

## Mikroprocesorové a vestavěné systémy

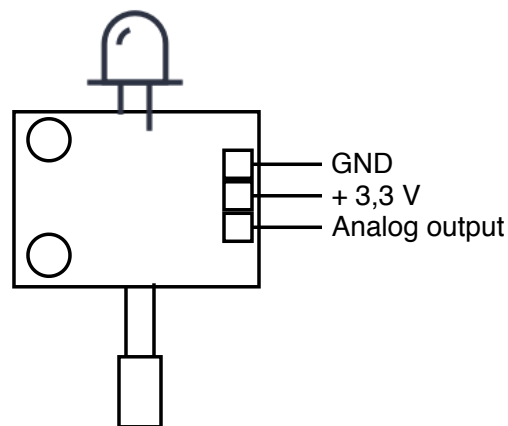
### Projekt – Měření srdečního tepu

Dominik Harmim (xharmi00)  
xharmi00@stud.fit.vutbr.cz  
30. prosince 2018

## 1 Úvod

Zadáním projektu byl návrh a implementace vestavné aplikace na výukové desce FitKit 3 [1], která obsahuje mikrokontrolér MCU Freescale KINETIS (MK60DN512ZVMD10), viz manuál [5]. K této desce se připojí modul pro měření srdečního tepu a modul se segmentovým LED displejem (zapojení těchto součástek bylo taktéž součástí řešení). Výsledná aplikace bude měřit frekvenci srdečního tepu (počet tepů za minutu) a bude zobrazovat výsledek na LED displeji.

Vestavná aplikace byla implementována v jazyce C v prostředí MCUXpresso IDE s využitím SDK pro MCU K60 [2].



Obrázek 1: Modul pro měření srdečního tepu

## 2 Zapojení hardware

Všechny externí moduly byly k desce FitKit 3 připojeny na skupinu pinů označených P1 (viz schéma zapojení FitKit 3 [6]).

### 2.1 Modul pro měření srdečního tepu

Modul pro měření srdečního tepu obsahuje tři vývody: napájení (+3,3 V), uzemnění a analogový výstup, jak demonstruje obrázek 1. Napájení bylo zapojeno na pin č. 1, uzemnění na pin č. 49 a analogový výstup na pin č. 14. Toto zapojení je znázorněno na obrázku 3.

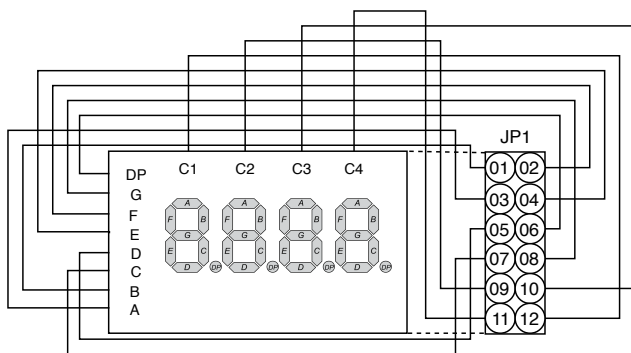
### 2.2 Modul se segmentovým displejem

Modul se segmentovým displejem obsahuje čtyři číslice skládající se ze sedmi segmenů standardně označených A-G a desetinnou tečkou za každou

z číslic standardně označenou jako DP. Modul obsahuje 8 vstupních pinů pro aktivaci těchto segmentů, včetně destinné tečky. Modul obsahuje další čtyři vstupní piny pro aktivaci jednotlivých číslic. Tyto piny jsou označeny C1-C4. V jednu chvíli může být aktivní pouze jedna číslice. Všechny výše uvedené piny jsou umístěny ve skupině pinů JP1 na zadní straně modulu. Význam jednotlivých pinů je uveden na obrázku 2. Skupina pinů JP1 byla zapojena na piny 17-28. Toto zapojení je naznačeno na obrázku 3.

### 2.3 Význam zapojení

Význam jednotlivých pinů a jejich použití v MCU, na které byly zapojeny výše uvedené moduly je popsán v tabulce 1.



