

STREDNÁ PRIEMYSELNÁ ŠKOLA STROJNÍCKA A ELEKTROTECHNICKÁ  
ULICA FRAŇA KRÁĽA 20, 949 01 NITRA

## ZOBRAZENIE GALTONOVEJ ROVNICE

Stredoškolská odborná činnosť

Číslo odboru: 2675 M



Riešitelia: Roman Mankovecký

Tomáš Nyiri

Nitra

2018

Ročník štúdia: 2017/2018

STREDNÁ PRIEMYSELNÁ ŠKOLA STROJNÍCKA A ELEKTROTECHNICKÁ  
ULICA FRAŇA KRÁĽA 20, 949 01 NITRA

## ZOBRAZENIE GALTONOVEJ ROVNICE

Stredoškolská odborná činnosť

Číslo odboru: 2675 M

Riešitelia: Roman Mankovecký

Tomáš Nyiri

Nitra

2018

Ročník štúdia: 2017/2018

Školitelia: Mgr. Eva Mokránová, PhD.

Mgr. Dušan Baláž

### Čestné vyhlásenie

Vyhlasujeme, že prácu stredoškolskej odbornej činnosti na tému „**Zobrazenie Galtonovej rovnice**“ sme vypracovali samostatne, s použitím uvedených literárnych zdrojov. Som si vedomý zákonných dôsledkov, ak v nej uvedené údaje nie sú pravdivé.

.....

**Roman Mankovecký**

.....

**Tomáš Nyiri**

## Pod'akovanie

Touto cestou by som sa chcel pod'akovať školiteľovi práce Mgr. Dušanovi Balážovi a konzultantovi práce Mgr. Eve Mokráňovej za pomoc, odborné vedenie, cenné rady a pripomienky pri vypracovaní a realizácii mojej práce: **Zobrazenie Galtonovej rovnice**

## OBSAH

<b>ZOZNAM TABULIEK, GRAFOV A ILUSTRÁCIÍ.....</b>	<b>7</b>
<b>ÚVOD.....</b>	<b>8</b>
<b>1 PROBLEMATIKA A PREHĽAD LITERATÚRY.....</b>	<b>9</b>
1.1 MIKROPROCESOR ATMEGA8.....	9
1.2 VLASTNOSTI MIKROPROCESORA ATMEGA8.....	10
1.3 BLOKOVÁ SCHÉMA PROCESORA ATMEGA8.....	11
1.4 VYSVETLENIE PINOV NA MIKROPROCESORE ATMEGA8.....	12
1.5 POSUVNÝ REGISTER.....	13
1.6 TRANZISTOR.....	14
1.6.1 BC170 NPN.....	14
1.6.2 BC817.....	15
1.7 LED DIÓDA.....	16
1.8 JAZYK C A JEHO VLASTNOSTI.....	17
1.9 EAGLE(SCHÉMATICKÝ PROGRAM).....	18
<b>2 CIELE PRÁCE SOČ.....</b>	<b>19</b>
2.1 HLAVNÝ CIEĽ.....	19
1.1 ČIASTKOVÉ CIELE.....	19
<b>3 PRAKTICKÁ ČASŤ.....</b>	<b>20</b>
3.1 POPIS ZARIADENIA.....	20
3.2 FUNKCIA ZARIADENIA.....	20
3.3 OPIS PRÁCE.....	21
3.4 POPIS POUŽITÝCH SÚČIASTOK.....	21
3.5 SCHÉMA.....	22
3.6 NÁVRH DOSKY.....	24
3.7 PROGRAM A JEHO OPIS.....	25

<b>4</b>	<b>VÝSLEDKY A DISKUSIA .....</b>	<b>29</b>
	<b>ZÁVERY PRÁCE .....</b>	<b>30</b>
	<b>ZHRNUTIE.....</b>	<b>31</b>
	<b>RESUME.....</b>	<b>32</b>
	<b>ZOZNAM BIBLIOGRAFICKÝCH ODKAZOV .....</b>	<b>33</b>

## ZOZNAM TABULIEK, GRAFOV A ILUSTRÁCIÍ

Obrázok 1	ATMEGA8-16PU.....	9
Obrázok 2	ATMEGA8-16PU.....	10
Obrázok 3	BLOKOVÁ SCHÉMA.....	11
Obrázok 4	ATMEGA8 VYSVETLENIE PINOV .....	12
Obrázok 5	74HC595.....	13
Obrázok 6	BV170.....	15
Obrázok 7	BC817 UMIESTNENIE PINOV .....	16
Obrázok 8	LED DIÓDA .....	17
Obrázok 9	LOGO PROGRAMOVACIEHO JAZYKA C .....	18
Obrázok 10	LOGO EAGLU .....	19
Obrázok 11	SCHÉMA .....	23
Obrázok 12	SCHÉMA .....	24
Obrázok 13	SCHÉMA .....	24
Obrázok 14	DPS LED MATICE.....	25
Obrázok 15	RIADIACA DPS .....	25

## ÚVOD

Táto práca obsahuje dokumentáciu k postupu zhotovenia práce s názvom „*Zobrazenie Galtonovej rovnice*“ pomocou led diód. V prvom rade sme podrobne preukázali potrebnú teóriu k danej práci a vysvetlili jej vyhotovenie ako teoreticky tak aj prakticky. Sú v nej zahrnuté všetky informácie ohľadom riešenia problémov, návrhu schémy, návrhu dosky a konečného vyhotovenia.



