V	400	400	400	

FUNCTION TABLE

Operation	Inputs					Resulting Function			
	Reset	Serial Input A	Shift Clock	Latch Clock	Output Enable	Shift Register Contents	Latch Register Contents	Serial Output SQ_H	Parallel Outputs $Q_A - Q_H$
Reset shift register	L	X	X	L, H, ↓	L	L	U	L	U
Shift data into shift register	H	D	↑	L, H, ↓	L	$D \rightarrow SR_A$; $SR_N \rightarrow SR_{N+1}$	U	$SR_G \rightarrow SR_H$	U
Shift register remains unchanged	H	X	L, H, ↓	L, H, ↓	L	U	U	U	U
Transfer shift register contents to latch register	H	X	L, H, ↓	↑	L	U	$SR_N \rightarrow LR_N$	U	SR_N
Latch register remains unchanged	X	X	X	L, H, ↓	L	*	U	*	U
Enable parallel outputs	X	X	X	X	L	*	**	*	Enabled
Force outputs into high impedance state	X	X	X	X	H	*	**	*	Z

SR: shift register contents
LR: latch register contents
X: don't care
Z: high impedance

D: data (L, H) logic level
U: remains unchanged
*: depends on Reset and Shift Clock inputs
**: depends on Latch Clock input

↑: Low-to-High
↓: High-to-Low

SWITCHING WAVEFORMS

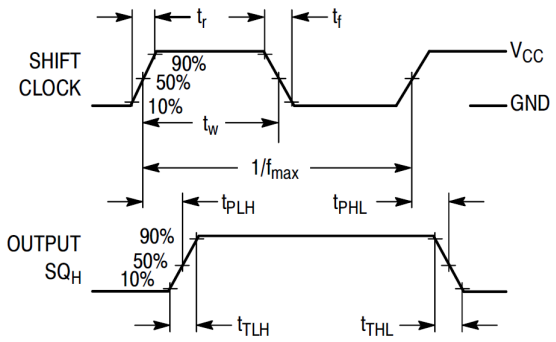


Fig. 1.

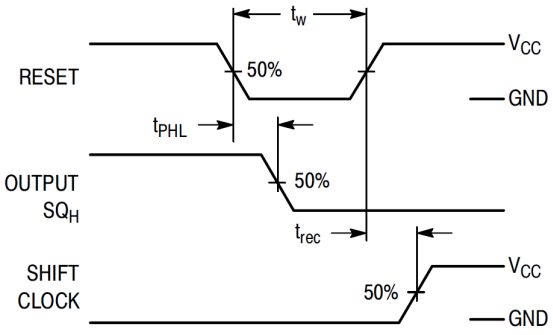


Fig. 2.

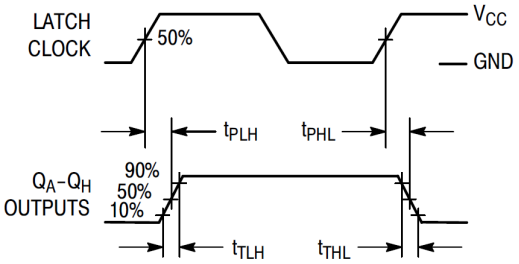


Fig. 3.

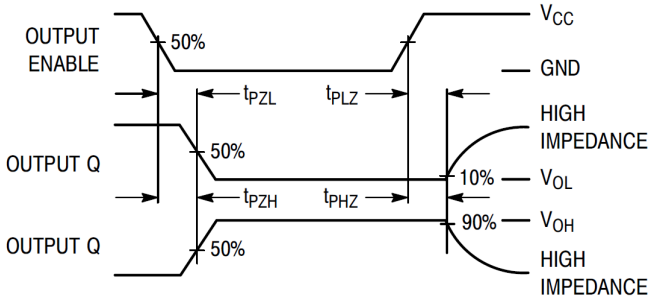


Fig. 4.

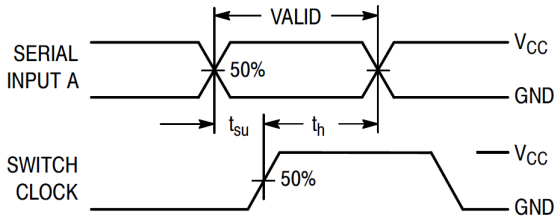


Fig. 5.