Obrázek 2: Modul se segmentovým displejem

Pin	Zapojení na desce	Označení v MCU	Použití v MCU
Senzor	P1 – 14	L1	ADC0 CH0 P
Disp A	P1 – 26	L9	PTA11
Disp B	P1 – 28	L8	PTA9
Disp C	P1 – 22	C2	PTD14
Disp D	P1 – 24	M9	PTA10
Disp E	P1 – 25	J7	PTA6
Disp F	P1 – 27	J8	PTA7
Disp G	P1 – 21	C1	PTD15
Disp DP	P1 – 23	K8	PTA8
Disp C1	P1 – 17	C9	PTD8
Disp C2	P1 – 20	C3	PTD13
Disp C3	P1 – 19	B1	PTD12
Disp C4	P1 – 18	B9	PTD9

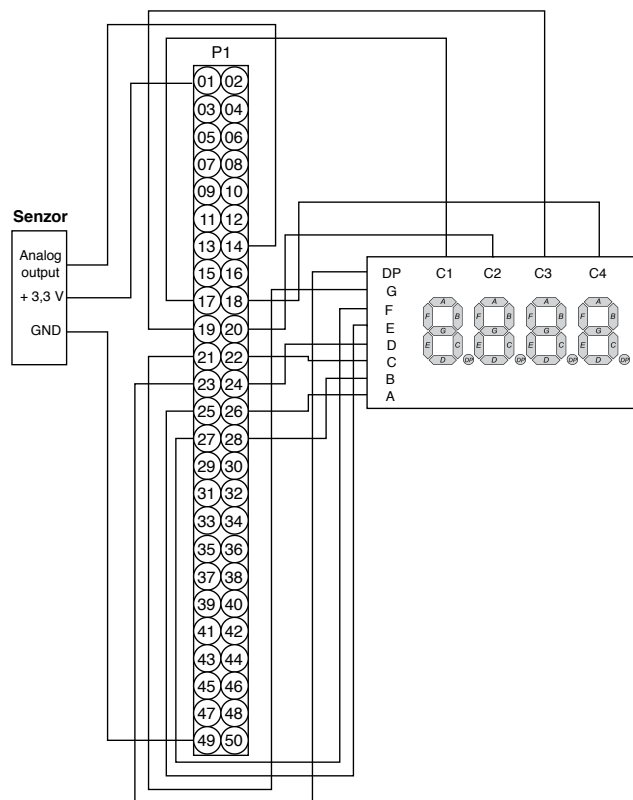
Tabulka 1: Význam zapojení

### 3 Způsob řešení

Implementace se nachází v souborech v adresářích board a source. Ostatní soubory byly vygenerovány při vytvoření projektu pro MCU K60 v prostředí MCUXpresso IDE.

V projektu jsou přímo využívány následující komponenty SDK pro MCU K60 (jsou umístěny v adresáři drivers):

- **ADC16** (16-bit SAR Analog-to-Digital Converter Driver) – čtení analogového signálu ze senzoru srdečního tepu a jeho převod do digitální podoby.
- **CLOCK** (Clock Driver) – nastavování hodin.
- **GPIO** (General-Purpose Input/Output Driver) – nastavování vstupních a výstupních pinů a zápis na vstupní piny LED displeje.
- **LPTMR** (Low Power Timer Driver) – počítání času měření srdečního tepu a času mezi jednotlivými srdečními pulsy.



Obrázek 3: Schéma zapojení

- **PIT** (Periodic Interrupt Timer Driver) – periodické obnovování LED displeje.
- **PORT** (Port Control and Interrupts) – nastavování vstupních a výstupních portů.

Vstupní bod aplikace (funkce main) se nachází v souboru source/main.c. Na začátku programu proběhne inicializace hardware a následně v nekonečné smyčce probíhá měření srdečního tepu a zobrazování tepové frekvence na displej.

#### 3.1 Konfigurace pinů

Konfiguraci vstupních a výstupních pinů byla nastavena v nástroji v prostředí MCUXpresso IDE, který na základě této konfigurace vytvořil soubory board/pin\_mux.h a board/pin\_mux.c. Tento nástroj k tomuto nastavení používá funkce komponent GPIO a PORT.

