ARM-FITKIT3: Aplikace modulu Watchdog Timer (WDOG)

Predmet: IMP – Mikroprocesorové a vestavěné systémy

Autor : René Bolf(xbolfr00) 18.12.2019

OBSAH

OBSAH 2
1 Úvod 1
1.1 Cieľ projektu 1
1.2 Watchdog Timer 1
2 Popis aplikácie a implementácia 1
2.1 Popis aplikácie 1
2.2 Demonštrácia Watchdogu 1
2.2.1 Použité registre 1
2.2.2 Watchdog Módy 2
2.3 Popis implementácie watchdogu 2
3 Záver 3
4 Literatúra 3
5 Obrázky 4

1 Úvod

1.1 Cieľ projektu

Cieľom projektu bolo demonštrovať možnosti modulu Watchdog Timer (WDOG) dostupného na mikrokontroléri Kinetis K60 z dosky platformy FITkit3.

1.2 Watchdog Timer

Watchdog je počítačová periféria, ktorá resetuje systém pri jeho za cyklení. To znamená, že Watchdog zabezpečí, že ak nastane akákoľvek chyba na mikrokontroléri tak pomocou resetu dokáže priviesť systém zo zaseknutého stavu späť k normálnej funkcií. Bez Watchdogu by musel byť mikrokontrolér resetovaný manuálne, čo by u čipov v automatických strojoch na odľahlých alebo špatne dostupných miestach mohol byť problém.

2 Popis aplikácie a implementácia

2.1 Popis aplikácie

Projekt je implementovaný v programovacom jazyku C v prostredí Kinetis Design Studio. Aplikácia bola vytvorená a testovaná pre mikrokontrolér Kinetis K60 z dosky platformy FITkit3.

Pre demonštrovanie možnosti modulu Watchdog Timeru je použitá jednoduchá vstavaná aplikácia prevzatá z demo cvičenia, ktorej autorom je pán doktor M. Bidlo. Táto aplikácia tvorí asi 60 % programu zvyšných 40 % je môj program pre demonštráciu WDOGu.

Aplikácia jednoduchým spôsobom demonštruje funkcionalitu Watchdogu, v nekonečnom cykle beží program, ktorý je zo začiatku indikovaný blikaním LED diódy LED_9, Táto základná funkcionalita je prevzatá z programu z democvičenia. Modul Watchdog očakáva periodické refresh správy, ktoré mu povedia, že systém beží v poriadku a teda nie je potrebný reset systému. Táto refresh správa je odoslaná pomocou tlačidla SW3. Po tomto stlačení sa odošle refresh správa, ktorá oznámi modulu WDOG, že systém beží v poriadku, okrem toho po stlačení tohto tlačidla SW3 začne LED dióda LED_9 len svietiť, ak sa pošle ďalšia refresh správa tak začne svietiť aj led dióda LED_10, ak sa pošle ďalšia refresh správa začne svietiť aj led dióda LED_11 atď. Ak budú svietiť už všetky 4 led diódy a pošle sa refresh správa tak sa postupne začnú vypínať až sa dostanú do počiatočného stavu kde bliká led dióda LED_9.

2.2 Demonštrácia Watchdogu

2.2.1 Použité registre

- **WDOG_STCRHL** – tento register sa používa na povolenie Watchdogu, nastavenie zdroja pre hodiny na LPO – Low- Power oscilátor, ktorý má frekvenciu 1kHz, nastavenie windowed (okienkového) módu.
- **WDOG_TOVALH / WDOG_TOVALL** – tento register slúži na nastavenie veľkosti periódy

- **WDOG_WINH / WDOG_WINL** – tento register slúži na nastavenie veľkosti okna, ktoré je potrebné pri použití okienkového módu Watchdogu
- **WDOG_UNLOCK** – tento register slúži na odomknutie možnosti zmien hodnôt registrov Watchdogu, používajú sa hodnoty 0xC520 a 0xD928
- **WDOG_REFRESH** – prijíma refresh správy pre watchdog, používajú sa hodnoty 0xA602 a 0xB480
- **WDOG_PRESC** – tento register slúži na nastavenie veľkosti pred deliča , v programe je využitá hodnota 0, to znamená že jeden watchdog cyklus trvá 1ms, preto pri nastavení 15 sekundovej periódy je hodnota 15000 čo symbolizuje 15 tisíc milisekúnd (15 sekúnd)

2.2.2 Watchdog Módy

Watchdog ponúka dva módy a to **periodický** mód a **okienkový** mód (windowed)

Periodický mód – tento mód je využívaný na kontrolu správneho behu programu. Nastaví sa perióda pomocou registrov WDOG_TOVALH a WDOG_TOVALL na nejakú hodnotu, v aplikácii je nastavená hodnota 10 sekúnd. V tejto perióde sa očakáva prijatie refresh správy, ak refresh správa nepríde nastane reset systému.