## 1.PROBLEMATIKA A PREHĽAD LITERATÚRY

### 1.1 Mikroprocesor Atmega8

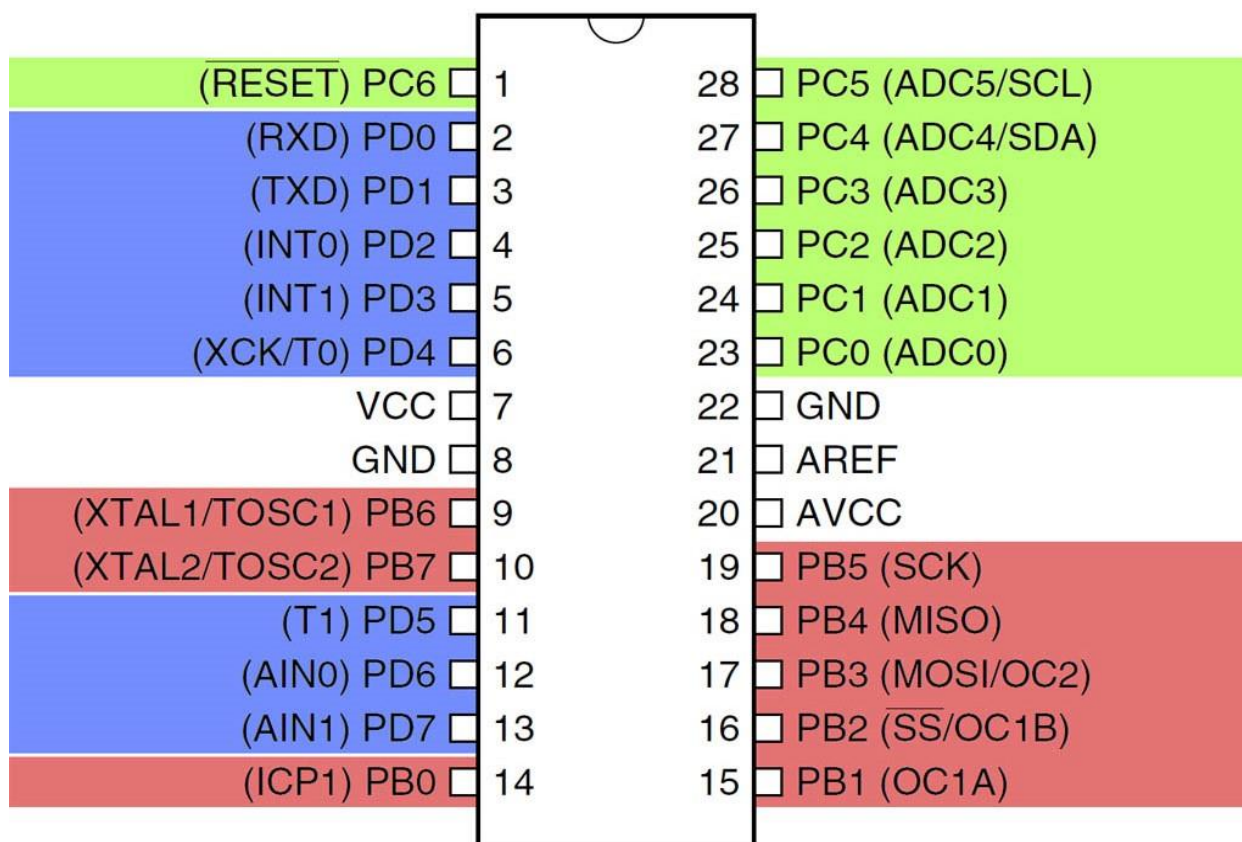
Atmega8 je mikroprocesor ktorý sa skladá z 28 pinov. Má 23 programovateľných vstupov / výstupov → I / O (porty), piny, ktoré môžu byť použité pre prepojenie s vonkajším svetom. Je možné ich nastaviť ako vstup alebo výstup nastavením určitej hodnoty v registry pomocou programovania. Mikroprocesor Atmega8 vyrába firma Atmel.

