AN3293

How to Use the SAMA5D2 GPIO Under Linux®

Introduction

This application note describes how to get started using the SAMA5D2 GPIO under Linux.

Accessing the GPIO pins in user space is easier since the GPIO chip model has been introduced into the kernel.

Refer to the section Hands-On for the two ways to interact with a GPIO device in Linux user space: as GPIO sysfs is deprecated since Linux 4.8, user space should use the GPIO device node.

Refer to the section Tools and Utilities for an easier way to interact with the GPIO device node using *libgpiod*.

Reference Documents

Title	Reference	Available
SAMA5D2 Series Datasheet	DS60001476	https://www.microchip.com/design-centers/32-bit-mpus
SAMA5D27 SOM1 Kit1 User Guide	DS50002667	https://www.microchip.com/DevelopmentTools/ProductDetails/PartNO/ATSAMA5D27-SOM1-EK1

Prerequisites

- Hardware
  - PC
  - SAMA5D27 SOM1 Evaluation Kit (Part Number: ATSAMA5D27-SOM1-EK1)
  - SDCard
- Software
  This demo runs on the AT91 Linux platform built by Buildroot. The first step is to set up the AT91 Buildroot development environment. Refer to the web site: http://www.at91.com/linux4sam/bin/view/Linux4SAM/BuildRoot

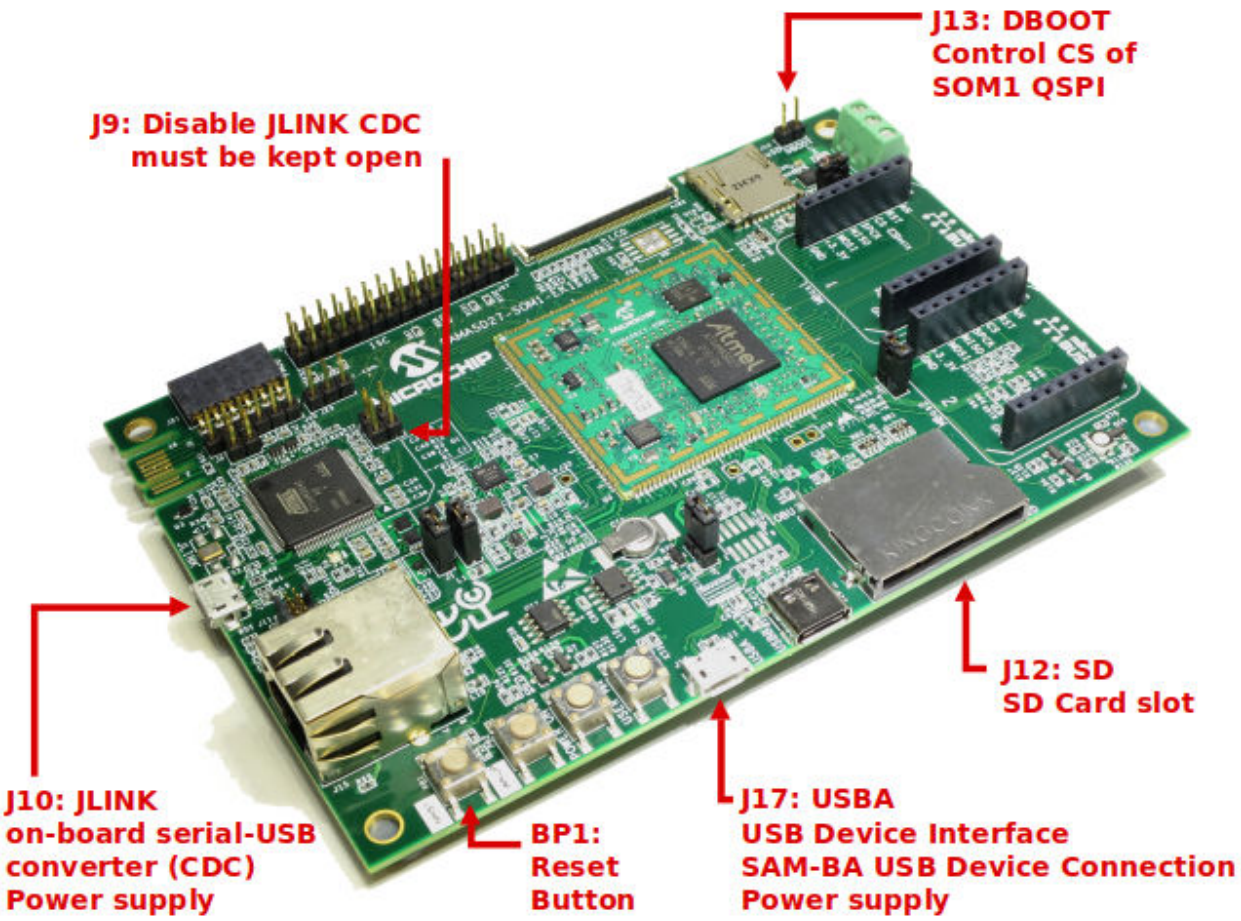
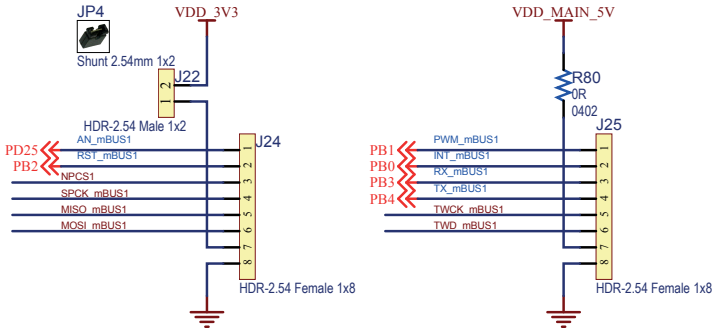

