

PulseRain
TECHNOLOGY

Doc# USG-0922-0159, Rev 1.0.0

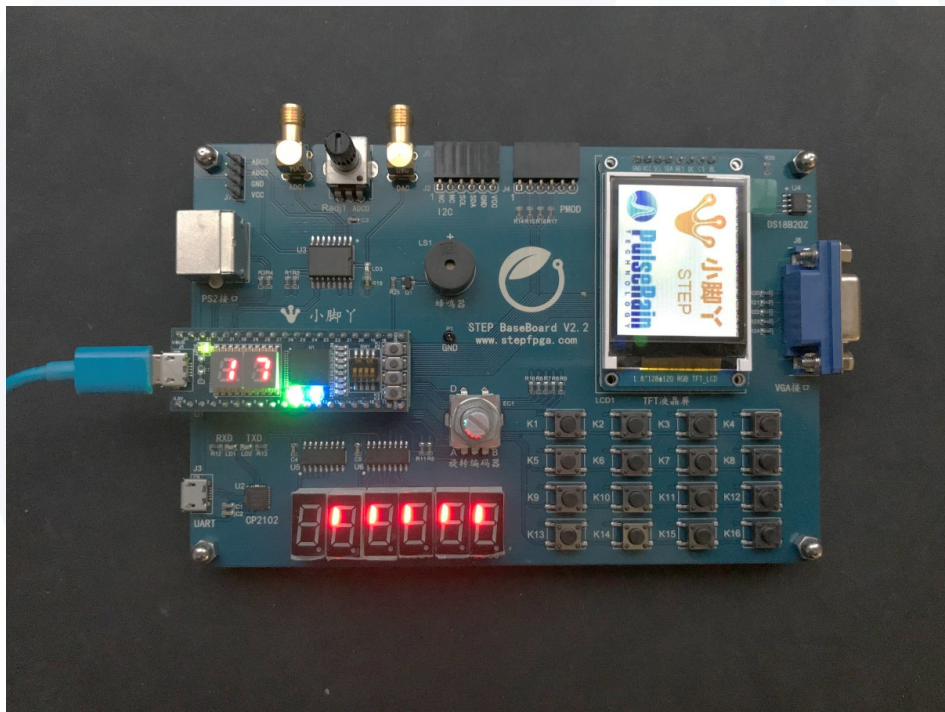
Copyright © 2018
PulseRain Technology, LLC.

10555 Scripps Trl, San Diego, CA 92131

☎ 858-877-3485 📠 858-408-9550
<http://www.pulserain.com>

Step FPGA M10

用户手册 (User Guide)



Mar, 2018

Table of Contents

| | | |
|-------|---|----|
| 1 | 前言 | 1 |
| 1.1 | 硬件概述 (Hardware Overview) | 1 |
| 1.2 | 工作流程 (Design Flow)..... | 2 |
| 2 | 软件设计(Software Design)..... | 3 |
| 2.1 | 连接开发板(Connect the Board)..... | 3 |
| 2.2 | 安装Arduino IDE | 4 |
| 2.3 | 编写Sketches | 8 |
| 2.4 | 安装和使用M10库 | 10 |
| 3 | 硬件设计(Hardware Design) | 12 |
| 3.1 | 代码(Repository for Step FPGA M10 RTL Design) | 12 |
| 3.2 | 编译综合(Build the FPGA Image) | 13 |
| 4 | 软硬件协作 | 14 |
| 4.1 | 片上闪存的配置(Onchip Flash Memory)..... | 14 |
| 4.2 | 外部设备和Arduino软件库 | 14 |
| 4.2.1 | 旋转编码器..... | 14 |
| 4.2.2 | 7段管显示 | 15 |
| 4.2.3 | PS/2键盘接口 | 16 |
| 4.2.4 | LCD (ST7735S) 接口 | 17 |
| 4.2.5 | 闪存内容读取..... | 18 |
| 4.2.6 | PWM 背光控制..... | 19 |
| 4.2.7 | 复位按键 | 19 |
| 4.3 | 样例(Examples)..... | 19 |
| 4.4 | 支持新的外部设备..... | 20 |

1 前言

在美国PulseRain Technology, LLC 和中国苏州思得普信息科技有限公司和合作努力下, PulseRain FP51-1T MCU 内核被成功地移植到了小脚丫Step FPGA 的 MAX10 开发板上. 同时针对Step FPGA的 Base Board, PulseRain Technology 也开发了新的外设硬件库(Peripheral RTL Library), 以支持旋转编码器(Rotary Encoder), PS2 接口, LCD 显示等一系列外部设备. 相应的Arduino软件库也得到了更新. 本文档是为该移植工作而编写的中文用户手册, 以方便查阅.

1.1 硬件概述 (Hardware Overview)

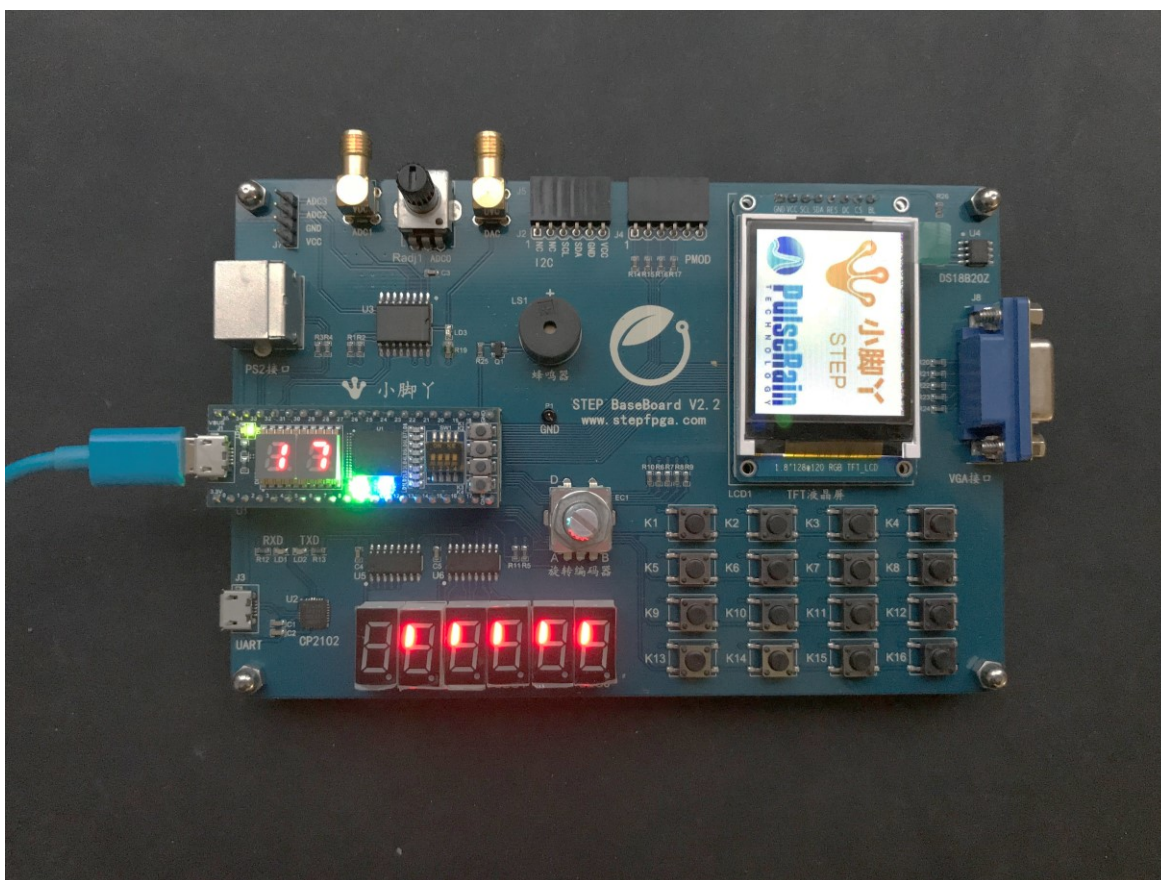


Figure 1-1 The Close View of Step FPGA M10

如Figure 1-1 所示, Step FPGA M10 平台主要包括两部分: 小脚丫STEP-MAX10 FPGA 开发板和STEP Base Board. 小脚丫STEP-MAX10 FPGA 开发板包含一片Intel 10M08SCM153 FPGA, 两个7段管显示和多个LED, 开关和按钮等. 同时FPGA开发板还通过40管脚的DIP接口同 STEP Base Board连接, 而STEP Base Board则包含有PS2, USB串行口, 旋转编码器, LCD等多种外部设备.

