LCD module.

1. Số segment (tức 1 số) dành cho:

Lưu ý: Dấu “:” hiển thị trên LCD nhưng ko được tính là 1 seg.

Frequency: seg 1 -> 5

CLOCK: seg1 -> 4

ALARML seg1 -> 4

PSNO: seg 5

(tham khao hàm Lcd\_blink() trong file Lcd\_main.c )

2. Có 2 chế độ nhấp nháy: nhấp nháy chu kì 500ms (500ms sáng, 500 ms tắt) và chu kì 1s. Và nhấp nháy ở 4 chế độ: FREQ, ALARM, CLOCK, PSNO.

3. Indicator “ST” (stereo)

On the FM band, ST will be displayed if the station is broadcasting in stereo. Stereo reproduction on AM is not available.

(Xem file *ST Indicator\_Stereo Indicator*)

4. Indicator “TI” (traffic information)

***Turning TI function On/Off***

**You can listen to traffic information. If you are listening to discs, they will continue playing until a traffic bulletin is broadcast.**

Press the [ TI ] button to turn the function on and off.

When the function is on, the TI indicator is displayed.

When a traffic bulletin begins to be broadcast, "Traffic Information" will be displayed.

If you are listening to a tape or disc when a traffic bulletin begins, they will automatically be turned off. When the bulletin ends or poor reception is experienced, the tape or disc will begin playing again.

(Nguồn: <https://www.manualslib.com/manual/85993/Kenwood-Excelon-Kvt-911dvd.html?page=30>)

5. Submode gồm: Clock (Alarm nằm trong CLOCK ), Volumn, Sound(mono, stereo)

Xem hàm Lcd\_mode() trong file lcd\_mode.c

6.

Cach kiem tra de hien thi mode se theo thu tu sau:

1. Check xem code area dung hay ko

2. Check ACCOFF

3. (ACCON)Check POWER ON/OFF

4. (ACCON v・POWERON)Check Version

5. (ACCON v・POWERON)Check Submode

6. Check Main Mode

(Xem hàm Lcd\_mode() trong file lcd\_mode.c)

7. Các seg hiển thị thời gian ở mode Alarm.

Lưu ý: vì SEG1 có kí tự ‘0’ nên sẽ ko hiển thị

Dấu “:” hiển thị trên LCD nhưng ko được tính là 1 seg.

VD: Nếu muốn LCD hiển thị 5:30 phút sáng

SEG1 SEG2 SEG3 SEG4 SEG5

5 ： 3 0 A

Indicator: AM

Nếu muốn LCD hiển thị 5:30 phút chiều.

