CONFIDENTIAL B

Keypad customization for Android O

Agenda

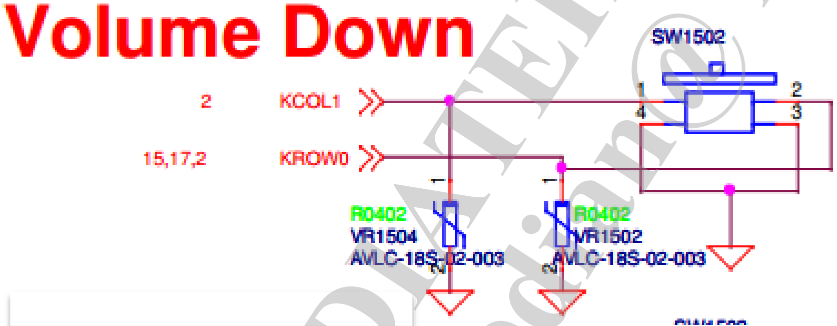
- keywords
- keypad hardware introduction
- keypad relative dws and dts setting
- Software of keypad dts
- kernel config items for keypad
- key use EINT mode
- long press reboot function
- keypad in lk and preloader
- keypad debug

Key words

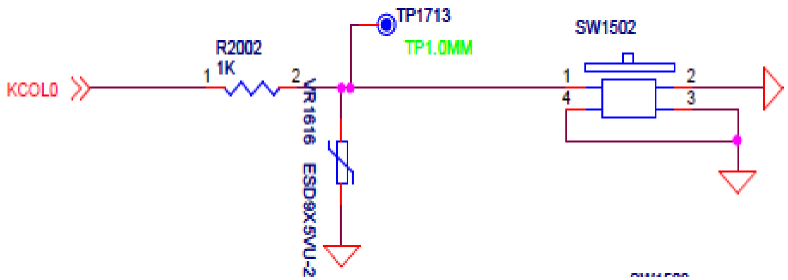
- <proj>
  - project name, e.g. tb8183m1_64_bsp
- <platform>
  - platform name, e.g. mt8183
- <kernel_ver>
  - linux kernel version e.g. kernel-4.4
- <arm_ver>:
  - arm or arm64

Keypad HW

- there are 3 types of circuit for one key： KCOL + KROW; KCOL + GND; GPIO(EINT) + GND.
- KCOL + KROW and KCOL + GND
  - Only GPIOs that with KCOL/KROW function can be used for this type.
  - keypad scanner will scan the whole matrix to detect which key is pressed or released.



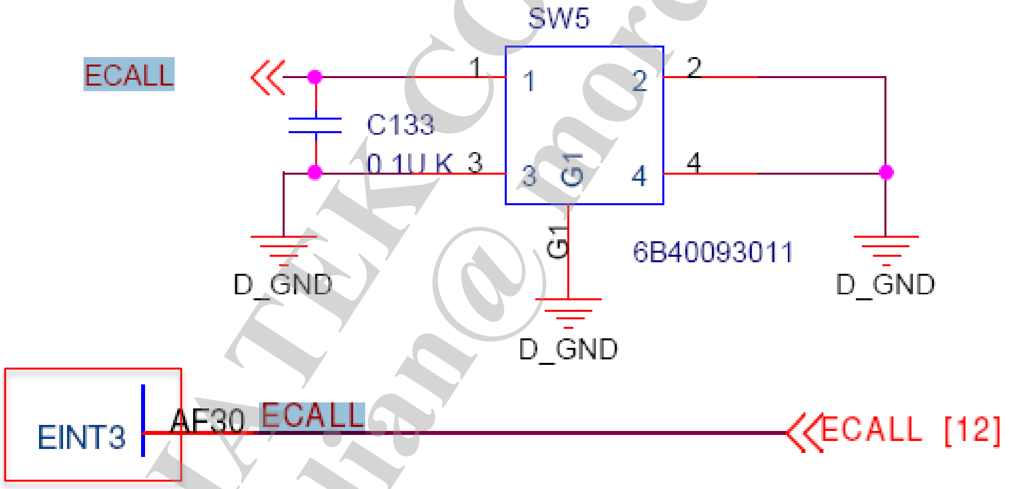
KCOL + KROW



KCOL + GND

Keypad HW

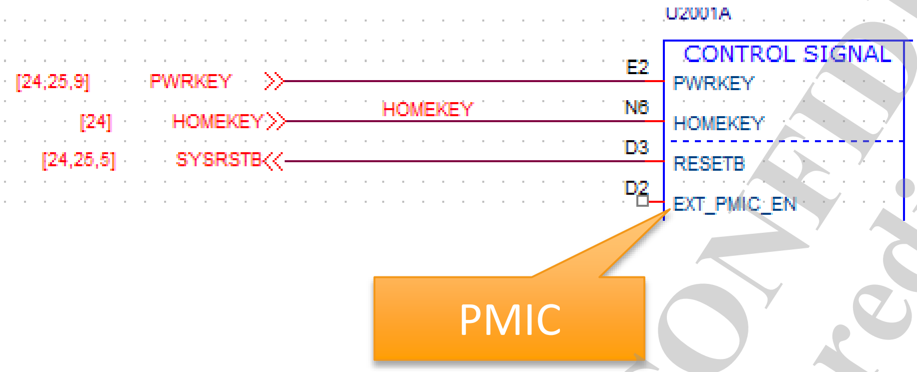
- GPIO(EINT) + GND
  - The GPIO must with EINT function.
  - Detect key press or release with interrupt handler.
  - HW circuit shows as follows

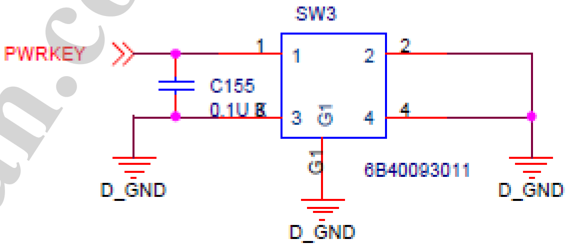


GPIO(EINT) + GND

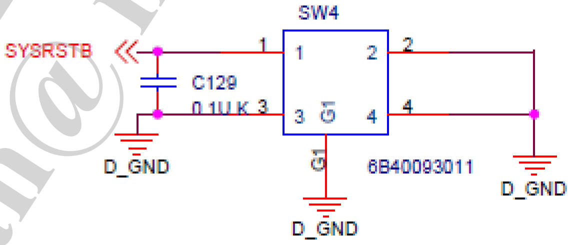
Keypad HW (PMIC key)

- Power Key connect to PMIC





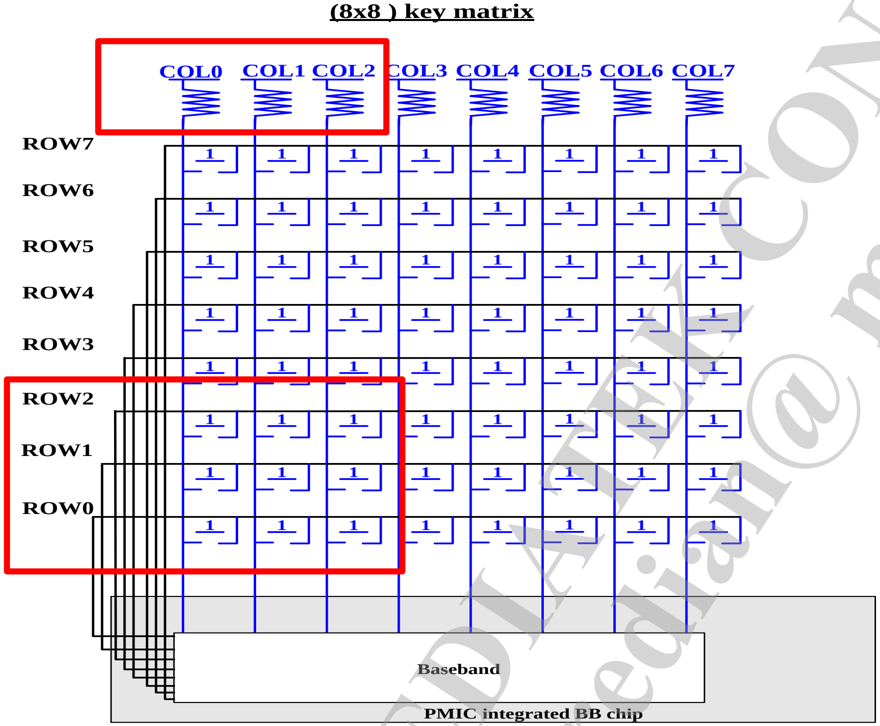
- PMIC RST key(Home key) can be used as volumeup or volumedown key



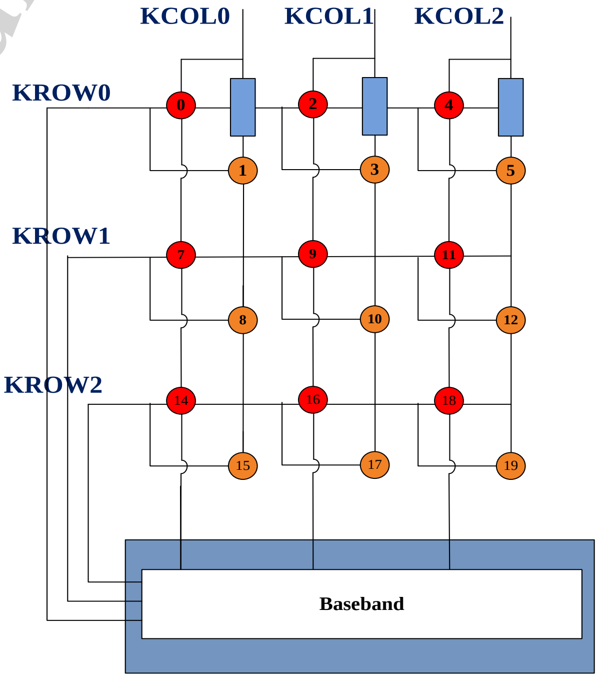
- Power key also can use GPIO(EINT) + GND mode

keypad scanner(1/5)

- MT8735 keypad scanner support 3*3 key matrix.
- support single key type and double key type
  - 3 x 3 single key (At most 9 keys)
  - 3 x 3 double key (At most 18 keys)