1.2 工作流程 (Design Flow)

在出厂的factory FPGA image里面已经包含有一个PulseRain FP51-1T MCU软核. 该软核支持与Arduino兼容的软件接口. 如果用户已经熟悉Arduino的开发环境, 用户可以按照如下步骤进行快速开发: (参见 Figure 1-2右半部)

- 1) 将小脚丫STEP-MAX10 FPGA 开发板插入STEP Base Board
- 2) 用一根microUSB电缆将小脚丫STEP-MAX10 FPGA 开发板与PC连接
- 3) 用另外一根microUSB电缆将STEP Base Board的USB串口与PC连接
- 4) 配置Arduino IDE, 并在其中编写sketch. 如有必要, 用户也可调用相关的外设软件库
- 5) 在调试结束后, 可以把完成的Sketch烧入FPGA 的Flash当中. 这样在上电后,新的Sketch会被直接调用

另外一方面, 如果用户需要客制化硬件,加入自己的外部设备控制器的话, 用户可以按照如下步骤操: (如 Figure 1-2左半部所示)

- 1) 编写RTL代码, 并且仿真
- 2) 用Intel Quartus Prime综合RTL代码, 产生.sof文件
- 3) 将.sof文件与.hex文件合并成.pof文件
- 4) 烧制FPGA

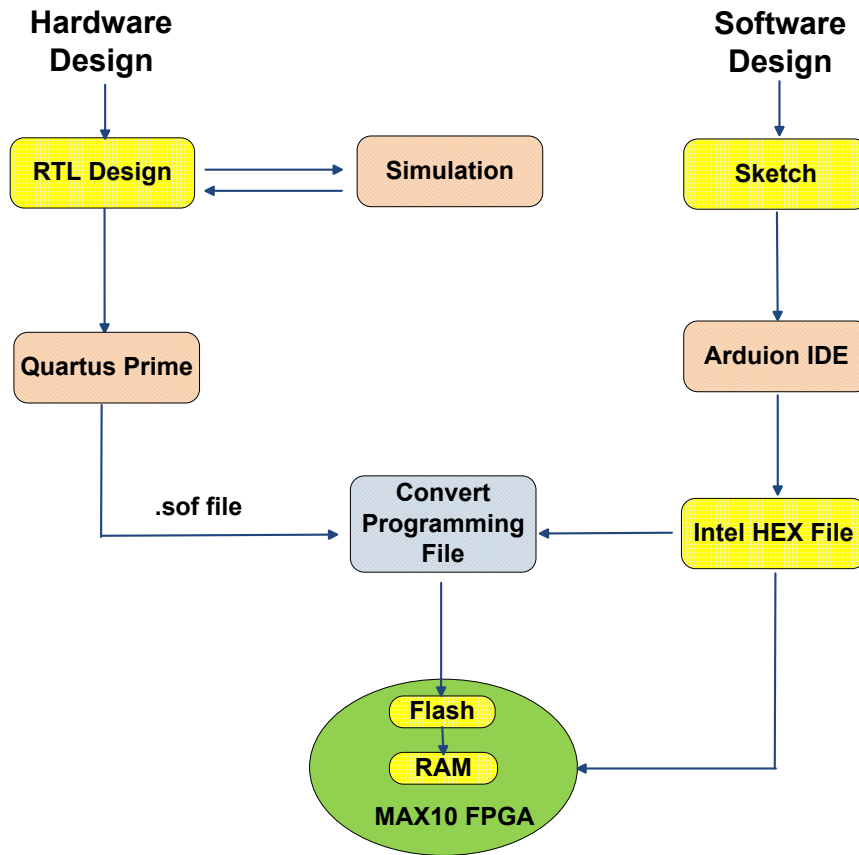


Figure 1-2 Design Flow for PulseRain M10

2 软件设计(Software Design)

本章节将详细介绍Figure 1-2中所示的软件设计流程。

2.1 连接开发板(Connect the Board)

按照Section 1.2中描述的方法,将Step FPGA开发系统和Windows PC 相连. 小脚丫的Base Board上有一个用Silicon Labs CP2102制作的USB/UART串口, 其End Point被设置成USB CDC Device.在Windows 10下,该设备会自动被确认为一个COM口,而无需驱动程序.(在其它Windows平台下,可能需要一个.inf 文件来配置驱动.) 如果一切正常的话, 在设备管理器里面可以看到如Figure 2-1所示的串口。

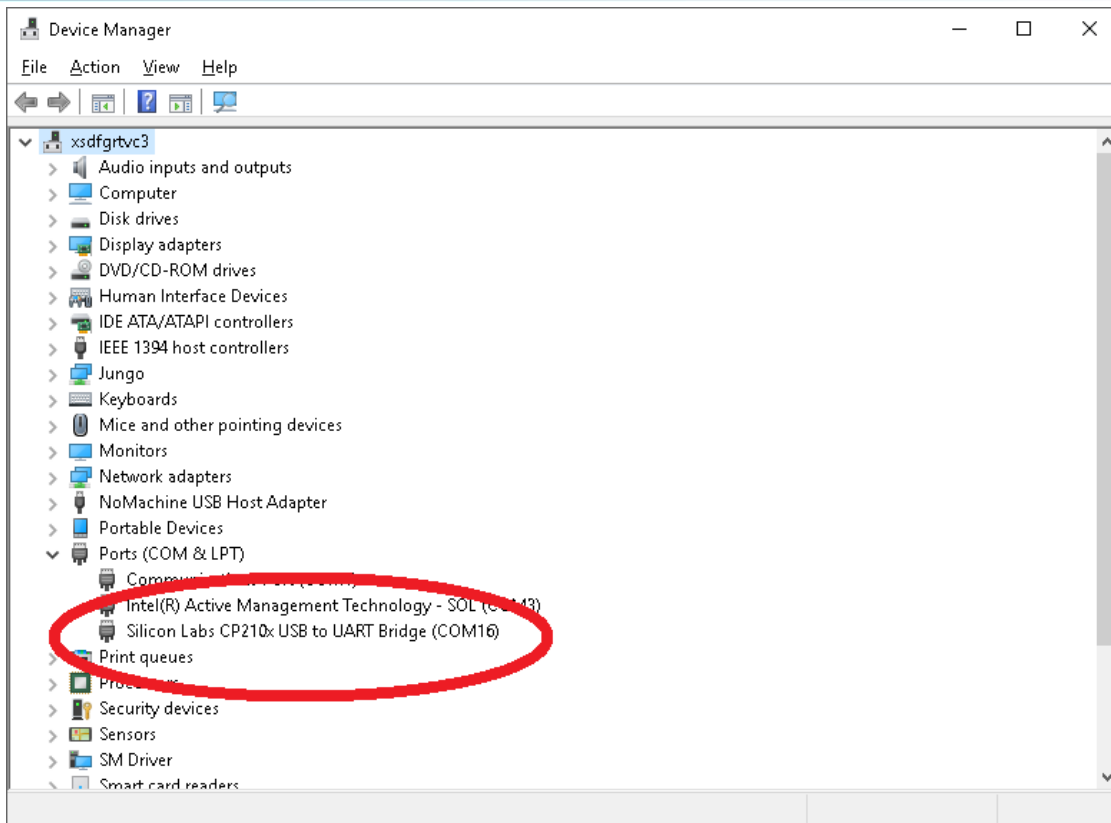


Figure 2-1 USB Serial Port in Windows Device Manager

2.2 安装Arduino IDE

- 1) 在Windows 10平台上, Arduino IDE可以从Windows App store里面直接安装. 在其它的Windows平台上, 可以从Arduino官网上下载Windows installer:
<https://www.arduino.cc/en/Main/Software>
- 2) 把Arduino IDE安装完毕并运行后, 请打开File / Preferences菜单, 如Figure 2-2所示:

