F²MC-8FX Family MB95200 Series Wake-Up on Key Stroke

This application note describes how to wake-up from standby mode on key stroke and gives one sample code.

1 Introduction

This document describes how to wake-up from standby mode on key stroke and gives one sample code.

2 Standby Modes

This chapter introduces standby modes and the STBC register.

Usually a device has limited power supply. To minimize power consumption, the device will transit to low power consumption modes (standby modes) when it is not in operation state.

The device in standby modes can be waken-up on key stroke to operating mode.

There are four standby modes: sleep mode, stop mode, timebase timer mode and watch mode. The device transits to a standby mode according to the setting of the standby control register (STBC).

The device can be released from a standby mode by an interrupt or a reset. Before transiting to normal operation, the device may wait for the oscillation stabilization wait time to elapse if necessary.

Figure 1 shows the Standby Control Register (STBC). This register is used to control transition from the RUN state to a standby mode, set the pin state in a standby mode, and control the generation of a software reset.

Figure 1. Standby Control Register (STBC)

Address	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
0008_H	STP	SLP	SPL	SRST	TMD	SCRDY	MCRDY	MRDY	$00000XXX_B$
	R0/W	R0/W	R/W	R0/W	R0/W	R/WX	R/WX	R/WX	

For detailed information on standby modes, please refer to “AN205271-F^{2}MC - 8FX Family, MB95200H/210H Series, Basic FW Setup” on Cypress website.

www.cypress.com/documentation/application-notes/mb95200-basic-fw-setup

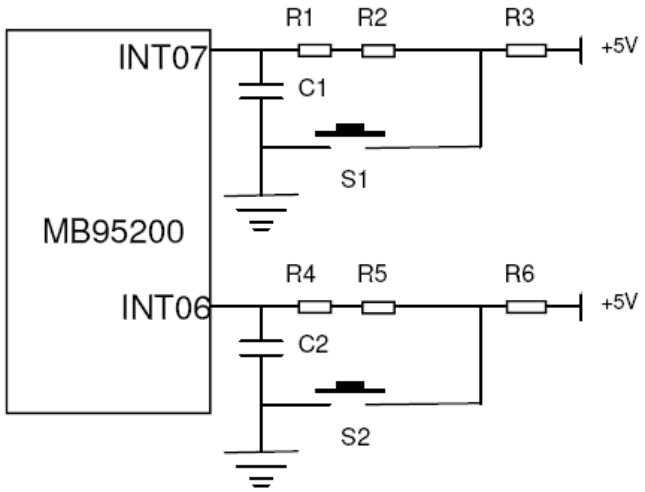
3 Wake-up on Key Stroke

This chapter introduces how to wake up a device from standby modes on key stroke.

To wake up the device from a standby mode on key stroke, first initialize the I/O register and enable the external interrupt, then the device enters a standby mode such as stop mode, sleep mode, timebase timer mode or watch mode. The device in a standby mode can wake up and return to operating mode by pressing an external key.

3.1 Hardware Setting

Figure 2. Hardware Setting



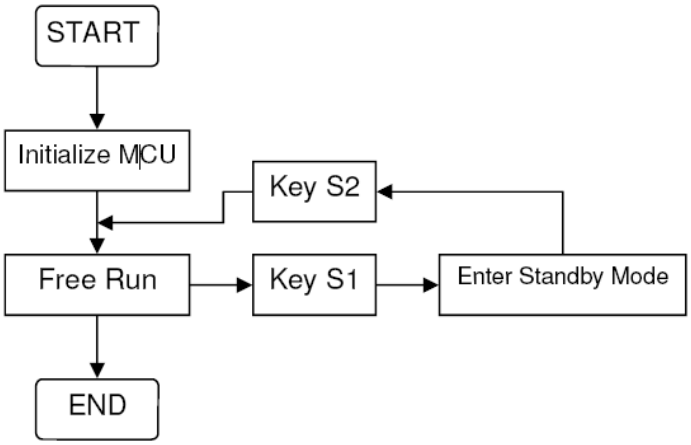
3.2 Software Design

When an interrupt request whose interrupt level is higher than "11B" is issued in a standby mode, the device is released from the standby mode.