J13: DBOOT
Control CS of
SOM1 QSPI
J9: Disable JLINK CDC
must be kept open
J12: SD
SD Card slot
J10: JLINK
on-board serial-USB
converter (CDC)
Power supply
BP1:
Reset
Button
J17: USBA
USB Device Interface
SAM-BA USB Device Connection
Power supply

Table of Contents

1. Hardware Design

1.1 Interface



The connector mikroBUS1 is used for easy testing and monitoring.

This application note describes how to control GPIO PB2 (J24 pin 2) under Linux.

2. Software Design

The Microchip Linux platform was built using Buildroot with the following configuration:

```
atmel_sama5d27_som1_ek_mmc_dev_defconfig
```

The GPIO device driver works under this default configuration without any adjustment.

2.1 Device Tree

Action: no need to change

Location: buildroot-at91/output/build/linux-linux4sam_6.0/arch/arm/boot/dts

Sources: sama5d2.dtsi

Device Tree for GPIO:

```
pioA: pinctrl@fc038000 {
    compatible = "atmel,sama5d2-pinctrl";
 // specify which driver will be used for this pioA device


    reg = <0xfc038000 0x600>;
 // pioA base address is 0xfc038000, size is 0x600


    interrupts = <18 IRQ_TYPE_LEVEL_HIGH 7>,
 // 128 gpios were divided into four banks


           <68 IRQ_TYPE_LEVEL_HIGH 7>,
 // each gpio bank has its own irq line
 //check buildroot-at91/output/build/linux-linux4sam_6.0/include/dt-bindings/interrupt-
 controller/irq.h
 // for the definitions of TRQ_TYPE….


           <69 IRQ_TYPE_LEVEL_HIGH 7>,
           <70 IRQ_TYPE_LEVEL_HIGH 7>;
    interrupt-controller;
    #interrupt-cells = <2>;
    gpio-controller;
    #gpio-cells = <2>;
    clocks = <&pioA_clk>;
 // definition for pioA clock source


};

pioA_clk: pioA_clk {
 #clock-cells = <0>;
 reg = <18>;
 // PID of pioA is 18, this definition of offset will be used to enable pioA clock in PMC


 atmel,clk-output-range = <0 83000000>;
 // pioA input clock, max frequency is 83MHz


};
```