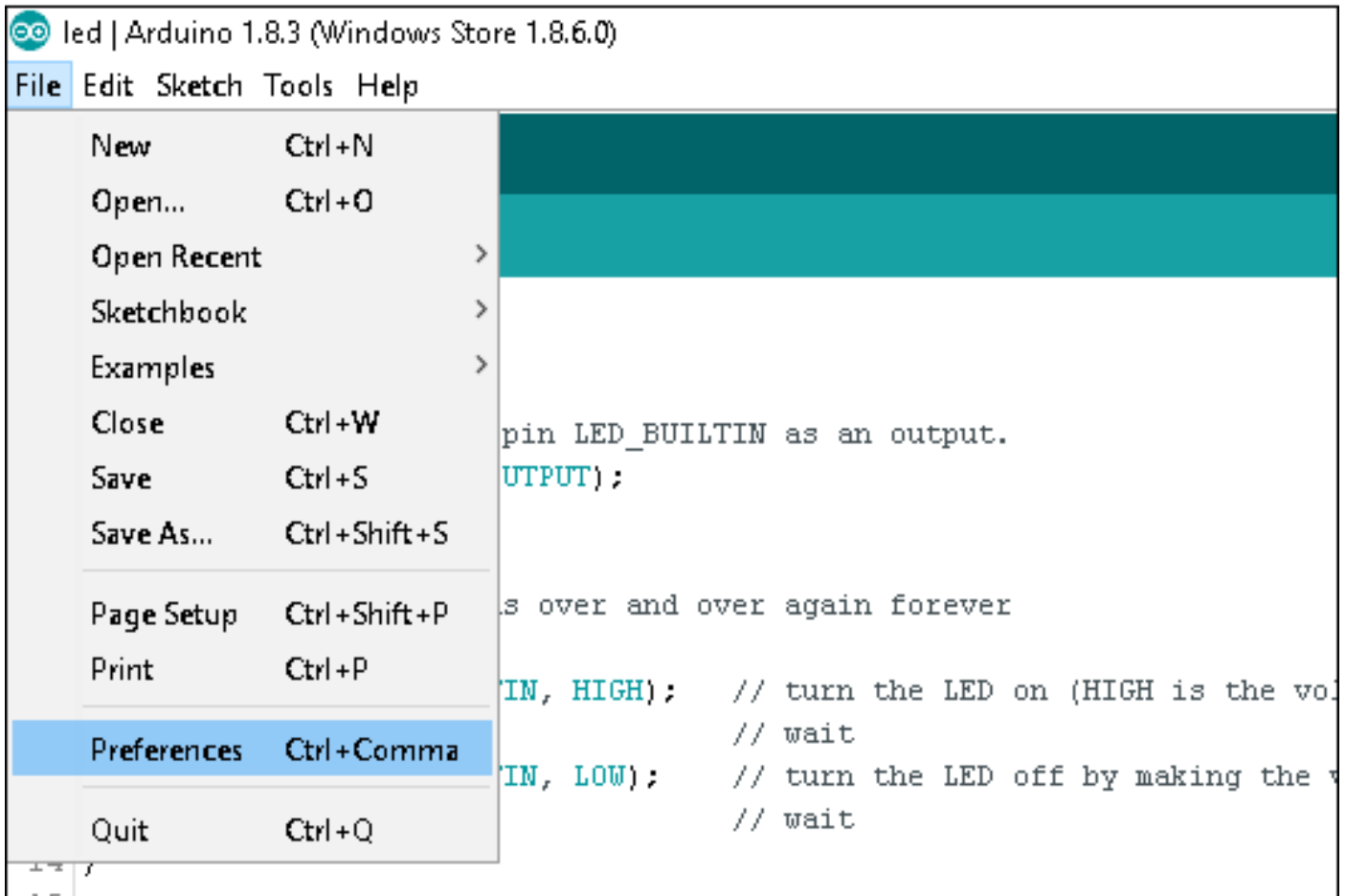


Figure 2-2 Arduino IDE, Preferences Menu

- File / Preferences菜单会开启一个如Figure 2-3所示的对话框. 请把"Additional Boards Managers URL"设置为
https://raw.githubusercontent.com/PulseRain/Arduino_M10_IDE/step_fpga/package_M10_index.json

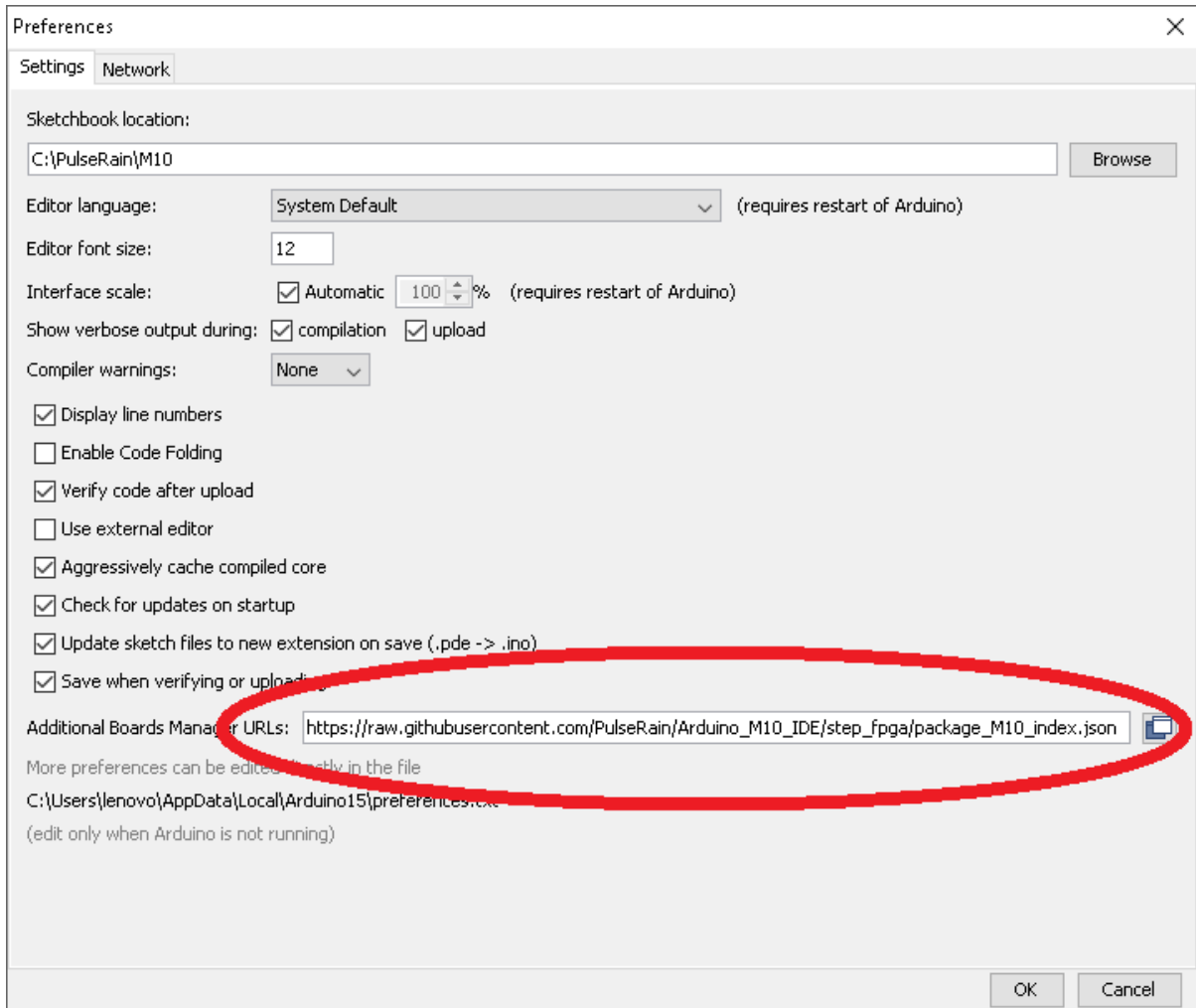


Figure 2-3 Arduino IDE, Preferences Dialogue

4) 关闭上一个对话框后,请打开Tools / Boards / Boards Manager菜单, 如Figure 2-4所示.

