

JPEG 2M Pixel Color Camera Serial Interface(TTL level)

From LinkSprite Playground

Contents

- 1 Description
- 2 Package List
- 3 Specification
- 4 Application Idea
- 5 Usage
 - 5.1 Hardware Installation
 - 5.2 Programming
- 6 Support
- 7 Resources
- 8 How to buy

Description

LS-Y201-2MP is LinkSprite's new generation high resolution serial port camera module. Its resolution is 2 million pixel. It can captures high resolution pictures using the serial port. LS-Y201-2MP is a modular design that outputs JPEG images through UART, and can be easily integrated into existing design.

Package List

- 1 X [CAM_JPEG_2M_TTLUART][201201004]
- 1 X [IT_CAM_IR_CAB][AA101103]

model:[CAM_JPEG_2M_TTLUART][201201004]



Cable included:[IT_CAM_IR_CAB][AA101103]



Specification

- 2 Million Pixels
- Support capture JPEG from serial port
- Default baud rate of serial port is 115200
- DC 5V power supply
- Size 32mm X 32mm
- Current consumption: 80-100mA

Sensor	2M cmos
Pixel Size	5.6um*5.6um
Shutter Type	Electronic Rolling Shutte
SNrmax	45DB
Dynamic Range	60DB
Maximum Frame Rate	16000*1200 15fps
rs-232 baud rate	MAX 115200
Power consumption	120MA
Power Supply	DC +5 V (+/-5%)
Operating Temperature	-20 ~ +60℃

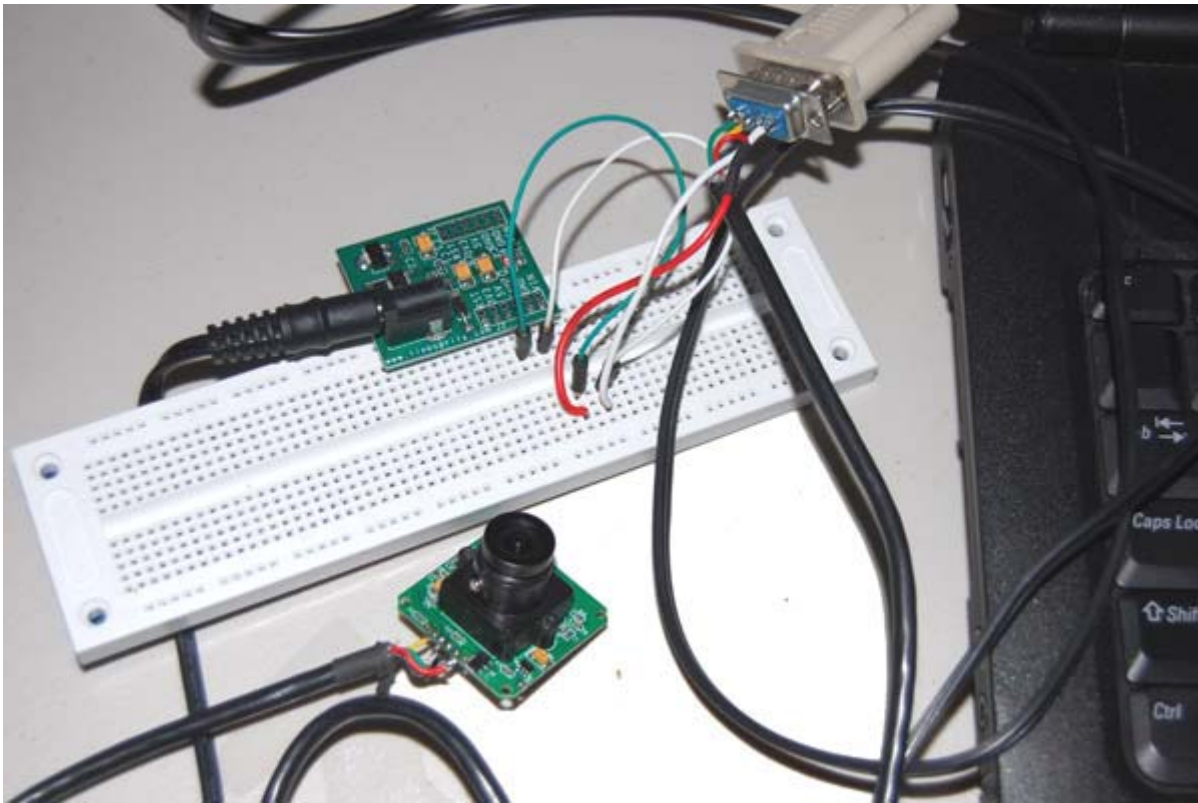
Application Idea

- Different image capture systems
- Environmental monitoring
- Industry monitoring
- Medical equipment
- Video phone
- Security
- Vehicle based GPS

Usage

Hardware Installation

Evaluation Setup



Programming



- Step 1: Click Open COM
- Step 2: Click Single Shot.

