

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Dokumentácia k projektu

ARM–KL05

LED náramkové hodinky na bázi RTC modulu

Obsah

1	Úvod	2
2	Popis implementácie	2
3	Záver	4

1 Úvod

Cieľom projektu bola implementácia vstavanej aplikácie v jazyku C pre mikrokontrolér Kinetis KL05, ktorá bude realizovať funkciu hodínok s nasledujúcimi požiadavkami:

- Informácia o čase sa zobrazí pomocou 7-segmentového LED displeja so 4 pozíciami pre jednotlivé číslice časového údaj.
- Na ovládanie funkcií hodínok k vyvolaniu aktuálneho času a jeho nastavovania bude slúžiť tlačítko na vývojovej doske.
- Čas bude meraný pomocou RTC–Real Time Clock periférie mikrokontroléru.
- Pri implementácii je nutné myslieť na dosiahnutie čo najnižšej spotreby pri využití vhodného režimu šetrenia energie tzv. sleep mód, ktorých mikrokontrolér ponúka viacero.

Na displeji sa bude zobrazovať čas, ktorý si užívateľ sám nastaví a po 15-sekundovej nečinnosti sa hodinky prepnú do sleep módu, čo znamená, že v tej dobe majú vypnutý displej.

2 Popis implementácie

V programe som implementovala nasledujúce funkcie:

InitMCU()

V tejto funkcii sa inicializujú základné časti mikrokontroléru a to nastavenie maximálnej frekvencie rozsahu časovača, jeho následné zapnutie a vypnutie watchdog-u.

InitPIN()

Funkcia inicializuje potrebné porty, zapína časovače na Porte A a Porte B, následne podľa schémy dostupnej k projektu nastaví jednotlivé piny pre prácu pomocou GPIO a ich smer.

InitRTC()

Funkcia inicializuje RTC perifériu nasledujúcim spôsobom. Zapnutím RTC časovača, resetovaním registrov, nastavením prerušení, zapnutím oscilátoru, nastavením počiatočných hodnôt pre počítadlo sekúnd a alarmu, povolenie prerušenia.

delay()

Krátka funkcia, ktorá vytvára omeškanie a využívam ju pri zobrazení čísiel na LED displeji.

get_time()

Funkcia zvyšuje hodnotu časového údaju po jednotlivých minútach.

increment_hours()

Funkcia využitá pri nastavovaní hodinovej časti číslicového údaju.

increment_minutes()

Funkcia využitá pri nastavovaní minútovej časti číslicového údaju.

RTC_IRQHandler()

Funkcia na obsluhu prerušenia vyvolanej RTC perifériou, ktorú dopĺňam o logiku pre inkrementáciu a nastavovanie časového údaju.

LLS_SleepMode()

Funkcia uvádzajúca mikrokontrolér do low-leakage stop módu. V tomto režime mikrokontrolér spí (prechádza z *RUN* módu do *STOP* módu). Uviesť ho späť do *RUN* módu je možné pomocou modulu LLWU – Low-Leakage Wake-Up Unit.

V tejto funkcii sa vypína *Clock monitor*, povoľuje sa PTB4 a RTC Alarm ako zdroj pre LLWU prerušenie, nastavuje sa aj register PMPROT s hodnotou pre LLS, taktiež aj SLEEPDEEP bit a v registri PMCTRL sa musí nastaviť STOPM na hodnotu 0x3. Pre zaistenie zápisu hodnôt do predošlých registrov je vhodné previesť jedno čítanie a jeden zápis (Read-After-Write Sequence). Nakoniec inštrukcia __WFI() uvedie mikrokontrolér do stavu, kde nevykonáva žiadny ďalší kód, len čaká na prerušenie.

main()

Funkcia pozostáva z prvotného volania InitMCU(), InitPIN(), InitRTC() a nekonečného cyklu, v ktorom pomocou konštrukcie switch prepínam medzi jednotlivými stavmi automatu, ktoré sú vysvetlené nižšie.

Popis konečného stavového automatu:

0. Displej je vypnutý, čo znamená, že neukazuje žiadny čas. Je volaná funkcia LLS_SleepMode(), ktorá uvádza mikrokontrolér do režimu spánku. Pomocou periférie RTC si počítam jednotlivé sekundy na základe počtu prerušení, po uplynutí 60 sekúnd pridám na hodnotu globálnych premenných reprezentujúcich jednotlivé číslice časových údajov. Zároveň čakáme na stlačenie tlačítka, po ktorom sa následne dostávam do stavu 1.

1. Zaznamenala som stlačenie tlačítka a rozsvetím LED displej, ktorý zobrazuje jednotlivé číslice časového údaju. Zároveň čakám na stlačenie tlačítka pre prechod do ďalšieho stavu alebo čakám 15 sekúnd pre opätovný prechod do stavu 0.
2. Po ďalšom stlačení tlačítka sa na displeji zobrazí iba hodinové číslice časového údaju, ktoré sa postupne inkrementujú. Po ďalšom stlačení sa presúvam do ďalšieho stavu s uloženými hodnotami pre číslice odpovedajúce hodinám časového údaju.
3. Začínam inkrementovať číslice na miestach minút a čakám na stlačenie tlačítka, ktorý ma privedie do stavu 1 a uloží nastavené hodnoty pre číslice na miestach minút časového údaju.

3 Záver

Projekt mi umožnil vyskúšať si programovanie mikrokontroléru s využitím RTC modulu, GPIO pinov a 7-segmentového displeja. Zároveň by som si priała viac takýchto interaktívnych a náučných projektov, kde si vieme vyskúšať rôzne mikrokontroléry.

Literatúra

[1] Drake, P.: Power Management for Kinetis and ColdFire+ MCUs. 2012.

Počas celej implementácie som sa riadila iba prednáškami, ukážkami a cvičeniami z predmetu IMP a referenčnými manuálmi pre môj typ mikrokontroléru [pdf](#).