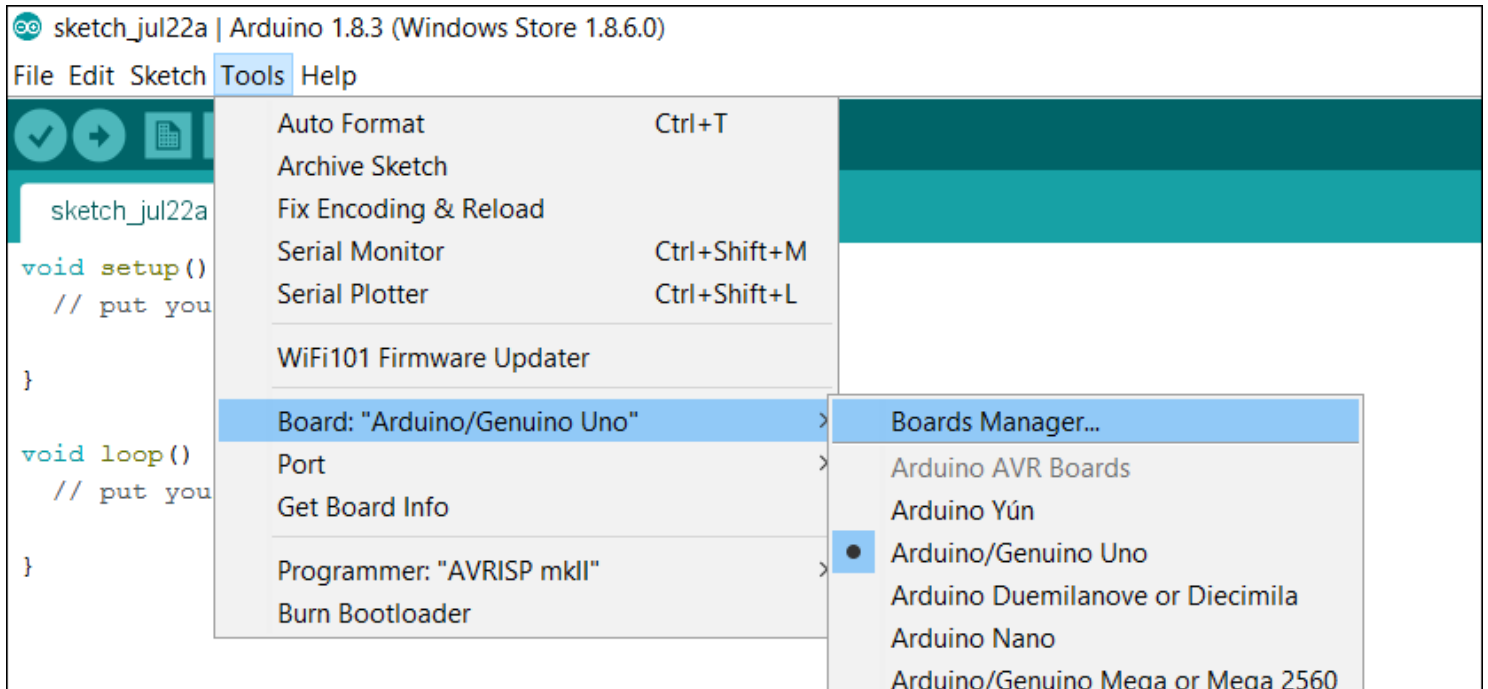


Figure 2-4 Arduino IDE, Boards Manager Menu

- 5) "Boards Manager"会启动一个如Figure 2-5所示的对话框. 在搜索条输入"step"来查找和安装所需的软件包

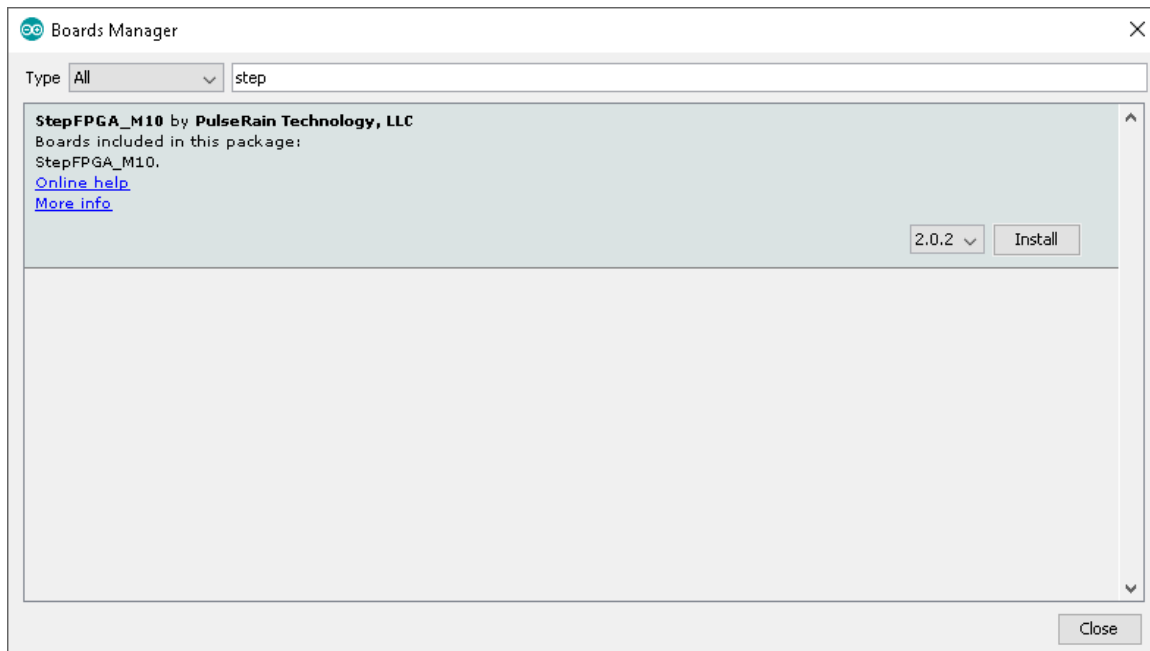


Figure 2-5 Arduino IDE, Boards Support Package for StepFPGA M10

2.3 编写Sketches

和所有的Arduino开发板一样, 在开始编写Sketches之前, 用户需要在Arduino IDE的Tool / Port菜单下选择正确的COM口和开发板名称, 如Figure 2-6所示.

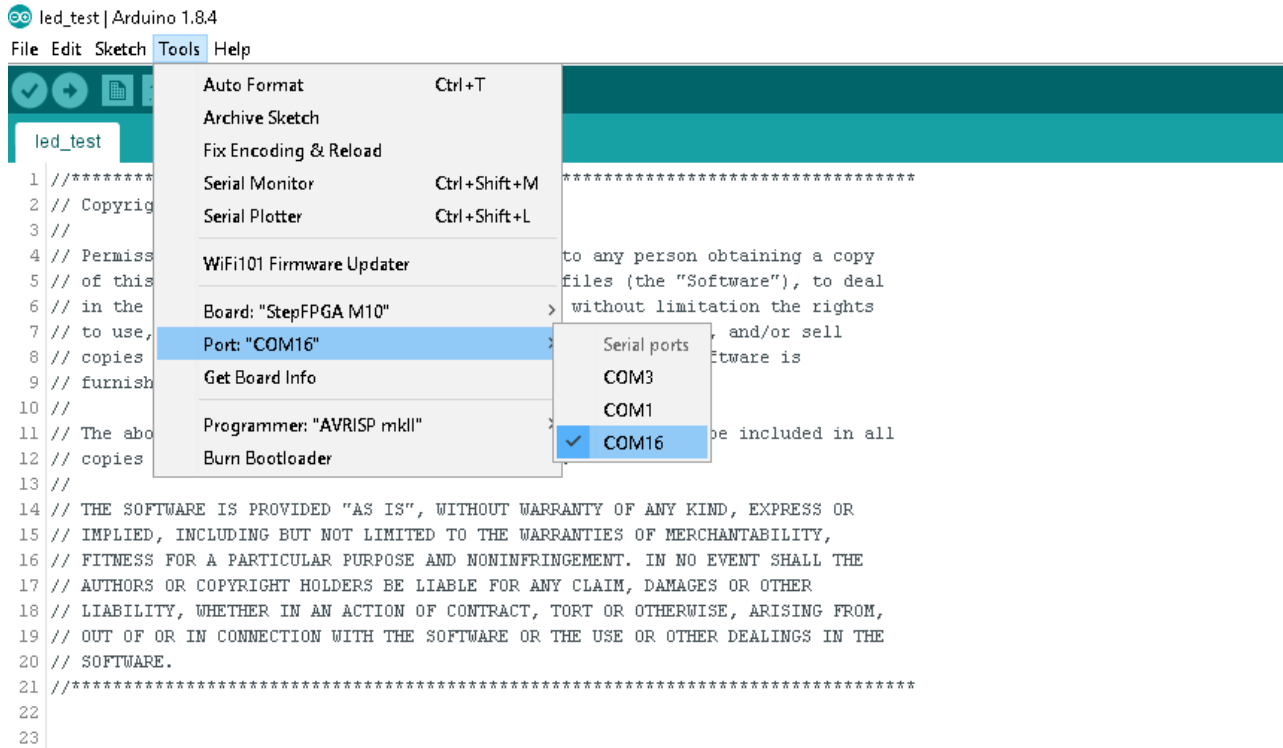


Figure 2-6 Arduino IDE, Select COM port

开发板的选择在Tool / Board菜单下, 如Figure 2-7所示.

```
void setup() {
  led_index = 1;
  P3_DIRECTION = 0xFF; // Set Port 3 direction
}

void loop() {
  P3 = led_index;

  if (led_index == 0x80) {
    led_index = 1;
  } else {
    led_index = led_index << 1;
  }

  delay(1000);
}
```