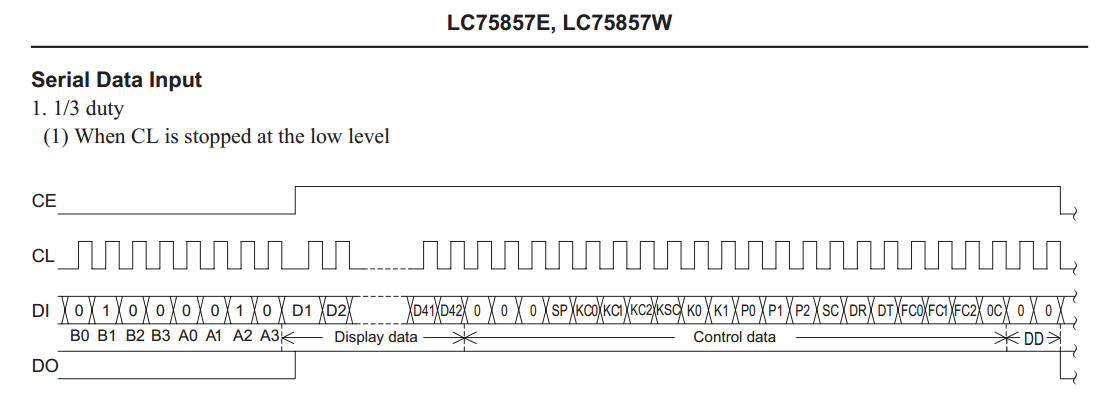
SEG1 SEG2 SEG3 SEG4 SEG5

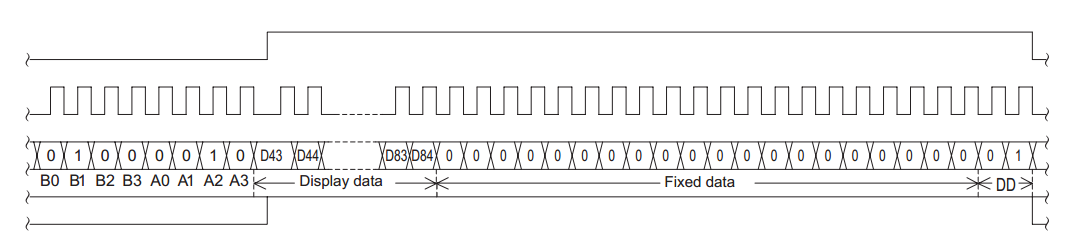
5 ： 3 0 A

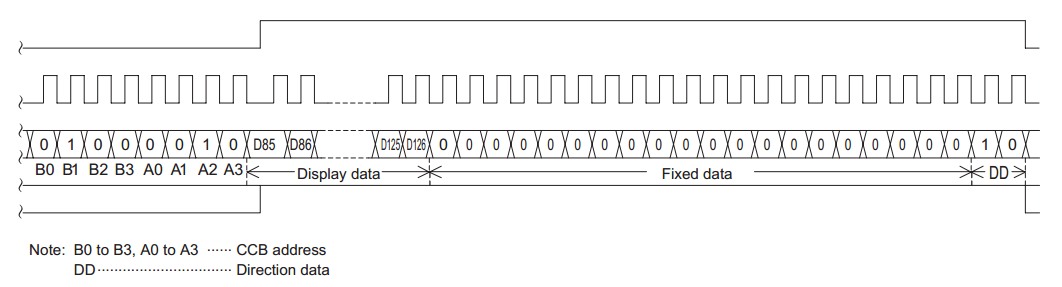
Indicator: PM

**8. 2 bit DD (Direction Data)** có thể là 2 bit để báo cho LCD driver biết lần truyền thứ mấy. Nếu lần truyền thứ 2 mà vẫn để

DD = 00 thì 42 bit data trước đó truyền đi sẽ đè lên 42 bit truyền đi ở lần truyền thứ nhất. Nếu vậy các segment ban đầu hiển thị đúng ở lần truyền thứ 1 sẽ sai sau lần truyền thứ 2.







9. AS(Auto Store) button trong mode TUNER.

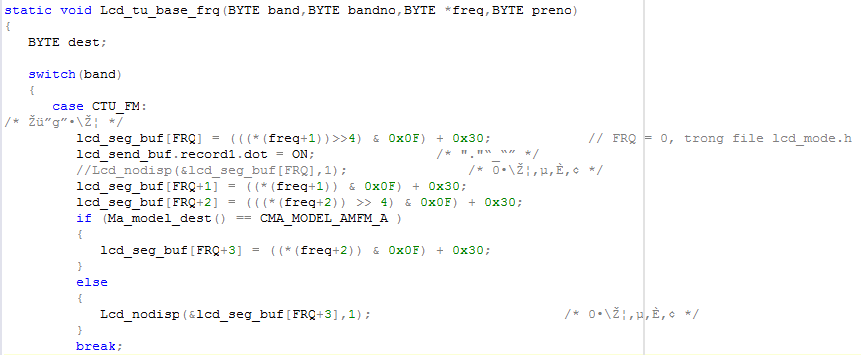
Say you go out of town, all your presets are no good anymore.  
  
Hit AS, it'll line up all the stations that come in, strongest signals first.  
  
Hit AS again, and you're original presets are returned.

