Hiwonder

Servo Secondary Development Driving Circuit Instruction

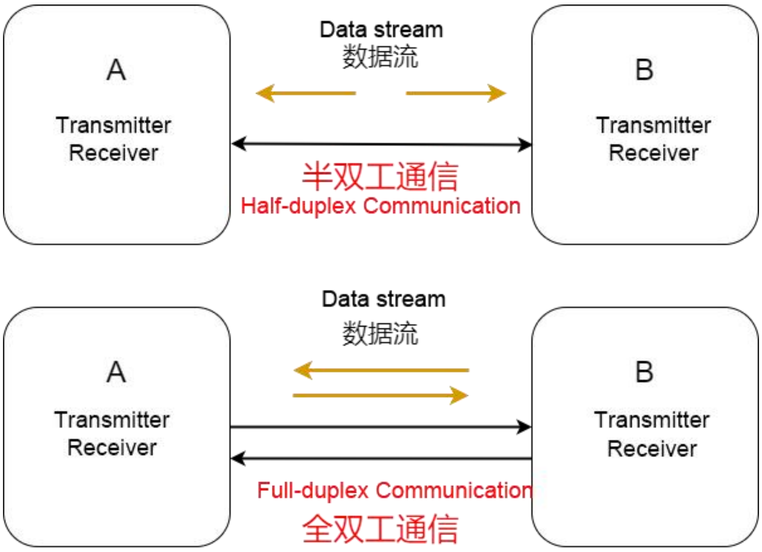
V1.0

1. Design Logic

If we want to design our own servo driver board, we need to first clarify the communication method we are going to use, which chips are required, and why these chips are necessary. Let's now delve into this information.

Bus servos use asynchronous serial bus communication, controlled uniformly through the UART asynchronous serial interface. Because the servo operates on a single signal line, our bus servo serial port adopts half-duplex asynchronous serial bus communication.

The difference between half-duplex and full-duplex is as follows:

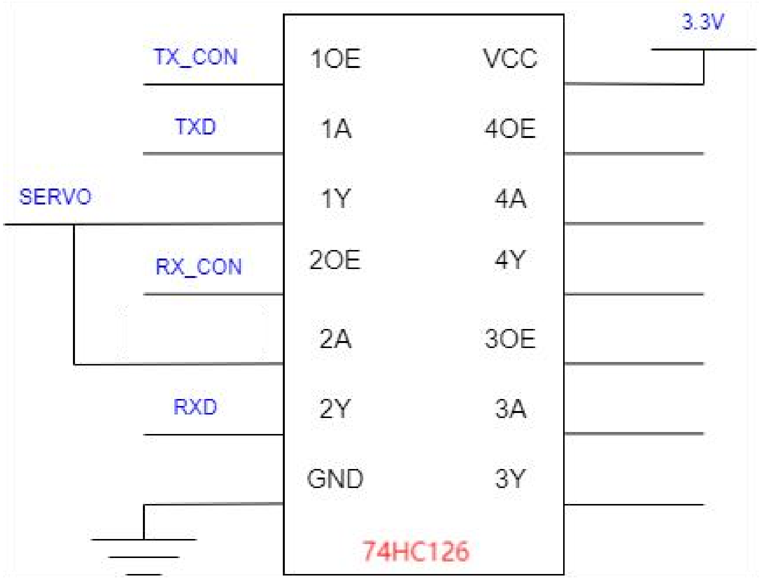


Half-duplex communication uses only one communication channel, where data travels bidirectionally on the same channel. It supports transmission from one end to the other, but the same end cannot send and receive simultaneously.

Full-duplex communication requires two independent communication channels—one for sending data and another for receiving data—allowing simultaneous bidirectional transmission.

After specifying the use of half-duplex serial communication, it is necessary for us to understand the communication protocol of bus servos. You can go to the path "**3 Servo Manual and Drawings\ Bus Servo Communication Protocol**" to view the servo communication protocol.

After understanding the bus servo communication protocol, we can now choose a suitable switch for data transmission in half-duplex communication, such as 74HC125 or 74HC126. Here, we choose 74HC126, which is a quad-channel tri-state buffer capable of controlling the transmission and reception of data on the half-duplex communication line.



In half-duplex communication, both sending and receiving data share the same communication line. To ensure correct data transmission, switching between sending and receiving is necessary, and the receiver must be disabled during transmission to prevent data collisions. Let's take a look at its user manual.

