



PMIC hardware components



Contents

1. PMIC hardware components	3
2. Category:ST boards	7
3. OP-TEE overview	7
4. Regulator overview	7
5. Watchdog overview	8
6. U-Boot overview	8
7. TF-A overview	8



PMIC hardware components

Stable: 11.06.2020 - 12:48 / Revision: 11.06.2020 - 12:24

Template:ArticleMainWriter Template:ArticleApprovedVersion

Contents

1 Article purpose	3
2 Software frameworks	3
3 STPMIC1	4
3.1 Description	4
3.2 Support in Linux Kernel	4
3.2.1 Core driver	4
3.2.2 Regulator driver	5
3.2.3 Watchdog driver	5
3.2.4 Input driver	5
3.2.5 Kernel Configuration	6
3.3 Support in U-BOOT	6
3.4 Support in Cortex-A7 Secure	6
3.4.1 TF-A	6
3.4.2 OP-TEE	7

1 Article purpose

The purpose of this article is to:

- list the PMIC hardware components that might be integrated in the different boards
- link these components to the corresponding software framework(s)
- point to the datasheet(s) of these components
- explain, when necessary, how to configure these components.

2 Software frameworks

Do	Peri	Software frameworks	Comment
mai Cor tex -A7	Cor tex -A7 no		



Do	Peri	Software frameworks			Comment
Main Secure (OP-TEE)	n-secure (Linux)	Cortex-M4 (STM32Cube)			
		OP-TEE overview	Regulator overview Watchdog overview		
PMIC	STPMIC1				

3 STPMIC1

3.1 Description

The STPMIC1 is a device that handles the power supplies for some STM32MP boards. The STPMIC1 main features are:

- configuration via I²C bus,
- 10 regulators of different kinds with over-current protection,
- 3 power switches for USB supplies,
- a power-on key input,
- a watchdog,
- a thermal protection,
- a NORMAL mode and an ALTERNATE mode that is used for system suspend.

The user manual is available here: NOT YET AVAILABLE

3.2 Support in Linux Kernel

The STPMIC1 driver is used to handle STPMIC1 Power Management Integrated Circuit (PMIC). The driver is composed of several parts:

- The core driver
- The regulator driver
- The watchdog driver
- The input driver

The thermal feature is not supported.

3.2.1 Core driver

The core implements the Multi-Function Devices Framework API.

- It is probed by the I²C framework.
- It handles communication with the PMIC via I²C.



PMIC hardware components

- It probes the others STPMIC1 drivers (input, watchdog...).
- It acts as an interrupt controller for the other drivers.

Source code:

```
include/linux/mfd/stpmic1.h  
drivers/mfd/stpmic1.c
```

Bindings are described in:

```
Documentation/devicetree/bindings/mfd/stpmic1.txt
```

3.2.2 Regulator driver

The driver implements the [Regulator framework API](#)

- Each BUCK, LDO or POWER SWITCH inside the STPMIC1 device is presented as a voltage regulator.
- Power switches are presented as fixed voltage sources. Voltage can not be set.
- The driver does not handle the suspend configuration. This is done by the Secure Monitor.

Source code:

```
drivers/regulator/stpmic1_regulator.c
```

Bindings are described in:

```
Documentation/devicetree/bindings/regulator/stpmic1-regulator.txt
```

3.2.3 Watchdog driver

The driver implements the [Watchdog framework API](#)

When enabled, a watchdog device is available for the user-land. As soon as a user has started to write in the watchdog it is armed in the PMIC. When the watchdog timer expires, the PMIC shuts down.

Source code:

```
drivers/watchdog/stpmic1_wdt.c
```

Bindings are described in:

```
Documentation/devicetree/bindings/watchdog/stpmic1-wdt.txt
```

3.2.4 Input driver

The driver implements the [Input framework API](#).

The driver exposes a single key (KEY_POWER) that can be used as any standard input device in /dev/input/eventX

Source code:



```
drivers/input/misc/stpmic1_onkey.c
```

Bindings are described in:

```
Documentation/devicetree/bindings/input/stpmic1-onkey.txt
```

3.2.5 Kernel Configuration

With kernel menuconfig, enable following configurations:

- core part: MFD_STPMIC1
- regulator: REGULATOR_STPMIC1
- watchdog: STPMIC1_WATCHDOG
- input: INPUT_STPMIC1_ONKEY

3.3 Support in U-BOOT

STPMIC1 is used by U-Boot to:

- Control regulators used by other drivers (mmc-supply for SDCard for example, usb vbus)
- Trace pmic startup reason

STPMIC1 is supported with existing uclass of the 'Driver Model' described in [doc/driver-model/pmic-framework.txt](#).

- pmic :
 - uclass: [drivers/power/regulator/regulator-uclass.c](#) , CONFIG_DM_PMIC
 - driver: [drivers/power/pmic/stpmic1.c](#) , CONFIG_PMIC_STPMIC1
- regulator:
 - uclass: [drivers/power/pmic/pmic-uclass.c](#) , CONFIG_DM_REGULATOR
 - driver: [drivers/power/regulator/stpmic1.c](#) , CONFIG_DM_REGULATOR_STPMIC1
- sysreset:
 - uclass: [drivers/sysreset/sysreset-uclass.c](#) , CONFIG_SYSRESET
 - code in pmic driver

The STPMIC1 is available with the existing commands:

- pmic (CONFIG_CMD_PMIC)
- regulator (CONFIG_CMD_REGULATOR)
- poweroff (CONFIG_CMD_POWEROFF)

STPMIC1 driver supports configuration via device-tree; the bindings, same as kernel, are described in:

```
doc/device-tree-bindings/mfd/stpmic1.txt
```

3.4 Support in Cortex-A7 Secure

3.4.1 TF-A

STPMIC1 is used by TF-A firmware to:

- Configure DDR power supplies.
- Configure the regulators for system suspend and system shutdown.



Driver source code:

```
include/drivers/st/stpmic1.h
drivers/st/pmic/stpmic1.c
```

SPTMIC1 driver supports configuration via device-tree, bindings are described in:

```
docs/fdts/stm32mp/st, stpmic1.txt
```

3.4.2 OP-TEE

STPMIC1 is used by OP-TEE OS to configure the regulators for system suspend and system shutdown.

Driver source code:

```
core/include/drivers/stpmic1.h
core/drivers/stpmic1.c
```

Power Management Integrated Circuit

Open Portable Trusted Execution Environment

Application programming interface

Low-dropout regulator

Multifunction device

Trusted Firmware for Arm Cortex-A

Doubledata rate (memory domain)

Operating System

Permission error

Stable: 17.06.2020 - 15:27 / Revision: 16.01.2020 - 13:36

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer

Permission error

Stable: 13.05.2020 - 08:56 / Revision: 13.05.2020 - 08:54

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer



Permission error

Stable: 11.06.2020 - 12:50 / Revision: 11.06.2020 - 12:12

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer

Permission error

Stable: 31.01.2020 - 13:22 / Revision: 31.01.2020 - 13:13

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer

Permission error

Stable: 25.05.2020 - 07:32 / Revision: 25.05.2020 - 07:25

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer

Permission error

Stable: 10.06.2020 - 06:49 / Revision: 04.05.2020 - 10:14

You do not have permission to read this page, for the following reason:

The action "Read pages" for the draft version of this page is only available for the groups ST_editors, ST_readers, Selected_editors, sysop, reviewer