(Nguồn: <https://www.audiforums.com/forum/audi-tt-7/what-does-button-do-radio-71340/>)

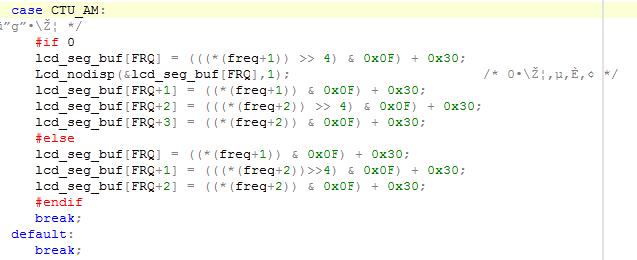
10. ma\_dest = 0x02 (AM: 522 -> 1629KHz FM: 76MHz -> 90Mhz)

11. Hiển thị tần số:

FM: seg1 -> 4 + dấu chấm



AM: tùy người lập trình, dùng seg1 -> 3 hay seg1 -> 4



File lcd\_mode.c

12. Main mode ISR

**Instant Station Recall (ISR)**

This function allows you to recall a specific radio station from any mode. Use it to find stations quickly when you want to hear specific information (such as traffic announcements) while driving. The ISR station is set to 87.5 MHz upon shipment from the factory.  
●**ISR memory**1) Press the [**FNC**] button to select the radio mode.

2) Tune in the station to be set as the ISR station.

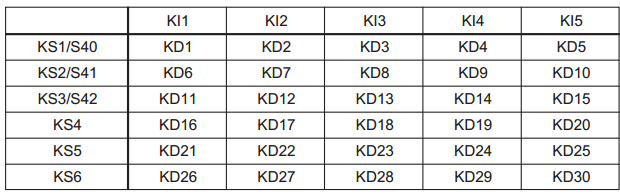
3) Press the [**ISR**] button for more than 2 seconds to store that station in the memory as the ISR station.

●**Recalling a station with ISR**Press the [**ISR**] button from any mode. **“ISR”** appears on the display and the set station is tuned in. To cancel, press the [**ISR**] button again.

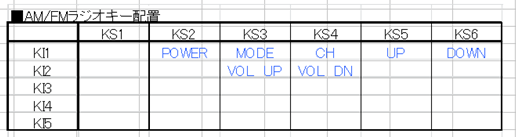
(Nguồn: <http://www.manualsdir.com/manuals/60264/clarion-ax430.html?page=8>)

Trong kadai6: Khi qua mode ISR sẽ chuyển qua tần số AM.

13.



Trong file LC75857E



File công ty

Từ 2 bảng trên, ta có tkey\_code\_tbl[] trong file key\_main.h

**14. Sự kiện xảy ra khi:**

**TH1: Nhấn nhanh key rồi thả**

**Th2: Nhấn giữ key lâu**

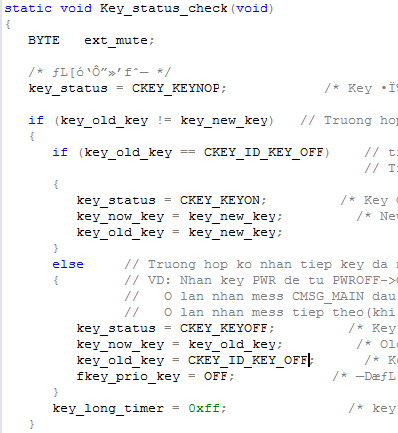
**TH3: ko nhấn key trong thời gian dài sau khi đã nhấn key trước đó**

TH1: Khi nhấn key chưa release, Module Key tại lần nhận mess CMSG\_MAIN đầu tiên sẽ có key\_status = KEYON

-> thực hiện function KEYON

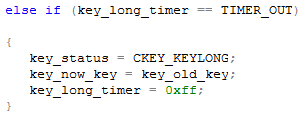
Khi release key, Module Key tại lần nhận mess CMSG\_MAIN thứ hai, key\_status = KEYOFF

-> thực hiện function KEYOFF.



