Serial control servo

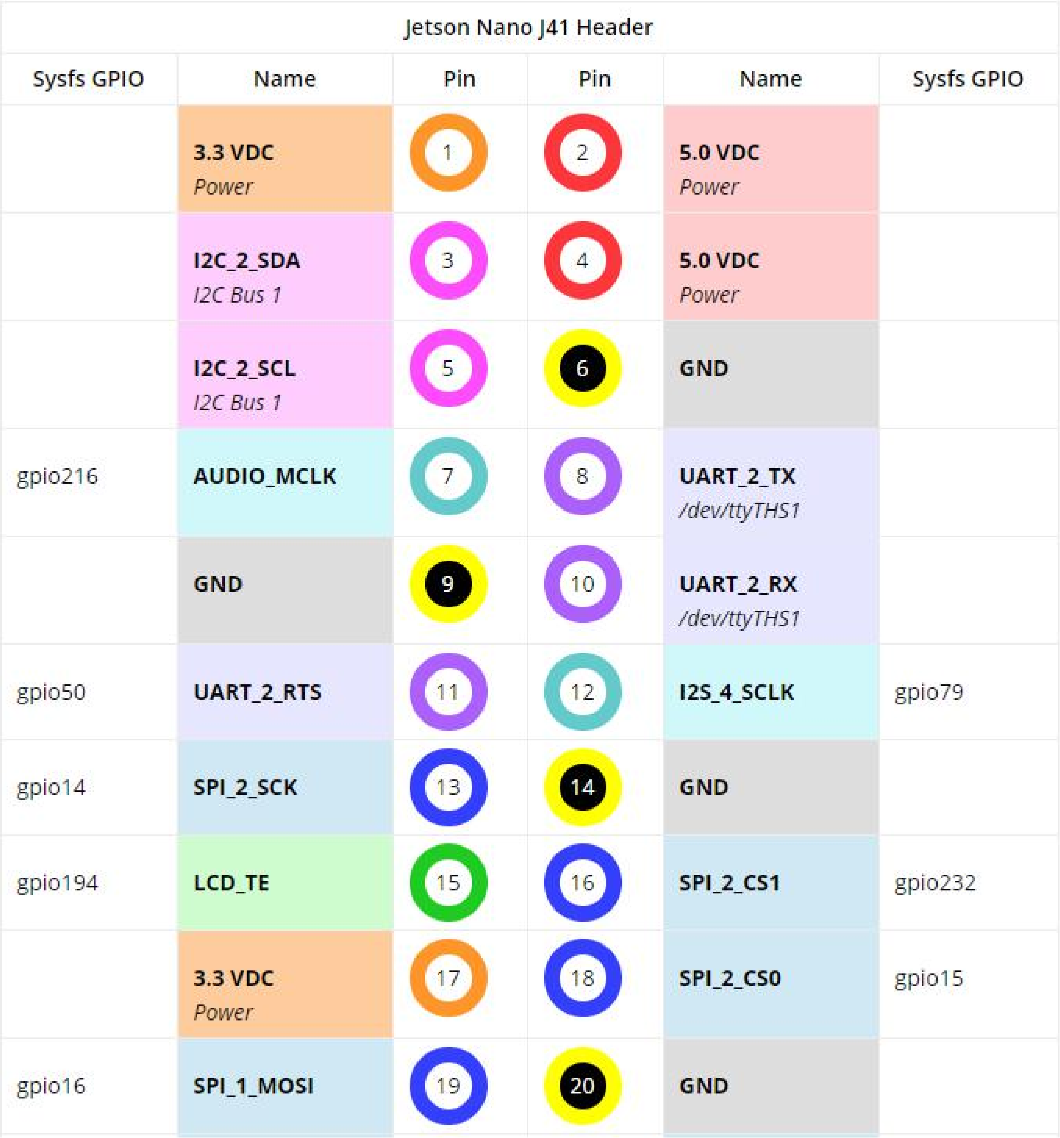
1. Learning goals

In this course, we mainly learn to use Jetson NANO and 16-channel servo debugging board to control the servo through IIC.

