Chipsmall
深圳市津利帝科技有限公司
Geelead Technology Limited
Chipsmall
采购部

Flymaple V1.1(SKU:DFR0188)

Contents

Introduction

Flymaple is a Quadcopter controller board, based on the Maple Project. The FlyMaple embeds a STM32F103RET6 (ARM Cortex-M3) as it's main MCU. It integrates a 3-Axis accelerometer,a 3-Axis gyroscope,a 3-Axis compass and a barometric pressure sensor. By using the 10 dgrees of freedom IMUs and powerful 72MHz high-performation microcontroller, flymaple is powerful enough to handle a complex algorithm and manage a robot motion system at the same time.

The design of flymaple is based on the maple, which is an arduino sytle ARM processor. The Maple IDE will make Processing/Arduino programmers feel right at home. If you're familiar with Arduino then you can get started with the Maple IDE quite easily.

Flymaple is aimed to run on the balancing robots,mobile platform,helicopters and quadcopters which require IMUs and high performance real time controllers. It has an amazing scalability, because it's compatible with the Arduino shields. FlyMaple extends 6 PWM channels and 8 channels for GPIO for controlling ESC/Servo and capturing RC receiver input.

Applications

- Aircraft
- Balancing robots
- Indoor inertial navigation
- Altimeter

Specification

- Working Voltage: 5v
- 7~12v External Power supply from the 2p JST connector
- Output Voltage: 3.3v/5v or External input power
- Microcontroller: STM32F103
- Running at 72Mhz with 32bit Arduino sytle ARM processor(Cortex-M3)
- Bootloader: Leaflabs Maple RET6 Edition

- 10 Degrees of Freedom on a single, flat board:
  ITG-3200 - triple-axis digital-output gyroscope
  ADXL345 - 13-bit resolution, ¡À16g, triple-axis accelerometer
  HMC5883L - triple-axis, digital magnetometer
  BMP085 - high-precision barometric pressure sensor

- Programable through an Arduino-based development environment - Maple IDE
- Compatible with most of Arduino shields
- Extends 6 channels PWM pins for controlling ESC/Servo
- Extends 8 channels GPIO for capturing RC receiver output

- 3x 4p JST connectors:
  Serial port 1
  GPS extension port
  I2C interface

- Size: 50x50x12mm
- Weight: 15g

Windows System Driver

- Plug the Maple board into the computer
- Open Device Manager
- Switch Maple board into continuous Bootloader mode (hit reset until you get the fast blinks followed by slower blinks;and immediately hit and hold the boot button until it changes into slow blink mode,release).
- It should show some broken USB device in your Devices, not a com-port.
- Download the Maple IDE for windows(http://static.leaflabs.com/pub/leaflabs/maple-ide/maple-ide-0.0.12-windowsxp32.zip),and extract somewhere.
- Select manually select the driver location, select Maple IDE directory "maple-ide-0.0.12-windowsxp32/drivers/mapleDrv / dfu directory, to install the driver.

Note:

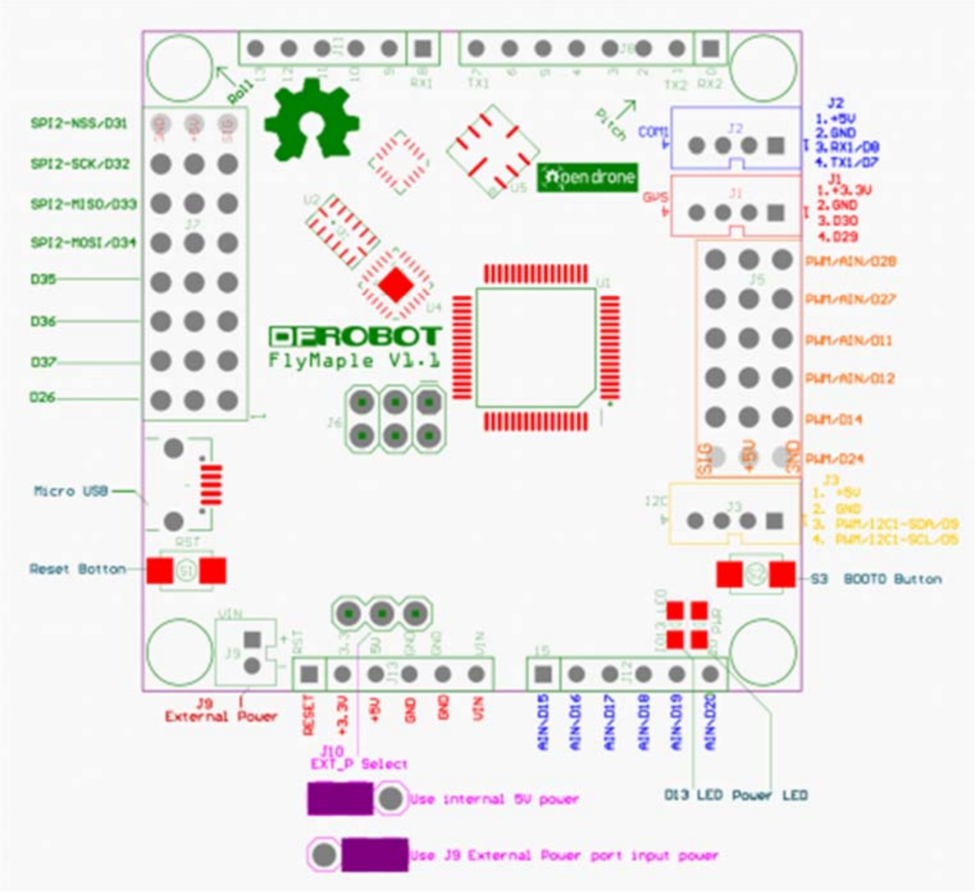
- It also supports WIN 7/8 64-bit.
- Note that while these instructions work on Windows XP, changes in Windows 7/8 mean that you won't be able to install the IDE without disabling driver signing on your computer.

- Download the binary libUSB package (http://sourceforge.net/projects/libusb-win32/files/), extract somewhere (we'll call it c:\LIBUSB)
- Run the libusb INF wizard (c:\LIBUSB\bin\inf-wizard.exe)
- Click through the dialogs, select maple,and the default values should be fine, or you can fill in some descriptive info.
- Save the INF and have it install. It'll produce a big red prompt about the driver being unsigned and problematic; however, it will still work.
- "Update" the MAPLE 003 device in your device manager and select the INF file you generated.

To load any program:

- Run the Maple IDE, when you hit the Upload, hit your Reset button on your Maple Board. It should detect and load the data on it. If your compile takes too long, try hitting the reset later in the upload. I used an alternating interval blink to prove it was working to me instead of the straight blink code.
- The serial drivers that came with the Maple IDE package do work fine, so you can use those drivers for the COM port directly.