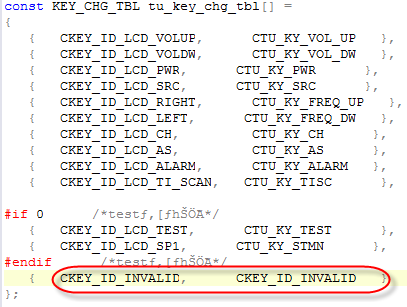
TH2: Khi nhấn giữ key lâu, LCD driver sẽ quét liên tục tất cả key theo chu kì 615T(s) và gửi key data tương ứng liên tục cho VDK sau 615T(s).

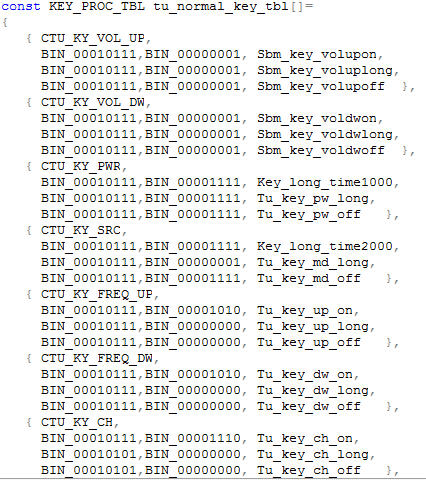
Vì luôn nhận được cùng 1 key data nên key\_old\_key luôn bằng key\_new\_key. Trong trường hợp này nó sẽ đợi key\_long\_timer = 0 rồi gán key\_status = KEYLONG. => thực hiên function KEYLONG

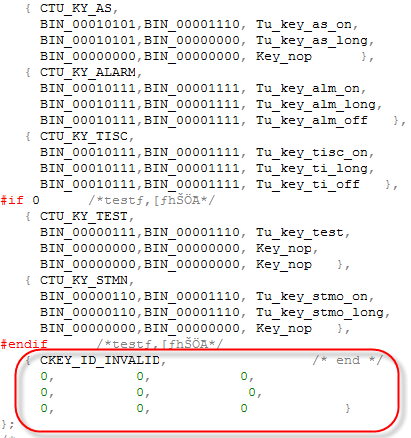


TH3: Nếu ko nhấn key trong thời gian dài, VDK sẽ ko nhận được bit key data nào, cho nên khi vào key table của mode nào đó, key code sẽ là CKEY\_ID\_INVALID dẫn đến function của key code đó cũng ko có => ko có sự kiện gì xảy ra.

VD: Sau khi nhấn thả key PWR ->PWRON -> Màn hình hiện thị nội dung mode TUNER. Trong trường hợp ko nhấn key thì keycode = CKEY\_ID\_INVALID và function liên quan cũng ko có.

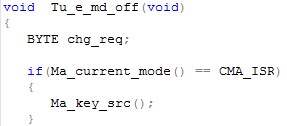


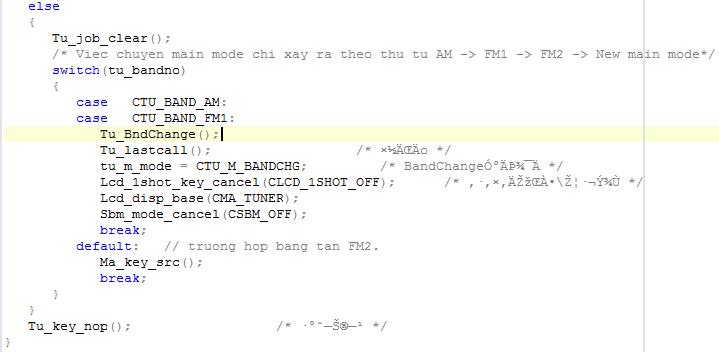




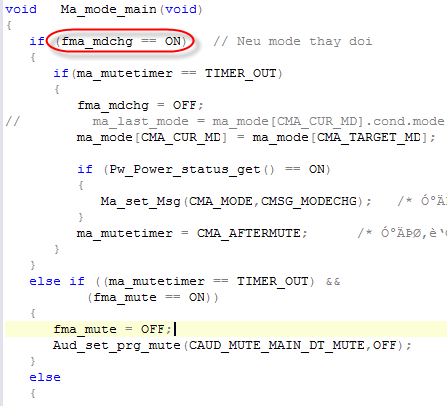
Trong file tu\_keytbl.h

15. Viec chuyen main mode chi xay ra theo thu tu AM -> FM1 -> FM2 -> New main mode.





Ma\_key\_src () sẽ set cờ fma\_mdchg = ON và cờ được sử dụng trong hàm Ma\_main\_mode().