Všetky(23) vstupy/výstupy → I / O (porty) Atmegy8 sú usporiadané do 3 skupín :

- Port B (PB0-PB7)
- Port C (PC0-PC7)
- Port D (PD0-PD7)

Farebné vyznačenie portov s pomenov s pomenovaním pinov:

### Atmega8-16PU



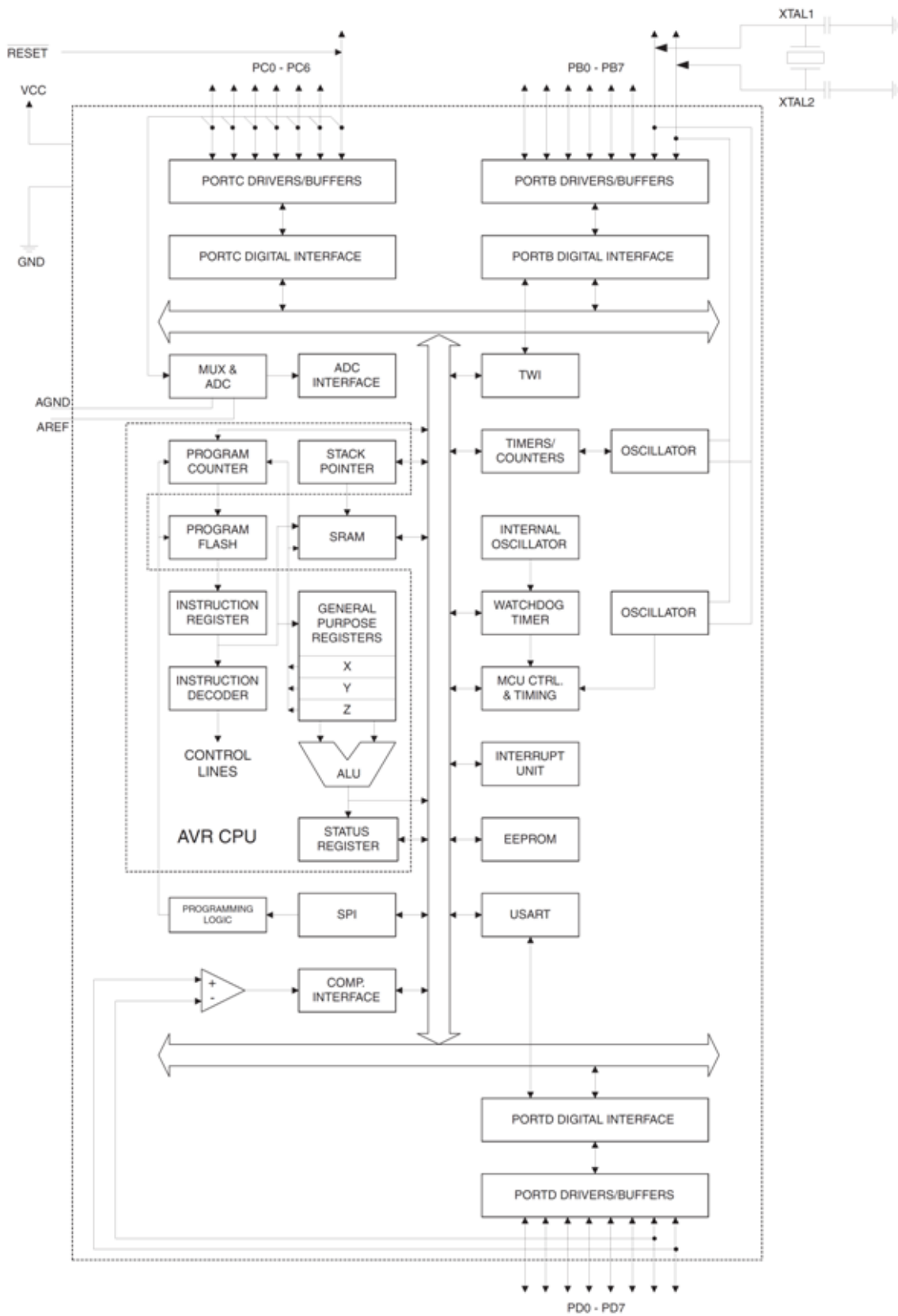
## 1.2 Vlastnosti mikroprocesora Atmega8

Tieto tzv. mikroradiče prinášajú jednoduchosť použitia, vysoký stupeň integrácie a nízku spotrebu energie. Atmel® AVR® 8-bit a 32-bitové mikrokontroléry (MCU) dopĺňajú sa Atmel® | SMART rad mikroprocesorov na báze ARM® procesora a MCU. Tieto zariadenia ponúkajú jedinečnú kombináciu výkonu, energetickej efektívnosti a flexibility návrhu. Sú založené na väčšine kódu úspornej architektúry priemyslu pre jazyk C, assembler a iné.

