
如何在 Linux® 下使用 SAMA5D2 SPI

简介

本应用笔记介绍在 Linux 下使用 SAMA5D2 SPI 的入门信息。

由于已在内核中引入了 SPI dev 接口，因此可轻松通过器件节点 “/dev/spidev” 访问用户空间中的 SPI 器件。有关 SPI dev 应用程序演示代码的信息，请参见[应用程序](#)一节。

SPI 器件具有受限的用户空间 API，支持对 SPI 从器件进行基本半双工 read() 和 write() 访问。此外，还可实现 ioctl() 请求、全双工传输和器件 I/O 配置。

参考文档

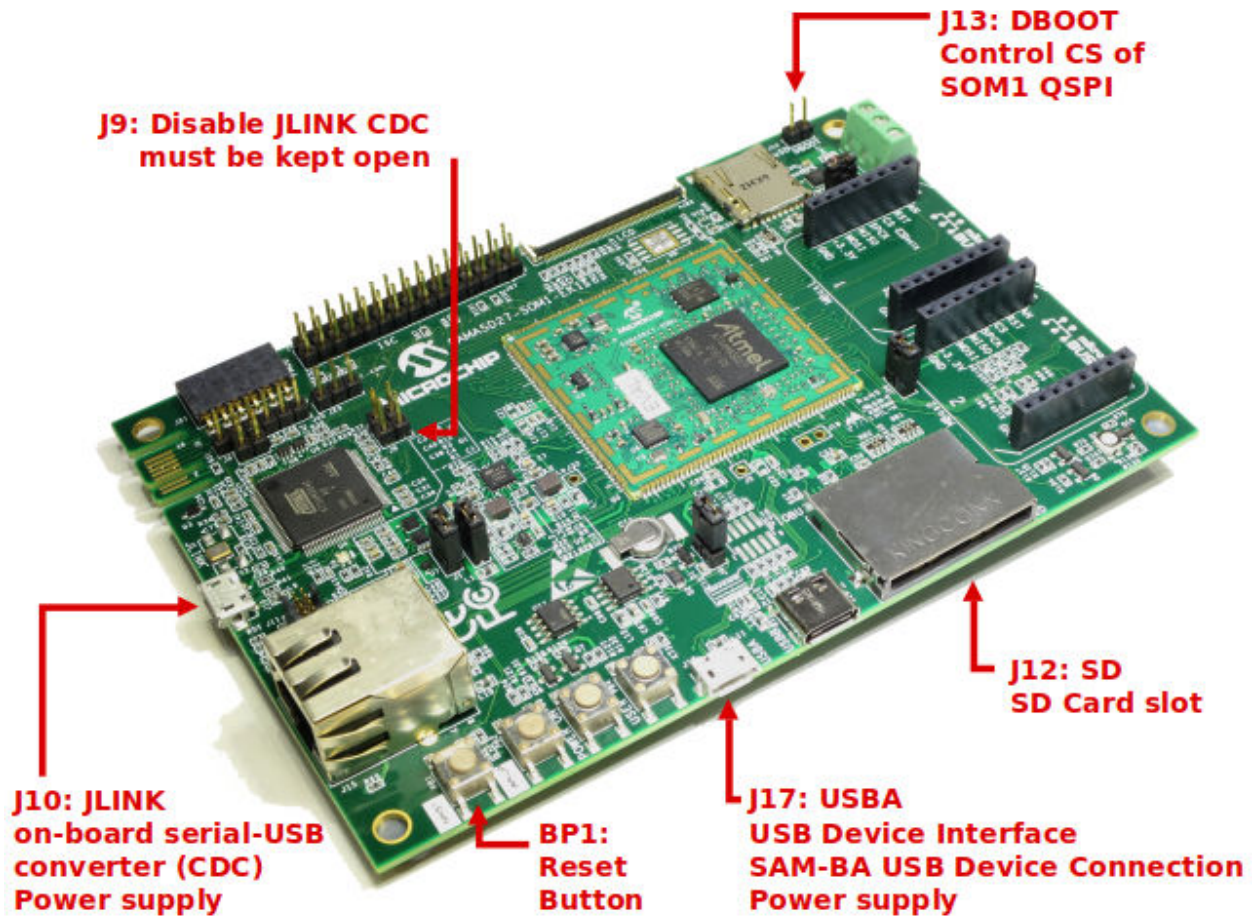
| 标题 | 编号 | 下载 |
|--------------------------|----------------|---|
| SAMA5D2 Series Datasheet | DS60001476 | https://www.microchip.com/design-centers/32-bit-mpus |
| SAMA5D27 SOM1 Kit1 用户指南 | DS50002667C_CN | https://www.microchip.com/DevelopmentTools/ProductDetails/PartNO/ATSAMA5D27-SOM1-EK1 |

前提条件

- 硬件
 - PC
 - SAMA5D27 SOM1 评估工具包（部件编号：ATSAMA5D27-SOM1-EK1）
 - SD 卡

- 软件

本演示在 Buildroot 编译的 AT91 Linux 平台上运行。第一步是建立 AT91 Buildroot 开发环境。请访问以下网站：
<http://www.at91.com/linux4sam/bin/view/Linux4SAM/BuildRoot>