Arduino sample code provided

```

/* Linksprite */
#include <NewSoftSerial.h>

byte incomingbyte;
NewSoftSerial mySerial(4,5);                                     //Configure pin 4 and 5 as soft serial port

```

```

int a=0x0000,j=0,k=0,count=0;                                //Read Starting address
uint8_t MH,ML;
boolean EndFlag=0;

void SendResetCmd();
void SendTakePhotoCmd();
void SendReadDataCmd();
void StopTakePhotoCmd();

void setup()
{
    Serial.begin(19200);
    mySerial.begin(38400);
}

void loop()
{
    SendResetCmd();
    delay(4000);                                              //After reset, wait 2-3 second to send take picture command

    SendTakePhotoCmd();

    while(mySerial.available()>0)
    {
        incomingbyte=mySerial.read();

    }
    byte a[32];

    while(!EndFlag)
    {
        j=0;
        k=0;
        count=0;
        SendReadDataCmd();

        delay(25);
        while(mySerial.available()>0)
        {
            incomingbyte=mySerial.read();
            k++;
            if((k>5)&&(j<32)&&(!EndFlag))
            {
                a[j]=incomingbyte;
                if((a[j-1]==0xFF)&&(a[j]==0xD9))    //Check if the picture is over
                {
                    EndFlag=1;
                    j++;
                    count++;
                }
            }

            for(j=0;j<count;j++)
            {
                if(a[j]<0x10)
                {
                    Serial.print("0");
                    Serial.print(a[j],HEX);
                    Serial.print(" ");
                }
                Serial.println();
            }
        }
        while(1);
    }

    //Send Reset command
    void SendResetCmd()
    {
        mySerial.print(0x56, BYTE);
        mySerial.print(0x00, BYTE);
        mySerial.print(0x26, BYTE);
        mySerial.print(0x00, BYTE);
    }

    //Send take picture command
    void SendTakePhotoCmd()
    {
        mySerial.print(0x56, BYTE);
        mySerial.print(0x00, BYTE);
        mySerial.print(0x36, BYTE);
        mySerial.print(0x01, BYTE);
        mySerial.print(0x00, BYTE);
    }

    //Read data
    void SendReadDataCmd()
    {
        MH=a/0x100;
        ML=a%0x100;
        mySerial.print(0x56, BYTE);
    }
}

```

```

mySerial.print(0x00, BYTE);
mySerial.print(0x32, BYTE);
mySerial.print(0x0c, BYTE);
mySerial.print(0x00, BYTE);
mySerial.print(0x0a, BYTE);
mySerial.print(0x00, BYTE);
mySerial.print(0x00, BYTE);
mySerial.print(MH, BYTE);
mySerial.print(ML, BYTE);
mySerial.print(0x00, BYTE);
mySerial.print(0x00, BYTE);
mySerial.print(0x00, BYTE);
mySerial.print(0x20, BYTE);
mySerial.print(0x00, BYTE);
mySerial.print(0x0a, BYTE);
a+=0x20;                                     //address increases 32, set according to buffer size
}

void StopTakePhotoCmd()
{
    mySerial.print(0x56, BYTE);
    mySerial.print(0x00, BYTE);
    mySerial.print(0x36, BYTE);
    mySerial.print(0x01, BYTE);
    mySerial.print(0x03, BYTE);
}

```

Support

If you have questions or other better design ideas, you can go to our forum (<http://www.linksprite.com/forum/index.php>) to discuss or creat a ticket for your issue at linksprite support (<http://www.linksprite.com/support/>).

Resources

- User Manual (<http://learn.linksprite.com/?p=2571>)
- Evaluation Software (https://s3.amazonaws.com/linksprite/camera/JPEG_camera_uartinterface_TTL/LS-Y201-2MP.exe)
- Linksprite Open Source Code for X86 (https://s3.amazonaws.com/linksprite/camera/JPEG_camera_uartinterface_TTL/LSY201.rar)
- .Net micro framework source code (<http://www.fezzzer.com/project/220/linksprite-camera-library/>)
- Arduino Tutorial (<http://learn.linksprite.com/jpeg-camera/tutorial-of-using-linksprite-2mp-uart-jpeg-camera-with-arduino/>)

How to buy

Here to buy on LinkSprite Cart (<http://www.linkspritedirect.com/index.php>)

Retrieved from "[http://linksprite.com/wiki/index.php5?title=JPEG_2M_Pixel_Color_Camera_Serial_Interface\(TTL_level\)&oldid=5371](http://linksprite.com/wiki/index.php5?title=JPEG_2M_Pixel_Color_Camera_Serial_Interface(TTL_level)&oldid=5371)"

-
- This page was last modified on 13 August 2014, at 03:53.
 - This page has been accessed 24,757 times.