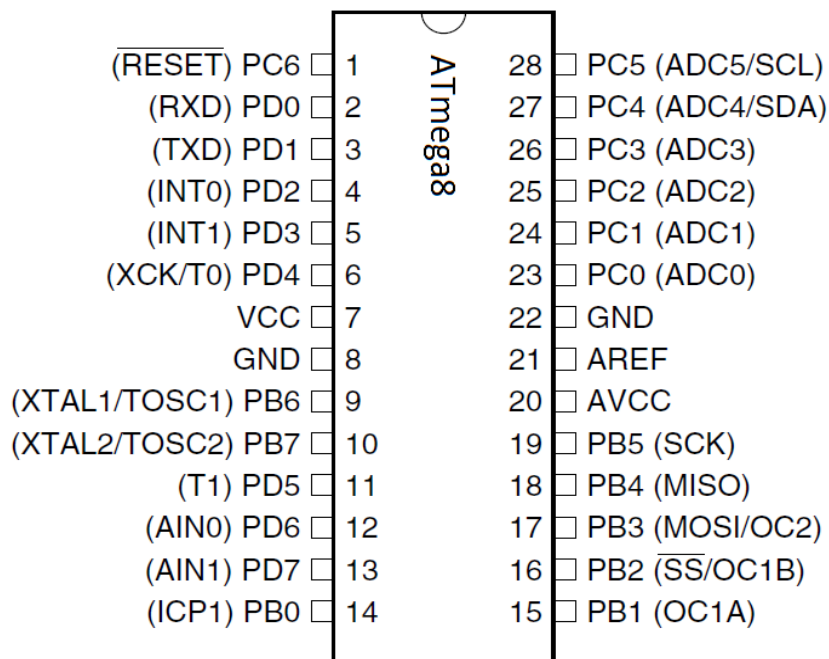
- 23 Programovateľné I / O porty
- 4,5 do prevádzky 5.5V
- Pracovná teplota -40...85°C
- Pracovná frekvencia 16 MHz
- 8K bytov v systéme, v programovateľnej pamäti
- 512 bajtov EEPROM, trvanie 100.000 cyklov zápisu / mazanie
- 8kB kapacita pamäte Flash
- 1K bajtov SRAM
- analógový komparátor
- watch dog
- 8 prevodníkov A/D 10 bit
- Dva 8-bitové časovače
- Jeden 16-bitový časovač / register s deličom
- Reálny čas registra s oddeleným oscilátorom
- Tri PWM kanále
- 6-kanálové ADC-štyri kanály 10-bitové presnosti a dva kanály 8-bitové presnosti
- Byte-orientovaný Dvojvodičový Sériové rozhranie
- Programovateľný Sériový USART
- Master / Slave SPI Serial Interface
- Power-on Reset a Programmable Brown-out detekcia
- Vonkajšie a vnútorné prerušenie Zdroje
- Päť režimov spánku: Idle, ADC Noise Reduction, Power-save, Power-down a Standby



### 1.3 Bloková schéma procesora Atmega8



## 1.4 Vysvetlenie pinov na mikroprocesore Atmega8



**PortB** (PB0 – PB7) vývodné piny → 9,10,14,15,16,17,18,19

**PortC** (PC0 – PC7) vývodné piny → 1,23,24,25,26,27,28

**PortD** (PD0 – PD7) vývodné piny → 2,3,4,5,6,11,12,13

**VCC** – Napájanie (5V) vývodný pin → 7

**GND** – Zem vývodné piny → 8,22

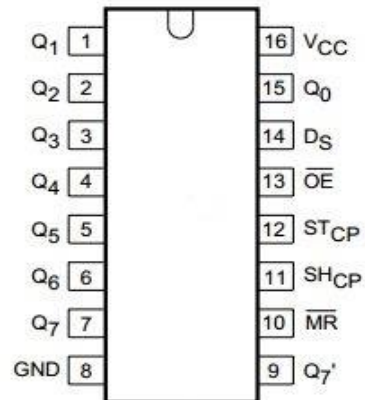
**AREF** - Nastavenie referenčného napätia vývodný pin → 21

**AVCC** – Napájacie napätie vývodný pin → 20

## 1.5 Posuvný register

Posuvný register je integrovaný obvod, ktorý umožňuje previesť informáciu zo sériovej na paralelnú. **74HC595** (použitý pri práci) umožňuje takto meniť 8 bitov. V praxi to znamená to, že namiesto ôsmich drôtov potrebujeme iba tri (plus dva na napájanie, ale tie by sa museli prirátavať aj k pôvodným ôsmim).

**74HC595** má 16 pinov:



Paralelné výstupy ( $Q_0$ - $Q_7$ ) piny  $\rightarrow$  1,2,3,4,5,6,7,15

GND (zem) pin  $\rightarrow$  8

VCC (napájanie 5V) pin  $\rightarrow$  16

Master reset pin  $\rightarrow$  10

Clock(hodinové impulzy) pin  $\rightarrow$  11

Storage register clock pin  $\rightarrow$  12

Output enable pin  $\rightarrow$  13

Výstup sériových bitov( $Q_7$ ) pin  $\rightarrow$  9

Vstup sériových bitov( $D_S$ ) pin  $\rightarrow$  14

$Q_7$  a  $D_S$  slúžia na prepojenie viacerých registrov. To znamená že prepojíme Výstup sériových bitov so vstupom na **d'alšom** posuvnom registri.