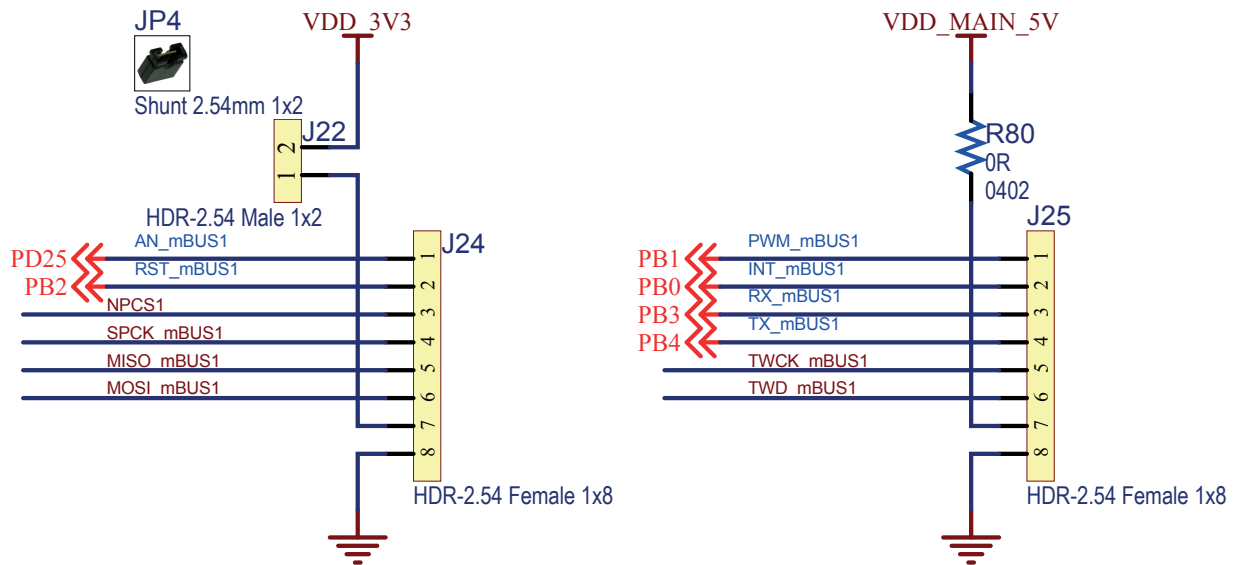


目录

| | |
|--|----|
| 简介..... | 1 |
| 参考文档..... | 1 |
| 前提条件..... | 1 |
| 1. 硬件设计..... | 4 |
| 1.1. 接口..... | 4 |
| 2. 软件设计..... | 5 |
| 2.1. 器件树..... | 5 |
| 2.2. 内核..... | 7 |
| 2.3. Rootfs..... | 8 |
| 2.4. 应用程序..... | 8 |
| 3. 动手实验..... | 11 |
| 4. 工具和实用程序..... | 12 |
| 5. Microchip 外设 I/O Python® (MPIO) | 14 |
| 5.1. Buildroot 中的 MPIO..... | 14 |
| 5.2. 示例..... | 16 |
| 6. 版本历史..... | 17 |
| 6.1. 版本 A——2019 年 9 月..... | 17 |
| Microchip 网站..... | 18 |
| 产品变更通知服务..... | 18 |
| 客户支持..... | 18 |
| Microchip 器件代码保护功能..... | 18 |
| 法律声明..... | 18 |
| 商标..... | 19 |
| 质量管理体系..... | 19 |
| 全球销售及服务网点..... | 20 |

1. 硬件设计

1.1 接口



mikroBUS1 连接器用于简化测试和监视。

要基于 Linux 控制 mikroBUS1 SPI, FLEXCOM4 (SPI 模式) 需与 SAMA5D27-SOM1-EK 上 mikroBUS1 接口的 SPI 总线相连接, 具体如下所述:

FLEXCOM4 SPI

- FLEXCOM4_IO4 → PD0 → NPCS1
- FLEXCOM4_IO2 → PC30 → SPCK_mBUS1
- FLEXCOM4_IO1 → PC29 → MISO_mBUS1
- FLEXCOM4_IO0 → PC28 → MOSI_mBUS1

有关 SAMA5D2 的引脚复用的更多详细信息, 请参见 SAMA5D2 数据手册中的“引脚说明(所有封装)”表。

FLEXCOM I/O 线说明

| 名称 | 说明 | | | 类型 |
|-------------|------------|-----------|------|-----|
| | USART/UART | SPI | TWI | |
| FLEXCOM_IO0 | TXD | MOSI | TWD | I/O |
| FLEXCOM_IO1 | RXD | MISO | TWCK | I/O |
| FLEXCOM_IO2 | SCK | SPCK | - | I/O |
| FLEXCOM_IO3 | CTS | NPCS0/NSS | - | I/O |
| FLEXCOM_IO4 | RTS | NPCS1 | - | O |

2. 软件设计

Microchip Linux 平台是使用 Buildroot 通过以下配置编译的：

`atmel_sama5d27_som1_ek_mmc_dev_defconfig`

SPI 总线驱动程序在此默认配置下工作。

有两种方法可以访问 SPI 总线驱动程序：

- 在内核空间中：
 - 通过 `spi_register_driver()` 接口注册您自己的 SPI 驱动程序，然后通过 `struct spi_device` 句柄访问 SPI 总线驱动程序。
- 在用户空间中：
 - 使能 SPIDEV 内核功能，然后通过器件节点 “/dev/spidev” 访问 SPI 总线驱动程序。

SPIDEV 是一个理想的选择，因为所有应用程序代码都在用户空间中运行，从而更易于进行开发。

在默认配置中，SPIDEV 未使能。以下各部分介绍了 SPIDEV 的使能步骤。

2.1 器件树

- 操作：需要更改
 - 更改 1：
 - 在器件树文件的 FLEXCOM4 器件节点下添加 `spidev` 器件的说明。
- 位置：`buildroot-at91/output/build/linux-linux4sam_6.0/arch/arm/boot/dts`
- 来源：
 - `sama5d2.dtsi`
 - `at91-sama5d27_som1_ek.dts`

sama5d2.dtsi 中 FLEXCOM4 的器件树：

```
flx4: flexcom@fc018000 {
    compatible = "atmel,sama5d2-flexcom"; // 指定哪个驱动程序将用于此
                                           // FLEXCOM 器件
    reg = <0xfc018000 0x200>; // FLEXCOM4 基址为 0xfc018000，大小为 0x200
    clocks = <&flx4_clk>; // uart4 时钟源的定义
    #address-cells = <1>;
    #size-cells = <1>;
    ranges = <0x0 0xfc018000 0x800>;
    status = "disabled"; // 默认为禁止状态，并将替换为 “okay”
};

flx4_clk: flx4_clk {
    #clock-cells = <0>;
    reg = <23>; // FLEXCOM4 的 PID 为 23，此偏移量定义将用于使能
               // PMC 中的 FLEXCOM4 时钟
    atmel,clk-output-range = <0 83000000>; // FLEXCOM4 输入时钟，最大频率为 83 MHz
};
```

