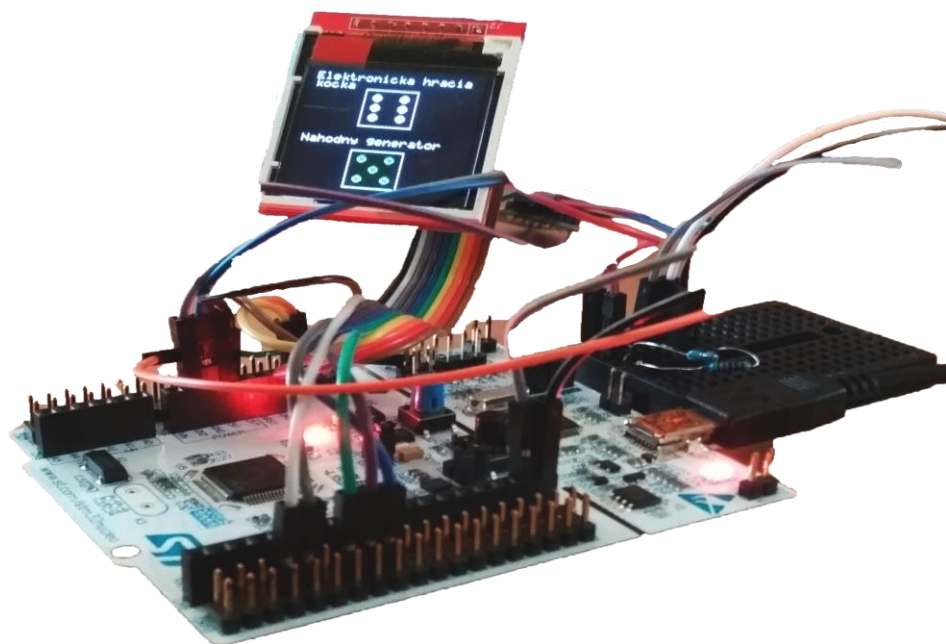


SLOVENSKÁ TECHNICKÁ UNIVERZITA

Fakulta Elektrotechniky a Informatiky



ELEKTRONICKÁ HRACIA KOCKA

(Zadanie z predmetu VRS)

Jerguš Frajt

Jozef Goga

Jaroslav Grejták

Zuzana Žemličková

2016/2017

Obsah

1	Špecifikácia zariadenia	3
1.1	Stručný popis.....	3
1.2	Technický popis	3
1.3	Rozšírený popis	3
1.4	Algoritmus generovania	4
1.4.1	Popis algoritmu	4
1.4.2	Používanie periférneho CRC pri čípoch STM32	5
1.4.3	Ukážka zdrojového kódu pre generovanie náhodných čísel	6
1.5	I ² C.....	7
1.6	Zoznam použitých komponentov	7
1.7	Komunikácia v zariadení.....	8
2	Používateľská príručka	9
2.1	Súčasti zariadenia	9
2.2	Popis zariadenia.....	9
2.3	Používanie zariadenia.....	10
3	Programátorská príručka	11

1 Špecifikácia zariadenia

1.1 Stručný popis

Elektronická hracia kocka je zariadenie, ktoré simuluje a zároveň nahrádza funkciu bežnej hracej kocky. Pracuje v dvoch základných režimoch, a to v režime keď náhodne generuje hodnoty bežnej hracej kocky, t.j. prirodzené čísla od 1 po 6, konkrétne množina {1, 2, 3, 4, 5, 6} pomocou algoritmu pre elektronické generovanie náhodných hodnôt zariadením STM Nucleo.

Druhý režim funguje na rovnakých princípoch ako bežná hracia kocka, a to na základe externého zariadenia v podobe akcelerometra (náhrada hracej kocky), ktorý sníma zrýchlenie v určitom smere a podľa toho vyhodnocuje, na ktorej hodnote práve kocka zastala, t.j. jedná sa o klasický prístup, avšak zobrazenie výslednej hodnoty a detegovanie vrchnej steny kocky je digitalizované.

1.2 Technický popis

Napájanie	5V / 2A
Displej	1,44"
Rozlíšenie	128 x 128

1.3 Rozšírený popis

Elektronická hracia kocka je zariadenie, ktoré nahrádza a rozširuje použiteľnosť bežnej hracej kocky. Kompletne riadiace centrum pre všetky akcie spojené s funkciou sa nachádza na tzv. embedded systéme na čipe od ST MicroElectronics – STM32L152RE. Zariadenie má možnosť fungovať v 2 režimoch.