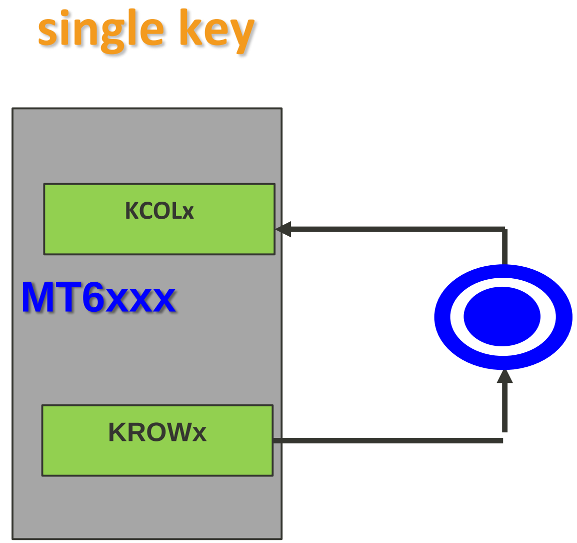


single key

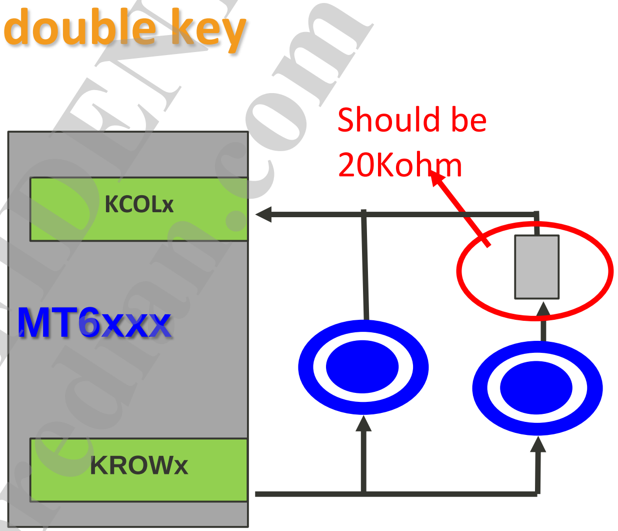


double key

keypad scanner(2/5)—double key



One KCOL and one KROW connected to one key

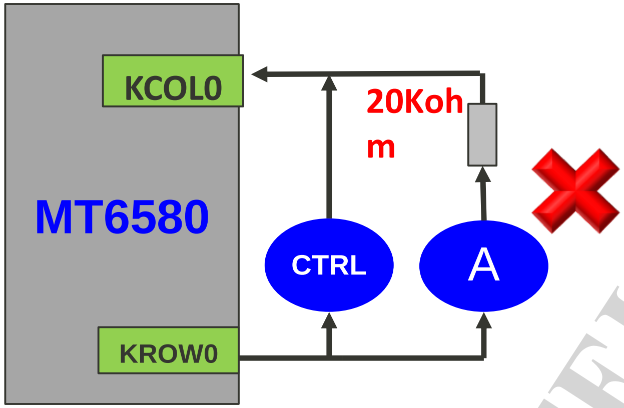


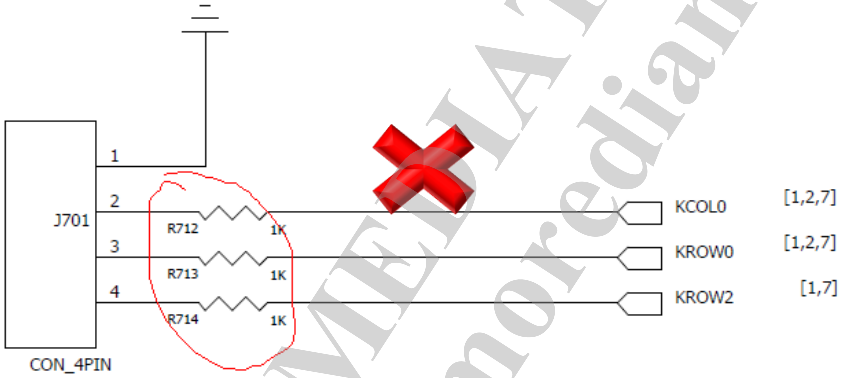
One KCOL and one KROW connected to 2 keys

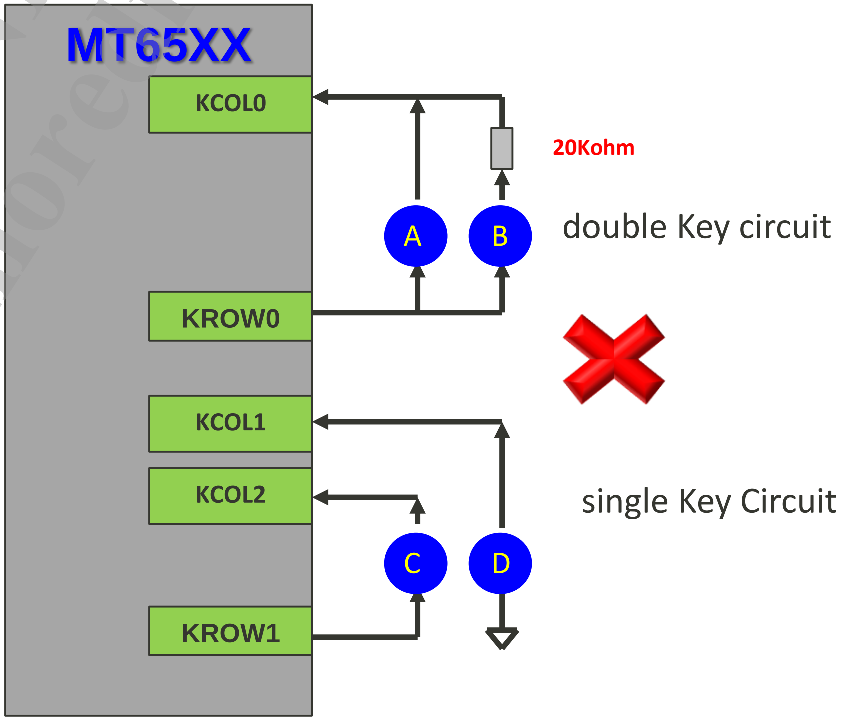
- double key type
  - Up to two keys can be connected to same KCOL and KROW simultaneously.
  - There should be a 20Kohm on one Key path

keypad scanner(3/5)—double key

- Don’t assign keys which will be pressed at the same time on same KCOL and KROW . (Ex: CTRL and A may be pressed at the same time.)
- Don’t implement the circuit in both key type(single and double key mixd)
- Do not add 1K ohm resistor to extend key layout, Voltage division would cause misjudgment

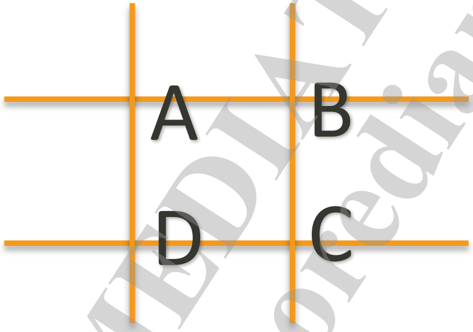






keypad scanner(5/5)--HW Limitation

- Single keypad
  - More than two keys pressed simultaneously in a specific pattern will retrieve the wrong information. e.g. A,B,C pressed, D would be detected too.
- Double keypad
  - It cannot detect two keys pressed simultaneously when the two keys are in the same group
  - Including the HW limitation of single keypad



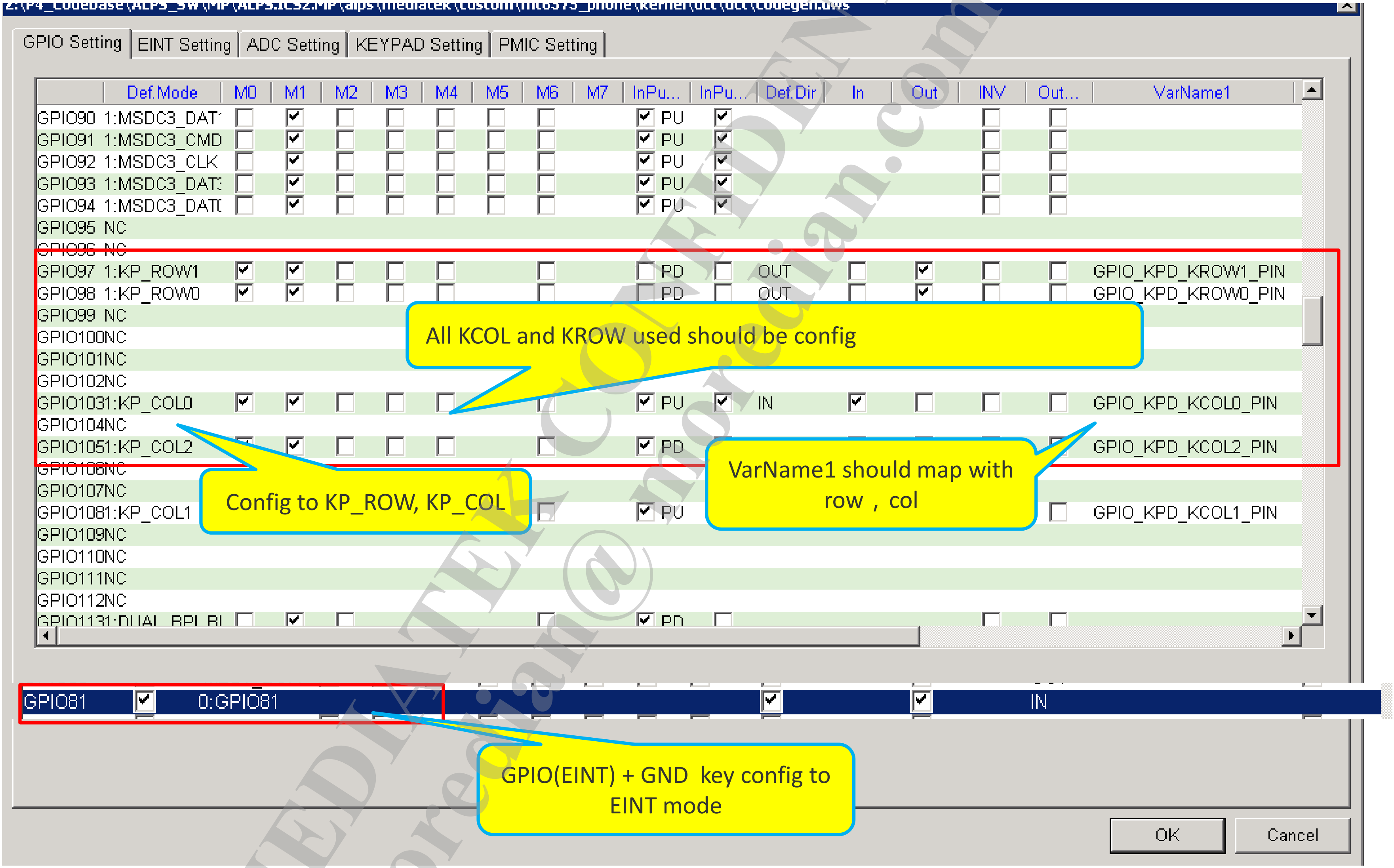
keypad dws setting

- GPIO setting in dws
- “KEYPAD” page setting in dws
  - keypad matrix
  - Power key setting
  - factory, recovery, download key... setting
  - extend type(double key)
  - PMIC home key(RST key)
- keypad scanner debounce

keypad dws setting--GPIO

- Only and all GPIOs that can work with KROW (or KCOL) mode need to check.
- DefMode and varName1 are important
- KCOL + GND only need to config KCOL
- set KCOL as: INPUT + PULL ENABLE + PULL UP
- set KROW as: OUTPUT + PULL ENABLE + PULL DOWN
- VarName1 should be just like
  - GPIO_KPD_KROW0_PIN
  - GPIO_KPD_KCOL0_PIN
  - GPIO_KPD_KCOL1_PIN
- GPIO(EINT) + GND type need to set that GPIO to EINT mode

keypad dws setting--GPIO



Z:\P4_Codebase\ALPS_SW\MP\ALPS.ICS2.MP\alps\mediatek\custom\mt6575_phone\kernel\dct\dct\codegen.dws

	Def.Mode	M0	M1	M2	M3	M4	M5	M6	M7	InPu...	InPu...	Def.Dir	In	Out	INV	Out...	VarName1
GPIO90	1:MSDC3_DAT'	□	☑	□	□	□	□	□		☑ PU	☑				□	□	
GPIO91	1:MSDC3_CMD	□	☑	□	□	□	□	□		☑ PU	☑				□	□	
GPIO92	1:MSDC3_CLK	□	☑	□	□	□	□	□		☑ PU	☑				□	□	
GPIO93	1:MSDC3_DAT:	□	☑	□	□	□	□	□		☑ PU	☑				□	□	
GPIO94	1:MSDC3_DAT(	□	☑	□	□	□	□	□		☑ PU	☑				□	□	
GPIO95	NC																
GPIO96	NC																
GPIO97	1:KP_ROW1	☑	☑	□	□	□		□		□ PD	□	OUT	□	☑	□	□	GPIO_KPD_KROW1_PIN
GPIO98	1:KP_ROW0	☑	☑	□	□	□		□		□ PD	□	OUT	□	☑	□	□	GPIO_KPD_KROW0_PIN
GPIO99	NC																
GPIO100	NC																
GPIO101	NC																
GPIO102	NC																
GPIO103	1:KP_COL0	☑	☑	□	□	□	[illegible]	□		☑ PU	☑	IN	☑	□	□	□	GPIO_KPD_KCOL0_PIN
GPIO104	NC																
GPIO105	1:KP_COL2	☑	☑	□	□	□		□		☑ PD	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	□	GPIO_KPD_KCOL2_PIN
GPIO106	NC																
GPIO107	NC																
GPIO108	1:KP_COL1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	□		☑ PU	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	□	GPIO_KPD_KCOL1_PIN
GPIO109	NC																
GPIO110	NC																
GPIO111	NC																
GPIO112	NC																
GPIO113	1:DUAL_BPI_BI	□	☑	□				□		☑ PD	□				□	□	