at91-sama5d27_som1_ek.dts 中 SPIDEV 功能的器件树：

```
flx4: flexcom@fc018000 {
    atmel,flexcom-mode = <ATMEL_FLEXCOM_MODE_SPI>; // 为此 flexcom 指定 SPI 模式
    status = "okay"; // 使能此器件

    uart6: serial@200 {
        .....
    };

    spi3: spi@400 {
```

```

compatible = "atmel,at91rm9200-spi"; // 指定哪个驱动程序将用于此
                                   // SPI 器件
reg = <0x400 0x200>; // 寄存器偏移地址为 0x400, 大小为 0x200
interrupts = <23 IRQ_TYPE_LEVEL_HIGH 7>; // FLEXCOM4 的 PID 为 23, 高级
                                   // 触发, 优先级为 7
                                   // 用于在 AIC 中配置 FLEXCOM4 中断
clocks = <&flx4_clk>; // FLEXCOM4 时钟源的定义
clock-names = "spi_clk";
pinctrl-names = "default";
pinctrl-0 = <&pinctrl_mikrobus_spi &pinctrl_mikrobus1_spi_cs
&pinctrl_mikrobus2_spi_cs>;
atmel,fifo-size = <16>; // FLEXCOM4 SPI 功能的引脚定义
status = "okay"; /* Conflict with uart6 and i2c3. */ // 将状态属性替换为
                                   // "okay", 使能 SPI 器件

spidev@1 { // 这是更改 1
    compatible = "spidev"; // 指定哪个驱动程序将用于此器件
    reg = <1>; // 此定义将用作 SPIDEV 的 CS 编号
    spi-max-frequency = <1000000>; // 为此 SPIDEV 指定时钟频率
// 检查 buildroot-at91/output/build/linux-linux4sam_6.0/drivers/spi/spi.c of_spi_parse_dt()
// (针对更多选项)
};

i2c3: i2c@600 {
    .....
};

pinctrl_mikrobus1_spi_cs: mikrobus1_spi_cs {
    pinmux = <PIN_PD0__FLEXCOM4_IO4>; // PD0 的多路开关将切换到 FLEXCOM4_IO4
    bias-disable;
};
pinctrl_mikrobus_spi: mikrobus_spi {
    pinmux = <PIN_PC28__FLEXCOM4_IO0>, // PC28 的多路开关将切换到 FLEXCOM4_IO0
    <PIN_PC29__FLEXCOM4_IO1>, // PC29 的多路开关将切换到 FLEXCOM4_IO1
    <PIN_PC30__FLEXCOM4_IO2>; // PC30 的多路开关将切换到 FLEXCOM4_IO2
    bias-disable;
};

```

建议不要直接将“spidev”用作器件树兼容名称。它可能会正常工作，但将显示以下警告：

```

# dmesg | grep spidev
spidev spi.1: buggy DT: spidev listed directly in DT
WARNING: CPU: 0 PID: 1 at drivers/spi/spidev.c:730 0xc045d630

```

由于“spidev”是 Linux 实现的一部分，而不是硬件说明，因此，请勿在没有特定名称的器件树中进行引用。

为避免出现此警告，应选择“spidev”以外的兼容名称，例如：

```

spidev@1 {
    compatible = "atmel,at91rm9200-spi";
    reg = <1>;
    spi-max-frequency = <1000000>;
};

```

然后编辑 SPIDEV 驱动程序文件 buildroot-at91/output/build/linux-linux4sam_6.0/drivers/spi/spidev.c。

向 SPIDEV 驱动程序兼容表中添加新的兼容名称：

```

static const struct of_device_id spidev_dt_ids[] = {
    { .compatible = "rohm,dh2228fv" },
    { .compatible = "lineartechnology,ltc2488" },
    { .compatible = "ge,achc" },
    { .compatible = "semtech,sx1301" },
    { .compatible = "atmel,at91rm9200-spi" },
    {},
};

```

2.2 内核

- 操作：需要更改
- 位置：buildroot-at91/output/build/linux-linux4sam_6.0/
- Defconfig: sama5_defconfig
- 驱动程序文件：
 - drivers/spi/spi.c
 - drivers/spi/spi-atmel.c
 - drivers/spi/spidev.c

添加 SPIDEV 功能的内核配置：

```
user@at91:~/buildroot-at91$ make linux-menuconfig
```

Device Drivers > SPI support > Atmel SPI Controller（器件驱动程序 > SPI 支持 > Atmel SPI 控制器）

这是 Atmel SPI 控制器的驱动程序。

在默认配置中，将选择此项。

```
.config - Linux/arm 4.14.73-linux4sam_6.0 Kernel Configuration
> Device Drivers > SPI support -
SPI support
Arrow keys navigate the menu. <Enter> selects submenus --- (or empty submenu ---). Highlighted letters are
hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < > module capable

--- SPI support
[ ] Debug support for SPI drivers
*** SPI Master Controller Drivers ***
< > Altera SPI Controller
[*] Atmel SPI Controller
< > Atmel USART Controller SPI driver
< > Analog Devices AXI SPI Engine controller
.* Utilities for Bitbanging SPI masters
< > Cadence SPI controller
< > DesignWare SPI controller core support
< * > GPIO-based bitbanging SPI Master
< > Freescale SPI controller and Aeroflex Gaisler GRLIB SPI controller
< > OpenCores tiny SPI
< > Rockchip SPI controller driver
< > NXP SC18IS602/602B/603 I2C to SPI bridge
< > Analog Devices AD-FMCOMMS1-EBZ SPI-I2C-bridge driver
< > Xilinx SPI controller common module
< > Xilinx ZynqMP CQSPI controller
*** SPI Protocol Masters ***
< * > User mode SPI device driver support
< > spi loopback test framework support
< > Infineon TLE62X0 (for power switching)
[ ] SPI slave protocol handlers

<Select> < Exit > < Help > < Save > < Load >
```

Device Drivers > SPI support > User mode SPI device driver support（器件驱动程序 > SPI 支持 > 用户模式 SPI 器件驱动程序支持）

这是 SPIDEV 的驱动程序。选择此项后可使能 SPIDEV 功能。

```

.config - Linux/arm 4.14.73-linux4sam_6.0 Kernel Configuration
> Device Drivers > SPI support -
SPI support
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenu --->). Highlighted letters are
hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module <> module capable

--- SPI support
[ ] Debug support for SPI drivers
*** SPI Master Controller Drivers ***
<> Altera SPI Controller
<*> Atmel SPI Controller
<> Atmel USART Controller SPI driver
<> Analog Devices AXI SPI Engine controller
-* Utilities for Bitbanging SPI masters
<> Cadence SPI controller
<> DesignWare SPI controller core support
<*> GPIO-based bitbanging SPI Master
<> Freescale SPI controller and Aeroflex Gaisler GRLIB SPI controller
<> OpenCores tiny SPI
<> Rockchip SPI controller driver
<> NXP SC18IS602/602B/603 I2C to SPI bridge
<> Analog Devices AD-FMCOMMS1-EBZ SPI-I2C-bridge driver
<> Xilinx SPI controller common module
<> Xilinx ZynqMP QSPI controller
*** SPI Protocol Masters ***
<*> User mode SPI device driver support
<> spi loopback test framework support
<> Infineon TLE62X0 (for power switching)
[ ] SPI slave protocol handlers

<Select> <Exit> <Help> <Save> <Load>