List 2-1 Sketch to Test LED

如果之前的步骤都正确实施的话，用户此刻就应该可以编写Sketch了。这里建议用户先试一下List 2-1 里所示的Sketch，来实现一个LED流水灯。

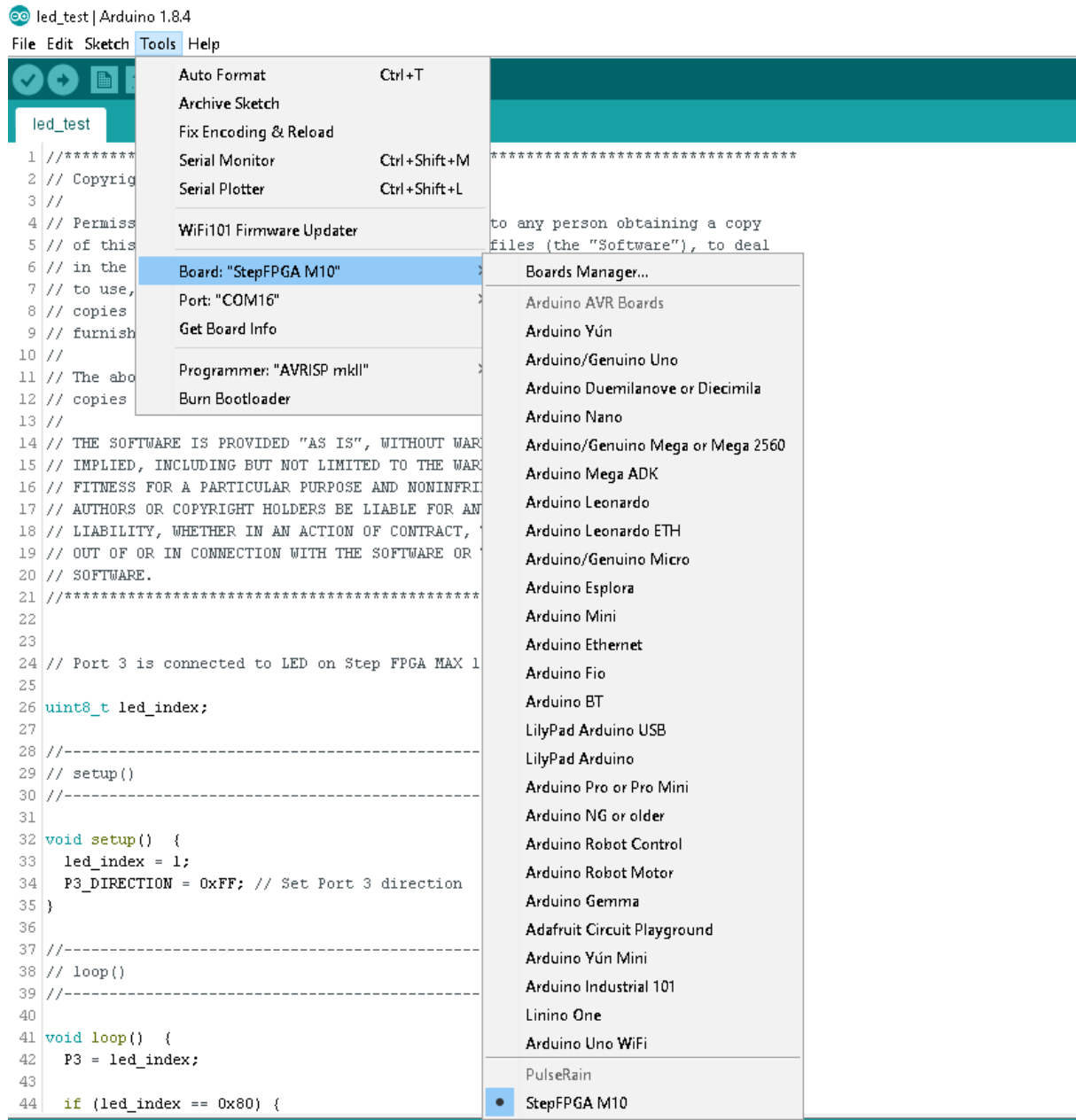


Figure 2-7 Arduino IDE, Select PulseRain M10 board

在把List 2-1粘帖到Arduino IDE里面以后，用户可以用快捷键"ctrl-U" (或者菜单Sketch/Upload) 来编译和下载sketch到Step FPGA开发板上。这里建议用户在Preferences对话框 (菜单File / Preferences) 中打

开"Show Verbose Output"选项, 如Figure 2-8所示. 这样的话, 就可以很容易地找到最终生成的.hex文件. 如果用户想把Sketch在上电时自动加载运行的话, 就需要把这个.hex文件烧录到MAX10 FPGA的UFM里面.

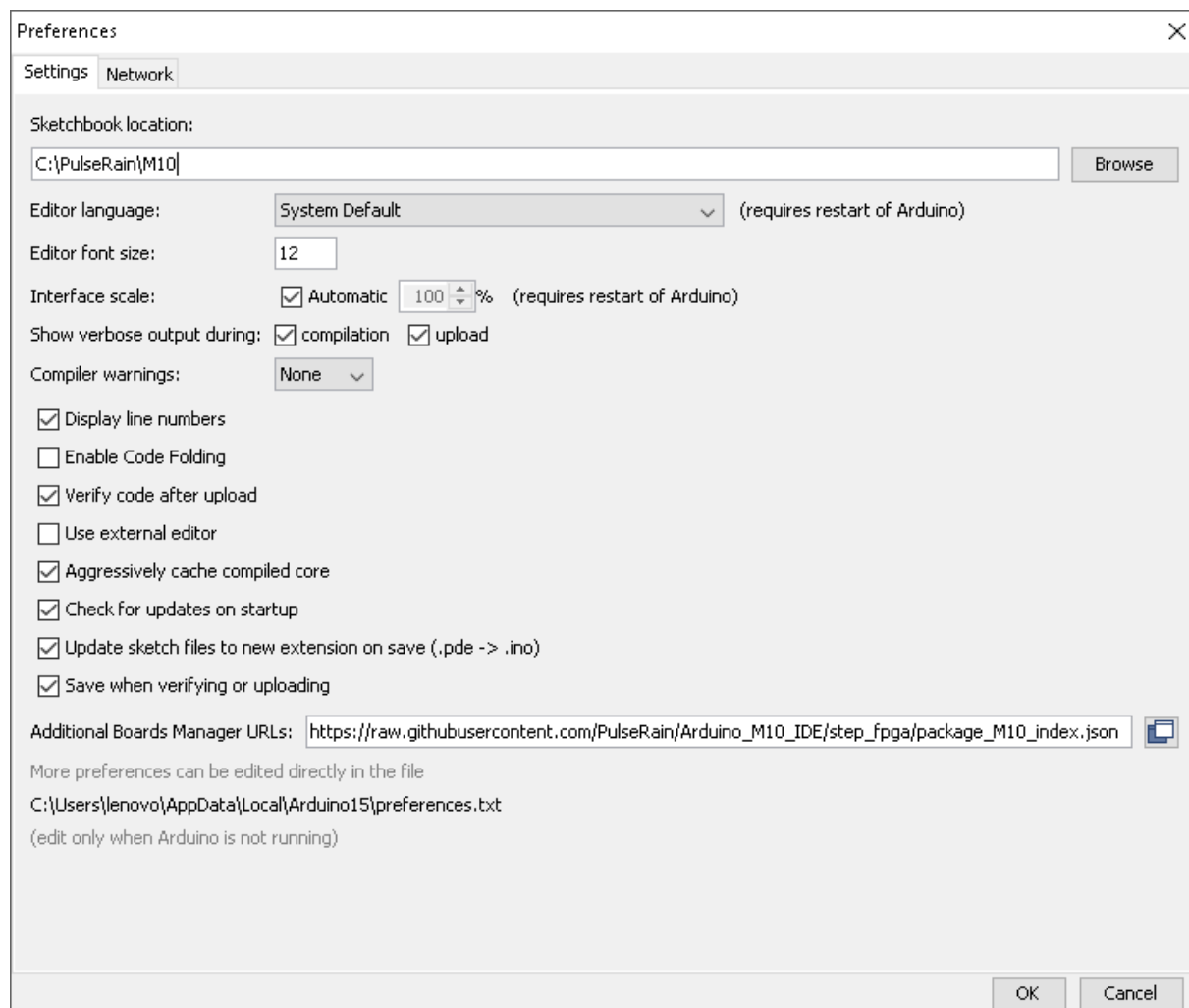


Figure 2-8 Turn on Verbose Option

2.4 安装和使用M10库

为方便用户操作外部设备, PulseRain Technology, LLC还为用户提供了M10软件库. 用户可以通过菜单 Sketch / Include Library / Manage Libraries ... 来安装软件库, 如Figure 2-9所示.