Jeden režim generuje náhodné hodnoty od 1 po 6, množina {1, 2, 3, 4, 5, 6} na základe vlastného softvérového riešenia využívajúceho náhodne generované hodnoty nezapojených vstupov z AD prevodníka, času behu aplikácie a ďalších subriešení, ktoré sa využívajú v algoritme sofistikovaného generovania náhodných hodnôt, pričom spúšťanie aj zastavenie náhodného generovania je realizované pomocou tlačidla na riadiacej jednotke

STM Nucleo. Viac podrobností o algoritme generovania náhodných hodnôt sa nachádza v časti *Algoritmus Generovania Náhodných Hodnôt*.

Druhý režim využíva algoritmus, ktorý pristupuje k externému zariadeniu – akcelerometru. Akcelerometer sníma hodnoty zrýchlenia v jednotlivých smeroch a podľa týchto hodnôt algoritmus vyhodnocuje, na ktorú „stenu“ hracej kocky akcelerometer smeruje a akú hodnotu má zariadenie ukázať. Z teoretického hľadiska ide o detegovanie smeru, v ktorom pôsobí značná poprípade celá zložka gravitačnej sily a túto je potrebné definovať ako nami definovanú stranu kocky a zobrazit' jej hodnotu na displeji.

Komunikácia medzi akcelerometrom a riadiacou jednotkou je typu I2C, kde využívame DMA prístup.

Na displeji je možné prepínanie jednotlivých režimov a zároveň slúži ako jedinečné zobrazenie celého procesu a výsledkov procesu.

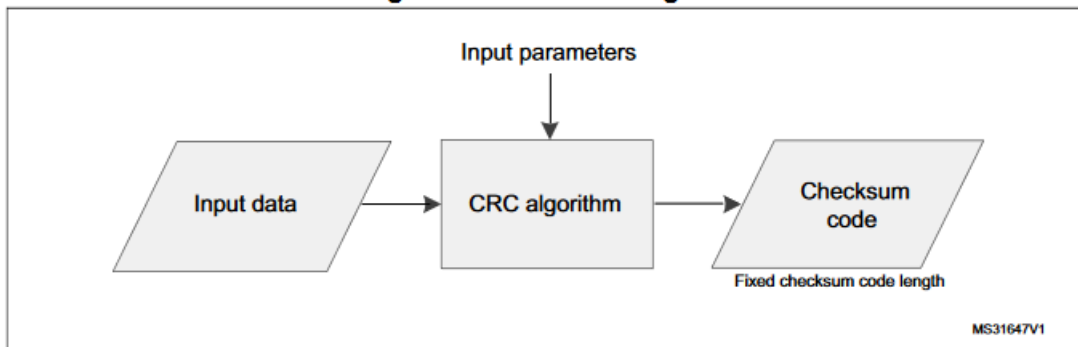
1.4 Algoritmus generovania

1.4.1 Popis algoritmu

Zariadenie STM nemá zabudovaný hardvérový RNG (random generator). Je však možné vytvoriť vlastný generátor. Obyčajný spôsob je meranie šumu nezávislých hodín. Nezávislé znamenajú, že dvojito hodín je istených 2 kryštálmi alebo RC oscilátormi a nepochádzajú obe z jedného. Je možné použiť systémový časovač a jeden z kHz RC oscilátorov. V našom algoritmickom riešení však používame iný spôsob na generovanie náhodných hodnôt.

Generovanie náhodných čísel bude zabezpečené pomocou čítania vnútornej teploty procesora pomocou ADC prevodníka. Tento signál je nepravidelný a ak budeme čítať túto hodnotu s najvyššou obnovovacou frekvenciou daného kanála, tak získame veľmi zašumené číslo prevodu. Následne toto číslo pošleme do hardvérového CRC (cyclic redundancy check) generátora, ktorý vygeneruje 32-bitové náhodné číslo. Pri niekoľkonásobnom opakovaní spustenia prevodu a vygenerovania CRC kódu získame pomerne kvalitný generátor náhodných čísel.