```

2.3 Rootfs

- 操作：无需更改
- 位置：buildroot-at91/output/images/rootfs.tar

器件树文件中不存在 SPI 总线编号的定义，并且 SPI 控制器的总线编号会在注册时自动分配。

例如，为第一个注册的 SPI 控制器分配总线编号 0，为第二个注册的 SPI 控制器分配总线编号 1，以此类推。

以下器件节点用于访问 SPI 总线驱动程序。第一个 1 表示总线编号 1，第二个 1 表示 CS 编号：

- /dev/spidev1.1

2.4 应用程序

本节提供通过器件节点 “/dev/spidev1.1” 访问 SPI 总线驱动程序的 C 语言演示：

编译方式

```
user@at91:~$ buildroot-at91/output/host/bin/arm-buildroot-linux-uclibcgnueabihf-gcc spi_dev.c
-o spi_test
```

源代码

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/ioctl.h>
#include <linux/spi/spidev.h>

#define DEV_SPI "/dev/spidev1.1"

int main(int argc, char *argv[])
{
    int fd;
    int ret;
    unsigned int mode, speed;
    char tx_buf[1];

```



```
char rx_buf[1];
struct spi_ioc_transfer xfer[2] = {0};

// 打开器件节点
fd = open(DEV_SPI, O_RDWR);
if (fd < 0) {
    printf("ERROR open %s ret=%d\n", DEV_SPI, fd);
    return -1;
}

// 设置 spi 模式
mode = SPI_MODE_0;
if (ioctl(fd, SPI_IOC_WR_MODE32, &mode) < 0) {
    printf("ERROR ioctl() set mode\n");
    return -1;
}
if (ioctl(fd, SPI_IOC_RD_MODE32, &ret) < 0) {
    printf("ERROR ioctl() get mode\n");
    return -1;
} else
    printf("mode set to %d\n", (unsigned int)ret);

// 设置 spi 速度
speed = 1*1000*1000;
if (ioctl(fd, SPI_IOC_WR_MAX_SPEED_HZ, &speed) < 0) {
    printf("ERROR ioctl() set speed\n");
    return -1;
}
if (ioctl(fd, SPI_IOC_RD_MAX_SPEED_HZ, &ret) < 0) {
    printf("ERROR ioctl() get speed\n");
    return -1;
} else
    printf("speed set to %d\n", ret);

// 传输数据
tx_buf[0] = 0xa5;
xfer[0].tx_buf = (unsigned long)tx_buf;
xfer[0].len = 1;
xfer[1].rx_buf = (unsigned long)rx_buf;
xfer[1].len = 1;

do {
    if (ioctl(fd, SPI_IOC_MESSAGE(2), xfer) < 0)
        perror("SPI_IOC_MESSAGE");
    usleep(100*1000);
} while (1);

// 关闭器件节点
close(fd);

return 0;
}
```

Buildroot 中提供其他 SPIDEV 应用程序:

```
user@at91:~/buildroot-at91$ make menuconfig
```

代码位于以下位置: buildroot-at91/output/build/spidev_test-v4.10/spidev_test.c

Target Packages > Debugging, profiling and benchmark > spidev_test (目标软件包 > 调试、性能分析和基准测试 > spidev_test)

```
/home/user/pr/bus_demo/bullroot/bullroot-at91/.config - Bullroot 2018.02 Configuration
> Target packages > Debugging, profiling and benchmark
Debugging, profiling and benchmark
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ---). Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> excludes a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [ ] feature is excluded

[*] ltrace
[ ] lttng-babeltrace
[ ] lttng-modules
[ ] lttng-tools
[ ] memstat
[ ] netperf
*** netsniff-ng needs a glibc or musl toolchain w/ threads, headers >= 3.0 ***
*** nmon needs a glibc toolchain ***
[ ] oprofile
[ ] pax-utils
[ ] pv
[ ] ranspeed/smp
[ ] ranspeed
[ ] rt-tests
[*] spidev test
[ ] strace
[ ] stress
*** stress-ng needs a glibc toolchain w/ dynamic library, headers >= 3.3 ***
[ ] sysdig
[ ] tcf-agent
[ ] tinymembench
[ ] trace-cmd
[ ] trinity
[ ] uclibc-ng-test
[ ] valgrind
[ ] whetstone

<Select> <Exit> <Help> <Save> <Load>
```

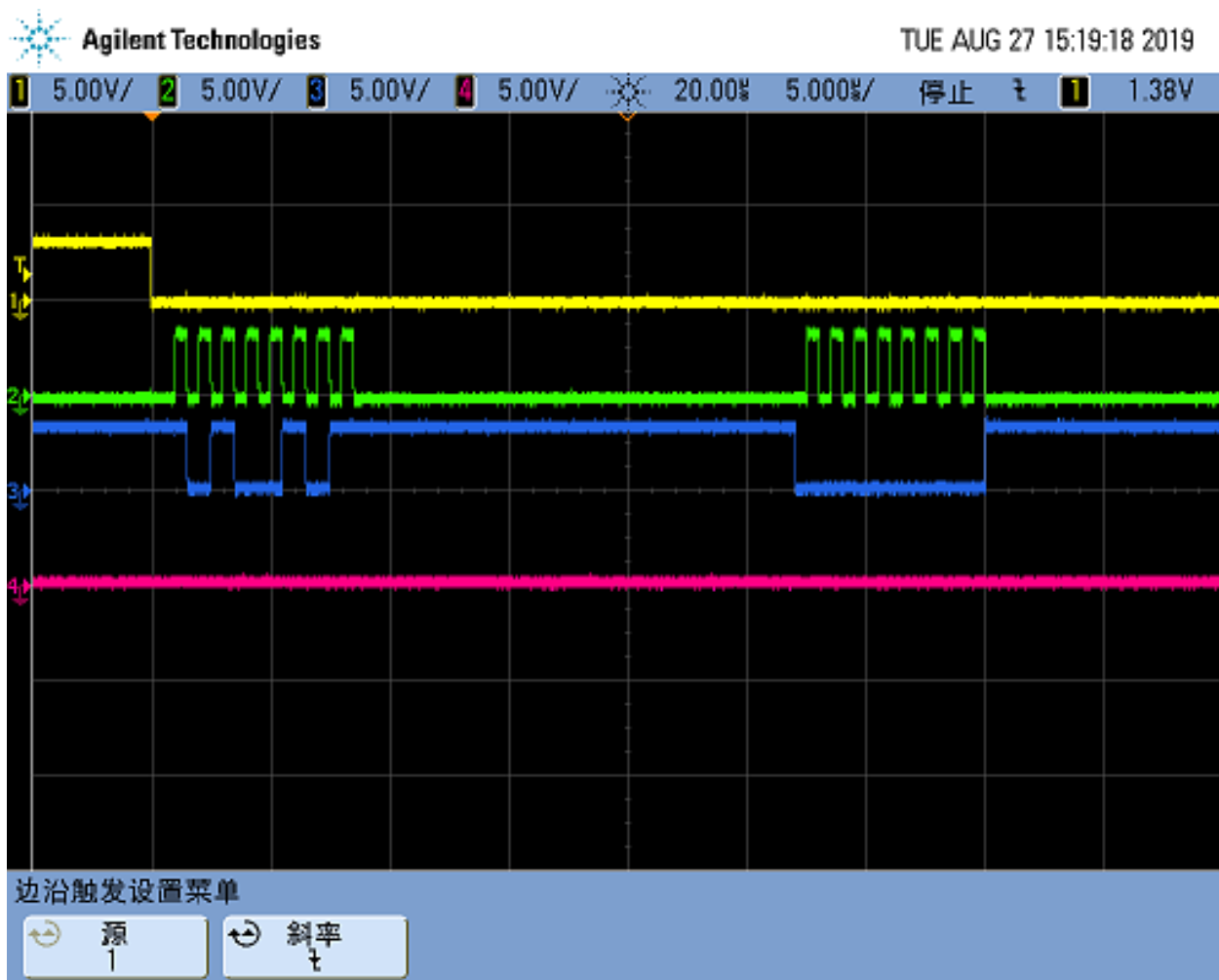
3. 动手实验

将 spi_test 应用程序复制到目标并执行，然后在 mikroBUS1 SPI 总线上监视 SPI 波形。