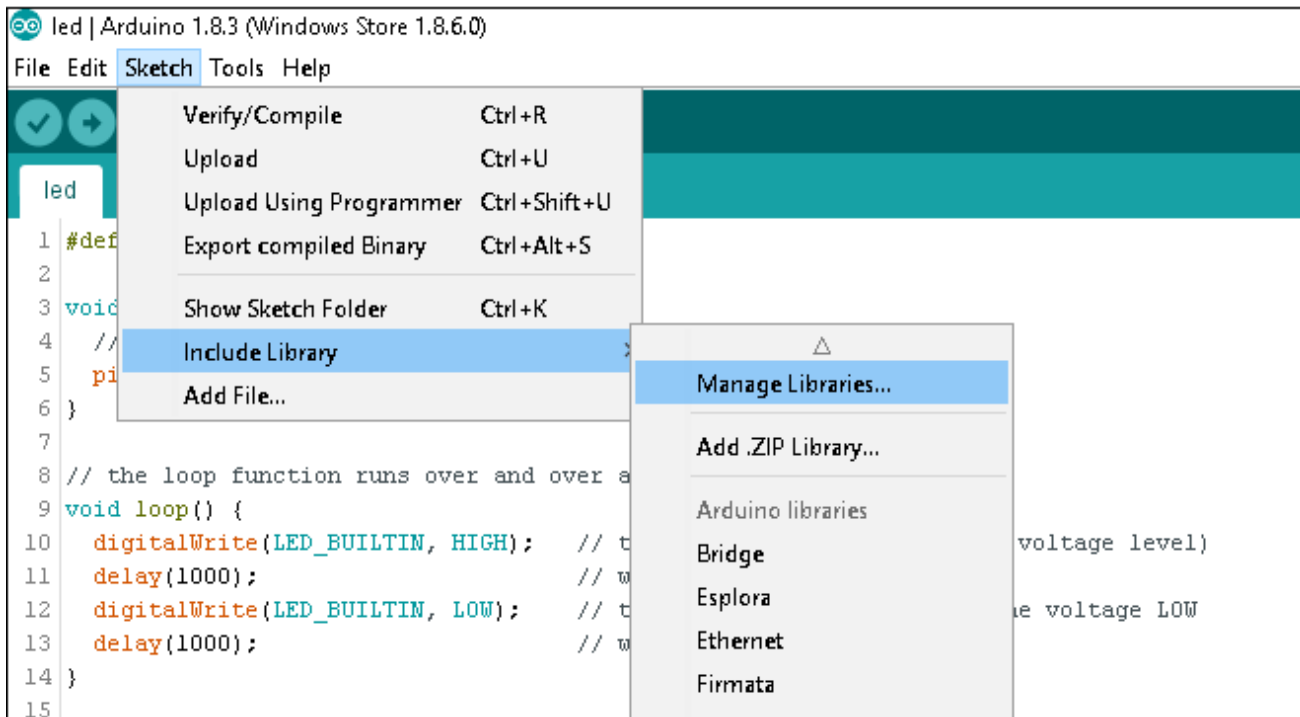


Figure 2-9 Arduino Manage Libraries Menu

该菜单会启动一个如Figure 2-10所示的对话框. 在搜索框中输入"M10"并安装相关的库(Figure 2-11).