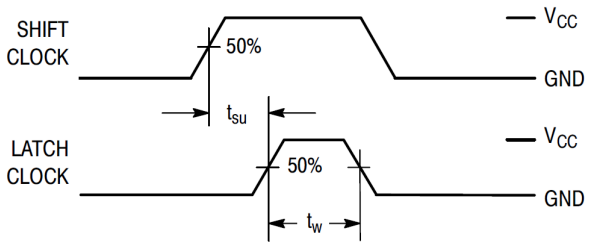
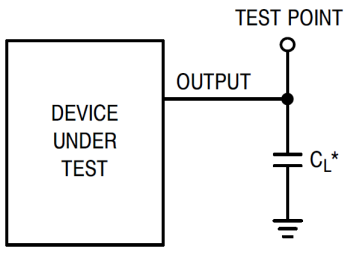


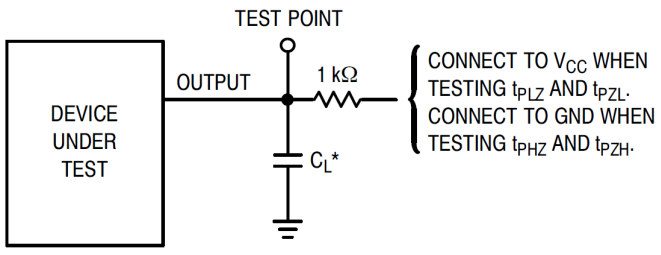
Fig. 6.

TEST CIRCUITS



*Includes all probe and jig capacitance

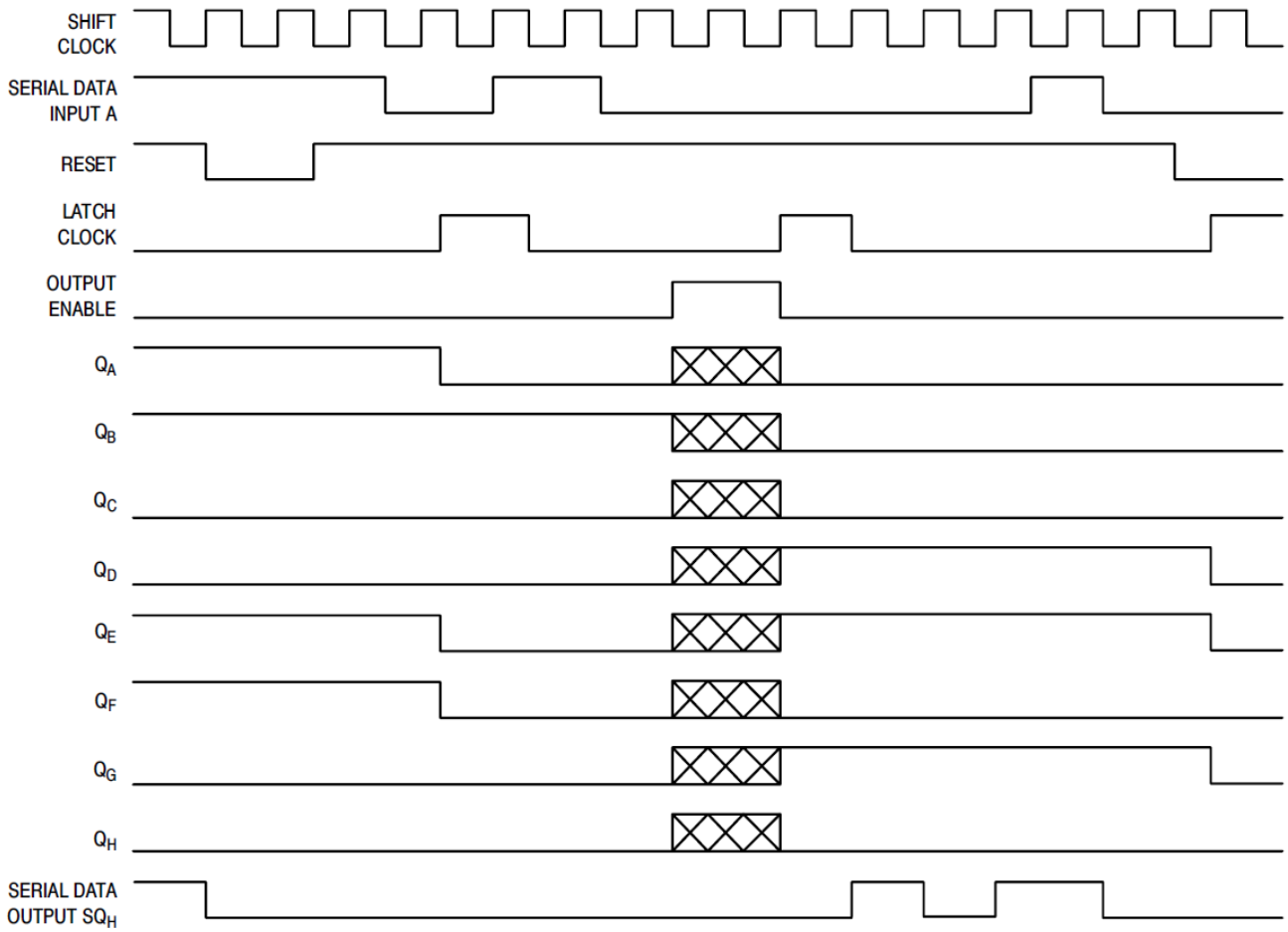
Fig. 7.