GPIO81 ☑ 0:GPIO81 ☑ ☑ IN

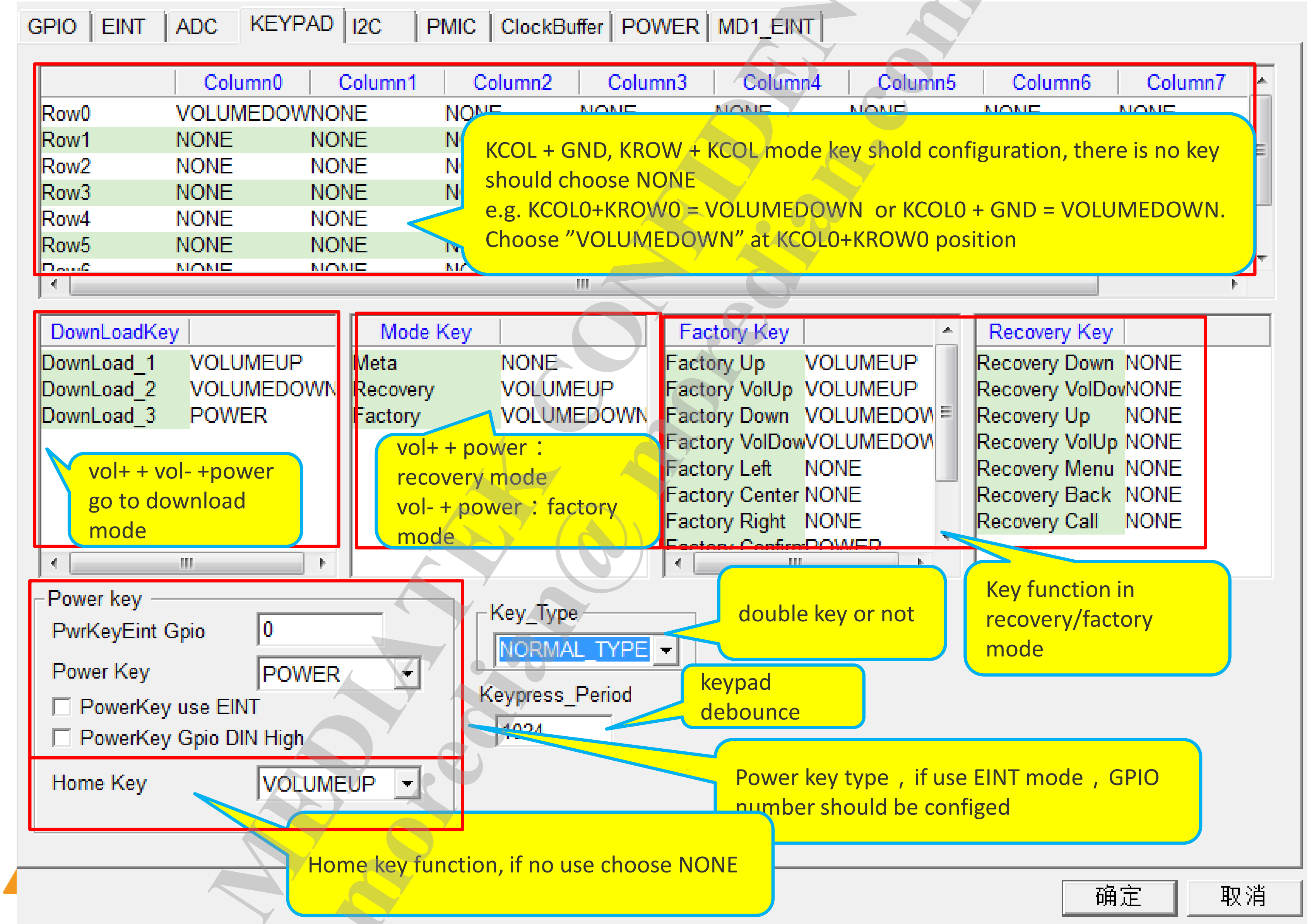
All KCOL and KROW used should be config

Config to KP_ROW, KP_COL

VarName1 should map with row , col

GPIO(EINT) + GND key config to EINT mode

Keypad dws setting—KEYPAD page



	Column0	Column1	Column2	Column3	Column4	Column5	Column6	Column7
Row0	VOLUMEDOWN	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row1	NONE	NONE	N					
Row2	NONE	NONE	N					
Row3	NONE	NONE	N					
Row4	NONE	NONE						
Row5	NONE	NONE	N					
Row6	NONE	NONE	NC					

KCOL + GND, KROW + KCOL mode key shold configuration, there is no key should choose NONE
e.g. KCOL0+KROW0 = VOLUMEDOWN or KCOL0 + GND = VOLUMEDOWN.
Choose ”VOLUMEDOWN” at KCOL0+KROW0 position

DownLoadKey	
DownLoad_1	VOLUMEUP
DownLoad_2	VOLUMEDOWN
DownLoad_3	POWER

vol+ + vol- +power go to download mode

Mode Key	
Meta	NONE
Recovery	VOLUMEUP
Factory	VOLUMEDOWN

vol+ + power ：recovery mode
vol- + power ：factory mode

Factory Key	
Factory Up	VOLUMEUP
Factory VolUp	VOLUMEUP
Factory Down	VOLUMEDOW
Factory VolDow	VOLUMEDOW
Factory Left	NONE
Factory Center	NONE
Factory Right	NONE
Factory Confirm	POWER

Recovery Key	
Recovery Down	NONE
Recovery VolDow	NONE
Recovery Up	NONE
Recovery VolUp	NONE
Recovery Menu	NONE
Recovery Back	NONE
Recovery Call	NONE

Key function in recovery/factory mode

Power key
PwrKeyEint Gpio: 0
Power Key: POWER
□ PowerKey use EINT
□ PowerKey Gpio DIN High

Home Key: VOLUMEUP

Key_Type: NORMAL_TYPE

double key or not

Keypress_Period: 1024

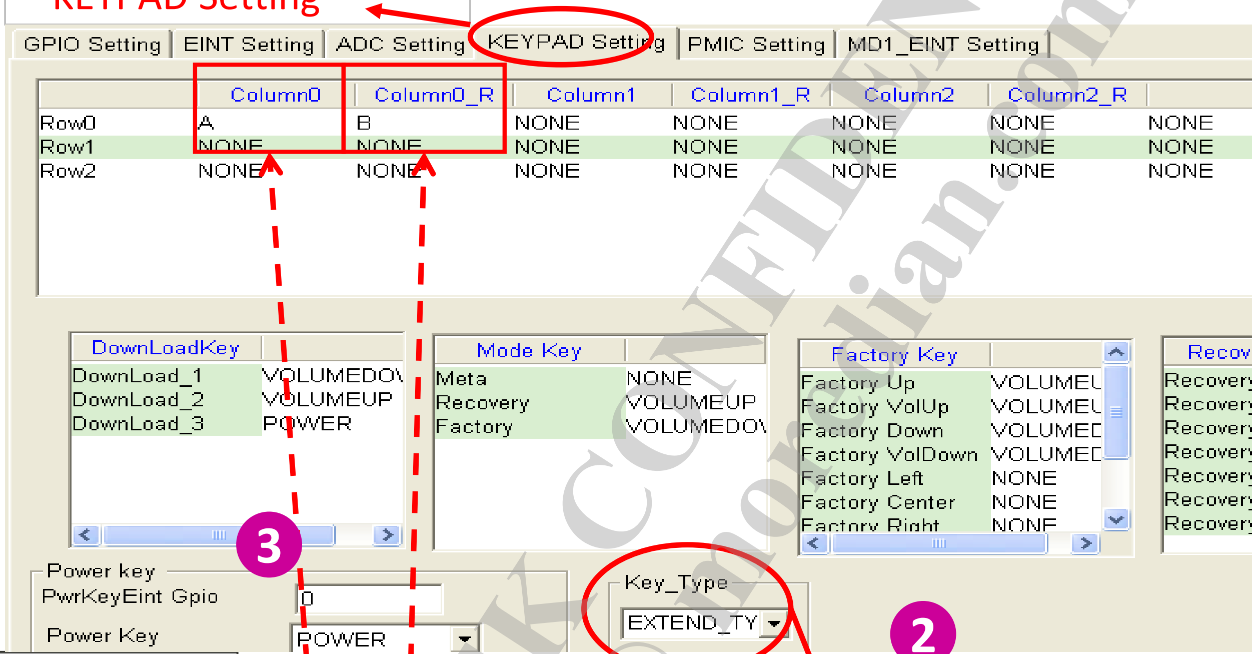
keypad debounce

Power key type，if use EINT mode，GPIO number should be configed

Home key function, if no use choose NONE

double key function

1 Choose Page “KEYPAD Setting”



GPIO Setting | EINT Setting | ADC Setting | KEYPAD Setting | PMIC Setting | MD1_EINT Setting

	Column0	Column0_R	Column1	Column1_R	Column2	Column2_R	
Row0	A	B	NONE	NONE	NONE	NONE	NONE
Row1	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row2	NONE	NONE	NONE	NONE	NONE	NONE	NONE

DownLoadKey	
DownLoad_1	VOLUMEDOV
DownLoad_2	VOLUMEUP
DownLoad_3	POWER

Mode Key	
Meta	NONE
Recovery	VOLUMEUP
Factory	VOLUMEDOV

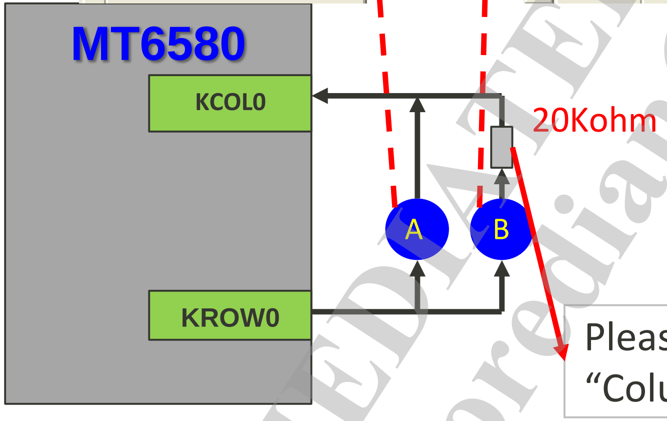
Factory Key	
Factory Up	VOLUMEU
Factory VolUp	VOLUMEU
Factory Down	VOLUMED
Factory VolDown	VOLUMED
Factory Left	NONE
Factory Center	NONE
Factory Right	NONE

Recov
Recovery
Recovery
Recovery
Recovery
Recovery
Recovery
Recovery

Power key
PwrKeyEint Gpio: 0
Power Key: POWER

Key_Type: EXTEND_TY

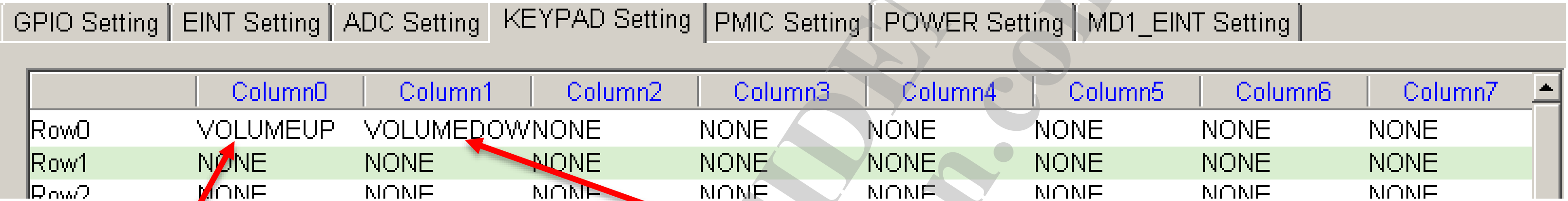
2 Choose “Key Type” as EXTEND_TYPE set bit[0]=1 of KP_SEL register in hal_kpd.c



Please notice that there’s new column named “Column0_R”. “R” stands for resistor(20K).