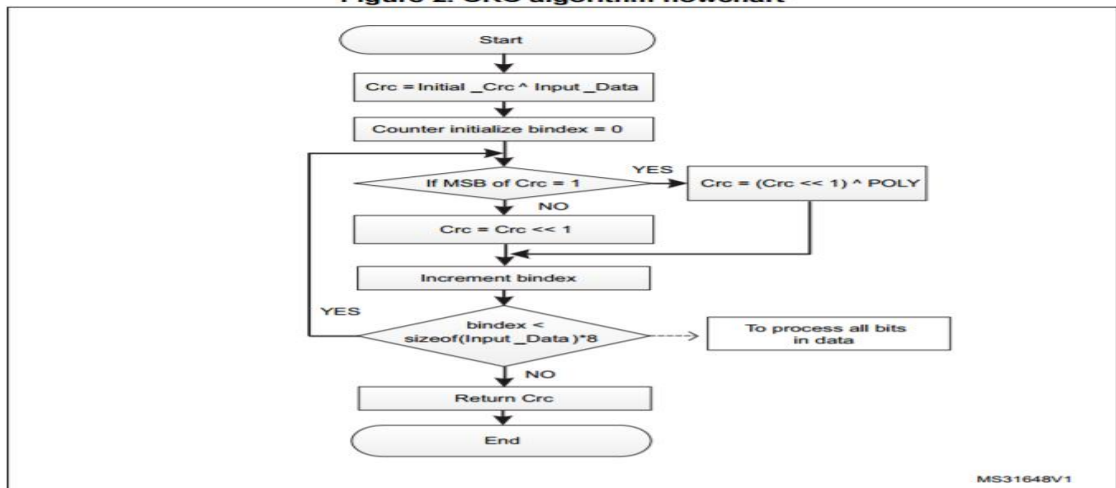
Figure 1. CRC block diagram



1.4.2 Používanie periférneho CRC pri čípoch STM32

Cyklická redundantná kontrola (CRC) je technika používaná pre detekciu chýb v digitálnych dátach ale bez vykonania opráv, ak sú zistené chyby. Používa sa pri prenose dát alebo pri ukladaní dát kontroly integrity. CRC je výkonný a ľahko implementovateľná technika na získanie spoľahlivých dát. Diagnostické pokrytie tejto techniky spĺňa požiadavky na základné bezpečnostné normy. To je dôvod, prečo je túto implementačnú funkciu CRC používajú v obsahu Flash integrity self-testu kontroly u ST firmvéru, ktorý je certifikovaný v súlade s IEC 60335-1 a IEC 607030-1 štandardy (označované ako požiadavky pre "Triedu B"). Pre viac informácií vid' manuál AN4187.

Figure 2. CRC algorithm flowchart



1.4.3 Ukážka zdrojového kódu pre generovanie náhodných čísel

```
uint32_t getTrueRandomNumber(void) {
    ADC_InitTypeDef ADC_InitStructure;
    //enable ADC1 clock
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_ADC1, ENABLE);
    // Initialize ADC 14MHz RC
    RCC_ADCCLKConfig(RCC_ADCCLK_HSI14);
    RCC_HSI14Cmd(ENABLE);
    while (!RCC_GetFlagStatus(RCC_FLAG_HSI14RDY));
    ADC_DeInit(ADC1);
    ADC_InitStructure.ADC_ContinuousConvMode = DISABLE;
    ADC_InitStructure.ADC_DataAlign = ADC_DataAlign_Right;
    ADC_InitStructure.ADC_Resolution = ADC_Resolution_12b;
    ADC_InitStructure.ADC_ScanDirection = ADC_ScanDirection_Backward;
    ADC_InitStructure.ADC_ExternalTrigConvEdge = ADC_ExternalTrigConvEdge_None;
    ADC_InitStructure.ADC_ExternalTrigConv = ADC_ExternalTrigConv_T1_TRGO; //default
    ADC_Init(ADC1, &ADC_InitStructure);
    //enable internal channel
    ADC_TempSensorCmd(ENABLE);
    // Enable ADC peripheral
    ADC_Cmd(ADC1, ENABLE);
    while (ADC_GetFlagStatus(ADC1, ADC_FLAG_ADEN) == RESET);
    ADC1->CHSELR = 0;
    //no channel selected
    //Convert the ADC1 temperature sensor, user shortest sample time to generate most noise
    ADC_ChannelConfig(ADC1, ADC_Channel_TempSensor, ADC_SampleTime_1_5Cycles);
    // Enable CRC clock
    RCC_AHBPeriphClockCmd(RCC_AHBPeriph_CRC, ENABLE);

    uint8_t i;
    for (i = 0; i < 8; i++) {
        //Start ADC1 Software Conversion
        ADC_StartOfConversion(ADC1);
        //wait for conversion complete
        while (!ADC_GetFlagStatus(ADC1, ADC_FLAG_EOC)) {
        }
        CRC_CalcCRC(ADC_GetConversionValue(ADC1));
        //clear EOC flag
        ADC_ClearFlag(ADC1, ADC_FLAG_EOC);
    }
    //disable ADC1 to save power
    ADC_Cmd(ADC1, DISABLE);
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_ADC1, DISABLE);
    return CRC_CalcCRC(0xBADA55E5);
}
```