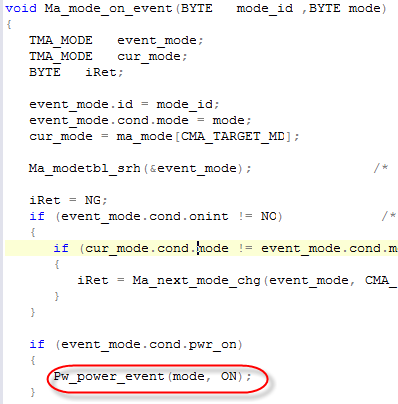


16. Main mode chỉ có 3 mode chính là: TUNER, AUX, ISR

Mode TUNER, AUX được chuyển qua lại bởi key MODE.

Mode ISR được chuyển tới khi nhấn key TI. Mode ISR được xử lí trong module Tuner.

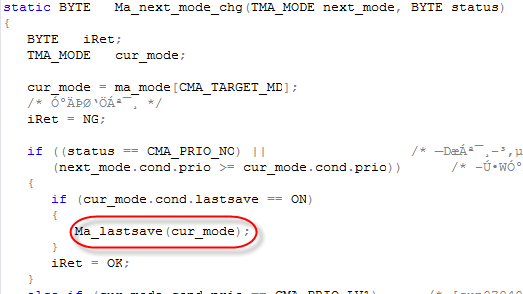
17. Khi đang POWEROFF, nếu nhấn key TI, mode ISR sẽ ON và POWER cũng ON.



18.

Khi chuyển main mode, VD từ Tuner -> AUX . Lastmode = TUNER.

Khi chuyển từ mode level thấp sang cao(VD: Tuner -> ISR) thì lastmode = mode level thấp.

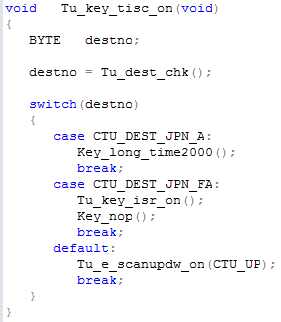


19. radio đang ở mode Tuner(TUNER ON), nhấn TI key để bật mode ISR. Có 3 giai đoạn:

GĐ1: Nhấn key nhưng chưa thả:

Nếu là vùng tần số Japan AM: thì sẽ quét 2s

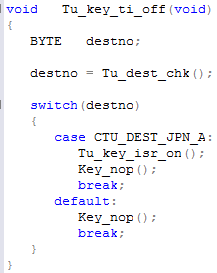
Nếu là vùng tần số Japan AM/FM: bật hay tắt ISR.



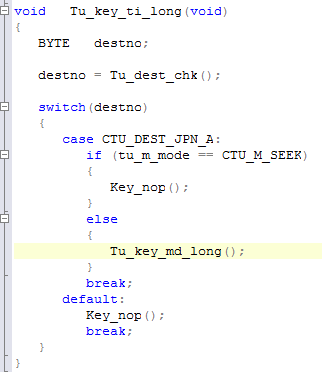
GĐ2: Thả key

Nếu là vùng tần số Japan AM: Bật hay tắt ISR

Nếu là vùng tần số Japan AM/FM: Ko thực hiên gì hết



GĐ3: Nhấn giữ key



Nếu là vùng tần số Japan AM:

+ Đang trong mode Tuner SEEK -> vô hiệu

+ Nếu ko fai mode SEEK: tạo tiếng beep

Nếu ko fai vùng tần số Japan AM: Việc nhấn giữ ko tạo ra hiệu ứng gì hết.

