

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Mikroprocesorové a vestavěné systémy  
Hra Life (celulární automat) na maticovém  
displeji

10. prosince 2016

David Kozák (xkozak15)

# **Obsah**

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Popis ovládání a implementace</b>	<b>2</b>
2.1	Popis ovládání . . . . .	2
2.2	Schéma zapojení . . . . .	3
2.3	Popis způsobu řešení . . . . .	4
<b>3</b>	<b>Závěrečné shrnutí</b>	<b>5</b>

# 1 Úvod

Tato dokumentace popisuje projekt do předmětu IMP, který v roce 2016 vytvořil student třetího ročníku FIT VUT David Kozák. V rámci projektu byl navržen a naimplementován projekt pro FITKIT[1] pracující s maticovým displayem modelujícím základní pravidla hry Life[2]. Hlavním úkolem v tomto projektu vytvořit řídící program pro mikrokontrolér msp430[3], který je součástí fitkitu.

## 2 Popis ovládání a implementace

Tato sekce tvoří jádro celé dokumentace, čtenář zde nalezne jak popis ovládání výsledné aplikace, tak i důležité technické detaily.

### 2.1 Popis ovládání

Celý projekt se ovládá za pomoci tlačítek, která jsou součástí fitkitu. Pro účely této sekce stačí následující znázornění.

1	2	3	A
4	5	6	B
7	8	9	C
*	0	#	D

Obrázek 1: Klávesnice na fitkitu

Jejich skutečnou podobu lze vidět na obrázku[4] v následující sekci popisující schéma zapojení zapojení. Klávesy označené červenou barvou jsou využity pro ovládání aplikace, klávesy zobrazené černou barvou se momentálně nevyužívají.

Klávesy 1,2,3 slouží pro zobrazení třech počátečních konfigurací, které jsou nastaveny přímo v kódu. Při stisknutí jednoho z těchto tlačítek se současné zobrazování zastaví a dojde k načtení příslušného výchozího stavu.

Klávesy 0 a # slouží pro ovládání běhu aplikace. Klávesa 0 funguje jako *pause/play*. Klávesa # slouží v režimu pause jako tlačítko pro krokování, které posune hru o jeden krok vpřed.

Klávsy 8 a 9 slouží pro přepínání mezi režimem 8-mi okolí a 9-ti okolí. Rozdíl mezi těmito módy spočívá v tom, že v 9-ti okolí je do okolí buňky počítána i buňka samotná.

Většina výše zmíněných kláves též jako vedlejší jev vypíše nějaký text na LCD display kitu. Pro smazání tohoto textu slouží klávesa C, která vymaže obsah displaye.

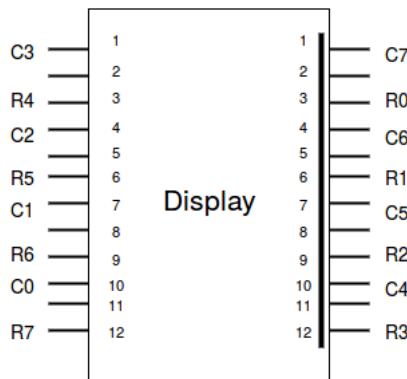
## 2.2 Schéma zapojení

Maticový display je připojen přímo na porty mikrokontroléru. Ten jej přes tyto porty ovládá. Zapojení můžete vidět v následující tabulce.

Řádek	Port	Sloupec	Port
0	P6M0	0	P4M0
1	P6M1	1	P4M1
2	P6M2	2	P4M2
3	P6M3	3	P4M3
4	P6M4	4	P4M4
5	P6M5	5	P4M5
6	P6M6	6	P1M4
7	P6M7	7	P2M5