1.5 I²C

I²C bus je skratka, ktorá vznikla z IIC bus, Internal-Integrated-Circuit Bus. Ako už názov napovedá, jedná sa o internú dátovú zbernicu slúžiacu pre komunikáciu a prenos dát medzi jednotlivými integrovanými obvodmi väčšinou v rámci jedného zariadenia.

Hlavnou výhodou je, že obojsmerný prenos prebieha iba po dvoch vodičoch - "dátá SDA (serial data)" a "hodiny SCL (serial clock)". To predovšetkým u mikrokontrolérov výrazne optimalizuje nároky na počet vstupno-výstupných pín a celkovo zjednodušuje výsledné zapojenie.

Na jednej zbernici môže byť pripojených viac integrovaných obvodov. V základnej verzii sú obvody adresované 7bitovo a v rozšírenej verzii 10bitovo. To umožňuje pripojenie 128 respektíve 1024 čipov s rôznou adresou na jednu spoločnú zbernicu. V praxi sú tieto čísla však podstatne nižšie, pretože adresa čipu väčšinou nie je možné určiť plnými 7 (10) bity ale napríklad iba tromi. Niekedy ju nie je možné určiť vôbec a je daná na pevno pre daný typ čipu - takých čipov teda na jednej zbernici nemôže byť viac ako jeden.

Prenosová rýchlosť zbernice je pre väčšinu aplikácií dostatočná aj v základnej verzii, kde je frekvencia hodín 100kHz. Vo vylepšených verziách to môže byť 400kHz alebo 1MHz, ale nie všetky integrované obvody túto verziu podporujú. Rýchlosť prenosu potom musí byť prispôbená pochopiteľne "najpomalšiemu" čipu na zbernici. Oba vodiče musia byť implicitne v logickej jednotke a to je zaistené pull-up rezistormi. Ich odpory majú hodnotu v rádoch kiloohmov. Čím je vyššia komunikačná frekvencia, tým musí byť nižšia hodnoty týchto odporov.

1.6 Zoznam použitých komponentov

Č.	Komponent	Funkcia
1.	STM Nucleo L152RE MB1136 C-03 2162200779	Riadiaca jednotka
2.	MPU 6050 – GY-521	Akcelerometer, gyroskop
3.	Display 1,44" SPI 128x128 V1.1	Zobrazovacia jednotka
4.	Kábel mini USB 5 pin	Pripojenie nabíjania
5.	Prepojovacie káble	Komunikácia medzi zariadeniami

1.7 Komunikácia v zariadení

Č.	Komponent č. 1	Komponent č. 2	Komunikácia
1.	Display 1,44" SPI 128x128 V1.1	STM Nucleo	I ² C
2.	MPU 6050 – GY-521	STM Nucleo	I ² C