**20. Từ ISR mode -> Tuner/ AUX mode.**

TH1: nhấn key TI: ISR ON-> OFF

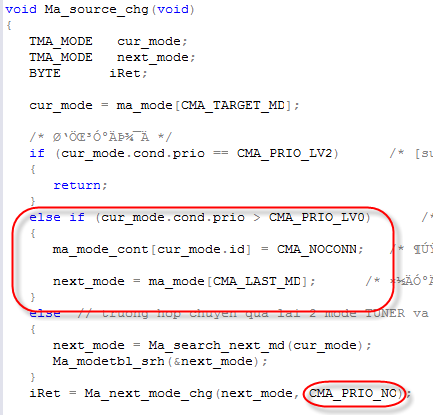
Giả sử từ Tuner -> ISR => lastmode = Tuner. Khi nhấn key TI, ISR OFF và sẽ chuyển sang lastmode được lưu trc đó, ở đây là Tuner.

Vì ISR ko có tính năng lastmode nên khi ISR OFF, hệ thống sẽ ko lưu lastmode là ISR mà sẽ giữ nguyên lastmode trc đó.

TH2: nhấn key MODE

Cũng tương tự trên, curmode hiện tại sẽ là CMA\_NOCONN. Và nextmode sẽ là lastmode được save trước đó.

Việc chuyển từ mode ISR sang Tuner/AUX khi nhấn key MODE sẽ ko check độ ưu tiên.



21. Lastmode xảy ra ở 3 trường hợp:

+ Từ main mode này sang main mode khác cùng độ ưu tiên

+ Từ main mode độ ưu tiên thấp lên main mode độ ưu tiên cao

+ Từ PWRON -> PWROFF

22.

Từ mode AUX ->TUNER: Chuyển sang band AM và hiện lại tần số và preset… trước khi vào AUX.

**23. Có thể chuyển từ mode AUX -> ISR**: Chuyển bình thường.

**23. Phân biệt 2 đoạn code trong hàm Key\_code\_process() và Key\_CodeCheck\_func() trong 2 hình dưới.**

Đoạn code trong Key\_CodeCheck\_func(): Nếu người dùng nhấn nhiều key cùng lúc thì sẽ vô hiệu

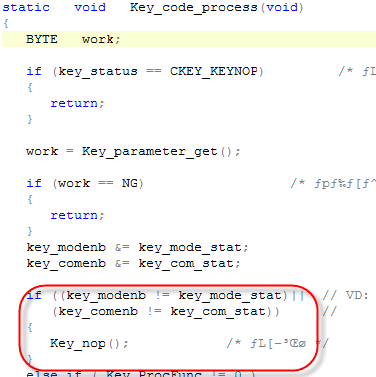
Đoạn code (khung đỏ) trong hàm Key\_code\_process() : Sauk hi đã check là chỉ 1 key được nhấn, ta sẽ xét xem với mode đó, những thao tác nào vô hiệu.

VD: đang trong mode TUNER SEEK => key\_mode\_stat = 0x02,

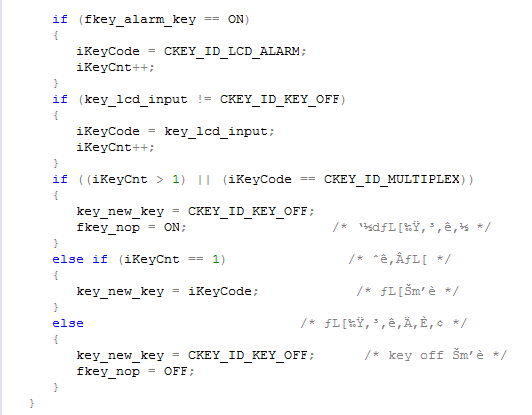
nếu nhấn key VOLUP/VOLDOWN thì key\_modenb = 0x01.(Xem file tu\_keytbl.h)

khi này key\_modenb = key\_modenb & key\_mode\_stat = 0x00 => thoa dieu kien => Key\_nop().