2.2 Kernel

Action: no need to change

Location: buildroot-at91/output/build/linux-linux4sam_6.0/

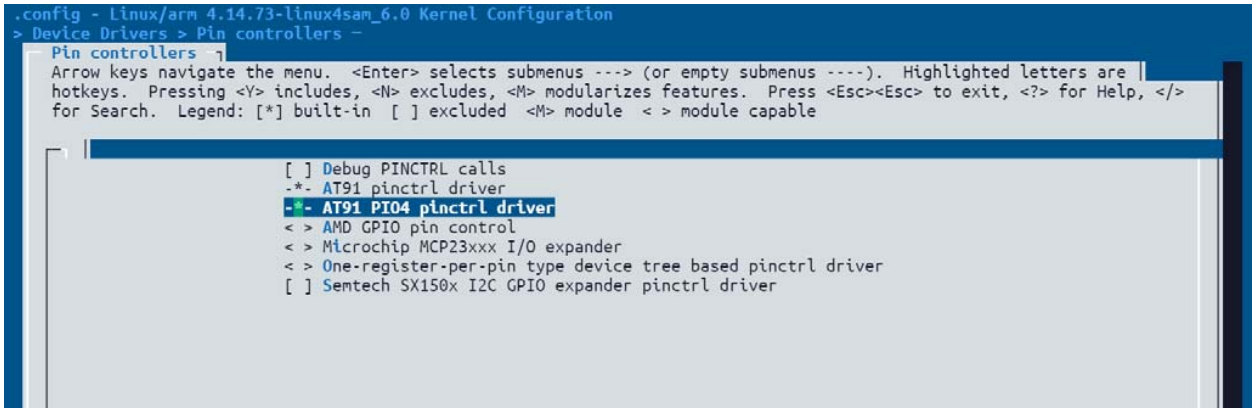
Defconfig: sama5_defconfig

Driver files: drivers/pinctrl/pinctrl-at91-pio4.c

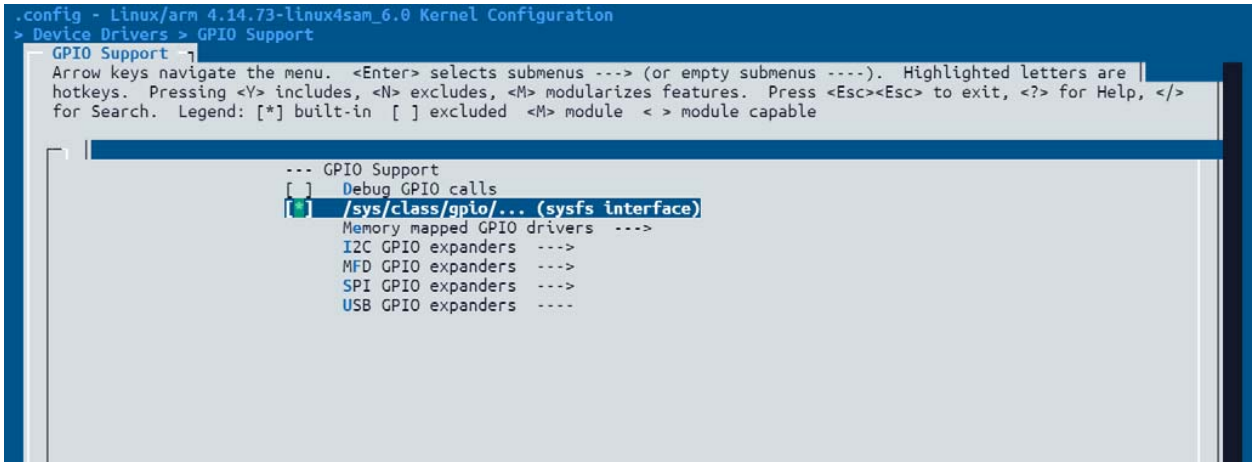
Check the kernel configuration for the GPIO function:

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

Device Drivers > Pin controllers > AT91 PIO4 pinctrl driver
With this setting, AT91 pinctrl and the GPIO driver are built into the kernel. The GPIO driver can then be accessed via a device node in rootfs (/dev/gpiochip0).



