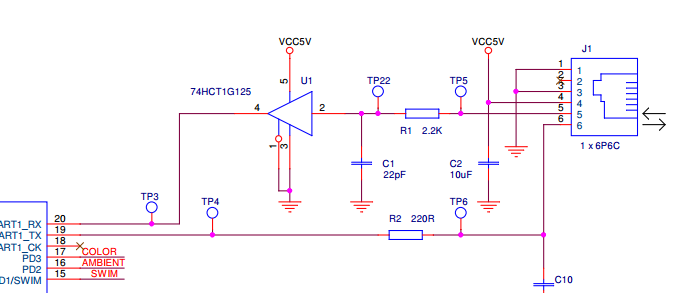
1. Lego color sensor 45506



1. 电路



Pin1 ---- gnd ---- 白色

Pin2 ---- NC ---- 黑色

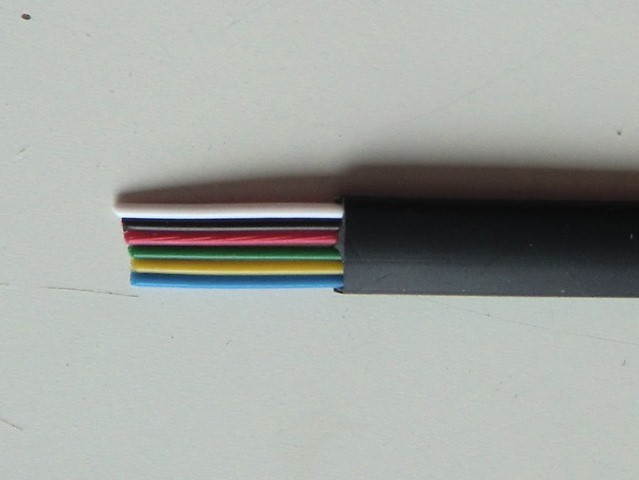
Pin3 ---- GND ---- 红色

Pin4 ---- VCC 5V ---- 绿色

Pin5 ---- RX ---- 黄色

Pin6 ---- TX ---- 蓝色





1. 直接和usb转串口连接

为简单起见，

Color sensor wire usb serial

Pin1 (gnd) 白色 gnd

Pin2 (NC)

Pin3 (NC)

