

# FAKULTA INFORMAČNÍCH TECHNOLOGIÍ VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ



## **Interpret jazyka IFJ16**

Dokumentace projektu do předmětu IMP  
Aplikace modulu Watchdog Timer, varianta 7

## **Obsah**

<b>1</b>	<b>Zadání</b>	<b>2</b>
<b>2</b>	<b>Watchdog Timer</b>	<b>2</b>
<b>3</b>	<b>Prvotní aplikace</b>	<b>2</b>
<b>4</b>	<b>Periodický režim</b>	<b>2</b>
<b>5</b>	<b>Okénkový režim</b>	<b>2</b>
<b>6</b>	<b>Nevčasná obsluha</b>	<b>3</b>
<b>7</b>	<b>Zdroje</b>	<b>3</b>

## 1 Zadání

Vytvoření aplikace, která demonstruje využití modulu watchdog timer na desce fitkit 3. Demonstrovat periodický (normální) a okénkový režim modulu watchdog timer. Dále demonstrujeme dopad nevčasné obsluhy watchdog timeru. Jako zdroj hodin předpokládáme Low Power Oscillator.

## 2 Watchdog Timer

Watchdog timer je hardwarová periférie starající se o správný chod systému u kterého není možný manuální reset, popřípadě je více pohodlné resetovat systém právě watchdogem. Funguje na principu čítače, který se pravidelně navyšuje a po dosažení určité hodnoty vyvolá restart systému. Pokud tento čítač nebudeme pravidelně nulovat, je vyvolán restart systému. Pro správné nulování je tedy zapotřebí správný chod programu a to jak softwarově, tak i hardwarově. Pokud se systém nějak zasekne (zacyklí, hardwarová chyba, přetečení) nebude docházet k pravidelnému nulování a watchdog tudíž vyvolá restart a následné zotavení systému. Nulování časovače se říká "refresh" watchdogu. Hodnota intervalu do které se může čítač inkrementovat dokud watchdog nevyvolá restart lze nastavit pomocí hodnot v registrech stejně tak jako ostatní hodnoty. Watchdog se používá u systémů, u kterých není možný manuální restart jako sondy ve vesmíru, podvodní zařízení a další podobné mechanismy.

Mezi důležité registry watchdogu v čipu Kinetis K60 patří: Watchdog Unlock register (WDOG\_UNLOCK) - provede se odemčení watchdogu pro následné upravování jeho registrů - odemčení se provede zápisem 0xC520 a 0xD928, Watchdog Status and Control Register High (WDOG\_STCTRLH) - nastavuje vypínání a zapínání watchdogu a nejrůznější možnosti watchdogu, Watchdog Time-out Value Register High (WDOG\_TOVALH), Watchdog Time-out Value Register Low (WDOG\_TOVALL) - nastavují horní limit pro časovač, po kterém se provede reset, Watchdog Window Register High (WDOG\_WINH), Watchdog Window Register Low (WDOG\_WINL) - určují dolní mez watchdogu nad kterou lze provádět refresh, Watchdog Refresh register (WDOG\_REFRESH) - určeny pro refresh watchdogu - refresh provedeme zápisem hodnot 0xA602 a 0xB480, Watchdog Prescaler register (WDOG\_PRESC) - určen pro přeskálování času - v naší aplikaci nastaven na 0. Tyto registry se budou nastavovat i v naší vestavěné aplikaci.

## 3 Prvotní aplikace

Prvotní aplikace využívá demoaplikaci, která rozblikává LEDku na fitkitu3. Frekvence blikání se dá navíc měnit pomocí tlačítek. Tato aplikace je poté rozšířena o inicializaci a pravidelné nulování watchdogu. Aplikace je převzata z přednášek IMP. V původní aplikaci se watchdog vypínal pomocí registru WDOG\_STCTRLH. Pro podrobnější informace o registrech je dobré zhlédnout dokumentaci k čipu K60 od strany 543.

## 4 Periodický režim

Přidává do aplikace inicializaci watchdogu ve funkci MCUInit a postupné refreshování v hlavní smyčce aplikace pomocí registru WDOG\_REFRESH. Inicializace se provádí tak, že nejprve odemkneme watchdog pomocí registru WDOG\_UNLOCK zápisem hodnot 0xC520 a 0xD928. Poté co watchdog odemkneme zapíšeme hodnotu timeoutu do registrů WDOG\_TOVALH (hodnota 0) WDOG\_TOVALL (hodnota 1200) - odpovídá přibližně jedné sekundě. Registr WDOG\_PRESC je nastaven na hodnotu 0. Registr WDOG\_STCTRLH nastaven na hodnotu 0xD1. Tato hodnota povoluje watchdog a zakazuje okénkový režim watchdogu. Pro periodický režim musí být makro win\_mod na hodnotě 0.

Refresh watchdogu poté provádíme v hlavní smyčce aplikace každou 400000 iteraci a to postupným zadáním hodnot 0xA602 a 0xB480 do registru WDOG\_REFRESH.

## 5 Okénkový režim

Pro demonstraci okénkového režimu je potřeba nastavit makro win\_mod na hodnotu 0. Při okénkovém režimu se může provést refresh watchdogu jen v určitém časovém intervalu určeného dolní mezí a horním timeoutem watchdogu. Pokud se provede refresh mimo tento interval nastane reset systému. Tato funkcionality se hodí pokud chceme např. zajistit, že systém není rychlejší ani pomalejší než by měl být. Spodní mez se nastavuje pomocí registrů

WDOG\_WINH (hodnota 0) a WDOG\_WINL (hodnota 600). Spodní mez je v naší aplikaci nastavena na přibližně 0.5s. Refresh watchdogu se tedy musí provést v rozmezí 0.5s až 1s.

## 6 Nevčasná obsluha

Demonstraci nevčasné obsluhy předvedeme pokud makro `err_mode` nastavíme na 1. Při tomto pokusu se aktivuje watchdog, ale již nedojde k jeho refreshi v hlavním cyklu aplikace. To způsobí doběhnutí časovače do nastaveného limitu a vyvolání resetu systému. Tomuto chování lze zabránit včasným a periodickým refreshováním watchdogu.

## 7 Zdroje

- Přednášky, opora, skripta a ukázkové aplikace z předmětu IMP na VUT FIT v Brně.
- <https://wis.fit.vutbr.cz/FIT/st/course-files-st.php?file=%2Fcourse%2FIMP-IT%2Flectures%2F06-imp-demo-FITkit3-src.tar&cid=12166>
- <https://mcuoneclipse.com/2015/05/27/semihosting-for-kinetis-design-studio-v3-0-0-and-gnu-arm-embedded-launchpad/>
- [http://cache.freescale.com/files/32bit/doc/ref\\_manual/K60P144M100SF2V2RM.pdf](http://cache.freescale.com/files/32bit/doc/ref_manual/K60P144M100SF2V2RM.pdf)
- Latex