Device Drivers > GPIO Support > /sys/class/gpio/… (sysfs interface)
With this setting, sysfs for the GPIO feature is built into the kernel. The GPIO driver can then be accessed via sysfs in rootfs (/sys/class/gpio).



2.3 Rootfs

Action: no need to change

Location: buildroot-at91/output/images/rootfs.tar

Two paths (file nodes) can be used to access the GPIO driver:

- /dev/gpiochip0
  - The dev node interface can only be accessed by the C language, because most operations must be done by ioctl().
- /sys/class/gpio
  - NOT recommended: since Linux 4.8, the GPIO sysfs interface is deprecated. See 4. Tools and Utilities. The sysfs interface is more user-friendly, since all required operations can be done by read() and write(). It is normally used in script programs or on command lines.

2.4 Application

The following is a C language demo for accessing the GPIO driver based on a dev node:

- Compilation

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

- Source code

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
#include <linux/gpio.h>
#include <sys/ioctl.h>

#define DEV_GPIO  "/dev/gpiochip0"

int main(int argc, char *argv[])
{
    int fd;
    int ret;

    struct gpiochip_info cinfo;
    struct gpioline_info linfo;
    struct gpiohandle_request req;
    struct gpiohandle_data data;

    /* open gpio */
    fd = open(DEV_GPIO, 0);
    if (fd < 0) {
        printf("ERROR: open %s ret=%d\n", DEV_GPIO, fd);
        return -1;
    }

    /* get gpio chip info */
    ret = ioctl(fd, GPIO_GET_CHIPINFO_IOCTL, &cinfo);
    if (ret < 0) {
        printf("ERROR get chip info ret=%d\n", ret);
        return -1;
    }
    printf("GPIO chip: %s, \"%s\", %u GPIO lines\n",
                    cinfo.name, cinfo.label, cinfo.lines);
    ret = ioctl(fd, GPIO_GET_LINEINFO_IOCTL, &linfo);
    if (ret < 0) {
        printf("ERROR get line info ret=%d\n", ret);
        return -1;
    }
    printf("line %2d: %s\n", linfo.line_offset,
                    linfo.name);

    /* set gpio_pb2 output */


 // 128 gpio in gpiochip0
     // 0   ~ 31    PA0 -> PA31
     // 32 ~ 63   PB0 -> PB31
     // 33 ~ 95   PC0 -> PC31
     // 96 ~ 127 PD0 -> PD31
```

```
    req.lineoffsets[0] = 34;
    req.lines = 1;
    req.flags = GPIOHANDLE_REQUEST_OUTPUT;
    strcpy(req.consumer_label, "RST_mBUS1");
    int lhfd = ioctl(fd, GPIO_GET_LINEHANDLE_IOCTL, &req);
    if (lhfd < 0) {
        printf("ERROR get line handle lhdf=%d\n", lhfd);
        return -1;
    }
    data.values[0] = 1;
    ret = ioctl(req.fd, GPIOHANDLE_SET_LINE_VALUES_IOCTL, &data);
    if (ret < 0) {
        printf("ERROR set line value ret=%d\n", ret);
        return -1;
    }

    while (1) {


 // set gpio_pb2 low


        data.values[0] = 0;
        ioctl(req.fd, GPIOHANDLE_SET_LINE_VALUES_IOCTL, &data);
        usleep(5*1000);



 // set gpio_pb2 high


        data.values[0] = 1;
        ioctl(req.fd, GPIOHANDLE_SET_LINE_VALUES_IOCTL, &data);
        usleep(5*1000);
    }

    /* close gpio */
    close(fd);

    return 0;
}
```

3. Hands-On

The GPIO driver can be accessed via dev node or sysfs.

3.1 Access Via dev node

Copy the gpio_test application to the target and execute it. This generates a 100 Hz wave via GPIO PB2 (J24 pin 2).