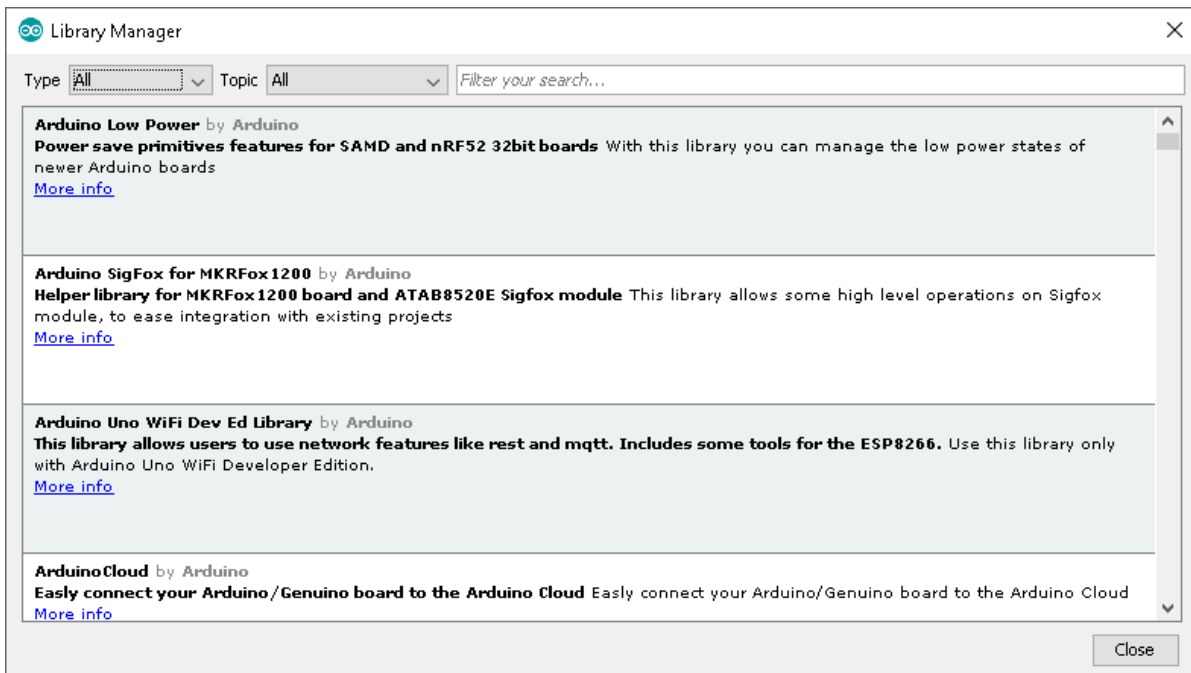


Figure 2-10 Arduino Library Manager

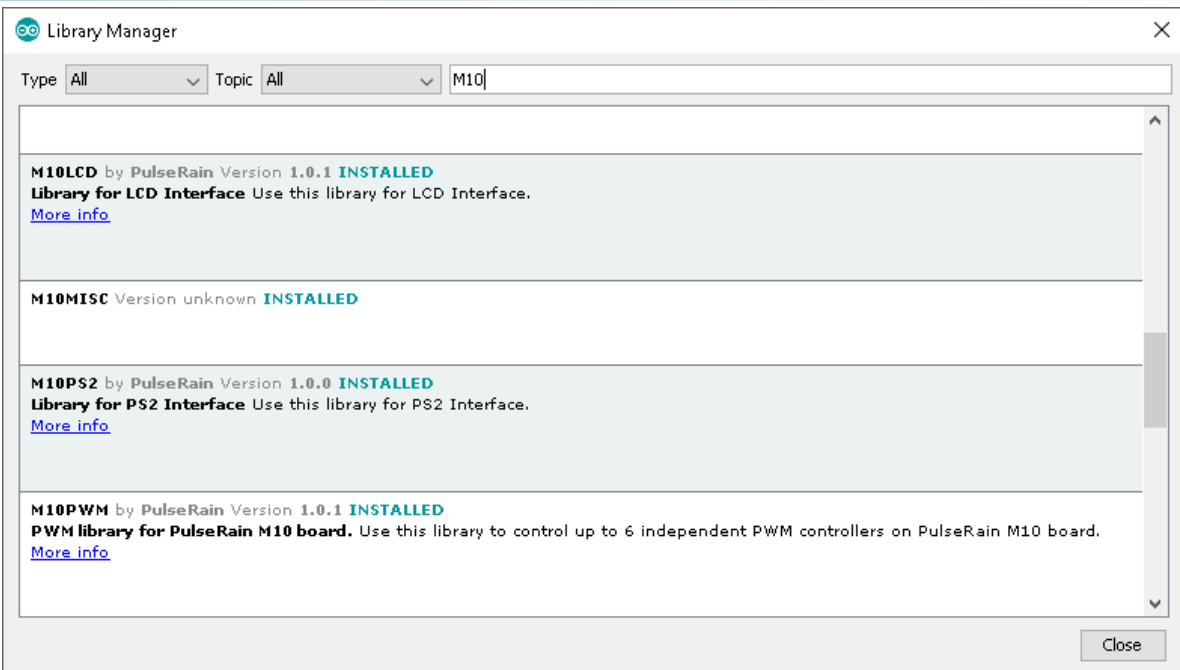


Figure 2-11 Search for M10 Library in Arduino Library Manager

在sketch中使用这些库的话，用户可以include相关的头文件即可。比如，如果用户需要使用PWM库的话，只需在Sketch的开头加入`#include "M10PWM.h"`，并在Sketch中调用相关的API。

3 硬件设计(Hardware Design)

本章节将详细介绍Figure 1-2中所示的硬件设计流程。

3.1 代码(Repository for Step FPGA M10 RTL Design)

和Step FPGA M10有关的RTL代码 (System Verilog)，可以在Github: <https://github.com/PulseRain/Mustang> 的step_fpga branch下面找到。可以用如下的命令得到代码：

```
>> git clone -b step_fpga https://github.com/PulseRain/Mustang.git step_fpga  
>> cd step_fpga  
>> git submodule update --init --recursive
```

3.2 编译综合(Build the FPGA Image)

用户可以用Intel Quartus Prime打开synth / Mustang_fast.qpf , 并点击"Start a new compilation", 如Figure 3-1所示.

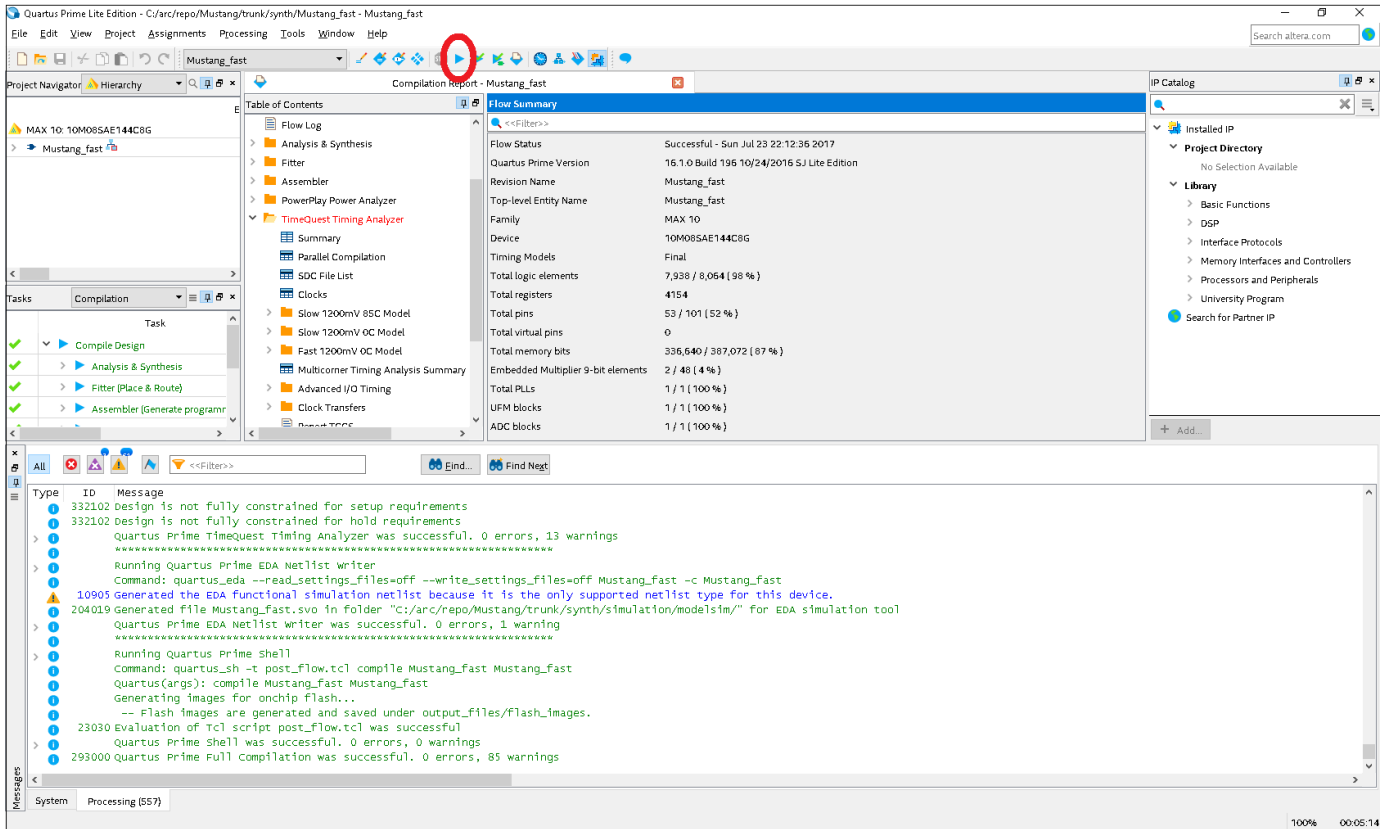


Figure 3-1 Building Mustang Project

Mustang project 内含一个叫post_flow.tcl的script.这个script会自动做文件合成与转换. 转换结束后,在synth/output_files/flash_images/full_image.pof就是最后需要的FPGA Image. 如果用户需要把数据放入UFM的话, 只需把对应的.hex文件命名成ufm.hex,并放到synth/output_files/之下, post_flow.tcl 会自动将其包含到最终的FPGA Image里面.

4 软硬件协作

4.1 片上闪存的配置(Onchip Flash Memory)

Step FPGA M10采用Single Compressed Image设置, 在该设置下, 其片上闪存的地址如Table 4-1所示:

| 分区号 | 闪存类型 | 地址 | 大小(字节) |
|-----|------|-------------------|--------|
| 1 | UFM | 0x00000 – 0x03FFF | 16384 |
| 2 | UFM | 0x04000 – 0x07FFF | 16384 |
| 3 | UFM | 0x08000 – 0x167FF | 59392 |
| 4 | CFM | 0x16800 – 0x397FF | 143360 |

Table 4-1 闪存配置

在 Table 4-1 所示的4个分区中, 分区1和2一共有32KB, 被用来存放代码. PulseRain FP51-1T MCU的缺省配置有32KB Instruction RAM和8KB Data RAM. 分区1和2 正好和FP51-1T MCU的 Instruction RAM 大小相对应.在上电启动时, `source / code_mem_power_on_loader.sv` 会把用户的代码从闪存分区1和2载入到 Instruction RAM 中并执行.

闪存分区3目前被用来存放LOGO图片, 该图片可以在LCD上显示. 小脚丫自带的LCD有132 x 162的分辨率. 这里建议用户存放一个大小为128 x 154的真彩图片在这个分区. 在 `synth / output_files`下有一个叫 `pic2hex.py`的Python Script, 可以把图片转换为 .hex 文件. 该目录下的 `bitmap.hex` 就是由同一目录下的 `pulse_step.png`生成的. 当用户用Arduino IDE 编译Sketch以后, 可以把Arduino生成的.hex 文件的最后一行删除, 并与`bitmap.hex`合并, 并将合并后的文件命名为`synth/output_files/ufm.hex`. 这样在生成FPGA Image时, `ufm.hex` 会被自动包含在最后生成的Image里面 (参见Section 3.2)

4.2 外部设备和Arduino软件库

针对Step FPGA的 Base Board, PulseRain Technology 增加了如下外部设备的支持:

4.2.1 旋转编码器

Step Base Board上自带一个旋转编码器. 在FP51-1T MCU之下, 其对应的控制寄存器地址如下:

| Address | Register Name |
|---------|----------------|
| 0xFF | ROTARY_ENCODER |

Table 4-2 Address Definition for Rotary Encoder

当ROTARY_ENCODER被写入时, 写入的值会被用作旋转编码器的初始值. 当用户按下旋转编码器后, 这个值会被载入计数器. 当ROTARY_ENCODER被读取时, 当前的计数器值会被读出. 计数器的值会始终保持在0到255的闭区间内.