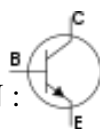
## 1.6 Tranzistor

Tranzistor alebo polovodičová trióda je polovodičová súčiastka, používaná ako zosilňovač, spínač, stabilizátor a modulátor elektrického napätia alebo prúdu. My ju budeme používať ako spínač.

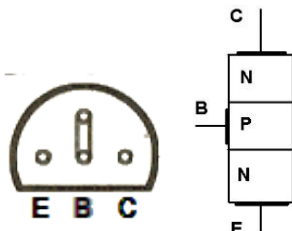
Hlavnou úlohou spínača je zopnúť v určitom čase a za určitý čas el. obvod. Ich hlavnou prednosťou je čas zopnutia a rozopnutia (čo najkratší - rádovo jednotky ms u spínacích tranzistorov až jednotky  $\mu$ s). Ďalšou vlastnosťou je spínací odpor má tiež byť čo najmenší v zopnutom stave a čo najväčší v rozopnutom.

### 1.6.1 BC170 NPN

Používame typ tranzistoru BC170 s polaritou NPN :



NPN tranzistor je druh tranzistora, ktorý obsahuje dva polovodiče typu N a jeden typu P. Ako z názvu vyplýva, polovodiče sú usporiadané tak, že dva polovodiče typu N obklopujú polovodič typu P. Takýto tranzistor potom funguje podobne ako dve diódy|diódy otočené smerom od seba (-|◀-▶|-)



#### Vlastnosti BC170 :

Collector-Base Voltage	→ max. 20[V]
Collector-Emitter Voltage	→ max. 20 [V]
Emitter-Base Voltage	→ max. 5 [V]
Collector Current	→ max. 100 [mA]

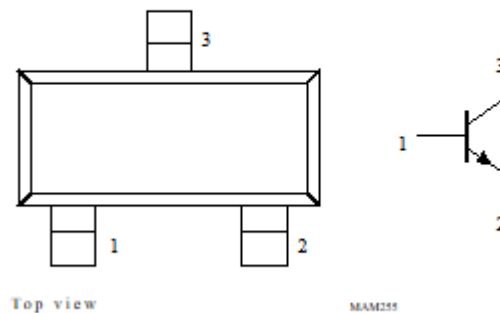


## 1.6.2 BC817

BC817 je tranzistor typu NPN použiteľný na malé napätie a prúd. Je to Tranzistor typu SMD.

Vlastnosti BC817:

Collector-Base Voltage	→ max. 50[V]
Collector-Emitter Voltage	→ max. 45 [V]
Emitter-Base Voltage	→ max. 5 [V]
Collector Current	→ max. 500 [mA]
Pracovná teplota	→ -65 ... +150°C



Názvy pinov:

1. Báza
2. Emitor
3. Kolektor

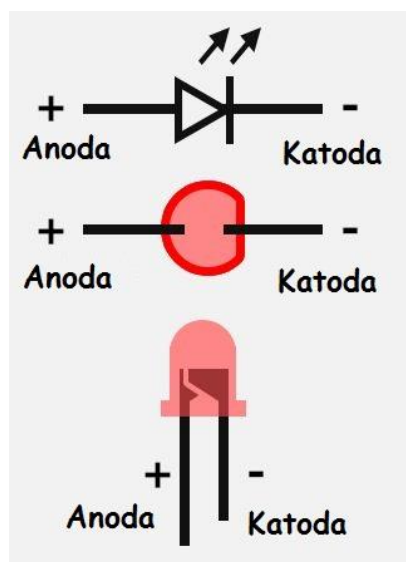
## 1.7 Led dióda

Luminiscenčná dióda, (ľudovo ledka) je polovodičová elektronická súčiastka, ktorá vyžaruje úzkospektrálne svetlo, keď ňou prechádza elektrický prúd v priepustnom smere. Svetiaci jav vzniká následkom žiarivej rekombinácie elektrónovo-dierového páru a je formou elektroluminiscencie. Farba vyžarovaného svetla závisí od štruktúry PN priechodu aj od použitého materiálu.

Dióda má 2 vývody:

- 1) Anoda – pozitívny typ polovodiča (je časťou typu P-priepustný)
- 2) Katoda – negatívny typ polovodiča (je časťou typu N-nepriepustný)

PN priechod led diódy zobrazený na obrázkoch:





## 1.8 Jazyk C a jeho vlastnosti

Jazyk C je pomerne minimalistický tzv. imperatívny programovací jazyk na úrovni blízkej hardvéru a je podobnejší strojovo orientovaným jazykom ako väčšina jazykov vyššej úrovne. O jazyku C sa niekedy hovorí ako o "prenosnom strojovom jazyku" (*portable assembler*). Na rozdiel od strojovo orientovaných jazykov je kód v jazyku C možné skompilovať (preložiť do strojového kódu) pre takmer každý počítač.