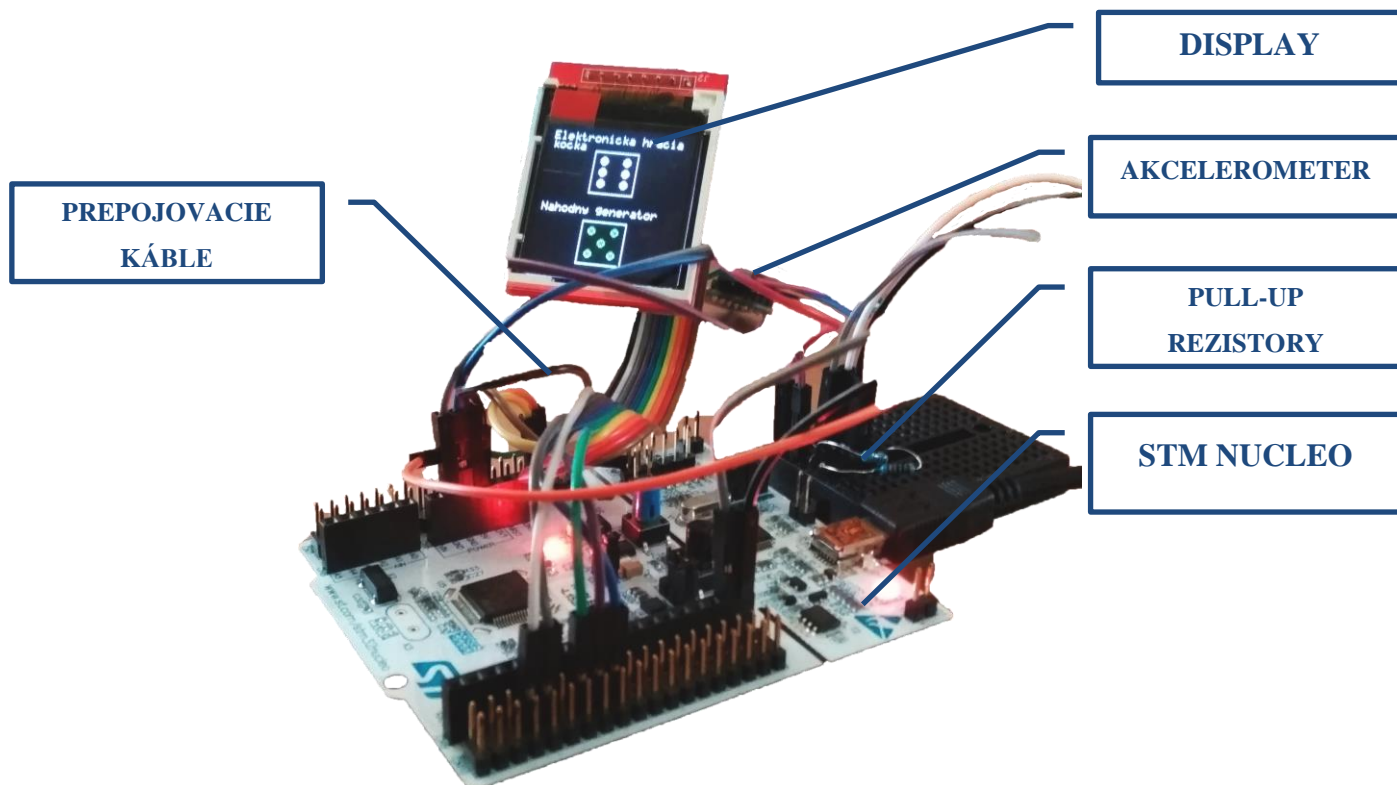
I²C (Inter-Integrated Circuit) je multi-master, multi-slave, single-ended, sériová zbernica počítača. Zvyčajne sa používa na pripojenia s nižšou rýchlosťou pre periférne obvody procesorov a mikroradičov v krátkej vzdialenosti a intra-board komunikáciu.

2 Používateľská príručka

2.1 Súčasti zariadenia

Č.	Komponent	Funkcia
1.	STM Nucleo L152RE MB1136 C-03 2162200779	Riadiaca jednotka
2.	MPU 6050 – GY-521	Akcelerometer, gyroskop
3.	Display 1,44" SPI 128x128 V1.1	Zobrazovacia jednotka
4.	Kábel mini USB 5 pin	Pripojenie nabíjania
5.	Prepojovacie káble	Komunikácia medzi zariadeniami

2.2 Popis zariadenia



2.3 Používanie zariadenia

1. Zariadenie aktivujeme zapojením USB kábla do zdroja napájania
2. Náhodné generovanie hodnôt elektronickej hracej kocky sa realizuje stlačením modrého tlačidla na základnej doske alebo prudkým pohybom akcelerometra
3. Virtuálny hod kocky docielime pootočením akcelerometra do ľubovoľnej polohy
4. Výsledky hodu sú zobrazované na displeji – v hornej časti sa nachádza informácia hodu kocky prostredníctvom akcelerometra zatiaľ čo v spodnej časti je vidieť výsledok náhodného generovania hodu
5. Zariadenie je možné resetovať stlačením čierneho tlačidla umiestneného na základnej doske

3 Programátorská príručka

Programátorská príručka sa nachádza v priloženom dokumente: PRILOHA_A.pdf