```
# chmod +x spi_test  
# ./spi_test
```

图注

- 黄线: NPCS1
- 绿线: SPCK_mBUS1
- 蓝线: MOSI_mBUS1
- 红线: MISO_mBUS1



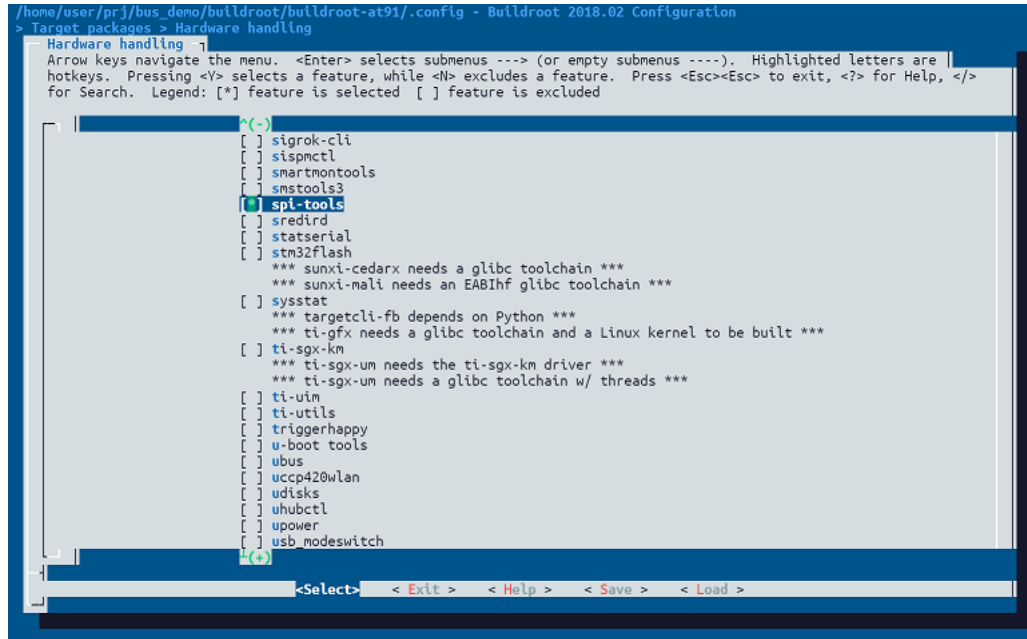
4. 工具和实用程序

Spi-tools 是一款用于执行 SPI 总线测试的工具，包含在 Buildroot 中。

使用默认的 Buildroot 配置时，将选择此工具。

```
user@at91:~/buildroot-at91$ make menuconfig
```

Target packages > Hardware handling > spi-tools (目标软件包 > 硬件处理 > spi-tools)



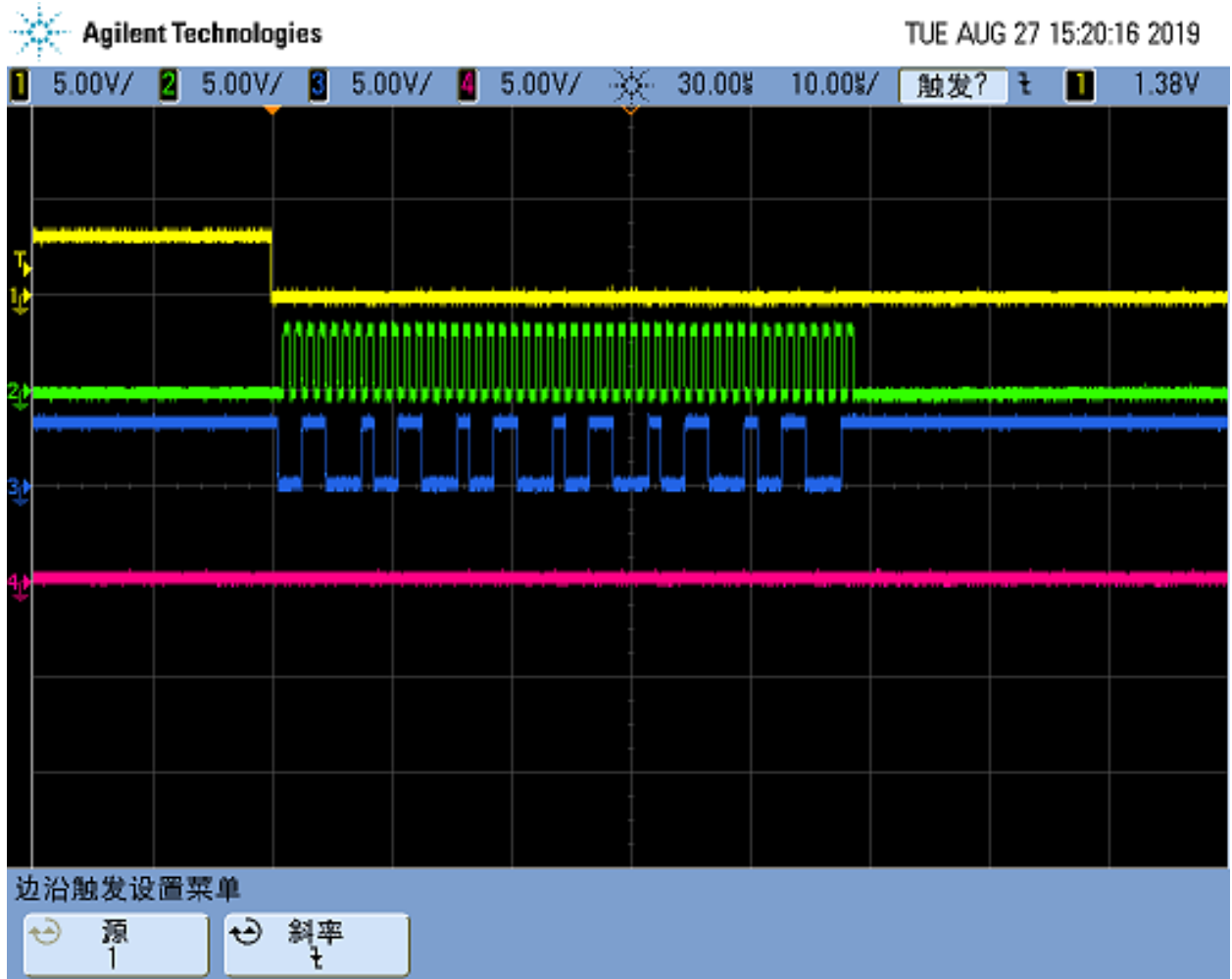
spi-tools 中有两条命令：