2. Preparation

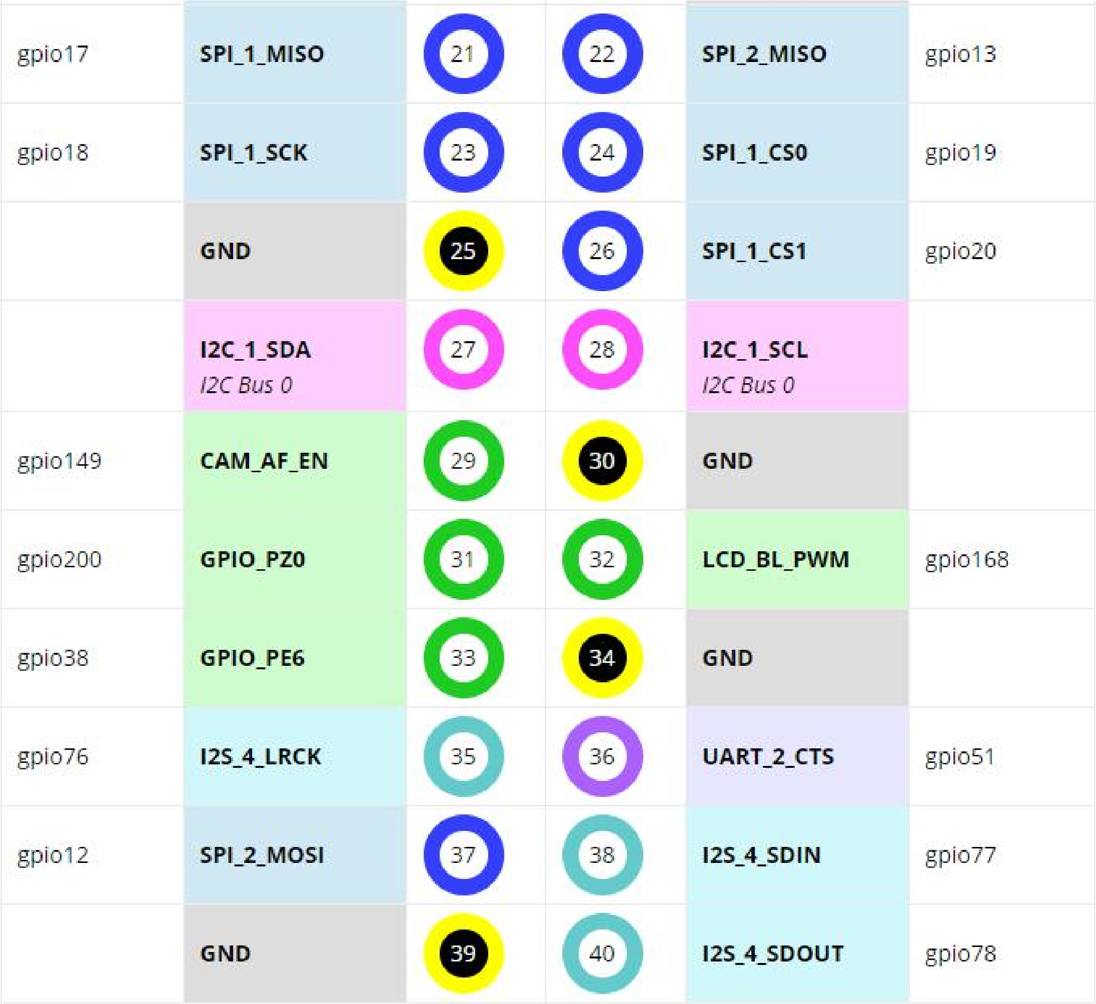
Connect the TX and RX of the module to the Pin10 and Pin8 pins of the Jetson NANO board. VCC and GND are connected to 3.3V and GND of Jetson NANO respectively.

Jetson NANO needs to open the serial port permission.



Jetson Nano J41 Header

Sysfs GPIO	Name	Pin	Pin	Name	Sysfs GPIO
	3.3 VDC *Power*	1	2	**5.0 VDC** *Power*	
	I2C_2_SDA *I2C Bus 1*	3	4	**5.0 VDC** *Power*	
	I2C_2_SCL *I2C Bus 1*	5	6	**GND**	
gpio216	**AUDIO_MCLK**	7	8	**UART_2_TX** */dev/ttyTHS1*	
	GND	9	10	**UART_2_RX** */dev/ttyTHS1*	
gpio50	**UART_2_RTS**	11	12	**I2S_4_SCLK**	gpio79
gpio14	**SPI_2_SCK**	13	14	**GND**	
gpio194	**LCD_TE**	15	16	**SPI_2_CS1**	gpio232
	3.3 VDC *Power*	17	18	**SPI_2_CS0**	gpio15
gpio16	**SPI_1_MOSI**	19	20	**GND**	



gpio17	SPI_1_MISO	21	22	SPI_2_MISO	gpio13
gpio18	SPI_1_SCK	23	24	SPI_1_CS0	gpio19
	GND	25	26	SPI_1_CS1	gpio20
	I2C_1_SDA *I2C Bus 0*	27	28	I2C_1_SCL *I2C Bus 0*	
gpio149	CAM_AF_EN	29	30	GND	
gpio200	GPIO_PZ0	31	32	LCD_BL_PWM	gpio168
gpio38	GPIO_PE6	33	34	GND	
gpio76	I2S_4_LRCK	35	36	UART_2_CTS	gpio51
gpio12	SPI_2_MOSI	37	38	I2S_4_SDIN	gpio77
	GND	39	40	I2S_4_SDOUT	gpio78