Diagram



Sample Code

This sample code is a piece of the whole library code for the motor functions, you need to download the library to run the whole sketch which contains this motors code.

```
#include <stdio.h>

#include "wirish.h"

#include "i2c.h"


extern uint16 MotorData[6];  //Motor control command


extern volatile unsigned int chan1PPM;  //PPM command

extern volatile unsigned int chan2PPM;

extern volatile unsigned int chan3PPM;

extern volatile unsigned int chan4PPM;
```

```
char str[512];

//////////////////////////////////////////////////////////////////////////////
///////
//Function:  void setup()
//Para: None
//Return:    None
//Initialize function
//////////////////////////////////////////////////////////////////////////////
//////
void setup()
{
  motorInit();       //Init Motors
  capturePPMInit();  //捕获遥控器接收机PPM输入信号功能初始化

  //configure I2C port 1 (pins 5, 9) with no special option flags (second arg
ument)
  i2c_master_enable(I2C1, 0);  //设置I2C1接口，主机模式

  initAcc();            //初始化加速度计
  initGyro();           //初始化陀螺仪
  bmp085Calibration();  //初始化气压高度计
  compassInit(false);   //初始化罗盘
  //compassCalibrate(1);  //校准一次罗盘，gain为1.3Ga
  //commpassSetMode(0);  //设置为连续测量模式

}
//////////////////////////////////////////////////////////////////////////////
///////
//函数原型：  void loop()
//参数说明：  无
```

```
//返回值：    无
//说明：      主函数，程序主循环
//////////////////////////////////////////////////////////////////////////////
//////
void loop()
{
  //***************ADXL345加速度读取测试例子*****************************
  int16 acc[3];
  getAccelerometerData(acc);  //读取加速度
  SerialUSB.print("Xacc=");
  SerialUSB.print(acc[0]);
  SerialUSB.print("    ");
  SerialUSB.print("Yacc=");
  SerialUSB.print(acc[1]);
  SerialUSB.print("    ");
  SerialUSB.print("Zacc=");
  SerialUSB.print(acc[2]);
  SerialUSB.print("    ");
  //delay(100);
  //**********************************************************/


  //***************ITG3205加速度读取测试例子*****************************
    int16 gyro[4];
    getGyroscopeData(gyro);    //读取陀螺仪

    SerialUSB.print("Xg=");
    SerialUSB.print(gyro[0]);
    SerialUSB.print("    ");
    SerialUSB.print("Yg=");
    SerialUSB.print(gyro[1]);
    SerialUSB.print("    ");
    SerialUSB.print("Zg=");
    SerialUSB.print(gyro[2]);
    SerialUSB.print("    ");
    //SerialUSB.print("temperature=");
    //SerialUSB.print(gyro[3]);
    //SerialUSB.print("    ");
    //delay(100);
  //*********************************************************************/
  //****************************BMP085 气压计测试例子******************
  int16 temperature = 0;
  int32 pressure = 0;
  int32 centimeters = 0;
  temperature = bmp085GetTemperature(bmp085ReadUT());
  pressure = bmp085GetPressure(bmp085ReadUP());
  centimeters = bmp085GetAltitude(); //获得海拔高度，单位厘米
  //SerialUSB.print("Temperature: ");
  SerialUSB.print(temperature, DEC);
  SerialUSB.print(" *0.1 deg C ");
  //SerialUSB.print("Pressure: ");
  SerialUSB.print(pressure, DEC);
  SerialUSB.print(" Pa ");
  SerialUSB.print("Altitude: ");
  SerialUSB.print(centimeters, DEC);
   SerialUSB.print(" cm ");
  //SerialUSB.println("    ");
  //delay(1000);
  //********************************************************************/
  //******************************HMC5883罗盘测试****************************
  float Heading;
  Heading = compassHeading();
  //SerialUSB.print("commpass: ");
  SerialUSB.print(Heading, DEC);
  SerialUSB.println(" degree");
```

```
  delay(100);
 //***************************************************************************/
  /*************************电机驱动测试***************************************************************
  MotorData[0] = 1;  //首先将PWM设置为最低以开启电调
  MotorData[1] = 1;
  MotorData[2] = 1;
  MotorData[3] = 1;
  MotorData[4] = 1;
  MotorData[5] = 1;
  motorCcontrol();   //计算各个电机控制量之差,将这个值用于定时器产生中断改变相应电机脉冲高电平时间
  delay(1000);
  MotorData[0] = 500;  //控制6个电调使电机按照一半速度运行
  MotorData[1] = 500;
  MotorData[2] = 500;
  MotorData[3] = 500;
  MotorData[4] = 500;
  MotorData[5] = 500;
  motorCcontrol();   //计算各个电机控制量之差,将这个值用于定时器产生中断改变相应电机脉冲高电平时间
  while(1);
  **********************************************************************************************************/
  /********************无线遥控器RC 的 PPM 捕获测试****************************************************
  SerialUSB.print("PPM Channel 1: ");
  SerialUSB.print(chan1PPM, DEC);
  SerialUSB.print("  ");
  SerialUSB.print("PPM Channel 2: ");
  SerialUSB.print(chan2PPM, DEC);
  SerialUSB.print("  ");
  SerialUSB.print("PPM Channel 3: ");

  SerialUSB.print(chan3PPM, DEC);

  SerialUSB.print("  ");

  SerialUSB.print("PPM Channel 4: ");

  SerialUSB.print(chan4PPM, DEC);

  SerialUSB.println("  ");

  delay(100);

  *****************************************************************************
*************************/

}
```