```
# spi-config -h
usage: spi-config options...
options:
  -d --device=<dev>  use the given spi-dev character device.
  -q --query          print the current configuration.
  -m --mode=[0-3]    use the selected spi mode:
                    0: low iddle level, sample on leading edge,
                    1: low iddle level, sample on trailing edge,
                    2: high iddle level, sample on leading edge,
                    3: high iddle level, sample on trailing edge.
  -l --lsb={0,1}    LSB first (1) or MSB first (0).
  -b --bits=[7...]  bits per word.
  -s --speed=<int>  set the speed in Hz.
  -h --help          this screen.
  -v --version       display the version number.
# spi-config -d /dev/spidev1.1 -q
/dev/spidev1.1: mode=0, lsb=0, bits=8, speed=1000000
# spi-pipe -h
usage: spi-pipe options...
options:
  -d --device=<dev>  use the given spi-dev character device.
  -b --blocksize=<int> transfer block size in byte.
  -n --number=<int>  number of blocks to transfer (-1 = infinite).
  -h --help          this screen.
  -v --version       display the version number.
# spi-pipe -d /dev/spidev1.1 -b 6 -n 1
111111
```

输入六个“1”，然后按 Enter 键。相应地将从示波器捕捉波形。

图注

- 黄线: NPCS1
- 绿线: SPCK_mBUS1
- 蓝线: MOSI_mBUS1
- 红线: MISO_mBUS1



5. Microchip 外设 I/O Python® (MPIO)

Microchip 外设 I/O (MPIO) Python 软件包可轻松访问运行 Linux 的 Microchip MPU 处理器和评估板上的各种硬件外设。这套 API 干净、一致、灵活、记录完备且经过全面测试，即使是最复杂的硬件外设，也可通过它来轻松浏览和使用。

更多信息，请参见 <https://github.com/linux4sam/mpio>。文件夹 `mpio/examples` 中提供了相关代码示例来说明如何使用 MPIO 接口模块。

5.1 Buildroot 中的 MPIO

要在 Buildroot 配置中充分利用 MPIO 的优势，请执行以下步骤：

1. 使能 Python

```
user@at91:~/buildroot-at91$ make menuconfig
```

选择 “python” 以使能 python 支持：

- Target packages > Interpreter languages and scripting > [*] python (目标软件包 > 解释程序语言和脚本 > [*] python)

然后进入 “python module format to install” (要安装的 python 模块格式) 并选择 “.py sources and .pyc compiled” (.py 源和已编译.pyc)。

- Target packages > Interpreter languages and scripting > python > python module format to install > .py sources and .pyc compiled (目标软件包 > 解释程序语言和脚本 > python > 要安装的 python 模块格式 > .py 源和已编译.pyc)

```

/home/user/tmp/mpio/test/buildroot-at91/.config - Buildroot 2018.02 Configuration
> Target packages > Interpreter languages and scripting - core python modules
Interpreter languages and scripting
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> selects a feature, while <N> excludes a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [ ] feature is excluded

[ ] 4th
[ ] enscript
[ ] erlang
[ ] execline
[ ] ftcl
[ ] gauche
[ ] guile
[ ] haserl
[ ] janvn
[ ] jintcl
[ ] lua
[ ] luajtt
[ ] micropython
[ ] moarvm
[ ] mono
[ ] nodejs
[ ] perl
[ ] php
[*] python
    python module format to install (.py sources and .pyc compiled) --->
        core python modules --->
        External python modules --->
[ ] ruby
[ ] tcl

```

必须选择一些其他的 python 模块。进入 “core python modules” (内核 python 模块)，然后选择 “curses module” (curses 模块)、“readline” 和 “hashlib module” (hashlib 模块)。

- Target packages > Interpreter languages and scripting > core python modules > [*] curses module (目标软件包 > 解释程序语言和脚本 > 内核 python 模块 > [*] curses 模块)
- Target packages > Interpreter languages and scripting > core python modules > [*] readline (目标软件包 > 解释程序语言和脚本 > 内核 python 模块 > [*] readline)

- Target packages > Interpreter languages and scripting > core python modules > [*] hashlib module

```

/home/user/tmp/mpio/test/buildroot-at91/.config - Buildroot 2018.02 Configuration
> Target packages > Interpreter languages and scripting > core python modules
core python modules
Arrow keys navigate the menu. <Enter> selects submenus --- (or empty submenu ----). Highlighted letters are hotkeys. Pressing <Y> selects a
feature, while <N> excludes a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [ ] feature is
excluded

*** The following modules are unusual or require extra libraries ***
[ ] bzip2 module (NEW)
[ ] bsddb module (NEW)
[ ] codescsjk module (NEW)
[*] curses module
[ ] ossaudiodev module (NEW)
[*] readline
[ ] ssl (NEW)
[ ] unicodedata module
[ ] sqlite module
[ ] xml module (NEW)
[ ] zlib module (NEW)
[*] hashlib module

```

(目标软件包 > 解释程序语言和脚本 > 内核 python 模块 > [*] hashlib 模块)

进入 “External python modules” (外部 python 模块)，然后选择 “python-setuptools”。

- Target packages > Interpreter languages and scripting > External python modules > [*] python-setuptools (目标软件包 > 解释程序语言和脚本 > 外部 python 模块 > [*] python-setuptools)