=> Đang trong quá trình SEEK, ko thể VOLUP/VOLDOWN



Hàm Key\_code\_process() trong file key\_main.c



Hàm Key\_CodeCheck\_func() trong file key\_main.c

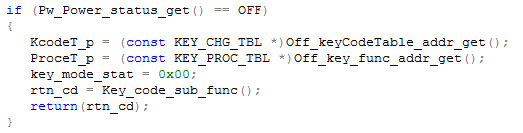
**24. fpw\_wakeup là cờ diễn tả trang thái NORMAL trong ACCOFF.**

fpw\_wakeup = ON => NORMAL ACCOFF

fpw\_wakeup = OFF => HOLDX

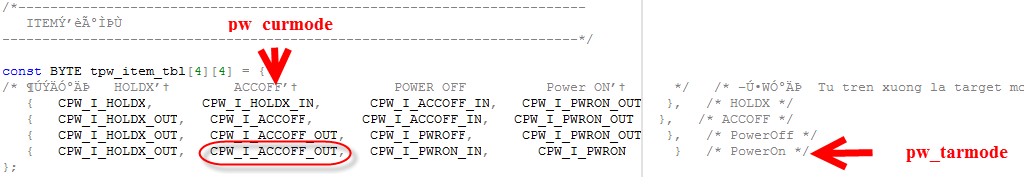
Được set ON trong hàm Clk\_time\_main().

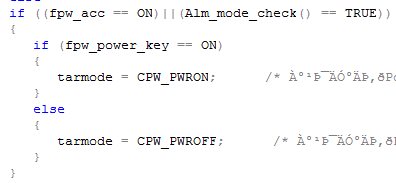
**25. Key module chỉ xét TH “ko là PWRON”(pw\_curmode != ON sẽ trả về OFF, tức pw\_curmode = PWROFF hay ACCOFF đều trả về OFF ). Khi nhấn key POWER (đang trong trạng thái ACCOFF) dù fpw\_power\_key = ON nhưng khi qua POWER module, nó cũng ko thể set power mode là PWRON vì khi này fpw\_acc = OFF.**



**=> Khi đang trong ACCOFF, nếu nhấn key POWER, fpw\_power\_key sẽ thay đổi. Nếu quay lại ACCON, nó sẽ dựa vào fpw\_power\_key để chuyển sang PWRON hay PWROFF.**

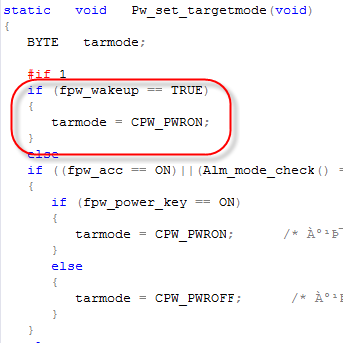
**Ta biết là khi từ PWRON -> ACCOFF, mặc dù có chuyển xuống PWROFF nhưng fpw\_power\_key ko bằng OFF, chứng tỏ nếu từ ACCOFF sang chuyển sang ACCON, nó sẽ chuyển qua PWRON thay vì PWROFF.**



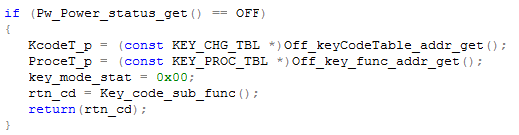


**26. Việc chuyển sang NORMAL ACCOFF khiến cho radio luôn luôn ở trạng thái PWRON.**

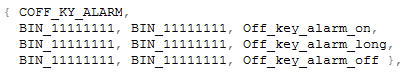
**fpw\_wakeup = FALSE khi đang ở HOLDX(ACCOFF)**



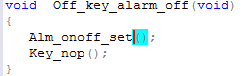
**27. Đang PWROFF hay ACCOFF(nói chung ko đang ở PWRON vì trong KEY module, nếu muốn vào OFF module thì ko dc pw\_curmode != PWRON)**