Keypad dws setting

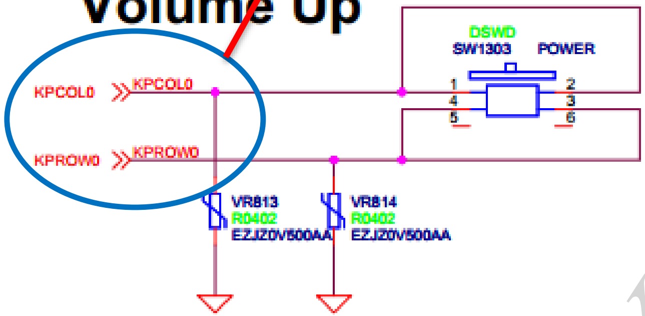
- config VOLUMEUP, VOLUMEDOWN key



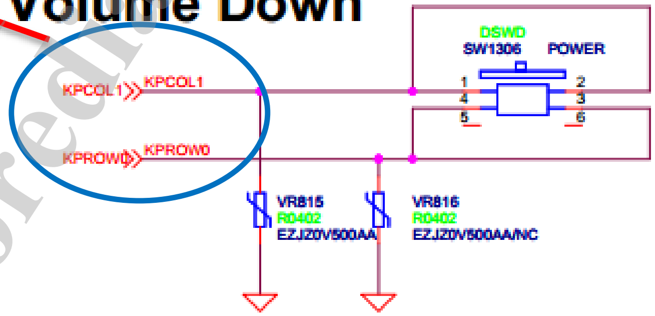
GPIO Setting | EINT Setting | ADC Setting | KEYPAD Setting | PMIC Setting | POWER Setting | MD1_EINT Setting

	Column0	Column1	Column2	Column3	Column4	Column5	Column6	Column7
Row0	VOLUMEUP	VOLUMEDOWN	NONE	NONE	NONE	NONE	NONE	NONE
Row1	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row2	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE

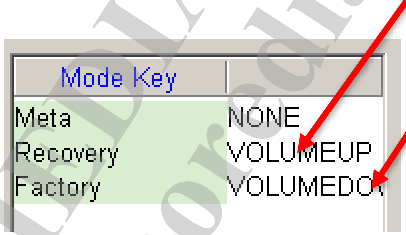
Volume Up



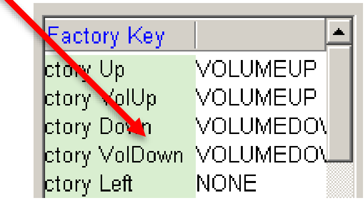
Volume Down



- combination key for boot mode and function key in special mode
  - PowerKey + VolumeDown = Factory mode
  - power key + volumeUp = recovery mode
  - move dwon functionin factory mode



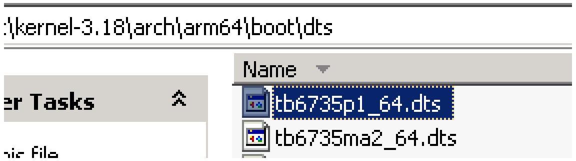
Mode Key	
Meta	NONE
Recovery	VOLUMEUP
Factory	VOLUMEDO



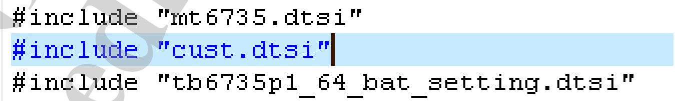
Factory Key	
ctory Up	VOLUMEUP
ctory VolUp	VOLUMEUP
ctory Down	VOLUMEDO
ctory VolDown	VOLUMEDO
ctory Left	NONE

Releationship between dts and dws

- please check dws file for your keypad setting, dws file will generate cust.dtsi when build.
- #include cust.dtsi in <proj>.dts or <platform>.dtsi



:\kernel-3.18\arch\arm64\boot\dts

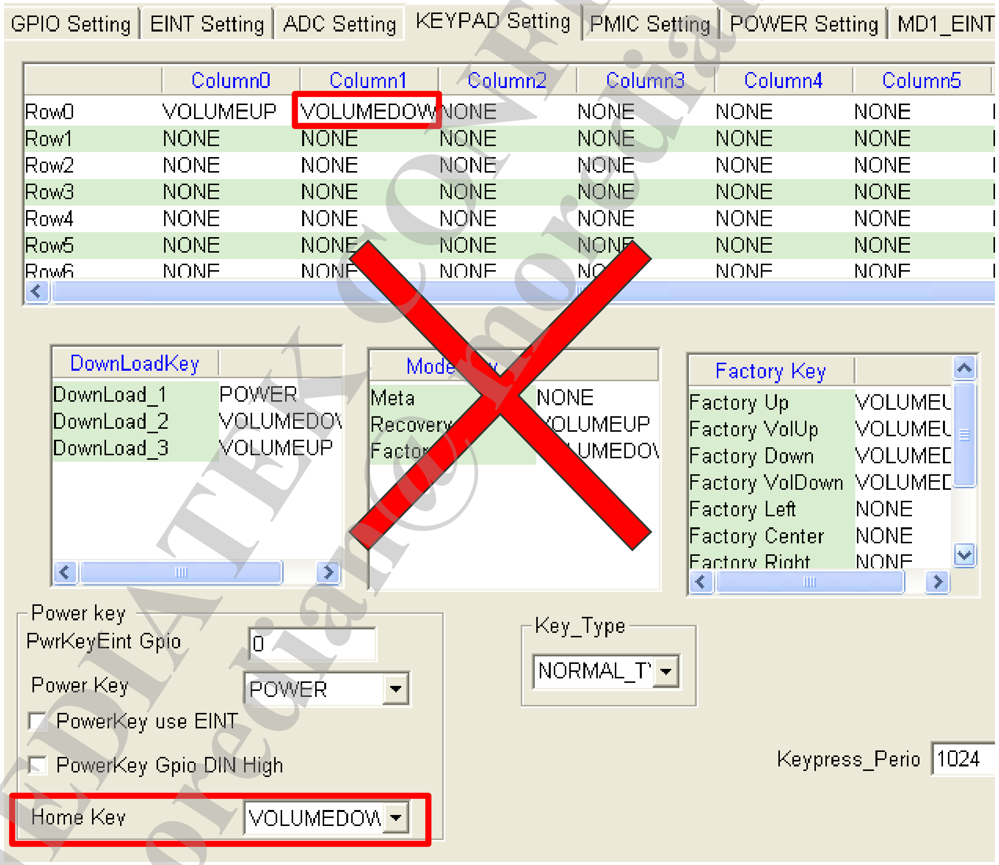


```
#include "mt6735.dtsi"
#include "cust.dtsi"
#include "tb6735p1_64_bat_setting.dtsi"
```

- relative files:
  - <kernel_ver>/drivers/misc/mediatek/dws/<platform>/<proj>.dws
  - out/target/<proj>/obj/KERNEL_OBJ/arch/< arm_ver >/boot/dts/cust.dtsi
  - <kernel_ver>/arch/< arm_ver >/boot/dts/<platform>.dtsi
  - <kernel_ver>/arch/< arm_ver >/boot/dts/<proj>.dts

Keypad —— Notice

- Please DO NOT configure the same key at both Home key field and ROW/COL field. Otherwise the key definition in dts would repeated



GPIO Setting | EINT Setting | ADC Setting | KEYPAD Setting | PMIC Setting | POWER Setting | MD1_EINT

	Column0	Column1	Column2	Column3	Column4	Column5
Row0	VOLUMEUP	VOLUMEDOW	NONE	NONE	NONE	NONE
Row1	NONE	NONE	NONE	NONE	NONE	NONE
Row2	NONE	NONE	NONE	NONE	NONE	NONE
Row3	NONE	NONE	NONE	NONE	NONE	NONE
Row4	NONE	NONE	NONE	NONE	NONE	NONE
Row5	NONE	NONE	NONE	NONE	NONE	NONE
Row6	NONE	NONE	NONE	NO	NONE	NONE

DownLoadKey	
DownLoad_1	POWER
DownLoad_2	VOLUMEDO\
DownLoad_3	VOLUMEUP

Mode	
Meta	NONE
Recovery	OLUMEUP
Factor	UMEDO\

Factory Key	
Factory Up	VOLUMEU
Factory VolUp	VOLUMEU
Factory Down	VOLUMED
Factory VolDown	VOLUMED
Factory Left	NONE
Factory Center	NONE
Factory Right	NONE

Power key

PwrKeyEint Gpio: 0

Power Key: POWER

□ PowerKey use EINT

□ PowerKey Gpio DIN High

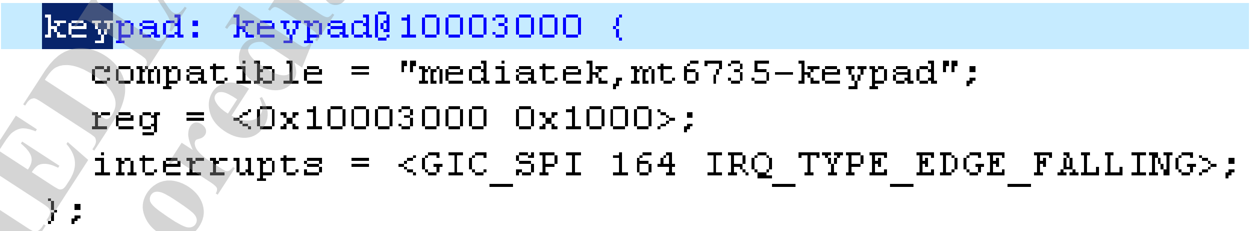
Home Key: VOLUMEDOW

Key_Type: NORMAL_T

Keypress_Perio: 1024

dts keypad node (1/3)