```

/home/user/tmp/mpio/test/buildroot-at91/.config - Buildroot 2018.02 Configuration
> Target packages > Interpreter languages and scripting > External python modules
External python modules
Arrow keys navigate the menu. <Enter> selects submenus --- (or empty submenu ----). Highlighted letters are hotkeys. Pressing <Y> selects a
feature, while <N> excludes a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [ ] feature is
excluded

^(-)- [ ] python-pyudev (NEW)
[ ] python-pyusb (NEW)
[ ] python-pyxb (NEW)
[ ] python-pyyaml (NEW)
[ ] python-pyzmq (NEW)
[ ] python-raven (NEW)
[ ] python-rem (NEW)
[ ] python-requests (NEW)
[ ] python-requests-oauthlib (NEW)
[ ] python-requests-toolbelt (NEW)
[ ] python-rpi-gpio (NEW)
[ ] python-rtslib-fb (NEW)
[ ] python-scandir (NEW)
[ ] python-schedule (NEW)
[ ] python-sdnottyfy (NEW)
[ ] python-secretstorage (NEW)
[ ] python-see (NEW)
[ ] python-serial (NEW)
[ ] python-service-identity (NEW)
[ ] python-setproctitle (NEW)
[*] python-setuptools
[ ] python-sh (NEW)
[ ] python-shutilwhich (NEW)

```

2.使能 MPIO 模块

进入 “External options” (外部选项) 并选择 “python-mpio”。

- External options > [*] python-mpio (外部选项 > [*] python-mpio)

```

/home/user/tmp/mpio/test/buildroot-at91/.config - Buildroot 2018.02 Configuration
> External options
External options
Arrow keys navigate the menu. <Enter> selects submenus --- (or empty submenu ----). Highlighted letters are hotkeys. Pressing <Y> selects a
feature, while <N> excludes a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [ ] feature is
excluded

*** MCHP (in /home/user/tmp/mpio/test/buildroot-external-microchip) ***
[ ] gl-decoder
*** gst1-at91 needs a toolchain w/ glibc ***
*** libplanes depends on libdrm, cairo, lua, and cJSON ***
[ ] ptc_examples
*** python-iocontrol depends on pyqt5 and mpio ***
[*] python-mpio
[ ] dt-overlay-at91
Init configuration (none) --->

```

3.完成 Buildroot 配置并进行编译

进入 “Filesystem images” (文件系统映像) 并将 rootfs 的确切大小设置为 120 MB。

- Filesystem images > (120M) exact size (文件系统映像 > (120M) 确切大小)

```

/home/user/tmp/mpio/test/buildroot-at91/.config - Buildroot 2018.02 Configuration
Filesystem Images
Filesystem Images
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----). Highlighted letters are hotkeys. Pressing <Y> selects a
feature, while <N> excludes a feature. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] feature is selected [ ] feature is
excluded

[ ] axfs root filesystem
[ ] cloop root filesystem for the target device
[ ] cpio the root filesystem (for use as an initial RAM filesystem)
[ ] cramfs root filesystem
[*] ext2/3/4 root filesystem
    ext2/3/4 variant (ext4) --->
    ( ) filesystem label
    (120M) exact size
    (0) exact number of inodes (leave at 0 for auto calculation)
    (5) reserved blocks percentage
    (-O ^64bit) additional mke2fs options
    Compression method (no compression) --->

```

保存后，以下新设置将添加到 Buildroot 的配置文件中：

```

.....
BR2_PACKAGE_PYTHON=y
BR2_PACKAGE_PYTHON_PY_PYC=y
BR2_PACKAGE_PYTHON_CURSES=y
BR2_PACKAGE_PYTHON_READLINE=y
BR2_PACKAGE_PYTHON_HASHLIB=y
BR2_PACKAGE_PYTHON_SETUPTOOLS=y
BR2_PACKAGE_PYTHON_MPIO=y
BR2_TARGET_ROOTFS_EXT2_SIZE="120M"
.....

```

然后重新配置并编译 buildroot：

```

user@at91:~/buildroot-at91$ make atmel_sama5d27_som1_ek_mmc_dev_defconfig
user@at91:~/buildroot-at91$ make

```

5.2 示例

编译成功后，使用 buildroot-at91/output/images/sdcard.img 烧写 SD 卡。

在目标板上执行 python 代码，例如：