```
# chmod +x gpio_test
# ./gpio_test
```

3.2 Access Via sysfs

1. Export PB2.

   ```
   # echo 34 > /sys/class/gpio/export
   ```

2. Set PB2 output.

   ```
   # echo out > /sys/class/gpio/PB2/direction
   ```

3. Set PB2 low.

   ```
   # echo 0 > /sys/class/gpio/PB2/value
   ```

4. Set PB2 high.

   ```
   # echo 1 > /sys/class/gpio/PB2/value
   ```

Pins are identified by numbers recognizable by the Linux kernel. For example, the four GPIO ports are identified as PAx, PBx, PCx and PDx. Each port covers 32 GPIOs, so PA0 corresponds to 0, PA5 to 5, PB0 to 32, etc. The calculation is done according to the number of pins and ports of the MPU GPIOs.

The following table shows the correspondence between the SAMA5D27 pins and the GPIO port numbers.

Table 3-1. SAMA5D27 Pin Numbers vs. GPIO Port Numbers

SAMA5D27 Pin Numbers	GPIO Port Numbers
0 - 31	PA0 - PA31
32 - 63	PB0 - PB31
33 - 95	PC0 - PC31
96 - 127	PD0 - PD31

4. Tools and Utilities

libgpiod is a C library and tools for interacting with the Linux GPIO character device via the /dev/gpiochipX device node. It encapsulates ioctl calls and data structures behind a straightforward API. *libgpiod* makes it easy to access the GPIO driver with less ioctl calls.

4.1 *libgpiod* Support

With the default configuration atmel_sama5d27_som1_ek_mmc_dev_defconfig, the *libgpiod* feature was not enabled.

Enable *libgpiod* support in Buildroot:

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

- Target packages > Libraries > Hardware handling > **[*] libgpiod**
- Target packages > Libraries > Hardware handling > **[*] install tools**

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

After building successfully, burn your SD card with an updated sdcard.img.

4.2 *libgpiod* APIs

libgpiod provides a simpler API to access the GPIO driver. Follow the C header file for these APIs:

```
user@at91:~/buildroot-at91$ vim output/build/libgpiod-0.3.2/include/gpiod.h
```

The *libgpiod* tools in the following folder can be used as demo codes to learn how to work with *libgpiod* APIs.

```
user@at91:~/buildroot-at91$ cd output/build/libgpiod-0.3.2/src/tools/
user@at91:~/buildroot-at91/output/build/libgpiod-0.3.2/src/tools$ ls *.c
gpiodetect.c  gpiofind.c  gpioget.c  gpioinfo.c  gpiomon.c  gpioset.c  tools-common.c
```

4.3 *libgpiod* Tools

libgpiod provides simple tools to access the GPIO driver via a command line.

It should be noted that for these *libgpiod* tools, the GPIO sysfs interface is deprecated from Linux 4.8. User space can use the character device instead. The *libgpiod* tools can then be used to access simply the GPIO driver via a command line instead of the GPIO sysfs interface.

Six commands are associated with the *libgpiod* tools:

- gpiodetect – lists all gpiochips present on the system, their names, labels and number of GPIO lines
- gpioinfo – lists all lines of specified gpiochips, their names, consumers, direction, active state and additional flags
- gpioget – reads values of specified GPIO lines
- gpioset – sets values of specified GPIO lines, potentially keeps the lines exported and waits until timeout, user input or signal
- gpiofind – finds the gpiochip name and line offset when given the line name
- gpiomon – waits for events on GPIO lines, specifies which events to watch, how many events to process before exiting or whether the events should be reported to the console

For details, visit https://git.kernel.org/pub/scm/libs/libgpiod/libgpiod.git/about/.

4.4 Using *libgpiod* Tools on SAMA5D27-SOM1-EK1

1. Detect the GPIO chip on the target.

```
# gpiodetect
gpiochip0 [fc038000.pinctrl] (128 lines)
```

2. Print information about each line.

```
# gpioinfo
gpiochip0 - 128 lines:
        line   0:        "PA0"       unused   input  active-high
        line   1:        "PA1"       unused   input  active-high
        line   2:        "PA2"       unused   input  active-high
        line   3:        "PA3"       unused   input  active-high
        line   4:        "PA4"       unused   input  active-high
        line   5:        "PA5"       unused   input  active-high
        line   6:        "PA6"       unused   input  active-high
        line   7:        "PA7"       unused   input  active-high
        line   8:        "PA8"       unused   input  active-high
        line   9:        "PA9"       unused   input  active-high
[..]
```

3. Find the GPIO chip name and offset for GPIO PB2.