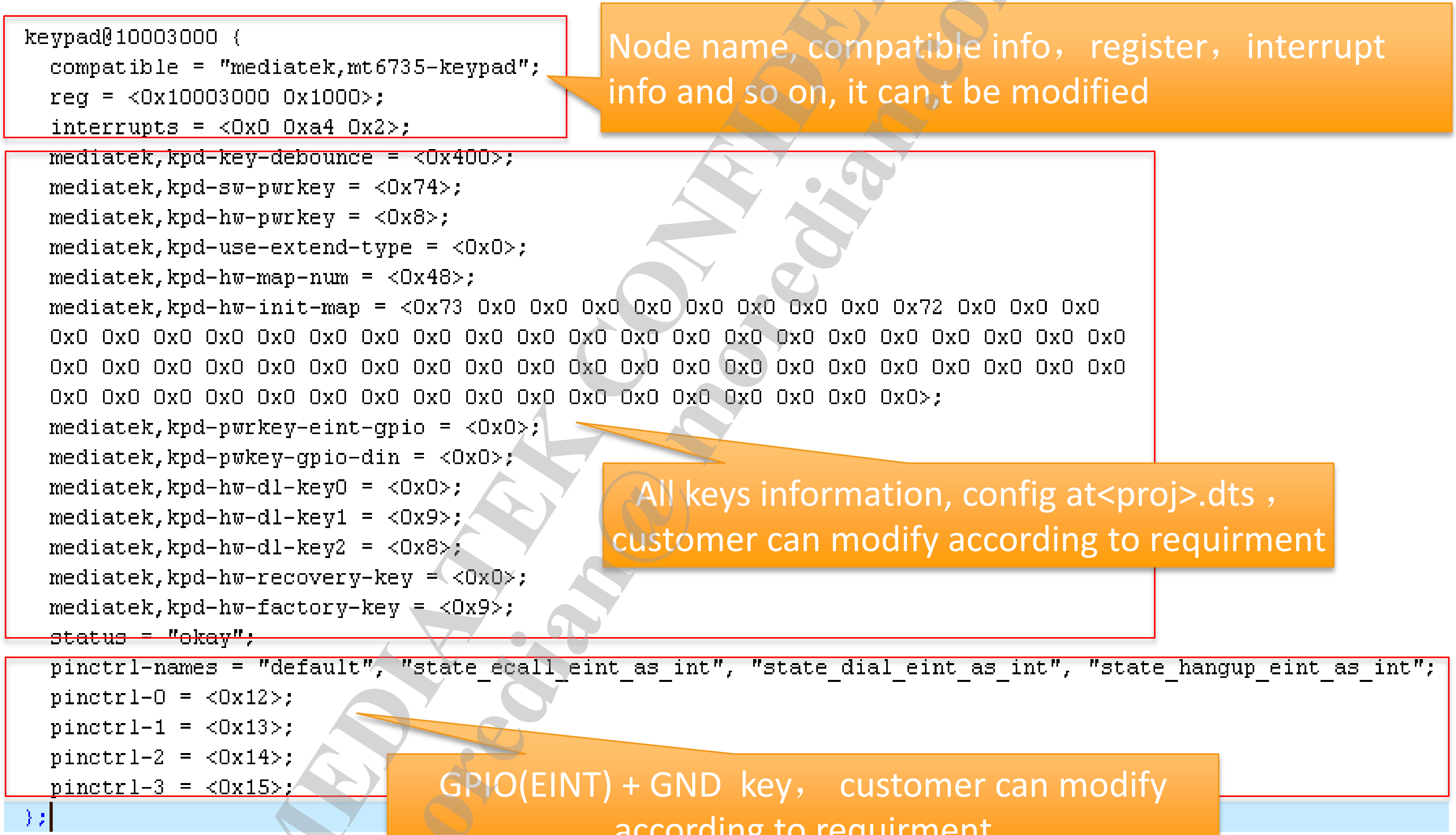
- Keypad information shows in dts node which named as ”keypad”
- “keypad” node is defined in <platform>.dtsi, don’t modifiy it.
  - path: <kernel_ver>/arch/< arm_ver >/boot/dts/<platform>.dtsi
- ”keypad” information is contained in cust.dtsi and <platform>.dtsi
  - cust.dtsi: generated by dws file when build, modify dws file but not cust.dtsi。
    - path: out/target/<proj>/obj/KERNEL_OBJ/arch/< arm_ver >/boot/dts/cust.dtsi
  - <proj>.dts:
    - path: <kernel_ver>/arch/< arm_ver >/boot/dts/<proj>.dts
- example of keypad node:



```
keypad: keypad@10003000 {
    compatible = "mediatek,mt6735-keypad";
    reg = <0x10003000 0x1000>;
    interrupts = <GIC_SPI 164 IRQ_TYPE_EDGE_FALLING>;
};
```

dts keypad node(2/3)

- keypad node information



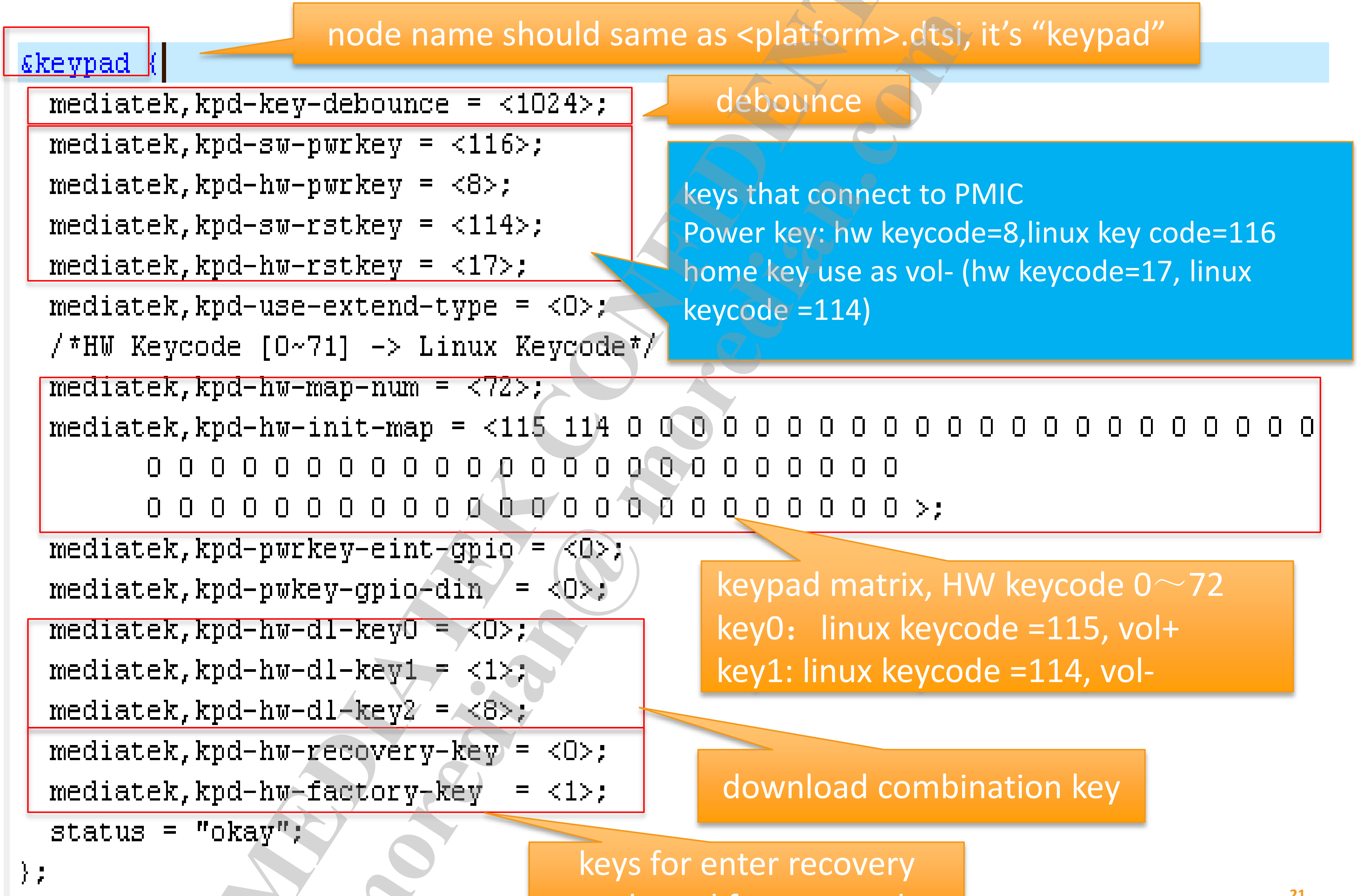
```
keypad@10003000 {
  compatible = "mediatek,mt6735-keypad";
  reg = <0x10003000 0x1000>;
  interrupts = <0x0 0xa4 0x2>;
  mediatek,kpd-key-debounce = <0x400>;
  mediatek,kpd-sw-pwrkey = <0x74>;
  mediatek,kpd-hw-pwrkey = <0x8>;
  mediatek,kpd-use-extend-type = <0x0>;
  mediatek,kpd-hw-map-num = <0x48>;
  mediatek,kpd-hw-init-map = <0x73 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x72 0x0 0x0 0x0
  0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0
  0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0
  0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0>;
  mediatek,kpd-pwrkey-eint-gpio = <0x0>;
  mediatek,kpd-pwkey-gpio-din = <0x0>;
  mediatek,kpd-hw-dl-key0 = <0x0>;
  mediatek,kpd-hw-dl-key1 = <0x9>;
  mediatek,kpd-hw-dl-key2 = <0x8>;
  mediatek,kpd-hw-recovery-key = <0x0>;
  mediatek,kpd-hw-factory-key = <0x9>;
  status = "okay";
  pinctrl-names = "default", "state_ecall_eint_as_int", "state_dial_eint_as_int", "state_hangup_eint_as_int";
  pinctrl-0 = <0x12>;
  pinctrl-1 = <0x13>;
  pinctrl-2 = <0x14>;
  pinctrl-3 = <0x15>;
};
```

Node name, compatible info， register， interrupt info and so on, it can,t be modified

All keys information, config at<proj>.dts，customer can modify according to requirment

GPIO(EINT) + GND key， customer can modify according to requirment

dts Keypad node(3/3)



```
&keypad {
    mediatek,kpd-key-debounce = <1024>;
    mediatek,kpd-sw-pwrkey = <116>;
    mediatek,kpd-hw-pwrkey = <8>;
    mediatek,kpd-sw-rstkey = <114>;
    mediatek,kpd-hw-rstkey = <17>;
    mediatek,kpd-use-extend-type = <0>;
    /*HW Keycode [0~71] -> Linux Keycode*/
    mediatek,kpd-hw-map-num = <72>;
    mediatek,kpd-hw-init-map = <115 114 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
          0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
          0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 >;
    mediatek,kpd-pwrkey-eint-gpio = <0>;
    mediatek,kpd-pwkey-gpio-din  = <0>;
    mediatek,kpd-hw-dl-key0 = <0>;
    mediatek,kpd-hw-dl-key1 = <1>;
    mediatek,kpd-hw-dl-key2 = <8>;
    mediatek,kpd-hw-recovery-key = <0>;
    mediatek,kpd-hw-factory-key  = <1>;
    status = "okay";
};
```

node name should same as <platform>.dtsi, it’s “keypad”

debounce

keys that connect to PMIC
Power key: hw keycode=8,linux key code=116
home key use as vol- (hw keycode=17, linux keycode =114)

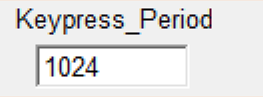
keypad matrix, HW keycode 0～72
key0： linux keycode =115, vol+
key1: linux keycode =114, vol-

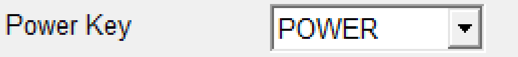
download combination key

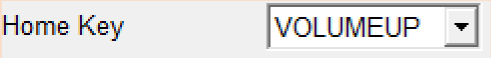
keys for enter recovery mode and factory mode

Keypad dts vs dws(1/3)

- dts vs dws

dts	dws
kpd-key-debounce = <1024>;	 Keypress_Period 1024

kpd-sw-pwrkey = <116>;
kpd-hw-pwrkey = <8>; (*note1)
	
	Power Key
	POWER

kpd-sw-rstkey = <115>;
kpd-hw-rstkey = <17>; (*note2)
	
	Home Key
	VOLUMEUP

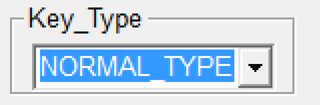
note1: HW keycode of Power key is 8, linux keycode is 116

note2: volume up use pmic homekey HW keycode is 17, linux key code is 115, it also defined as reset key(Power + Volume up)

- HW keycode
  - keypad matrix key code, 0~71, col0 + row0=0, col1 + row0=1 ... col1+row1=9...
- linux key code
  - include/linux/input.h

Keypad dts vs dws(2/3)