Pin4 (VCC 5V) 绿色 5v

Pin5 (RX) 黄色 tx

Pin6 (TX) 蓝色 rx

注意： RX和TX必须交叉连接

波特率使用2400， 获得sensor数据如下：

|  |
| --- |
| 0x00 0x00 0x40 0x1D 0xA2 0x49 0x05 0x02 0xB1 0x52 0x00 0xE1 0x00 0x00 0x4C 0x9D 0x00  0x10 0x43 0x4F 0x4C 0x2D 0x43 0x41 0x4C 0x00 0x41 0x9D 0x01 0x00 0x00 0x00 0x00 0x00  0x20 0xFF 0x7F 0x47 0xA4 0x9D 0x03 0x00 0x00 0x00 0x00 0x00 0xFF 0x7F 0x47 0xA6 0x95  0x30 0x80 0x04 0x01 0x05 0x00 0xEA 0x9C 0x00 0x52 0x47 0x42 0x2D 0x52 0x41 0x57 0x00  0x40 0x5D 0x9C 0x01 0x00 0x00 0x00 0x00 0x00 0x0C 0x7F 0x44 0x55 0x9C 0x03 0x00 0x00  0x50 0x00 0x00 0x00 0x0C 0x7F 0x44 0x57 0x94 0x80 0x03 0x01 0x04 0x00 0xED 0x9B 0x00  0x60 0x52 0x45 0x46 0x2D 0x52 0x41 0x57 0x00 0x5C 0x9B 0x01 0x00 0x00 0x00 0x00 0x00  0x70 0x0C 0x7F 0x44 0x52 0x9B 0x03 0x00 0x00 0x00 0x00 0x00 0x0C 0x7F 0x44 0x50 0x93  0x80 0x80 0x02 0x01 0x04 0x00 0xEB 0xA2 0x00 0x43 0x4F 0x4C 0x2D 0x43 0x4F 0x4C 0x4F  0x90 0x52 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x6D 0x9A 0x01 0x00 0x00 0x00 0x00 0x00  0xa0 0x00 0x00 0x41 0x25 0x9A 0x03 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x41 0x27 0x9A  0xb0 0x04 0x63 0x6F 0x6C 0x00 0x00 0x00 0x00 0x00 0x01 0x92 0x80 0x01 0x00 0x02 0x00  0xc0 0xEE 0xA1 0x00 0x43 0x4F 0x4C 0x2D 0x41 0x4D 0x42 0x49 0x45 0x4E 0x54 0x00 0x00  0xd0 0x00 0x00 0x00 0x6B 0x99 0x01 0x00 0x00 0x00 0x00 0x00 0x00 0xC8 0x42 0xED 0x99  0xe0 0x03 0x00 0x00 0x00 0x00 0x00 0x00 0xC8 0x42 0xEF 0x99 0x04 0x70 0x63 0x74 0x00  0xf0 0x00 0x00 0x00 0x00 0x05 0x91 0x80 0x01 0x00 0x03 0x00 0xEC 0xA0 0x00 0x43 0x4F  0x100 0x4C 0x2D 0x52 0x45 0x46 0x4C 0x45 0x43 0x54 0x00 0x00 0x00 0x00 0x00 0x7D 0x98  0x110 0x01 0x00 0x00 0x00 0x00 0x00 0x00 0xC8 0x42 0xEC 0x98 0x03 0x00 0x00 0x00 0x00  0x120 0x00 0x00 0xC8 0x42 0xEE 0x98 0x04 0x70 0x63 0x74 0x00 0x00 0x00 0x00 0x00 0x04  0x130 0x90 0x80 0x01 0x00 0x03 0x00 0xED 0x04 |

1. Sensor 消息通信协议

消息分为四种类型： 系统消息，命令消息，数据消息，信息消息

一个字节8位，最高两位代表消息类型

| **XX** | **Description** |
| --- | --- |
| 00 | System message |
| 01 | Command message |
| 10 | Info message |
| 11 | Data message |

1. 系统消息

0x00 --> 0b 00 00 00 00 , 结构为：0bXXLLLYYY， 现在XX = 00 ， 所以这个消息是系统消息， LLL为bit3-bit5， 代表payload的长度，也就是2lll。 低三位YYY代表消息类型,所以根据下表，000表示此消息为SYNC

所以0x00表示此消息为系统消息， SYNC，payload=0

**System Messages**

| **Message Byte** | **Description** |
| --- | --- |
| 0b00000000 | SYNC |
| 0b00000010 | NACK |
| 0b00000100 | ACK |
| 0b00LLL110 | ESC (reserved for future use) |

系统消息只有一个字节

1. 命令消息

|  |  |  |
| --- | --- | --- |
| **Message Byte** | Payload | **Description** |
| 0b01000000 | T | TYPE: sensor type T is the type of the sensor (0-255) |
| 0b01001001 | M, V | MODES:  sensor modes  M+1 is the number of modes supported (1-8)  V+1 is the number of modes to be shown (1-M) |
| 0b01010010 | SSSS | SPEED: maximum sensor baud rate SSSS (32 bit integer) is the maximum baud rate supported by the sensor |
| 0b01000011 | M | SELECT: change sensor mode M specifies the desired sensor mode (0-7) |
| 0b01LLL100 | <data> | WRITE: Send data to the sensor <data> consists of exactly 20bLLL bytes |

0x40 ---> 0b 01 00 00 00 , sensor type, 结构为：0bXXLLLYYY， 现在XX = 01, 所以这个消息是命令消息，参照上表， LLL = 000， 所以payload为20 = 1，接下来的字节是0x1d(29)，所以是一个color sensor，





接下来的字节是0xa2，实际上command message有payload+2个字节，最后一个字节是checksum， 也就是0xa2是checksum

注：checksum计算(0x40 xor 0x1d xor 0xff)

0x49 ---> 0b01 00 10 01 , sensor modes, 结构为：0bXXLLLYYY， 现在XX = 01, 所以这个消息是命令消息，modes， LLL = 001, 所以payload为21 = 2，接下来的字节是0x05， 0x02， checksum为0xb1（0x49 xor 0x05 xor 0x02 xor 0xff）。 0x05 为M ， 0x02 为V,