Okienkový mód (windowed) – tento mód je využívaný na kontrolu rýchlosti behu programu. Okrem nastavenia periódy je nutné nastaviť aj veľkosť okna. Veľkosť okna sa nastavuje registrami WDOG_WINH a WDOG_WINL, v aplikácii je nastavená hodnota 2,5 sekundy. Pri tomto móde sa refresh správa očakáva v intervale od nastavenej hodnoty okna až po hodnotu periódy.

Napríklad ak máme nastavenú periódu na hodnotu 10 sekúnd a okno má hodnotu 2,5 sekundy, refresh správa je očakávaná od 2,5 sekundy po 10 sekundu. Ak by správa refresh prišla mimo tento interval alebo by neprišla vôbec nastane reset systému.

2.3 Popis implementácie watchdogu

Program obsahuje funkciu s názvom void **WdogInit()**, ktorá slúži na nastavenie (inicializáciu) watchdogu. V tejto funkcií sa vykonáva odomknutie Watchdogu a to pomocou registra WDOG_UNLOCK tento register musí mať hodnoty 0xC520 a 0xD928. Ďalej je nastavený pred delič na hodnotu 0 WDOG_PRESC = 0x00;

Nastavenie módu watchdogu sa nastavuje pomocou registru WDOG_STCTRLH. **Periodický mód** má nastavenú hodnotu **WDOG_STCTRLH = 0x5** (0101) –

- 0 – windowed mod je vypnutý
- 1 – IRQRSTEN je zapnutý a teda WDOG najprv vygeneruje prerušenie, až potom generuje reset
- 0 – nastavenie WDOG hodín zdroja na LPO
- 1 – zapnutie WDOG

Windowed mód má nastavenú hodnotu **WDOG_STCTRLH = 0xD** (1101), jediný rozdiel je, že je nastavený windowed mód. V tejto funkcií sa nastavuje aj veľkosť periódy a to registrami WDOG_TOVALH = 0 WDOG_TOVALL = 10000 čo je 10 sekúnd. Samozrejme táto hodnota sa dá meniť.

Ak je zvolený okienkový mód tak je potreba nastaviť okienko a to pomocou WDOG_WINH = 0 WDOG_WINL = 2500 a to znamená že je to 2,5 sekundy.

Okrem tejto funkcie program obsahuje aj funkciu, ktorá má názov **kickingTheDog()** a v tejto funkcie sa nastavuje len refresh správy a to WDOG_REFRESH = 0xA602 a WDOG_REFRESH = 0xB480; Tento refresh sa nazýva „kopnutie do psa“ a teda ak sa nepošle refresh správa resetuje sa systém. V hlavnej časti programu v maine() sa vykonáva program v nekonečnom cykle, kde pri stlačení tlačidla **SW3** nastane refresh správy a teda zavolá sa funkcia **kickingTheDog(),** okrem toho pri tomto stlačení sa postupne rozsvecujú led diódy podľa počtov refresh správ, ak ich bolo 4 svietia všetky štyri led diódy. Následne pri ďalšej refresh správy sa postupne vypínajú led diódy.

3 Záver

Program na jednoduchej aplikácií demonštruje použitie Watchdogu a to okienkového módu a periodického módu. Pomocou aplikácií je možné demonštrovať dopad série nevčasných obslúh WDOG na chod vstavanej aplikácie a tak isto je možné demonštrovať aj včasnú obsluhu.

4 Literatúra

K60 Sub-Family Reference Manual kapitola 24: Watchdog Timer (WDOG) to jest strana od 542 do 564

5 Obrázky

Všetky obrázky sú prevzaté zo zdroja literatúry K60 Sub-Family Reference Manual.

24.7.1 Watchdog Status and Control Register High (WDOG_STCTRLH)

Address: 4005_2000h base + 0h offset = 4005_2000h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read	0	DISTESTWDOG	BYTESEL[1:0]		TESTSEL	TESTWDOG	0	Reserved	WAITEN	STOPEN	DBGEN	ALLOWUPDATE	WINEN	IRQRSTEN	CLKSRC	WDOGEN
Write																
Reset	0	0	0	0	0	0	0	1	1	1	0	1	0	0	1	1