```
# gpiofind PB2
gpiochip0 34
```

4. Set PB2 output high.

```
# gpioset gpiochip0 34=1
```

5. Set PB2 output low.

```
# gpioset gpiochip0 34=0
```

6. Toggle PB2 high for 1 second.

```
# gpioset --mode=time --sec=1 gpiochip0 34=0
```

7. Monitor the PB2 pin status.

```
# gpiomon gpiochip0 34
event: FALLING EDGE offset: 34 timestamp: [1325983345.255958082]
event:  RISING EDGE offset: 34 timestamp: [1325983345.256686960]
event: FALLING EDGE offset: 34 timestamp: [1325983348.205010375]
event:  RISING EDGE offset: 34 timestamp: [1325983348.577229302]
event: FALLING EDGE offset: 34 timestamp: [1325983348.657488131]
event:  RISING EDGE offset: 34 timestamp: [1325983348.695700717]
event: FALLING EDGE offset: 34 timestamp: [1325983348.830615058]
event:  RISING EDGE offset: 34 timestamp: [1325983349.395371156]
event: FALLING EDGE offset: 34 timestamp: [1325983349.938427156]
event:  RISING EDGE offset: 34 timestamp: [1325983349.938456229]
```

5. Microchip Peripheral I/O Python® (MPIO)

The Microchip Peripheral I/O (MPIO) Python package provides easy access to various hardware peripherals found on Microchip MPU processors and evaluation boards running Linux. The API is clean, consistent, flexible, documented, and well tested. It makes navigating and exercising even the most complex hardware peripherals a trivial task.

For more information, see https://github.com/linux4sam/mpio. Code examples showing how to work with the MPIO interface modules are provided in the folder mpio/examples.

5.1 MPIO in buildroot

In order to benefit from MPIO in your buildroot configuration, follow the steps below:

1. Enable Python

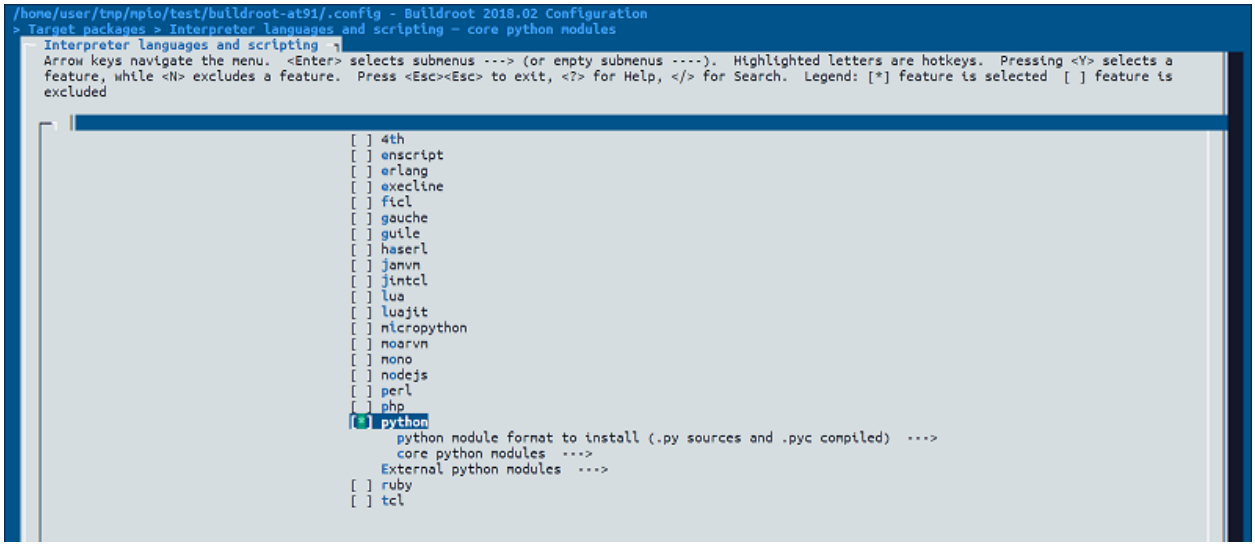
```