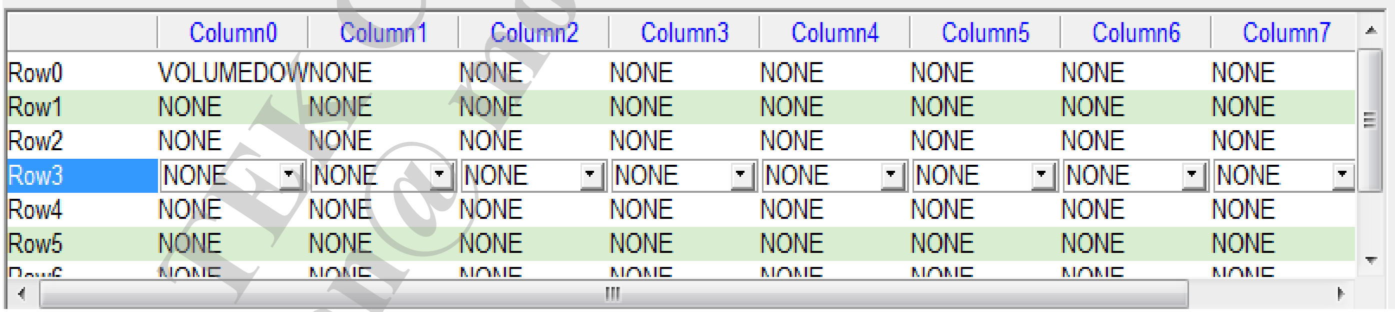
- dtsi VS dws
  - dtsi: kpd-use-extend-type = <0>;
    dws:



Key_Type

NORMAL_TYPE

  - dtsi: kpd-hw-init-map = <114 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 >;
    dws:

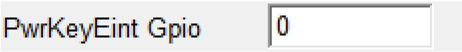


	Column0	Column1	Column2	Column3	Column4	Column5	Column6	Column7
Row0	VOLUMEDOW	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row1	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row2	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row3	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row4	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row5	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE
Row6	NONE	NONE	NONE	NONE	NONE	NONE	NONE	NONE

    - volumedown key allocateed at col0 + row0: linux keycode is 114, HW keycode is “0”.
    - The matrix contains 72 keys. dws=“NONE” => dts=“0”.

Keypad dts vs dws(3/3)

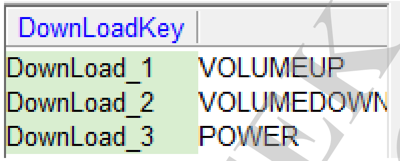
- dtsi VS dws
  - dtsi : kpd-pwrkey-eint-gpio = <0>;kpd-pwkey-gpio-din = <0>;

    dws: 

PwrKeyEint Gpio	0

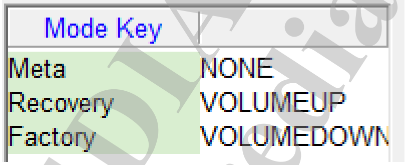
  - dtsi: kpd-hw-dl-key0 = <17>; kpd-hw-dl-key1 = <0>;

    kpd-hw-dl-key2 = <8>;

    dws: 

DownLoadKey	
DownLoad_1	VOLUMEUP
DownLoad_2	VOLUMEDOWN
DownLoad_3	POWER

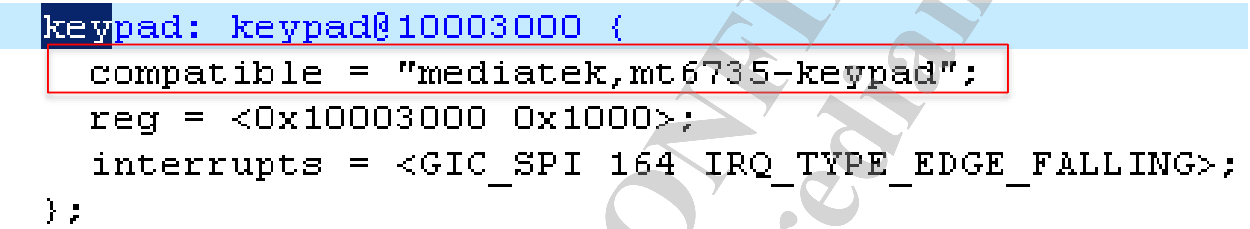
  - dtsi: kpd-hw-recovery-key = <17>;   kpd-hw-factory-key = <0>;

    dws: 

Mode Key	
Meta	NONE
Recovery	VOLUMEUP
Factory	VOLUMEDOWN

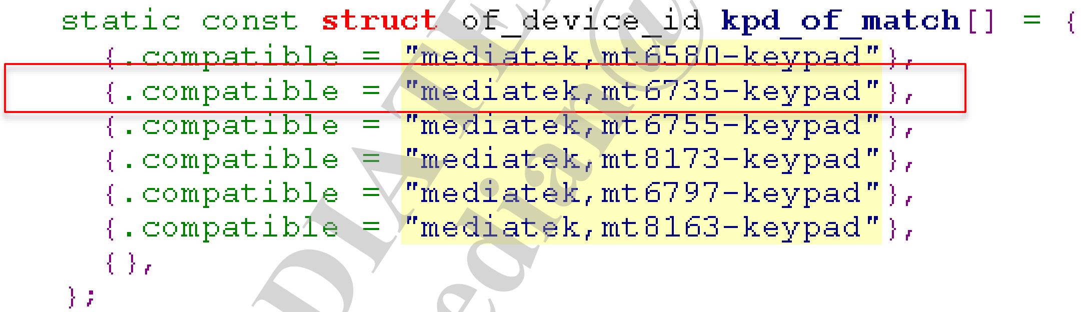
SW for dts(1/2)

- compatible information in software should be same as dts
  - compatible strings of dts is in keypad root node. e.g.”mediatek,mt6735-keypad”



```
keypad: keypad@10003000 {
  compatible = "mediatek,mt6735-keypad";
  reg = <0x10003000 0x1000>;
  interrupts = <GIC_SPI 164 IRQ_TYPE_EDGE_FALLING>;
};
```

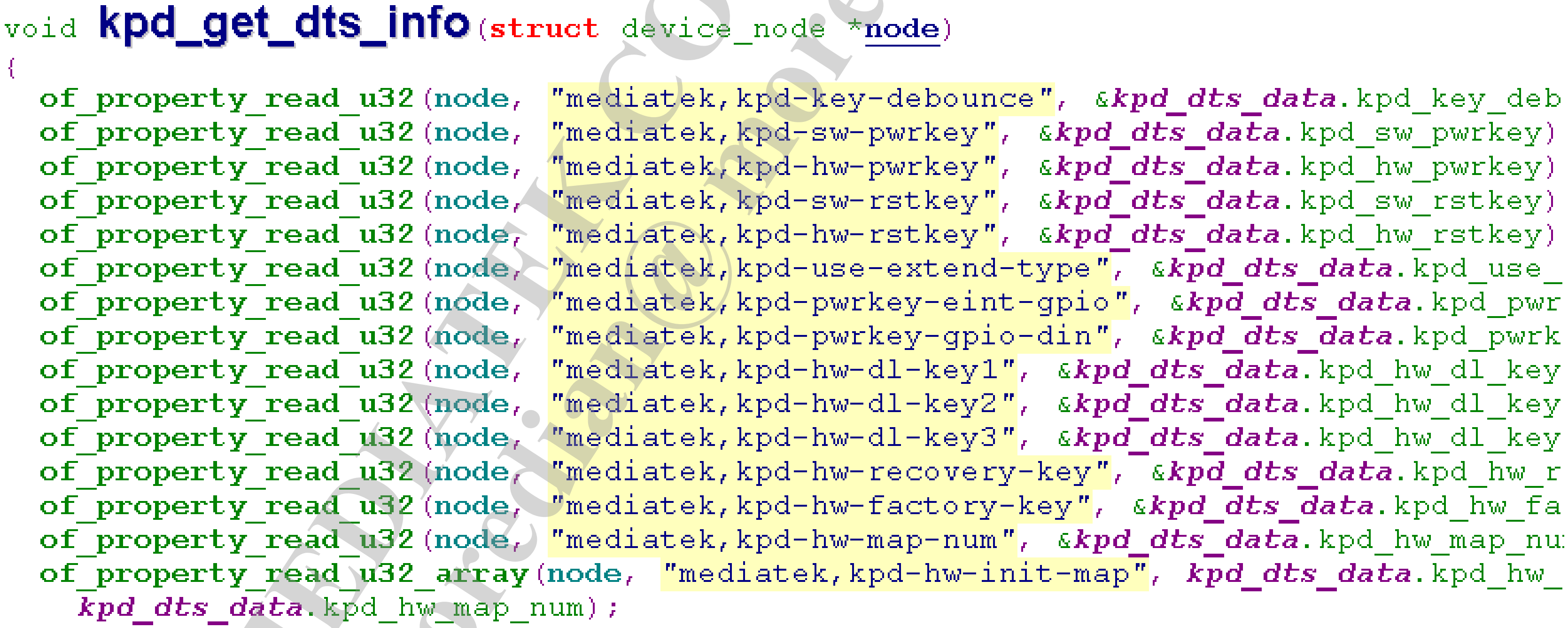
- compatible strings of software is in kpd.c
  - <kernel_ver>/drivers/input/keyboard/mediatek/kpd.c



```
static const struct of_device_id kpd_of_match[] = {
    {.compatible = "mediatek,mt6580-keypad"},
    {.compatible = "mediatek,mt6735-keypad"},
    {.compatible = "mediatek,mt6755-keypad"},
    {.compatible = "mediatek,mt8173-keypad"},
    {.compatible = "mediatek,mt6797-keypad"},
    {.compatible = "mediatek,mt8163-keypad"},
    {},
};
```

SW for dts(2/2)

- if you want to check and add a new key, please refer and modify kpd_get_dts_info() in kpd.c
- path: <kernel_ver>/drivers/input/keyboard/mediatek/kpd.c



```
void kpd_get_dts_info(struct device_node *node)
{
  of_property_read_u32(node, "mediatek,kpd-key-debounce", &kpd_dts_data.kpd_key_deb
  of_property_read_u32(node, "mediatek,kpd-sw-pwrkey", &kpd_dts_data.kpd_sw_pwrkey)
  of_property_read_u32(node, "mediatek,kpd-hw-pwrkey", &kpd_dts_data.kpd_hw_pwrkey)
  of_property_read_u32(node, "mediatek,kpd-sw-rstkey", &kpd_dts_data.kpd_sw_rstkey)
  of_property_read_u32(node, "mediatek,kpd-hw-rstkey", &kpd_dts_data.kpd_hw_rstkey)
  of_property_read_u32(node, "mediatek,kpd-use-extend-type", &kpd_dts_data.kpd_use_
  of_property_read_u32(node, "mediatek,kpd-pwrkey-eint-gpio", &kpd_dts_data.kpd_pwr
  of_property_read_u32(node, "mediatek,kpd-pwrkey-gpio-din", &kpd_dts_data.kpd_pwrk
  of_property_read_u32(node, "mediatek,kpd-hw-dl-key1", &kpd_dts_data.kpd_hw_dl_key
  of_property_read_u32(node, "mediatek,kpd-hw-dl-key2", &kpd_dts_data.kpd_hw_dl_key
  of_property_read_u32(node, "mediatek,kpd-hw-dl-key3", &kpd_dts_data.kpd_hw_dl_key
  of_property_read_u32(node, "mediatek,kpd-hw-recovery-key", &kpd_dts_data.kpd_hw_r
  of_property_read_u32(node, "mediatek,kpd-hw-factory-key", &kpd_dts_data.kpd_hw_fa
  of_property_read_u32(node, "mediatek,kpd-hw-map-num", &kpd_dts_data.kpd_hw_map_nu
  of_property_read_u32_array(node, "mediatek,kpd-hw-init-map", kpd_dts_data.kpd_hw_
    kpd_dts_data.kpd_hw_map_num);
```