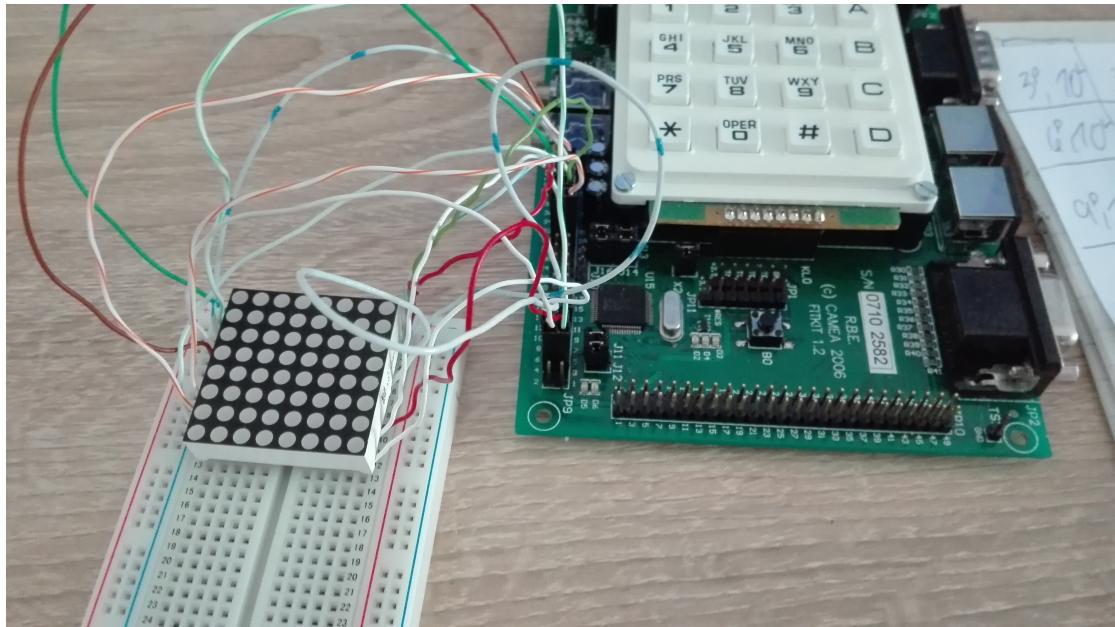
Obrázek 2: Připojení portů na display

Jelikož displej nabízí možnost svítit dvěma různými barvami a není z dokumentace přesně definován přesný význam všech 24 pinů, bylo potřeba experimentálně ověřit, jak displej funguje, a přiřadit jednotlivým pinům jejich význam. Z dvou možných barev zvolil autor červenou pro její větší výraznost. Výsledné znalosti o displeji můžete vidět na následujícím obrázku, *C* značí sloupce, *R* značí řádky.



Obrázek 3: Popis využitých portů displaye, černá úsečka značí stěnu, na které je napsán typ displeje

Pro větší názornost jsou též přiloženy dva obrázky samotného zapojení.



Obrázek 4: Příklad skutečného zapojení

### 2.3 Popis způsobu řešení

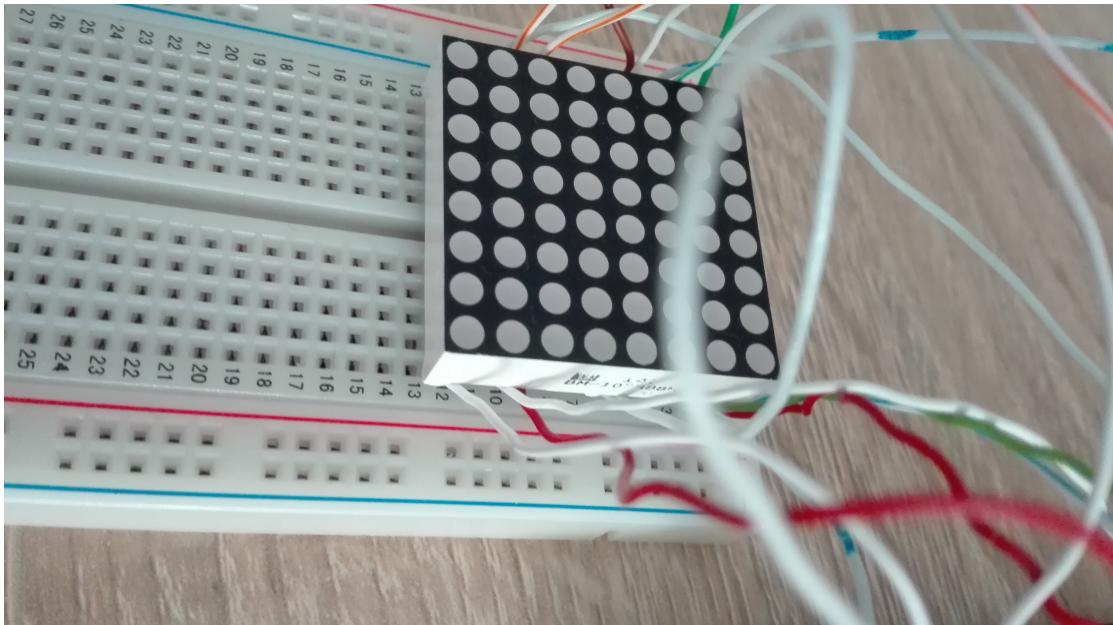
Stav programu je uchováván v globálních proměnných. Nejdůležitější proměnné jsou pole  $8 \times 8$  *cells* reprezentující současný stav a také proměnná booleovského typu *isRunning* určující, zda se simulace má sama chýbat kupředu či nikoliv. Hlavní funkce main nainiciuluje mcu i fpga a povolí přerušení od časovače. Následně také vybere defaultní výchozí konfiguraci pro hru a přejde do nekonečné smyčky, ve které obsluhuje terminál, klávesnici a také ovládá displej.