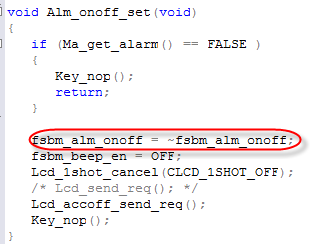


- Nhấn key ALARM: cho phép alarm hoạt động hay ko



Off module





fsbm\_alm\_onoff được check ở Clk\_time\_main() trong file clock\_mode.c

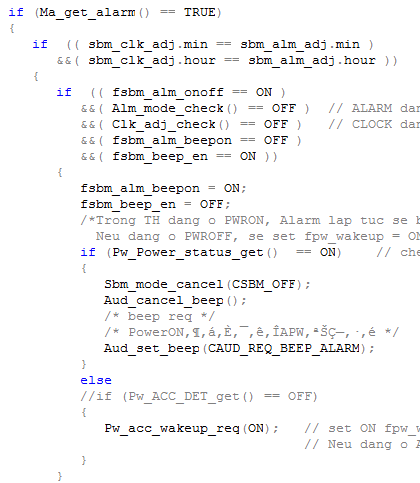
- Nhấn key POWER:

+ Nhấn thả: fpw\_power\_key thay đổi

+ Nhấn giữ: bắt đầu chế đố điều chỉnh clock(nếu ACCON)

- Nhấn key MODE: fpw\_power\_key thay đổi

28.



Clk\_time\_main()

Nếu cho phép ALARM hoạt động và clock chạy đúng thới gian ALARM dc set, nếu ko đang ở PWRON(PWROFF hay ACCOFF) thì:

Sẽ chuyển sang PWRON và giữ nguyên trạng thái vì khi này fpw\_wakeup = ON. (Xem mục 24 và 26).

**29. Khi đang ở ACCOFF, LCD hiển thị sẽ có 2 trường hợp:**

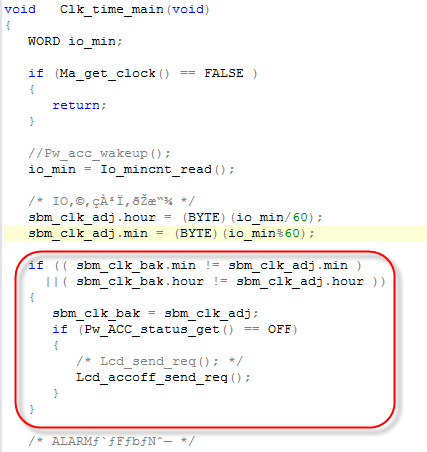
TH1: Đang trong mode ALARM ADJUST. => LCD hiển thị nhấp nháy thời gian ALARM

TH2: Đang trong mode ALARM NORMAL => Hiển thị thời gian CLOCK. Thời gian được cập nhật thông qua hàm Clk\_time\_main() trong clock\_mode.c. LCD ko hiển thị nhấp nháy TG khi đang ở CLOCK ADJUST

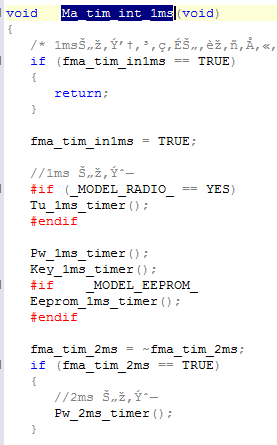
Xem hình dưới, ta thấy io\_min luôn cập nhật

=> **sbm\_clk\_adj.min** và **sbm\_clk\_adj.hour** luôn khác **sbm\_clk\_bak.min** và **sbm\_clk\_bak.hour**

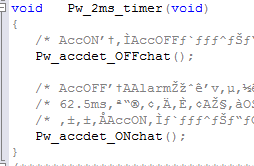
=> Luôn vào if.



**30. Cờ fpw\_acc được gán trong hàm Pw\_2ms\_timer(), và cứ mỗi 2ms thì hàm này được gọi trong Ma\_tim\_int\_1ms().**



Trong ma\_tim.c



Trong pw\_funct\_ext.c. Hàm Pw\_accdet\_OFFchat() check ACCOFF và Pw\_accdet\_ONchat()check ACCON