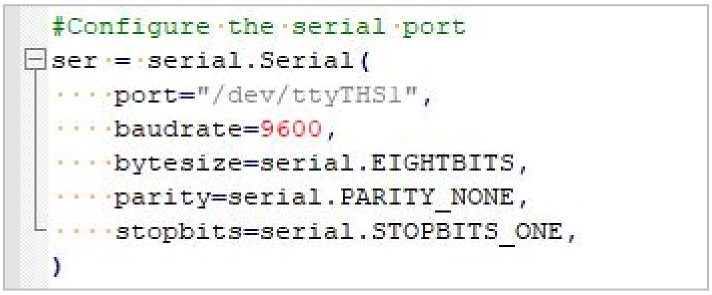
Input following command to open the serial port permission.

sudo chmod 777 /dev/ttyTHS1

3. Code

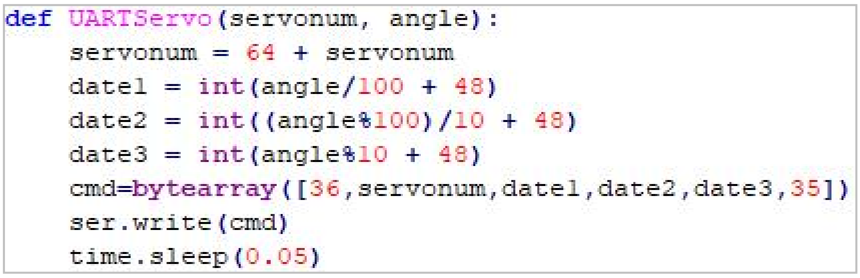
About code, please view **16CServo-uart.py** file.

3.1 Define the device address of the module



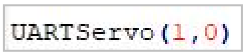
```
#Configure the serial port
ser = serial.Serial(
    port="/dev/ttyTHS1",
    baudrate=9600,
    bytesize=serial.EIGHTBITS,
    parity=serial.PARITY_NONE,
    stopbits=serial.STOPBITS_ONE,
)
```

3.2 The serial port controls the servo function.



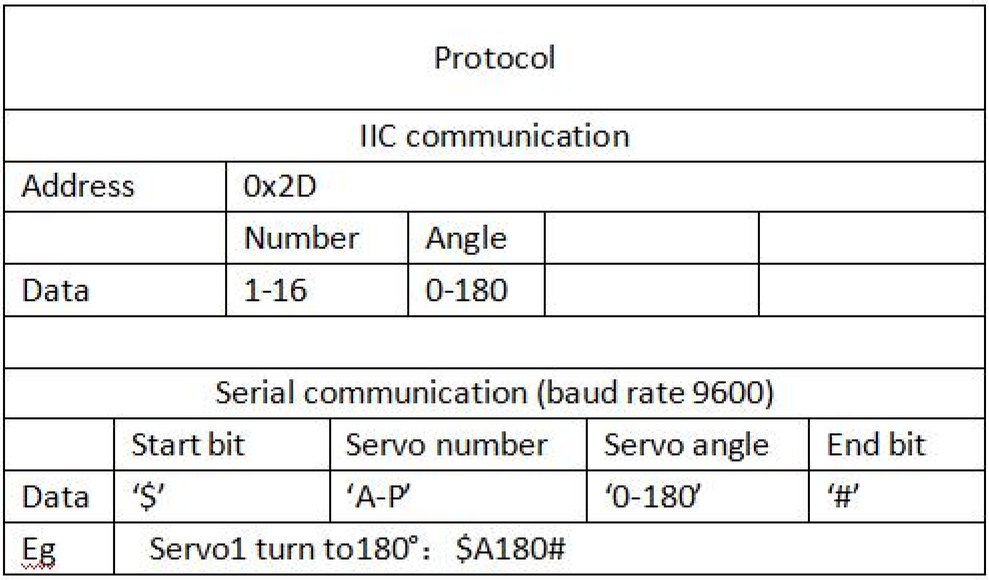
```
def UARTServo(servonum, angle):
    servonum = 64 + servonum
    date1 = int(angle/100 + 48)
    date2 = int((angle%100)/10 + 48)
    date3 = int(angle%10 + 48)
    cmd=bytearray([36,servonum,date1,date2,date3,35])
    ser.write(cmd)
    time.sleep(0.05)
```

3.3 Set the angle of the servo S1 to 0



```
UARTServo(1,0)
```

4. Module protocol



Protocol				
IIC communication				
Address	0x2D			
	Number	Angle		
Data	1-16	0-180		
Serial communication (baud rate 9600)				
	Start bit	Servo number	Servo angle	End bit
Data	‘$’	‘A-P’	‘0-180’	‘#’
Eg	Servo1 turn to180°: $A180#			

5. Running code

Input following command in command terminal of jetson nano.

python3 16CServo-uart.py

6. Experimental phenomena

After the program is run successfully. The servo will rotate 0°, after 2s it will rotate 180°.