keypad dct Tool and dws file

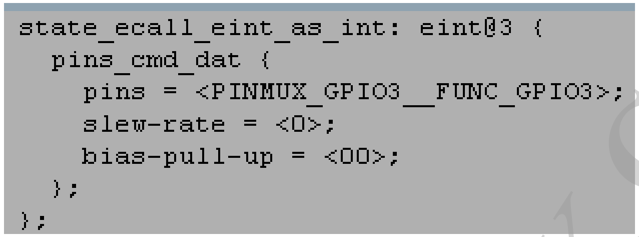
- DCT Tool, DrvGen.exe
  - kernel:
    - <kernel_ver>/tools/dct/ old_dct
  - lk：
    - vendor/mediatek/proprietary/bootable/bootloader/lk/scripts/dct/old_dct
  - preloader：
    - vendor/mediatek/proprietary/bootable/bootloader/preloader/tools/dct/old_dct
- codegen.dws file
  - path:
    - vendor/mediatek/proprietary/bootable/bootloader/lk/target/<proj>/dct/dct/codegen.dws
    - vendor/mediatek/proprietary/bootable/bootloader/preloader/custom/<proj>/dct/dct/codegen.dws
    - <kernel_ver>/drivers/misc/mediatek/dws/<platform>/<proj>.dws
    - vendor/mediatek/proprietary/custom/<proj>/kernel/dct/dct/codegen.dws

kernel config of keypad

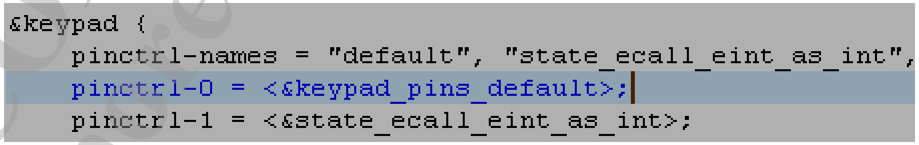
- path of config file:
  - <kernel_ver>/arch/<arm_ver>/configs/<proj>_defconfig
  - <kernel_ver>/arch/<arm_ver>/configs/<proj>_debug_defconfig
- kernel config items:
  - CONFIG_KPD_PWRKEY_USE_PMIC: power key use pmic power key
  - CONFIG_KPD_PWRKEY_USE_EINT: power key use GPIO(EINT) + GND
  - long press reboot mode:
    - CONFIG_ONEKEY_REBOOT_NORMAL_MODE
    - CONFIG_ONEKEY_REBOOT_OTHER_MODE
    - CONFIG_TWO_KEY_REBOOT_NORMAL_MODE
    - CONFIG_TWO_KEY_REBOOT_OTHER_MODE
    - CONFIG_KPD_PMIC_LPRST_TD

key use EINT type

- enable function in kernel config
  - e.g. power key use EINT
    - CONFIG_KPD_PWRKEY_USE_EINT=y
- set GPIO(EINT) information in dts



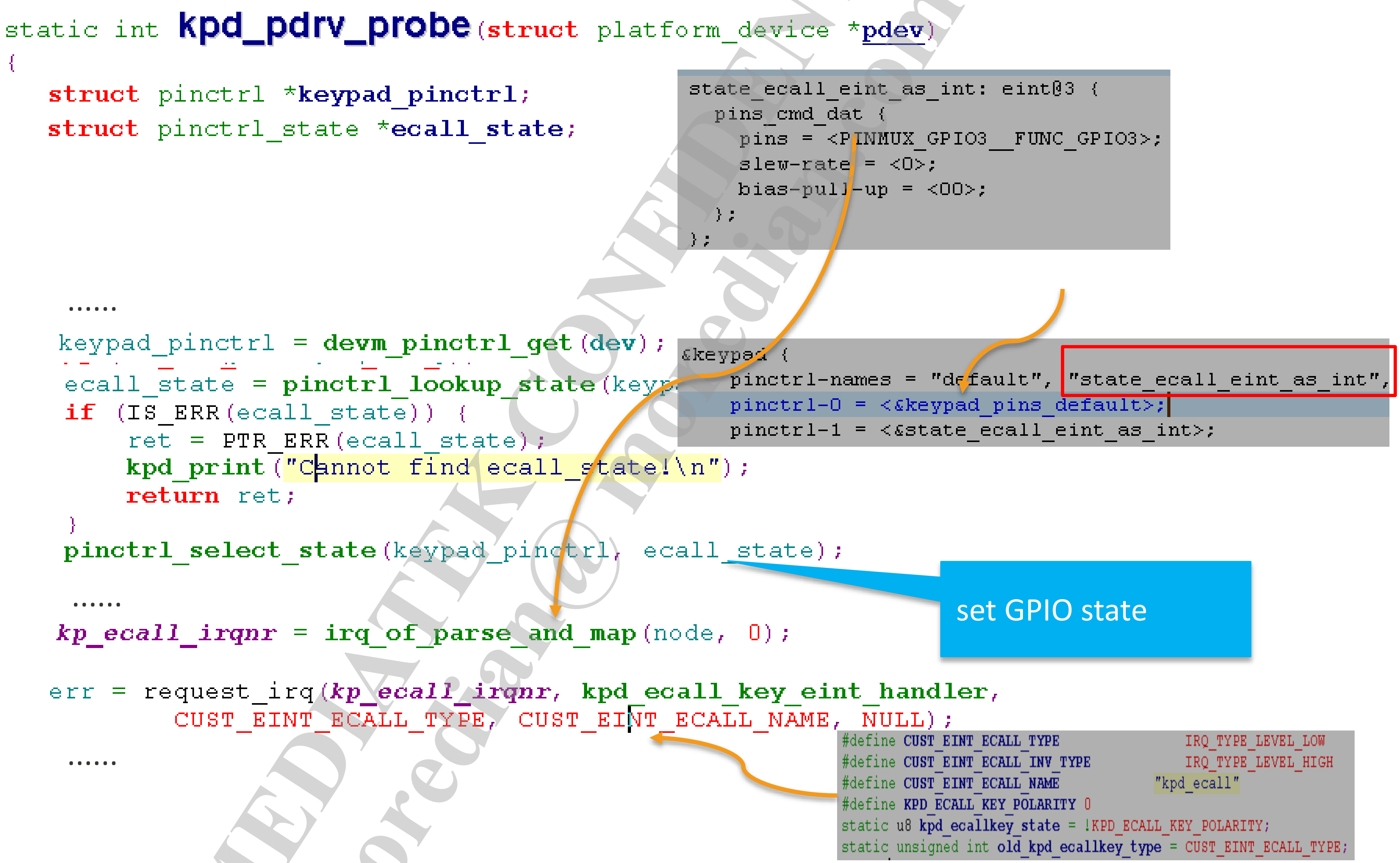
```
state_ecall_eint_as_int: eint@3 {
  pins_cmd_dat {
    pins = <PINMUX_GPIO3__FUNC_GPIO3>;
    slew-rate = <0>;
    bias-pull-up = <00>;
  };
};
```



```
&keypad {
    pinctrl-names = "default", "state_ecall_eint_as_int",
    pinctrl-0 = <&keypad_pins_default>;
    pinctrl-1 = <&state_ecall_eint_as_int>;
```

- keypad probe function
  - initialize GPIO and input device
  - register irq info
- implement eint handler function in kpd.c
  - handle irq and report key event to input device

key use EINT type--SW



```
static int kpd_pdrv_probe(struct platform_device *pdev)
{
    struct pinctrl *keypad_pinctrl;
    struct pinctrl_state *ecall_state;

     ......
    keypad_pinctrl = devm_pinctrl_get(dev);
     ecall_state = pinctrl_lookup_state(keyp
     if (IS_ERR(ecall_state)) {
          ret = PTR_ERR(ecall_state);
          kpd_print("Cannot find ecall_state!\n");
          return ret;
     }
     pinctrl_select_state(keypad_pinctrl, ecall_state);

     ......
    kp_ecall_irqnr = irq_of_parse_and_map(node, 0);

   err = request_irq(kp_ecall_irqnr, kpd_ecall_key_eint_handler,
            CUST_EINT_ECALL_TYPE, CUST_EINT_ECALL_NAME, NULL);

     ......
```

```
state_ecall_eint_as_int: eint@3 {
    pins_cmd_dat {
        pins = <PINMUX_GPIO3__FUNC_GPIO3>;
        slew-rate = <0>;
        bias-pull-up = <00>;
    };
};
```

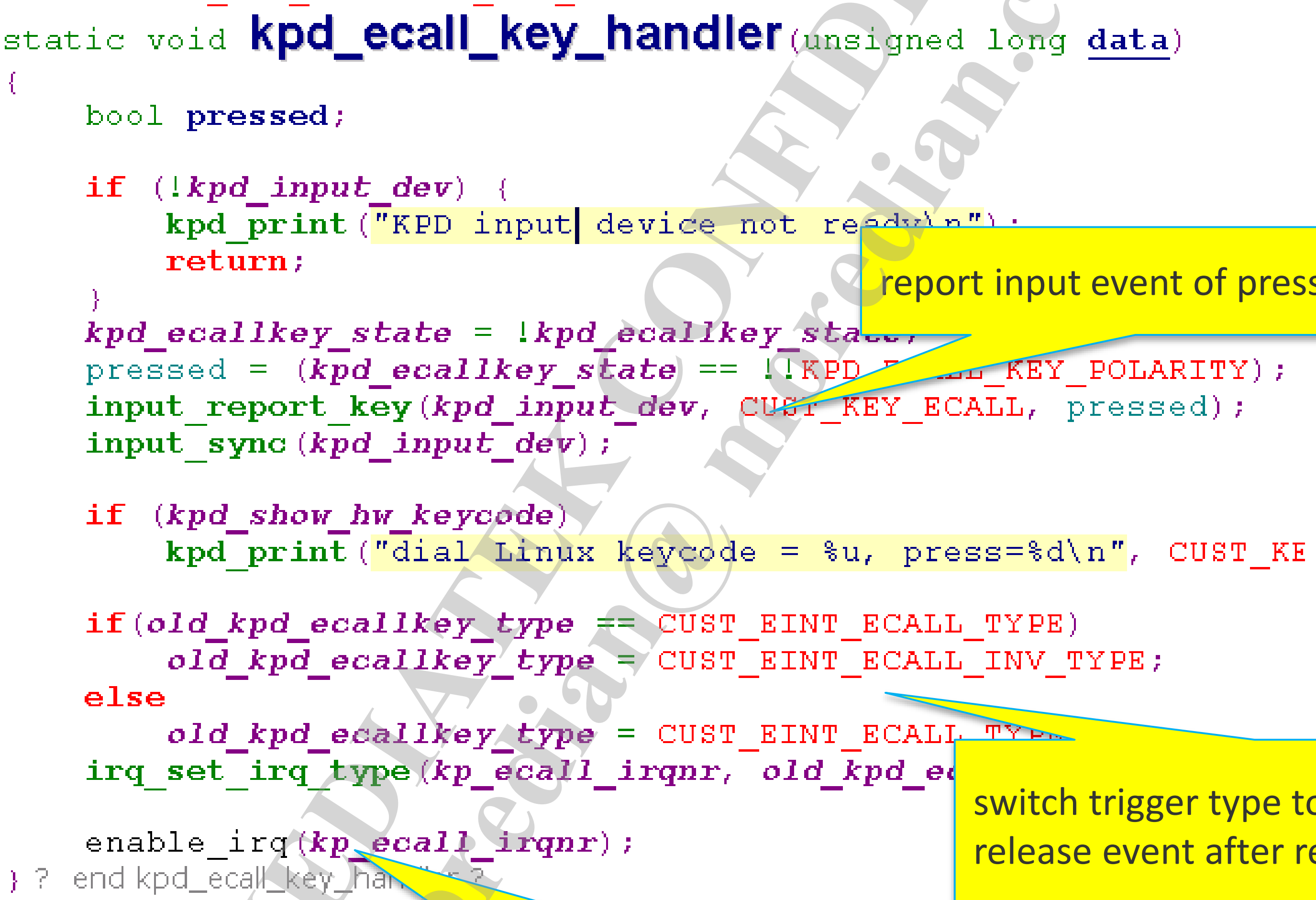
```
&keypad {
    pinctrl-names = "default", "state_ecall_eint_as_int",
    pinctrl-0 = <&keypad_pins_default>;
    pinctrl-1 = <&state_ecall_eint_as_int>;
```