Table 3. Function selection[1]

Inputs		Output
nOE	nA	nY
H	L	L
H	H	H
L	X	Z

[1] H = HIGH voltage level; L = LOW voltage level; X = don't care; Z = high-impedance OFF-state

H (High Level): Represents the logic level in a high voltage state, typically corresponding to logic high level 1.

L (Low Level): Represents the logic level in a low voltage state, typically corresponding to logic low level 0.

X (Don't Care): Indicates that the logic level status is not of concern in specific situations, and can be either a high or low level.

Z (High Impedance): Indicates that the output is in a high impedance (OFF) state under specific conditions, meaning it does not affect the circuit. When the enable pin is disabled, the output is in a high impedance state, providing no driving force to the signal line.

Here, we can simply understand that when the OE (Output Enable) pin is high, if the input pin is high, the output pin will also be high; if the input pin is low, the output pin will also be low.

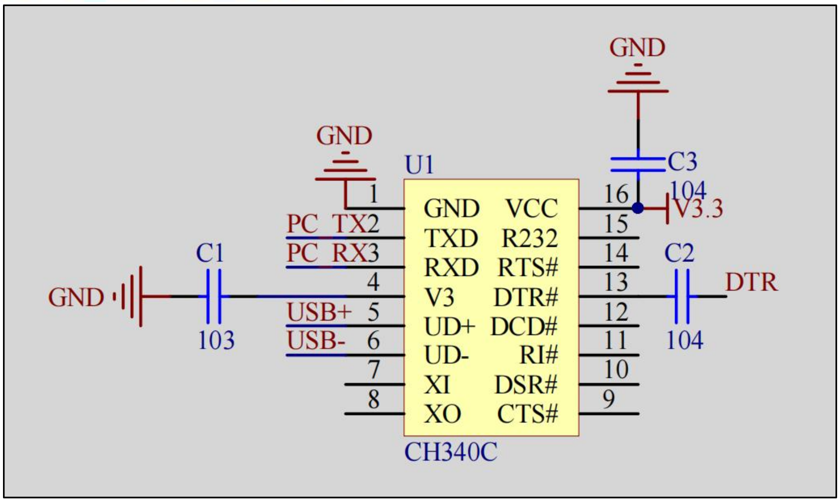
When the OE pin is low, regardless of the state of the input pin, the output pin is in a high impedance state, providing no driving force to the signal line (can be understood as floating).

High impedance output means that the digital circuit output is neither high level nor low level, but rather in a state similar to being disconnected, with the output potential determined by the subsequent circuit.

Control mode and communication process of the chip: When writing is required, pull up the TX_CON (Transmit Control) pin and pull down the RX_CON (Receive Control) pin. Connect the SERVO signal line to the servo's signal line. The signal transmitted by the serial TX (Transmitter) is directly passed to the SERVO, while the serial RX (Receiver) is effectively in a floating state and does not affect data transmission.

The same principle applies when receiving data.

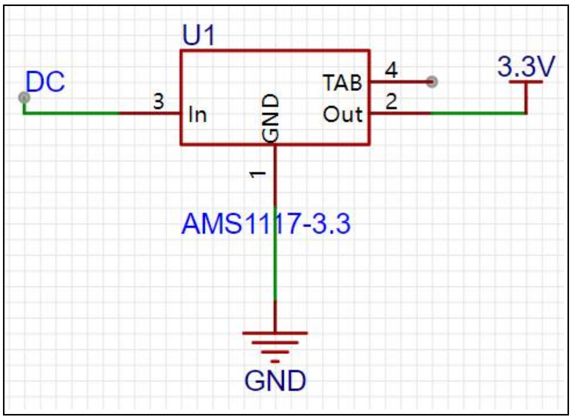
1. When connecting to the serial port, we can use a CH340 chip. Data transmission can be achieved by connecting the USB+ and USB- pins of the CH340 to USB for data transmission. At the same time, connecting the serial port pins of the 74HC126 chip enables data transmission and controls the enable state of the 74HC126.



2. Voltage and Current Requirements

Having solved the main servo control issues, now let's discuss the voltage and current requirements for the control board circuit.

1. According to the manual, the operating voltage of the chip is 3.3V. However, our bus servos operate at a minimum voltage of 7V. If we directly connect the chip, it will damage the chip. Therefore, we need a voltage regulator chip:



Here, we can regulate the DC input to stabilize at 3.3V for power supply, preventing damage to the chip. For specific details, please refer to the corresponding voltage regulator chip manual.