*Includes all probe and jig capacitance

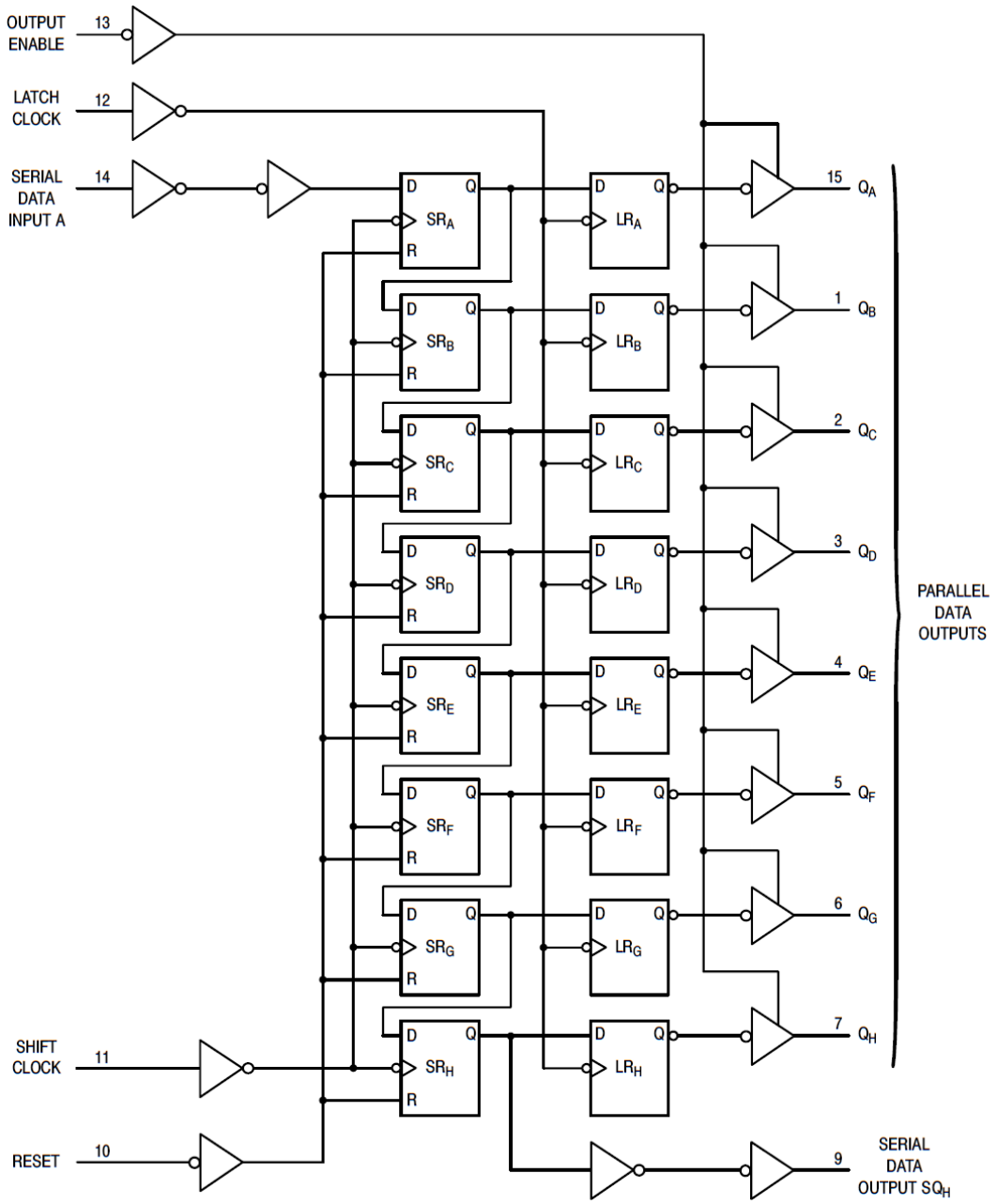
Fig. 8.

TIMING DIAGRAM



NOTE: ⊠ implies that the output is in a high–impedance state.

FUNCTIONAL LOGIC DIAGRAM



TYPICAL OPERATING CHARACTERISTICS

T.B.D.

REVISION NOTICE

The description in this datasheet is subject to change without any notice to describe its electrical characteristics properly.