```

# ./adc2.py DEVICE
# ./gpio1.py PIN
# ./pwm_led.py DEVICE CHANNEL
.....

```

注：python 示例代码可从 <https://github.com/linux4sam/mpio/examples> 获得

6. 版本历史

6.1 版本 A——2019 年 9 月

第一版。

Microchip 网站

Microchip 网站 (<http://www.microchip.com/>) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。我们的网站提供以下内容：

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及归档软件
- **一般技术支持**——常见问题解答 (FAQ)、技术支持请求、在线讨论组以及 Microchip 设计伙伴计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

产品变更通知服务

Microchip 的产品变更通知服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请访问 <http://www.microchip.com/pcn>，然后按照注册说明进行操作。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师 (ESE)
- 技术支持

客户应联系其代理商、代表或 ESE 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过 <http://www.microchip.com/support> 获得网上技术支持。

Microchip 器件代码保护功能

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿意与关心代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 (Digital Millennium Copyright Act)》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

法律声明

提供本文档的中文版本仅为为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担

保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和/或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。除非另外声明，否则在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、Microchip 徽标、Adaptec、AnyRate、AVR、AVR 徽标、AVR Freaks、BesTime、BitCloud、chipKIT、chipKIT 徽标、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、HELDO、IGLOO、JukeBlox、KeeLoq、Kleer、LANCheck、LinkMD、maXStylus、maXTouch、MediaLB、megaAVR、Microsemi、Microsemi 徽标、MOST、MOST 徽标、MPLAB、OptoLyzzer、PackerTime、PIC、picoPower、PICSTART、PIC32 徽标、PolarFire、Prochip Designer、QTouch、SAM-BA、SenGenuity、SpyNIC、SST、SST 徽标、SuperFlash、Symmetricom、SyncServer、Tachyon、TempTrackr、TimeSource、tinyAVR、UNI/O、Vectron 及 XMEGA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的注册商标。

APT、ClockWorks、The Embedded Control Solutions Company、EtherSynch、FlashTec、Hyper Speed Control、HyperLight Load、IntelliMOS、Libero、motorBench、mTouch、Powermite 3、Precision Edge、ProASIC、ProASIC Plus、ProASIC Plus 徽标、Quiet-Wire、SmartFusion、SyncWorld、Temux、TimeCesium、TimeHub、TimePictra、TimeProvider、Vite、WinPath 和 ZL 均为 Microchip Technology Incorporated 在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BlueSky、BodyCom、CodeGuard、CryptoAuthentication、CryptoAutomotive、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherGREEN、In-Circuit Serial Programming、ICSP、INICnet、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNet 徽标、memBrain、Mindi、MiWi、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、SAM-ICE、Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、ViewSpan、WiperLock、Wireless DNA 和 ZENA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的商标。

SQTP 为 Microchip Technology Incorporated 在美国的服务标记。

Adaptec 徽标、Frequency on Demand、Silicon Storage Technology 和 Symmcom 均为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

GestIC 为 Microchip Technology Inc. 的子公司 Microchip Technology Germany II GmbH & Co. KG 在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2020, Microchip Technology Incorporated 版权所有。

ISBN:

AMBA、Arm、Arm7、Arm7TDMI、Arm9、Arm11、Artisan、big.LITTLE、Cordio、CoreLink、CoreSight、Cortex、DesignStart、DynamIQ、Jazelle、Keil、Mali、Mbed、Mbed Enabled、NEON、POP、RealView、SecurCore、Socrates、Thumb、TrustZone、ULINK、ULINK2、ULINK-ME、ULINK-PLUS、ULINKpro、µVision 和 Versatile 均为 Arm Limited（或其子公司）在美国和/或其他国家/地区的商标或注册商标。

质量管理体系

有关 Microchip 的质量管理体系的信息，请访问 <http://www.microchip.com/quality>。

全球销售及服务中心

| 美洲 | 亚太地区 | 亚太地区 | 欧洲 |
|--|---|--|--|
| 公司总部 2355 West Chandler Blvd. Chandler, AZ 85224-6199 电话: 480-792-7200 传真: 480-792-7277 技术支持: http://www.microchip.com/support 网址: http://www.microchip.com | 澳大利亚 - 悉尼 电话: 61-2-9868-6733 中国 - 北京 电话: 86-10-8569-7000 中国 - 成都 电话: 86-28-8665-5511 中国 - 重庆 电话: 86-23-8980-9588 中国 - 东莞 电话: 86-769-8702-9880 中国 - 广州 电话: 86-20-8755-8029 中国 - 杭州 电话: 86-571-8792-8115 中国 - 香港特别行政区 电话: 852-2943-5100 中国 - 南京 电话: 86-25-8473-2460 中国 - 青岛 电话: 86-532-8502-7355 中国 - 上海 电话: 86-21-3326-8000 中国 - 沈阳 电话: 86-24-2334-2829 中国 - 深圳 电话: 86-755-8864-2200 中国 - 苏州 电话: 86-186-6233-1526 中国 - 武汉 电话: 86-27-5980-5300 中国 - 西安 电话: 86-29-8833-7252 中国 - 厦门 电话: 86-592-2388138 中国 - 珠海 电话: 86-756-3210040 | 印度 - 班加罗尔 电话: 91-80-3090-4444 印度 - 新德里 电话: 91-11-4160-8631 印度 - 浦那 电话: 91-20-4121-0141 日本 - 大阪 电话: 81-6-6152-7160 日本 - 东京 电话: 81-3-6880-3770 韩国 - 大邱 电话: 82-53-744-4301 韩国 - 首尔 电话: 82-2-554-7200 马来西亚 - 吉隆坡 电话: 60-3-7651-7906 马来西亚 - 槟榔屿 电话: 60-4-227-8870 菲律宾 - 马尼拉 电话: 63-2-634-9065 新加坡 电话: 65-6334-8870 台湾地区 - 新竹 电话: 886-3-577-8366 台湾地区 - 高雄 电话: 886-7-213-7830 台湾地区 - 台北 电话: 886-2-2508-8600 泰国 - 曼谷 电话: 66-2-694-1351 越南 - 胡志明市 电话: 84-28-5448-2100 | 奥地利 - 韦尔斯 电话: 43-7242-2244-39 传真: 43-7242-2244-393 丹麦 - 哥本哈根 电话: 45-4485-5910 传真: 45-4485-2829 芬兰 - 埃斯波 电话: 358-9-4520-820 法国 - 巴黎 电话: 33-1-69-53-63-20 传真: 33-1-69-30-90-79 德国 - 加兴 电话: 49-8931-9700 德国 - 哈恩 电话: 49-2129-3766400 德国 - 海尔布隆 电话: 49-7131-72400 德国 - 卡尔斯鲁厄 电话: 49-721-625370 德国 - 慕尼黑 电话: 49-89-627-144-0 传真: 49-89-627-144-44 德国 - 罗森海姆 电话: 49-8031-354-560 以色列 - 若那那市 电话: 972-9-744-7705 意大利 - 米兰 电话: 39-0331-742611 传真: 39-0331-466781 意大利 - 帕多瓦 电话: 39-049-7625286 荷兰 - 德卢内市 电话: 31-416-690399 传真: 31-416-690340 挪威 - 特隆赫姆 电话: 47-72884388 波兰 - 华沙 电话: 48-22-3325737 罗马尼亚 - 布加勒斯特 电话: 40-21-407-87-50 西班牙 - 马德里 电话: 34-91-708-08-90 传真: 34-91-708-08-91 瑞典 - 哥德堡 电话: 46-31-704-60-40 瑞典 - 斯德哥尔摩 电话: 46-8-5090-4654 英国 - 沃金厄姆 电话: 44-118-921-5800 传真: 44-118-921-5820 |
| 亚特兰大 德卢斯, 佐治亚州 电话: 678-957-9614 传真: 678-957-1455 奥斯汀, 德克萨斯州 电话: 512-257-3370 波士顿 韦斯特伯鲁, 马萨诸塞州 电话: 774-760-0087 传真: 774-760-0088 芝加哥 艾塔斯卡, 伊利诺伊州 电话: 630-285-0071 传真: 630-285-0075 达拉斯 阿迪森, 德克萨斯州 电话: 972-818-7423 传真: 972-818-2924 底特律 诺维, 密歇根州 电话: 248-848-4000 休斯顿, 德克萨斯州 电话: 281-894-5983 印第安纳波利斯 诺布尔斯特维尔, 印第安纳州 电话: 317-773-8323 传真: 317-773-5453 电话: 317-536-2380 洛杉矶 米慎维荷, 加利福尼亚州 电话: 949-462-9523 传真: 949-462-9608 电话: 951-273-7800 罗利, 北卡罗来纳州 电话: 919-844-7510 纽约, 纽约州 电话: 631-435-6000 圣何塞, 加利福尼亚州 电话: 408-735-9110 电话: 408-436-4270 加拿大 - 多伦多 电话: 905-695-1980 传真: 905-695-2078 | | | |