Táto vlastnosť spolu s tým, že samotný prekladač jazyka C i väčšina jadra UNIXu boli prepísané do C, uľahčilo ich rozšírenie na rôzne počítače. Z týchto dôvodov nazvali jazyk C jazykom strednej úrovne. Hlavným dôvodom vytvorenia jazyka C bolo uľahčiť písanie programov s menším množstvom chýb procedurálnym spôsobom, ale bez dodatočnej záťaže na kompilátor, ktorého tvorbu komplikujú komplexné vlastnosti jazyka. C je z najpoužívanejších programovacích jazykov. C sa cení vďaka jeho efektívnosti a je najpopulárnejším jazykom na písanie systémového softvéru, hoci sa používa aj na tvorbu aplikačného softvéru. Tiež sa bežne používa pri výuke programovania, hoci nie je určený pre úplných začiatočníkov.

Z tohto dôvodu má C nasledujúce dôležité vlastnosti :

- Jednoduché jadro jazyka s dôležitou funkcionalitou ako matematické funkcie alebo obsluha súborov presunutou do množiny knižničných rutín namiesto toho, aby bola súčasťou jazyka
- Zameranie na paradigmu procedurálneho programovania s prvkami pre programovanie štruktúrovaným štýlom
- Jednoduchý systém typov, ktorý predchádza ich chybnému použitiu
- Použitie preprocesorového jazyka, C preprocesor, pre úlohy ako definícia makier a viacnásobné vkladanie (*include*) zdrojového súboru
- Nízkoúrovňový prístup k pamäti počítača pomocou ukazovateľov (smerníkov)
- Minimalistická množina kľúčových slov
- Parametre sa odovzdávajú hodnotou, odovzdávanie parametrov odkazom sa realizuje pomocou mechanizmu ukazovateľov.
- Ukazovatele na funkcie, ktoré umožňujú zárodočnú formu uzáveru a polymorfizmu
- Zmena lexikálneho významu premennej podľa miesta výskytu
- Záznamy, alebo užívateľsky definované zložené údajové typy (*struct*, *union*), ktoré umožňujú kombináciu súvisiacich údajov a manipuláciu s nimi ako s celkom
- Čo najmenej iného



## 1.9 EAGLE ( schematický program )

**EAGLE** je program pre automatizáciu elektronickej aplikácie so schematickými zachyteniami dosky s plošnými spojmi a jej rozloženie. Uľahčenie návrhy dosiek s plošnými spojmi ( DPS ). Program je od firmy CadSoft. Táto firma sídli v Nemecku.

**EAGLE** → Easily Applicable Graphical Layout Editor ( Lahko použiteľné rozmiestnenie grafických prvkov ).

Program je rozdelený do troch častí :

- 1) Spoločné
- 2) Editor schém (sch) → schéma
- 3) Editor spojov (brd) → board

**Editor schém** ( sch ), slúži na projektovanie schémy. Diely môžu byť umiestnené na mnohých listoch a sú spojené dohromady cez porty.

**Editor dosiek plošných spojov** ( board ), je vlastne ucelený balík pre profesionálny návrh dosiek plošných spojov. Vlastný návrh plošného spoja je postupnosť viacerých krokov ako definovanie obrysov dosky, rozmiestnenie súčiastok, vedenie spojov, tvorba výkresovej dokumentácie, technologických dát...

EAGLE obsahuje autorouter ( automatické generovanie DPS ( dosky plošných spojov)).

V Eagly nájdeme aj Libraries(knižnicu súčiastok, pinov,...) ktoré si môžeme doupravovať popripradne vytvoriť nové súčiastky. Rôzne knižnice súčiastok sa dajú dodatočne stiahnuť.

Light verzia Eaglu je voľne dostupná aj na oficiálnych stránkach.



## 2. CIELE PRÁCE SOČ

### 2.1 Hlavný cieľ

Hlavný cieľ práce bolo zhotoviť pomôcku pre nasimulovanie modulu matematickej rovnice o pravdepodobnosti určitých kombinácií. Vytvoriť maticu led diod ktorú bude možné univerzálne naprogramovať.

### 2.2.Čiastkové ciele

1. Navrhnutie schémy
2. Navrhnutie riadiacej DPS a led matice
3. Zhotovenie riadiacej DPS a led matice
4. Prepojenie oboch DPS
5. Otestovať funkčnosť oboch DPS
6. Naprogramovať mikroprocesor
7. Uzavrieť DPS do obalu

### 3. PRAKTICKÁ ČASŤ

#### 3.1 Popis zariadenia:

Zariadenie obsahuje riadiacu jednotku, ktorá umožňuje rozsvetovať LED diódy a LED maticu na ktorej sa nachádzajú LED diódy. Riadiacia jednotka obsahuje mikroprocesor (Atmega8), štyri posuvné registre a dve tlačidlá pomocou ktorých sa dá zariadenie spúšťať a resetuje. Tiež sa na riadiacej jednotke nachádzajú tranzistory ktoré nám dopomáhajú rozsvetovať určité LEDky.