set GPIO state

```
#define CUST_EINT_ECALL_TYPE                    IRQ_TYPE_LEVEL_LOW
#define CUST_EINT_ECALL_INV_TYPE                IRQ_TYPE_LEVEL_HIGH
#define CUST_EINT_ECALL_NAME                 "kpd_ecall"
#define KPD_ECALL_KEY_POLARITY 0
static u8 kpd_ecallkey_state = !KPD_ECALL_KEY_POLARITY;
static unsigned int old_kpd_ecallkey_type = CUST_EINT_ECALL_TYPE;
```

key use EINT type--SW

- handler for EINT type key
  - for one polarity hall sensor, it may not need to change trigger type and just report both press and release key event for every interrupt



```
static void kpd_ecall_key_handler(unsigned long data)
{
    bool pressed;

    if (!kpd_input_dev) {
        kpd_print("KPD input device not ready\n");
        return;
    }
    kpd_ecallkey_state = !kpd_ecallkey_state;
    pressed = (kpd_ecallkey_state == !!KPD_ECALL_KEY_POLARITY);
    input_report_key(kpd_input_dev, CUST_KEY_ECALL, pressed);
    input_sync(kpd_input_dev);

    if (kpd_show_hw_keycode)
        kpd_print("dial Linux keycode = %u, press=%d\n", CUST_KE

    if(old_kpd_ecallkey_type == CUST_EINT_ECALL_TYPE)
        old_kpd_ecallkey_type = CUST_EINT_ECALL_INV_TYPE;
    else
        old_kpd_ecallkey_type = CUST_EINT_ECALL_TYPE;
    irq_set_irq_type(kp_ecall_irqnr, old_kpd_ec

    enable_irq(kp_ecall_irqnr);
} ? end kpd_ecall_key_handler ?
```

report input event of press or release

switch trigger type to detect press or release event after release or pressed

enable irq

One key long press for reset

- long press power key for reset
  - kernel config items:
    - CONFIG_ONEKEY_REBOOT_NORMAL_MODE=y
    - CONFIG_ONEKEY_REBOOT_OTHER_MODE=y
    - CONFIG_KPD_PMIC_LPRST_TD=1
      - the value for this item is 0,1,2,3, means long press time is 7s, 11s, 14s, 5s
  - Kconfig path:
    - <kernel_ver>/drivers/input/keyboard/Kconfig

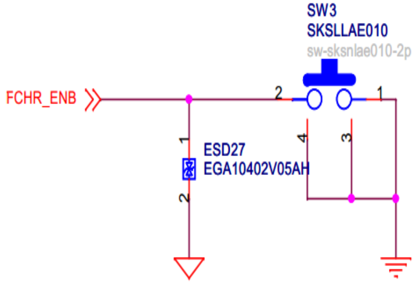
two key long press for reset

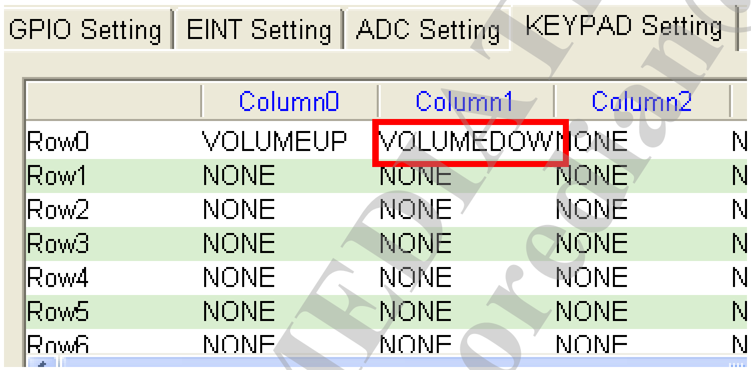
- HW should support Vol- or vol+ should connect to HOME-KEY(FCHR_ENB pin) of PMIC
- Kernel Config items：

```
CONFIG_TWO_KEY_REBOOT_NORMAL_MODE=y
CONFIG_TWO_KEY_REBOOT_OTHER_MODE=y
CONFIG_KPD_PMIC_LPRST_TD=1
```

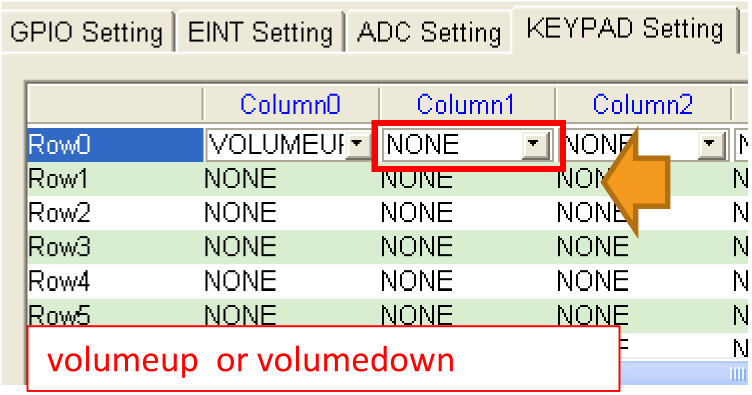
(value 0～3 stands for7s/11s/14s/5s)

Volume Down





	Column0	Column1	Column2	
Row0	VOLUMEUP	VOLUMEDOWN	NONE	N
Row1	NONE	NONE	NONE	N
Row2	NONE	NONE	NONE	N
Row3	NONE	NONE	NONE	N
Row4	NONE	NONE	NONE	N
Row5	NONE	NONE	NONE	N
Row6	NONE	NONE	NONE	N



	Column0	Column1	Column2	
Row0	VOLUMEUP	NONE	NONE	N
Row1	NONE	NONE	NONE	N
Row2	NONE	NONE	NONE	N
Row3	NONE	NONE	NONE	N
Row4	NONE	NONE	NONE	N
Row5	NONE	NONE	NONE	N

volumeup or volumedown

keypad in preloader

- In pl/lk, boot mode is decided by pressed keys, so combination key should be correct
- preloader
  - .../preloader/platform/<platform>/src/drivers/inc/keypad.h
  - dws will generate cust_kpd.h
  - cust_kpd.h(.../preloader/custom/<proj>/dct/dct)
  - .../preloader/platform/<platform>/src/drivers/keypad.c
    - Related function：mtk_kpd_gpios_set, set_kpd_pmic_mode, mtk_detect_key...
  - PMIC related function:
    - .../bootloader/preloader/platform/mt6735/src/drivers/pmic.c

keypad in lk

- keypad related files in lk
  - .../lk/platform/<platform>/include/platform/mtk_key.h
  - dws generate cust_kpd.h
    - .../lk/target/<platform>/dct/dct
    - MT65XX_FACTORY_KEY, MT65XX_PMIC_RST_KEY...
  - .../lk/platform/<platform>/mtk_key.c
    - Related code： set_kpd_pmic_mode, mtk_detect_key...
  - PMIC related function:
    - .../bootloader/lk/platform/mt6735/mt_pmic.c

Keypad debug

- Log analysis
- getevent and GPIO check adb command
- Review config files
  - Review dws、 dts、 kernel config， to check related config is right or not
- source code review
  - Check source code(eg.kpd.c) whether realize the function
- HW check
  - Check whether sch is mapping with SW
  - Check whether keypad signal has shake

keypad debug--Log analysis

- Kernel log
  - keyword： kpd
  - In keypad log will find key press and key release
  - press and release will appear at the same time

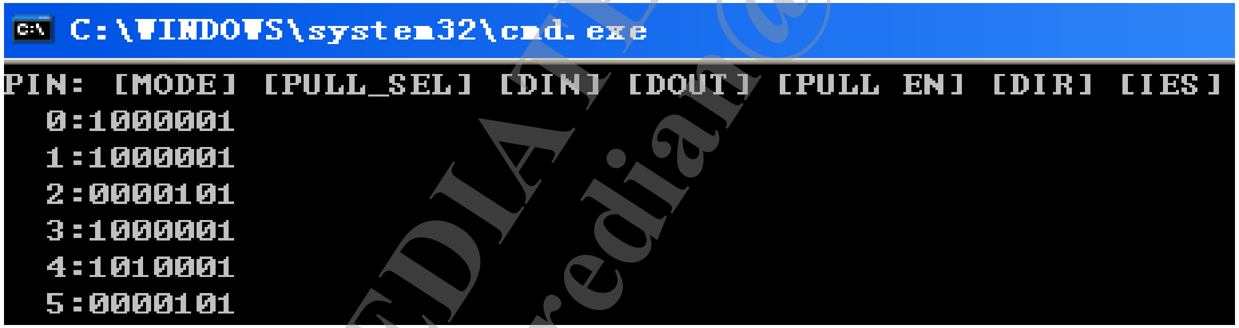
```
<4>[253.828234]kpd: register = fffe ffff ffff ffff ff
<4>[253.828825]kpd: (pressed) HW keycode = 0
<4>[253.829348]kpd: report Linux keycode = 115
<4>[253.829857]kpd: save new keymap state
<4>[254.030814]kpd: register = ffff ffff ffff ffff ff
<4>[254.031405]kpd: (released) HW keycode = 0
<4>[254.031936]kpd: report Linux keycode = 115
<4>[254.032445]kpd: save new keymap state
```

bit0 =0 means hw keycode=0 的key有动作

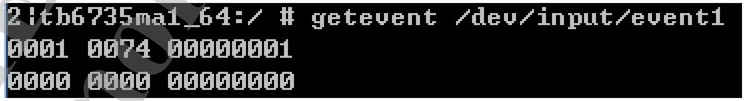
linux keycode = 115，也就是volume up key

Keypad debug—adb command

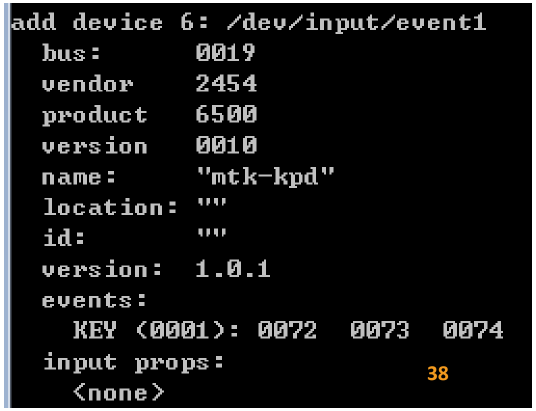
- get kpd input events(“adb shell” first to enter shell)
  - getevent –i: find which input event is for keypad
  - getevent /dev/input/event?: get key event
- catch and change GPIO status
  - cat /sys/devices/virtual/misc/mtgpio/pin
    - refer to bringup sop of gpio


C:\WINDOWS\system32\cmd.exe

```
PIN: [MODE] [PULL_SEL] [DIN] [DOUT] [PULL EN] [DIR] [IES]
   0:1000001
   1:1000001
   2:0000101
   3:1000001
   4:1010001
   5:0000101
```



```
2!tb6735mal_64:/ # getevent /dev/input/event1
0001 0074 00000001
0000 0000 00000000
```



```
add device 6: /dev/input/event1
  bus:      0019
  vendor    2454
  product   6500
  version   0010
  name:     "mtk-kpd"
  location: ""
  id:       ""
  version:  1.0.1
  events:
    KEY (0001): 0072  0073  0074
  input props:
    <none>
```

Appendix

- some FAQ
  - FAQ13931-[keypad]怎样在Android L版本添加新Key
  - FAQ13908-[keypad]怎样实现单按PowerKey重启功能？
  - FAQ13560-[BMT]MT6735上长摁powerkey(powerkey+home key)shutdown 或者reset phone
  - FAQ05859-[keypad]关于89 jb2上的自动设置按键唤醒系统的功能以及L版本上设置唤醒键

MEDIATEK

everyday genius