1. Voltage: Typically, bus servos operate around 7.4V, but there are some high-voltage bus servos that can use higher voltages, so the voltage regulator

chip's range can be increased to around 12V.

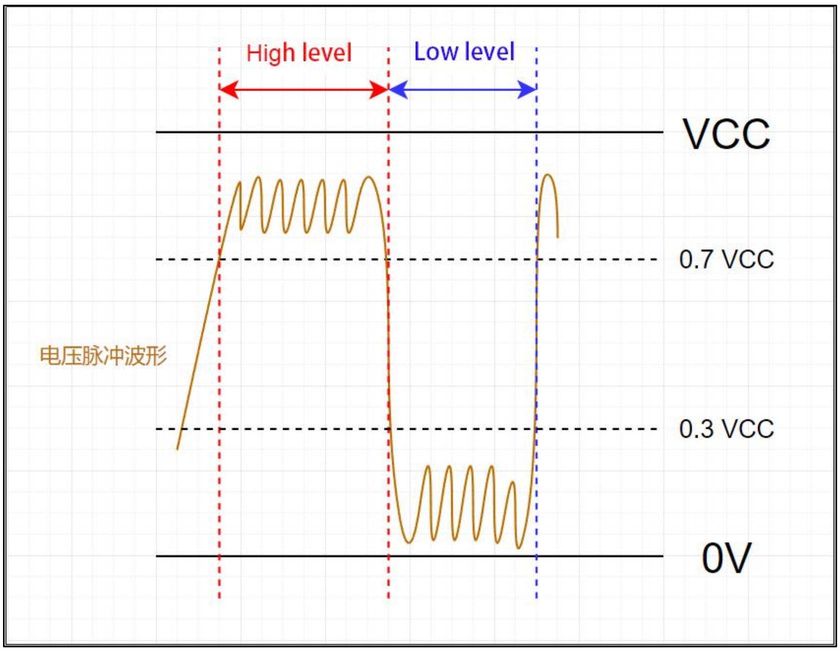
2. Current: Under normal operation, connecting a servo at idle can draw several hundred milliamps. If the current draw reaches about 1A every 10 seconds, it's advisable to use thicker wires and include a large capacitor to smooth the current and store charges. Note that this is for normal usage; if a servo becomes stalled accidentally, the stall current can be significantly higher than the idle current. When a servo's output shaft is prevented from rotating, it attempts to generate more torque to overcome resistance, leading to increased current. In such cases, it's recommended to add a fuse to protect against increased current. Specific selection should be based on the usage scenario.

3. Input Signal Range: For the input side of the 74HC126, the typical operating range is between 0V and VCC (supply voltage), which in this case is 3.3V.

4. Output Signal Range: For the output side of the 74HC126, when the enable pin is active, the output signal range typically matches the supply voltage (VCC), which is between 0V and 3.3V.

5. High Level (High Level): For input signals, if the input voltage is higher than the input threshold (usually 0.7Vcc), it is considered a high level. For output signals, when the enable pin is active, the output voltage is close to VCC and considered a high level.

6. Low Level (Low Level): For input signals, if the input voltage is lower than half of the input threshold (usually 0.3Vcc), it is considered a low level. For output signals, when the enable pin is inactive, the output voltage is close to ground (0V) and considered a low level.



7. If conditions allow, you can add a surge protection circuit to prevent surges from damaging the circuit and equipment.

3. Notices

1. A servo typically requires 1A of current to operate normally and can draw up to 3A during stall conditions. Therefore, as a general rule, design the power supply for each servo with a capacity of 1A. For example, powering 10 servos would require a 10A power supply, and so on, calculating the current accordingly (excluding extreme stall conditions).

2. What should be done to account for stall conditions? Generally, adding a current recovery of 1A to 2A at the power supply connection of each servo should be sufficient.

3. Self-designed circuit boards require capacitors of 1000uF or more for power filtering. Use short and thick wires for power supply lines to prevent circuit boards or servos from restarting or experiencing servo jitter.

4. During the servo power supply process, the current carrying capacity of

the power lines may only be a few hundred milliamps. When connecting power lines for supply, it's important to consider the current carrying capacity of the power lines, which can be calculated based on factors such as conductor cross-sectional area, length, material, and temperature. Specific calculation methods and measurement techniques can be learned from the referenced materials.

5. Due to the high torque of our servos and the need for high driving power, it's recommended to use a rated voltage for power supply. This avoids situations where a voltage is too low to control the servo to the target angle or too high, potentially damaging circuit components or the servo itself.