WDOG_STCTRLH field descriptions

Field	Description
15 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
14 DISTESTWDOG	Allows the WDOG's functional test mode to be disabled permanently. After it is set, it can only be cleared by a reset. It cannot be unlocked for editing after it is set. 0 WDOG functional test mode is not disabled. 1 WDOG functional test mode is disabled permanently until reset.
13–12 BYTESEL[1:0]	This 2-bit field selects the byte to be tested when the watchdog is in the byte test mode. 00 Byte 0 selected 01 Byte 1 selected 10 Byte 2 selected 11 Byte 3 selected
11 TESTSEL	Effective only if TESTWDOG is set. Selects the test to be run on the watchdog timer. 0 Quick test. The timer runs in normal operation. You can load a small time-out value to do a quick test. 1 Byte test. Puts the timer in the byte test mode where individual bytes of the timer are enabled for operation and are compared for time-out against the corresponding byte of the programmed time-out value. Select the byte through BYTESEL[1:0] for testing.
10 TESTWDOG	Puts the watchdog in the functional test mode. In this mode, the watchdog timer and the associated compare and reset generation logic is tested for correct operation. The clock for the timer is switched from the main watchdog clock to the fast clock input for watchdog functional test. The TESTSEL bit selects the test to be run.
9 Reserved	This field is reserved. This read-only field is reserved and always has the value 0.
8 Reserved	This field is reserved.
7 WAITEN	Enables or disables WDOG in Wait mode. 0 WDOG is disabled in CPU Wait mode. 1 WDOG is enabled in CPU Wait mode.
6 STOPEN	Enables or disables WDOG in Stop mode.

Table continues on the next page...

Obrázok 1 : Register WDOG_STCTRLH

WDOG_STCTRLH field descriptions (continued)

Field	Description
	0 WDOG is disabled in CPU Stop mode. 1 WDOG is enabled in CPU Stop mode.
5 DBGEN	Enables or disables WDOG in Debug mode. 0 WDOG is disabled in CPU Debug mode. 1 WDOG is enabled in CPU Debug mode.
4 ALLOWUPDATE	Enables updates to watchdog write-once registers, after the reset-triggered initial configuration window (WCT) closes, through unlock sequence. 0 No further updates allowed to WDOG write-once registers. 1 WDOG write-once registers can be unlocked for updating.
3 WINEN	Enables Windowing mode. 0 Windowing mode is disabled. 1 Windowing mode is enabled.
2 IRQRSTEN	Used to enable the debug breadcrumbs feature. A change in this bit is updated immediately, as opposed to updating after WCT. 0 WDOG time-out generates reset only. 1 WDOG time-out initially generates an interrupt. After WCT, it generates a reset.
1 CLKSRC	Selects clock source for the WDOG timer and other internal timing operations. 0 WDOG clock sourced from LPO . 1 WDOG clock sourced from alternate clock source.
0 WDOGEN	Enables or disables the WDOG's operation. In the disabled state, the watchdog timer is kept in the reset state, but the other exception conditions can still trigger a reset/interrupt. A change in the value of this bit must be held for more than one WDOG_CLK cycle for the WDOG to be enabled or disabled. 0 WDOG is disabled. 1 WDOG is enabled.

Obrázok 2 : Register WDOG_STCTRLH pokračovanie

24.7.7 Watchdog Refresh register (WDOG_REFRESH)

Address: 4005_2000h base + Ch offset = 4005_200Ch

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	WDOGREFRESH															
Reset	1	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0

WDOG_REFRESH field descriptions

Field	Description
15–0 WDOGREFRESH	Watchdog refresh register. A sequence of 0xA602 followed by 0xB480 within 20 bus clock cycles written to this register refreshes the WDOG and prevents it from resetting the system. Writing a value other than the above mentioned sequence or if the sequence is longer than 20 bus cycles, resets the system, or if IRQRSTEN is set, it interrupts and then resets the system.

24.7.8 Watchdog Unlock register (WDOG_UNLOCK)

Address: 4005_2000h base + Eh offset = 4005_200Eh

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read Write	WDOGUNLOCK															
Reset	1	1	0	1	1	0	0	1	0	0	1	0	1	0	0	0

WDOG_UNLOCK field descriptions

Field	Description
15–0 WDOGUNLOCK	Writing the unlock sequence values to this register to makes the watchdog write-once registers writable again. The required unlock sequence is 0xC520 followed by 0xD928 within 20 bus clock cycles. A valid unlock sequence opens a window equal in length to the WCT within which you can update the registers. Writing a value other than the above mentioned sequence or if the sequence is longer than 20 bus cycles, resets the system or if IRQRSTEN is set, it interrupts and then resets the system. The unlock sequence is effective only if ALLOWUPDATE is set.

Obrázok 3: Registre WDOG_REFRESH a WDOG_UNLOCK