Řízení běho programu je prováděno s využitím přerušení. Využívá se přerušení od časovače, které je generováno s frekvencí XYZ Hz a které v případě, že proměnná *isRunning* má hodnotu *true*, vypočítá novou generaci a výsledek uloží do *cells*.

Pokud byla stisknuta klávesa, která má v aplikaci definovaný význam, je provedena rutina ošetřující stisk dané klávesy. Stisknutí klávesy *0* pouze invertuje hodnotu proměnné *isRunning*, čímž zastaví či spustí automatické posunování vpřed. Stiskutí klávesy *#* v režimu *pause* provede posunutí o jeden krok vpřed. Klávesy *1,2,3* bez ohledu na situaci nastaví *isRunning* na false a poté načtou příslušnou počáteční konfiguraci do *cells*.

Klávesy *8* a *9* umožňují přepínat mezi 8-mi okolím a 9-ti okolím .

Samotné vypočítání nové generace probíhá dvouprůchodově. V prvním průchodu dojde k výpočtu nových hodnot do pomocné tabulky, v druhém průchodu se tyto hodnoty přepíší do hlavní tabulky.



Obrázek 5: Příklad skutečného zapojení - detail na displej

### 3 Závěrečné shrnutí

V tomto projektu byla implementována hra Life(celuární automat) s využitím fitkitu a externího maticového displeje. Projekt obsahuje tři výchozí kofigurace, běh je možno pozastavit a krokovat. Ze zadání byly splněny všechny body. Jako možné rozšíření se jeví například možnost ovládat aplikaci z terminálu, konkrétně autor navrhuje možnost pomocí speciálního příkazu zvolit počáteční konfiguraci. Vzhledem k časovému presu ovšem toto rozšíření nebylo implementováno. Demonstraci projektu můžete vidět na videích [4] a [5].

## Reference

- [1] Webové stránky projektu FITKIT [cit. 2016-12-09][Online]  
<https://http://merlin.fit.vutbr.cz/FITkit/>
- [2] Popis hry Game Of Life [cit. 2016-12-09] [Online]  
[https://en.wikipedia.org/wiki/Conway's\\_Game\\_of\\_Life](https://en.wikipedia.org/wiki/Conway's_Game_of_Life)
- [3] Uživatelský manuál mikrokontroléru MSP430 [cit. 2016-12-09] [Online]  
<http://www.ti.com/lit/ug/slau144j/slau144j.pdf>

- [4] Ukázka módu 9-ti okolí [cit. 2016-12-09] [Online]  
<https://www.youtube.com/watch?v=oLCLrJIVXTk>
- [5] Ukázka módu 8-ti okolí [cit. 2016-12-09] [Online]  
<https://www.youtube.com/watch?v=EAMCaPm-iR4>