#### 3.2 Funkcia zariadenia:

Spočíva v naprogramovaní mikroprocesora Atmega8, ktorý pomocou tlačidiel TL1, TL2 vyšle bity na registre a tie podľa hodnoty signálu, ktoré prijali sa naplnia do určitej hodnoty a za pomoci tranzistorov zapnú príslušné LED diódy. Toto zariadenie je pripojené na 5V. Tlačidlo TL1-set a TL2- reset. Po viac násobnom stlačení tlačidla TL1 môžeme vidieť na matici LED preukázať Gaussovej krivky.

### 3.3 Opis práce

Na začiatku práce (zobrazenie Galtonovej rovnice) som si najskôr navrhol dve schémy, schému zapojenia súčiastok s mikroprocesorom (Atmega8), tlačítkami, registrami a ďalšími potrebnými súčiastkami pre realizáciu a samotnú činnosť práce. Druhá dosku (led maticu“), ktorá je obojstranná som si navrhol v poradí druhú. Na led matici sa nachádzajú usporiadané ledky, ktoré sú potrebné na preukázanie Gaussovej krivky. Obe schémy som si navrhol v programe (EAGLE), po vyhotovení oboch schém som si navrhol dosky plošných spojov (DPS) pre obe schémy. Návrh (DPS) vytvorený v počítači som si vytlačil na kriedový papier, ktorý som pomocou žehličky nažehlil na medennú dosku. Žehlenie prebiehalo zhruba v troch intervaloch po 20 sekúnd. Po úspešnom nažehlení som dosky vyleptal v leptacom roztoku. Dosky sa nachádzali v roztoku približne 30 min. Samozrejme to záleží aj od koncentrácie daného roztoku. Po vyleptaní oboch dosiek som dosky očistil a ošetril chyby ktoré nastali pri leptaní, ďalej som si na dosky navítal potrebné diery na osadenie súčiastok, ktoré slúžia na osadenie súčiastok a prepojovacích káblov či medzi doskami alebo na doske a pokračoval som osadzovaním najskôr led matice a neskôr riadiacej dosky. Hotové dosky plošných spojov (DPS) som po naspajkovaní a prepojení oboch dosiek skontroloval pred prípadnými nechcenými prepojami (chybami). Po uskutočnení tejto kontroli som dosky pripevnil a uzatvoril do obalu (púzdra) aby nedošlo k devastácii dosiek prípadným prenášaním. Mikroprocesor Atmega8 som pomocou programu v jazyku C naprogramoval.