To wake up the device on key stroke, use the external interrupt 6 and 7. Press key S1, enter the external interrupt 7 and set the STBC register, and the device enters a standby mode.

Press key S2 and enter the external interrupt 6, the device will release from the standby mode and transit to normal operation.

The flow chart is illustrated as below:



4 Sample Code

This chapter illustrates how to wake up a device from a standby mode on key stroke with an example.

The following code can make a device transit to stop mode and be released from stop mode.

```
#include"mb95200.h"
#define LED2 PDR0_P05
#define LED3 PDR6_P64
#define LED4 PDR6_P63
#define Light 0 /*LED light*/
#define Unlight 1 /*LED unlight*/
/****************vDelay routine**********************/
unsigned char vDelay(unsigned int uiCount)
{
       int i;
       while(uiCount--)
       {
              for(i=0;i<100;i++)
              asm("\tNOP");
       }
}
/*************************************************************************
NAME: __interrupt void external_int06(void)
FUNCTION: LED3,LED4 Lihgt _ Unlight
*************************************************************************/
__interrupt void external_int06(void)
{
       EIC30_EIR0=0;       /*CLEAR INTERRUPT REQUEST*/
       /*LED3 LIGHT TO UNLIGHT,LED4 LIGHT TO UNLIGHT*/
       LED3=Light;
       LED3=Unlight;
       LED4=Light;
       vDelay(10);
       LED4=Unlight;
}
/*************************************************************************
NAME: __interrupt void external_int07(void)
FUNCTION: ENTERY STANDBY MODE (STOP MODE)
*************************************************************************/
__interrupt void external_int07(void)
{
       EIC30_EIR1=0; /*CLEAR INTERRUPT REQUEST*/
       /*LED2,3,4 KEEP LIGHT*/
       LED2=Light;
       LED3=Light;
       LED4=Light;
       STBC_STP=1; /*ENTERY STOP MODE */
}
```

```
/*******************MCU initialization*********************/
void MCU_initialization(void)
{
       __DI();
       SYSC=0x03;
       DDR6_P63=1;          /* Enable output */
       DDR6_P64=1;          /* Enable output */
       DDR0_P05=1;          /* Enable output */
       LED2=Unlight;
       LED3=Unlight;
       LED4=Unlight;
       /* external interrupt */
       EIC30=0x55;          /* INT06 INT07 enable falling edge */
       InitIrqLevels();     /* INIT INTERRUPT LEVELS SET */
       __EI();              /* ENABLE ALL INTERRUPT */
}
/*********************main routine*************************/
void main(void)
{
       MCU_initialization();
       STBC_SPL=0;          /* EXTERN PIN HOLD PRECEDING */
       while(1)
       {
              vDelay(5);
              LED2=~LED2;
              LED3=~LED3;
              LED4=~LED4;
              vDelay(5);
       }
}
```

5 Additional Information

For more information about how to use MB9595200H/210H EV-board, BGM Adaptor and SOFTUNE, please refer to SKT MB2146-410A-01-E User Guide, or visit websites:

www.cypress.com/supporttools/8fx

www.cypress.com/documentation/application-notes/mb95200-wake-key-stroke

Document History

Document Title: AN205512 - F²MC-8FX Family MB95200 Series Wake-Up on Key Stroke

Document Number: 002-05512

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	—	HUAL	04/13/2009 04/15/2009	Initial release. Add detailed description on standby modes.
*A	5262013	HUAL	05/09/2016	Migrated Spansion Application note from MCU-AN-500039-E-11 to Cypress format.
*B	5844606	AESATMP9	08/04/2017	Updated logo and copyright.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2009-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.