user@at91:~/buildroot-at91$ make menuconfig
```

Select “**python**” to enable python support:

- Target packages > Interpreter languages and scripting > [*] **python**

Then enter “**python module format to install**” and select “**.py sources and .pyc compiled**”.

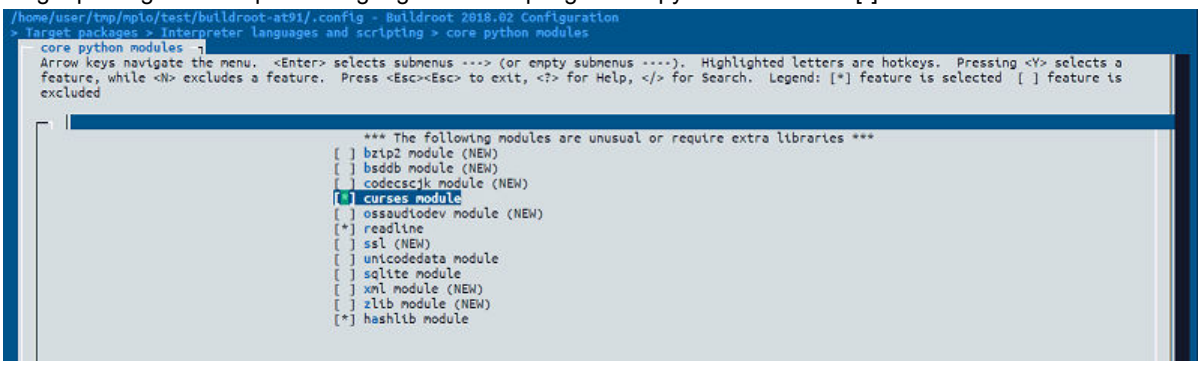
- Target packages > Interpreter languages and scripting > python > python module format to install > **.py sources and .pyc compiled**



Some additional python modules must be selected. Enter “**core python modules**” and select “**curses module**”, “**readline**” and “**hashlib module**”.

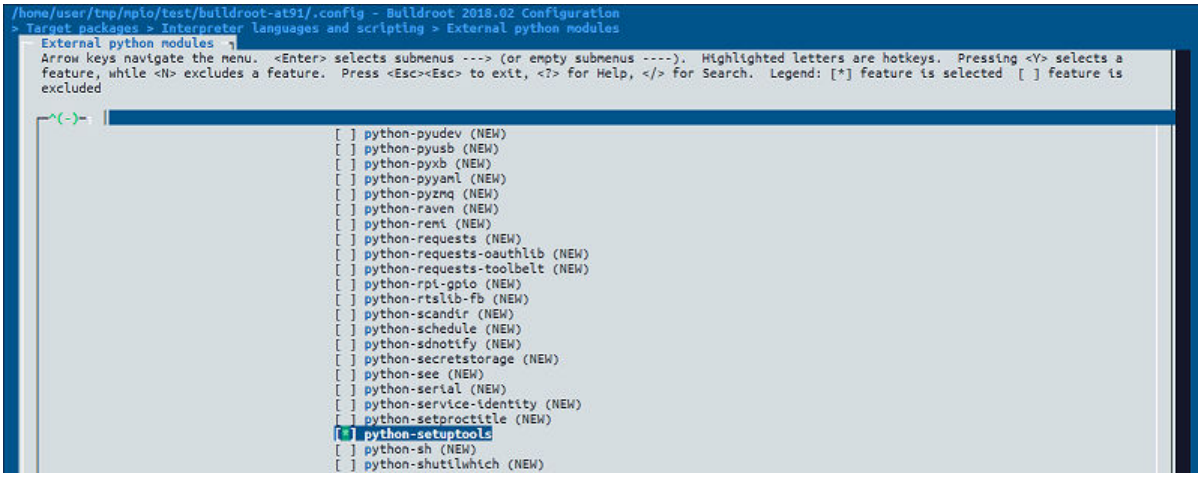
- Target packages > Interpreter languages and scripting > core python modules > [*] **curses module**
- Target packages > Interpreter languages and scripting > core python modules > [*] **readline**

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



Enter “**External python modules**” and select “**python-setuptools**”.

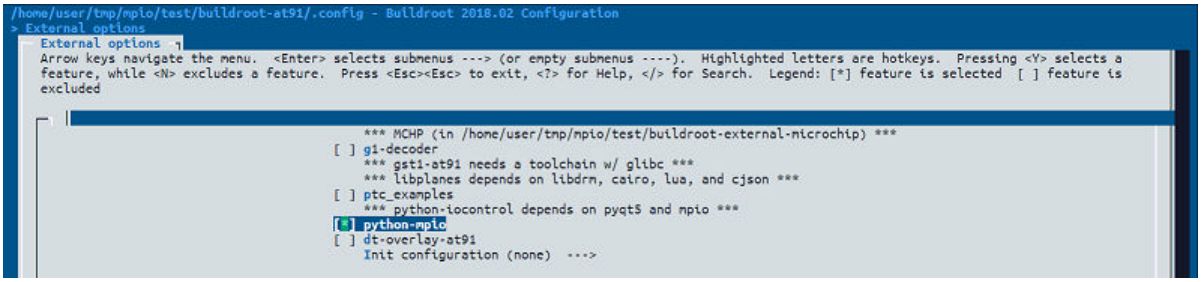
- Target packages > Interpreter languages and scripting > External python modules > **[*] python-setuptools**



2. Enable the MPIO Module

Enter “**External options**” and select “**python-mpio**”.

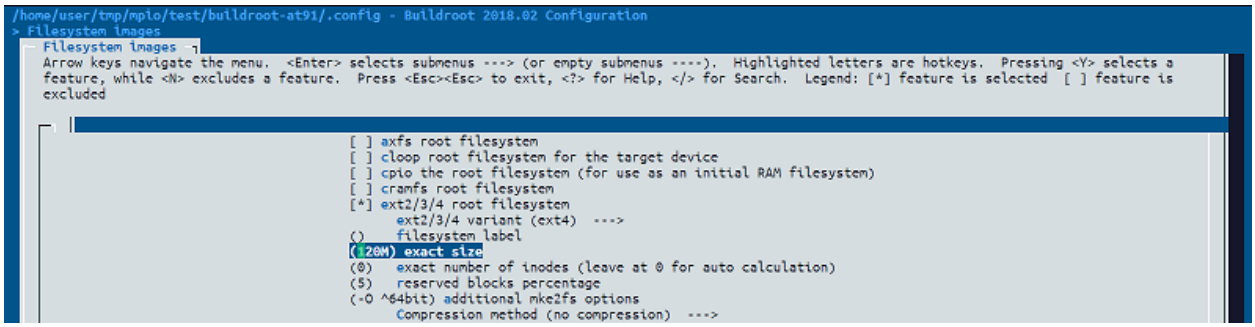
- External options > **[*] python-mpio**



3. Finish the buildroot Configuration and Build

Enter “**Filesystem images**” and set the exact size of rootfs to 120MB.

- Filesystem images > **(120M) exact size**



After saving, the following new settings are added to the configuration file of buildroot:

……

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"

……

Then re-configure and build buildroot:

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

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

5.2 Examples

After building successfully, burn your SD card with buildroot-at91/output/images/**sdcard.img**.

Execute the python codes on the target board, for example:

./adc2.py DEVICE

./gpio1.py PIN

./pwm_led.py DEVICE CHANNEL

……

Note: The python example code can be found in https://github.com/linux4sam/mpio/examples

6. Revision History

6.1 Rev. A - 10/2019

First issue.

The Microchip Website

Microchip provides online support via our website at http://www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to http://www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: http://www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with

your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PackeTime, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TempTrackr, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

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 logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, Vite, WinPath, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

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 logo, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, 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, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

ISBN: 978-1-5224-5181-5

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, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit http://www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: http://www.microchip.com/support Web Address: http://www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4450-2828 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820