#### 3.2 Segmentový LED displej

Na začátku běhu aplikace se provádí inicializace časovače PIT, který periodicky v pravidelných intervalech zobrazuje jednotlivé číslice na LED displeji. Každých 4 166  $\mu$ s se generuje přerušení, které

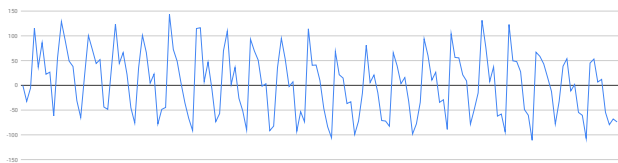
zobrazí další číslici. Toto odpovídá obnovovací frekvenci 60 Hz celého displeje. Nastavování jednotlivých pinů displeje se dělá funkcemi komponenty GPIO. Implementace displeje se nachází v souborech `source/display.h` a `source/display.c`.

### 3.3 Měření srdečního tepu

Na začátku běhu aplikace se provádí inicializace časovače LPTMR, který je použit pro měření času mezi jednotlivými měřeními srdečního tepu a času mezi jednotlivými srdečními impulsy. Tento časovač je vynulován při každém novém měření. Dále se provádí inicializace analogově-digitálního převodníku ADC16, který se používá pro čtení signálu ze senzoru a jeho převodu do digitální podoby. Samotný výpočet tepové frekvence je implementován v souborech `source/sensor.h` a `source/sensor.c`.

Po přečtení signálu ze senzoru srdečního tepu funkcemi komponenty ADC16 se signál filtruje. Provádí se tzv. dolní propust [4], která v signálu potlačí vyšší frekvence, tzn. bude odstraněn šum na vyšších frekvencích. Následně se provádí tzv. horní propust [3], která v signálu potlačí nižší frekvence, tzn. signál se bude pohybovat okolo nuly. Již filtrovaný signál je ilustrován na obrázku 4.

Vždy, když signál v nejvyšším bodě začne klesat, je detekován srdeční impuls. Počet srdečních tepů za jednu minutu je vypočítán vztahem  $rate = \frac{60}{t}$ , kde proměnná  $t$  je čas mezi dvěma impulsy. Tímto způsobem je zprůměrováno pět po sobě jdoucích hodnot a výsledek je zobrazen na LED displej. Navíc je každý nový výsledek průměrován i s několika předchozími, takže pokud je na senzoru přiložený prst delší dobu, tak by zobrazovaná tepová frekvence měla konvergovat k přesnějším výsledkům. Měření tepové frekvence je restartováno pár sekund po odejmutí prstu ze senzoru.



Obrázek 4: Signál ze senzoru srdečního tepu

## 4 Závěr

V projektu se podařilo implementovat všechny body zadání. Řešení je v zásadě funkční, pouze je potřeba

dát si pozor na to, jak je vzdálená dioda od senzoru u modulu pro měření tepu a taky je nutné s prstem na senzoru co nejméně pohybovat v průběhu měření, v opačném případě naměřené hodnoty nemusí odpovídat skutečnosti. Při testování implementace jsem naměřené výsledky porovnával s naměřenými výsledky ze sportovních chytrých hodinek Xiaomi Amazfit Pace. Při testování za ideálních podmínek se naměřené hodnoty lišily jen nepatrně.

## Literatura

- [1] FitKit 3 (Minerva). [online], [vid. 2018-12-30]. Dostupné z: <https://minerva.php5.cz>
- [2] Kinetis SDK 2.0 API Reference Manual. [online], [vid. 2018-12-30]. Dostupné z: <http://mcuxpresso.nxp.com/apidoc/index.html>
- [3] High-pass filter. [online], listopad 2018, [vid. 2018-12-30]. Dostupné z: [https://en.wikipedia.org/wiki/High-pass\\_filter](https://en.wikipedia.org/wiki/High-pass_filter)
- [4] Low-pass filter. [online], prosinec 2018, [vid. 2018-12-30]. Dostupné z: [https://en.wikipedia.org/wiki/Low-pass\\_filter](https://en.wikipedia.org/wiki/Low-pass_filter)
- [5] K60 Sub-Family Reference Manual. [online], rev. 2. červen 2012, [vid. 2018-12-30]. Dostupné z: [http://cache.freescale.com/files/32bit/doc/ref\\_manual/K60P144M100SF2V2RM.pdf](http://cache.freescale.com/files/32bit/doc/ref_manual/K60P144M100SF2V2RM.pdf)
- [6] Šimek, V.: FitKit 3 (Minerva) – schéma zapojení. [online], září 2012, [vid. 2018-12-30]. Dostupné z: <https://minerva.php5.cz/doc/schemata/Minervafull.pdf>