0x52---> 0b01 01 00 10, sensor speed, 结构为：0bXXLLLYYY， 现在XX = 01, 所以这个消息是命令消息，spead, LLL = 010, 所以payload为22 = 4，接下来的字节是0x00 0xE1 0x00 0x00， 代表支持的最大波特率， checksum为0x4C(0x52 xor 0x00 xor 0xe1 xor 0x00 xor 0x00 xor 0xff)

1. 信息消息

|  |  |  |  |
| --- | --- | --- | --- |
| **Message Byte** | **Info Byte** | **Payload** | **Description** |
| 0b10LLLMMM | 0 | string | NAME: name of mode 0bMMM  <string> is an ASCII string, padded with zeros to a length of exactly 20bLLL characters |
| 0b10011MMM | 1 | LLLL, HHHH | RAW: range of raw sensor readings LLLL (32bit float) is the lowest raw value the sensor returns HHHH (32 bit float) is the highest raw value the sensor returns |
| 0b10011MMM | 2 |  |  |
| 0b10011MMM | 3 | LLLL, HHHH | SI: range of readings in SI units LLLL (32bit float) value in SI units corresponding to the lowerst raw value HHHH (32 bit float) value in SI units corresponding to the hightest raw value |
| 0b10LLLMMM | 4 |  |  |
| 0b10010MMM | 0x80 | S, T, F, D | FORMAT: format of the sensor data in mode 0bMMM S: the number of items (at least 1) T: the data type of the items (see table) F: the number of digits to show (0-15, including decimals and the decimal point) D: the number of decimals to show (0-15) |

0x9D ---> 0b10011101，结构为：0bXXLLLYYY， 现在XX = 10， 所以这个消息是信息消息, MMM = 0b101, 此sensor的model为5，下一个字节是0x00， 根据上表， payload代表string，LLL = 011, 所以payload为23 = 8，所以接下来8个字节为0x43 0x4F 0x4C 0x2D 0x43 0x41 0x4C 0x00（COL-CAL,最后一个0x00为字节对齐）.checksum为0x41(0x9d xor 0x00 xor 0x43 xor 0x4F xor 0x4C xor 0x2D xor 0x43 xor 0x41 xor 0x4C xor 0x00 xor 0xff)

0x9D ---> 0b10011101，结构为：0bXXLLLYYY， 现在XX = 10， 所以这个消息是信息消息, MMM = 0b101, 此sensor的model为5，下一个字节是0x01， 根据上表，payload代表范围， LLL = 011, 所以payload为23 = 8，所以接下来8个字节为0x00 0x00 0x00 0x00 0x00 0xFF 0x7F 0x47， checksum为0xa4（0x9d xor 0x01 xor 0x00 xor 0x00 xor 0x00 xor 0x00 xor 0x00 xor 0xFF xor 0x7F xor 0x47 xor 0xff）

0x9D ---> 0b10011101，结构为：0bXXLLLYYY， 现在XX = 10， 所以这个消息是信息消息, MMM = 0b101, 此sensor的model为5，下一个字节是0x03， 根据上表，payload代表SI， LLL = 011, 所以payload为23 = 8，所以接下来8个字节为0x00 0x00 0x00 0x00 0x00 0xFF 0x7F 0x47， checksum为0xa6（0x9d xor 0x03 xor 0x00 xor 0x00 xor 0x00 xor 0x00 xor 0x00 xor 0xFF xor 0x7F xor 0x47 xor 0xff）

0x95 ---> 0b10010101，结构为：0bXXLLLYYY， 现在XX = 10， 所以这个消息是信息消息, MMM = 0b101, 此sensor的model为5，下一个字节是0x80， 根据上表，根据上表，payload代表S，T，F，D，LLL = 0101, 所以payload为22 = 4，所以接下来4个字节为0x04 0x01 0x05 0x00， checksum为0xEA（0x95 xor 0x80 xor 0x04 xor 0x01 xor 0x05 xor 0x00 xor 0xff）