4.2.2 7段管显示

小脚丫Step FPGA 的 MAX10 开发板上自带两个7段管显示. 它们和FP51-1T MCU的 Port 0和Port 1端口对应. Port2[1:0] 被用来作为这两个7段管显示的使能控制. 为方便用户, PulseRain Technology已经开发提供M10Seven库. 用户可以在Sketch中 #include "M10SevenSeg.h" 来调用该库(库的安装请参见Section 2.4). 该库包含有以下的API :

- *void init ()*

Parameters:

None.

Return Value:

None

调用该函数来初始化硬件.

- *void onOff(uint8_t on_off_mask)*

Parameters:

on_off_mask: bit 1 和 bit 0 被用来对这两个7段显示做使能控制

Return Value:

None

调用该函数来对7段显示做使能控制.

- *void byteHex (uint8_t num)*

Parameters:

num: 需要显示的字节

Return Value:

None

调用该函数来用16进制显示一个字节.

- `void decimalPoint (uint8_t mask)`

Parameters:

小数点: bit 1 和 bit 0 被用来对小数点做使能控制

Return Value:

None

调用该函数来用对小数点做使能控制.

4.2.3 PS/2键盘接口

Step Base Board上自带一个PS/2键盘接口. 在FP51-1T MCU之下, 其对应的控制寄存器地址如下:

| Address | Register Name |
|---------|---------------|
| 0xD5 | PS2_CSR |
| 0xD6 | PS2_DATA |

Table 4-3 Address Definition for Rotary Encoder

- CSR (Control and Status Register), CSR各位定义如下:

| Bits | R/W | Default | Description |
|-------|-----|---------|------------------------------|
| 0 | RW | 0 | 当有新的数据被收到时,该位会被设为1. 写0来清除该位. |
| 7 : 1 | WO | 0 | RESERVED |

Table 4-4 Bit Map for CSR (Control Status Register)

- DATA (8 bit, RO),当有新的数据被收到时, 会被放入该寄存器.

为方便用户, PulseRain Technology已经开发提供M10PS2库. 用户可以在Sketch中 `#include "M10PS2.h"` 来

调用该库(库的安装请参见Section 2.4). 该库包含有以下的API :

- `void init ()`

Parameters:

None.

Return Value:

None

调用该函数来初始化硬件.

- `uint8_t dataAvailable ()`

Parameters:

None

Return Value:

当接收FIFO中有数据时, 函数会返回1, 否则返回0

调用该函数来确认是否有未被读取的数据在FIFO中.

- `uint8_t read ()`

Parameters:

None

Return Value:

从接收FIFO中读取的数据

调用该函数来用从接收FIFO中读取数据.

4.2.4 LCD (ST7735S) 接口

Step Base Board上自带一个1.8英寸TTF LCD(ST7735S) 显示器. 在FP51-1T MCU之下, 其对应的控制寄存器地址如下:

| Address | Register Name |
|---------|---------------|
| 0xC1 | LCD_CSR |
| 0xC2 | LCD_DATA |

Table 4-5 Address Definition for Rotary Encoder

- CSR (Control and Status Register), CSR各位定义如下:

| Bits | R/W | Default | Description |
|-------|-----|---------|---|
| 0 | RW | 0 | 对该位写入0时,表示是要发送命令操作.对该位写入1时,表示是要发送命令操作.当命令或数据发送完成时,该位为会被置为1.当有新的数据写入LCD_DATA寄存器时,该位会被清除. |
| 1 | RW | 0 | 对硬件的复位操作由该位控制. 在正常工作状态下该位应被置为0 |
| 7 : 2 | WO | 0 | RESERVED |

Table 4-6 Bit Map for CSR (Control Status Register)

- DATA (8 bit, WO),将要发送的命令或数据放入该寄存器.

为方便用户, PulseRain Technology已经开发提供M10LCD库. 用户可以在Sketch中 #include "M10LCD.h" 来调用该库(库的安装请参见Section 2.4). 该库包含有以下的API :

- *void reset ()*

Parameters:

None.

Return Value:

None

调用该函数来复位硬件.

- *void writeCMD (uint8_t cmd)*

Parameters:

cmd: 需要发送的命令

Return Value:

None

调用该函数来给LCD发送命令.

- *void writeDAT (uint8_t dat)*

Parameters:

dat: 需要发送的数据

Return Value:

None

调用该函数来给LCD发送数据.

4.2.5 闪存内容读取

在FP51-1T MCU之下, 用户可以通过以下的控制寄存器来读取闪存内容:

| Address | Register Name |
|---------|-----------------|
| 0xC3 | FLASH_READ_CSR |
| 0xC4 | FLASH_READ_DATA |

Table 4-7 Address Definition for Rotary Encoder

- CSR (Control and Status Register), CSR各位定义如下:

| Bits | R/W | Default | Description |
|-------|-----|---------|--|
| 0 | RW | 0 | 对该位写入时, 会启动读取操作. 当读取完成时, 该位会被置为1. 当新操作开始时, 该位会被清除. |
| 7 : 1 | WO | 0 | RESERVED |

Table 4-8 Bit Map for CSR (Control Status Register)

- DATA (8 bit, RW)
在读取数据之前, 将字节地址从高字节开始依次写入该寄存器. 读取操作完成后, 从该寄存器读出结果数据.

4.2.6 PWM 背光控制

LCD 的背光亮度是由PWM控制的, 具体操作请参见:

https://github.com/PulseRain/M10PWM/raw/master/extras/M10_PWM_TRM.pdf

4.2.7 复位按键

小脚丫Step-MAX10 FPGA开发板的KEY1被用来作为复位键, 并对该输入做了硬件防抖处理.

4.2.8 LED

小脚丫Step-MAX10 FPGA开发板带有8个红色LED, 它们被连接到了FP51-1T MCU 的Port 3 上.

4.3 样例(Examples)

用户可以在下面的Repository找到和小脚丫Step FPGA M10 平台有关的样例:

https://github.com/PulseRain/step_fpga_example

特别, 其中的demo_all样例包含有对以上外设和库的综合操作. demo_all会从闪存中读取LOGO图片, 并按18位真彩方式在LCD上显示. 用户也可以通过旋转编码器来调节背光, 旋转编码器的数值也会在7段管上显示, LED也会做相应变化. 而从PS/2键盘上得到的扫描码, 则会通过串口显示. 在Arduino IDE 里, 用户可以通过Serial Monitor在115200bps下看到结果.

4.4 支持新的外部设备

如果用户需要增加新的外设的话, 具体步骤如下 (以LCD为例)

1. 编写RTL模块, 并且simulate, 比如

GitHub\step_fpga\submodules\PulseRain_FP51_MCU\submodules\PulseRain_rtl_lib\LCD 下面的那些文件

2. 给新增加的外设分配SFR 地址, 地址分配见

GitHub\step_fpga\submodules\PulseRain_FP51_MCU\common\SFR.svh

```
parameter unsigned [DATA_WIDTH - 1 : 0] LCD_CSR_ADDR = 8'hC1;
```

```
parameter unsigned [DATA_WIDTH - 1 : 0] LCD_DATA_ADDR = 8'hC2;
```

3. 更改 GitHub\step_fpga\submodules\PulseRain_FP51_MCU\peripherals\peripherals.sv

把新外设增加到Peripheral bus 上. 由于新的外设增加了新的管脚, 所以除了peripherals.sv 之外,

GitHub\step_fpga\submodules\PulseRain_FP51_MCU\source\PulseRain_FP51_MCU.sv 和

GitHub\step_fpga\source\Mustang_fast.sv 也要相应修改, 在port list 中增加管脚. 比如

```
output wire          LCD_BL,
output wire          LCD_DAT_CMD,
output wire          LCD_RES,
inout wire           LCD_SDA,
inout wire           LCD_SCL,
```

4. 对新增的管脚做Constrain, 比如Pin assignment, voltage level 等

5. 由于新增加了SFR, Arduino 编译器的SFR定义也要相应修改. 编译器的SFR头文件通常被安装在

C:\Users\...\AppData\Local\Arduino15\packages\StepFPGA_M10\tools\M10_compiler\...\SDCC\include\mcs51\8051.h