### 3.4 Popis použitých súčiastok :

Led 8mm cervena 145

74hc595            4

BC817            16

BS170            16

Atmega8            1

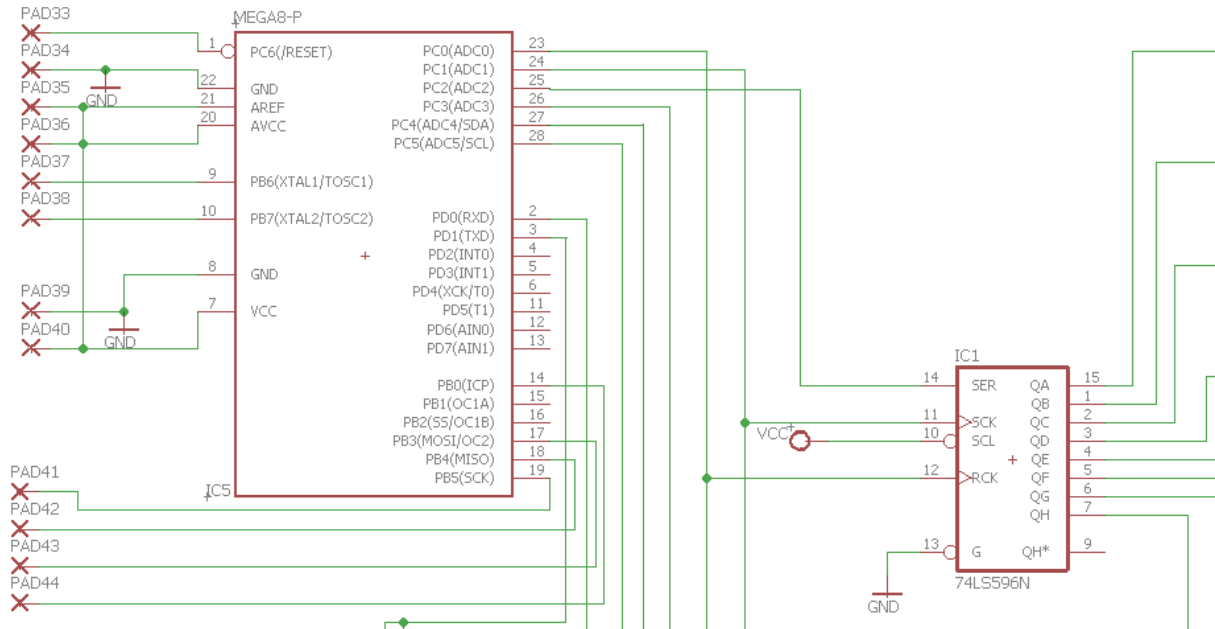
Odpor 3k3 smd    16

Odpor 1r smd    16

Tlačítko            2

### 3.5 Schéma

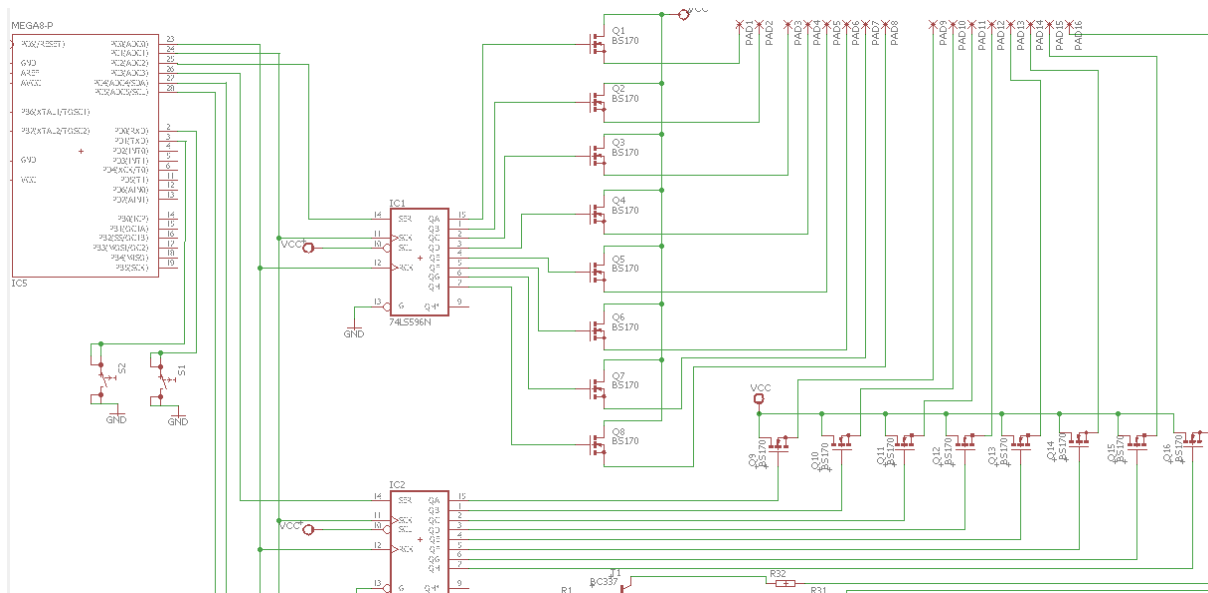
(Riadiaca doska)



Riadiaca doska

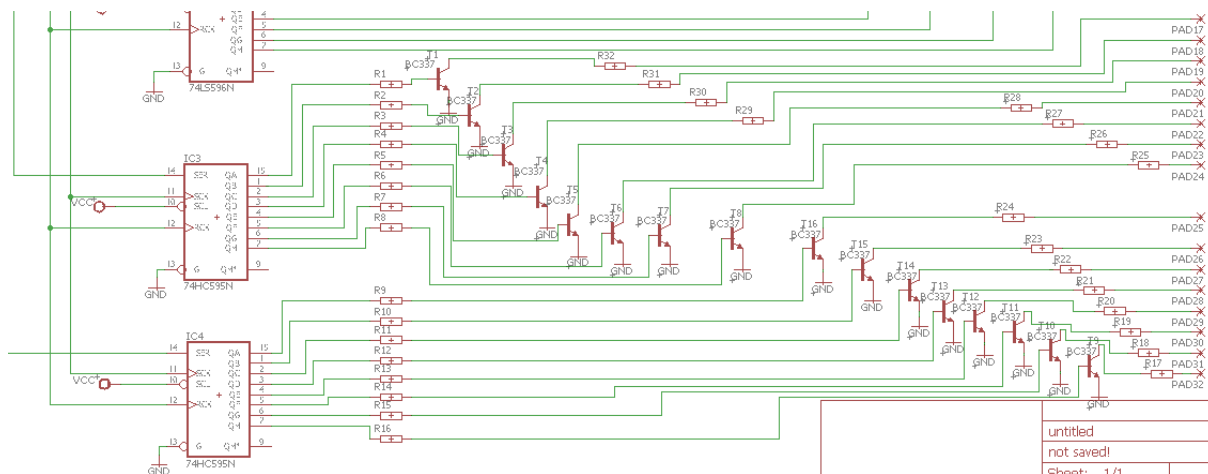
Atmega8 piny:

- Programovacie piny (prepájajú sa s programovacou jednotkou) → 1,8,14,17,18,19
- Piny vyvedené na tlačítka TL1 a TL2 → 2,3
- VCC(stačí ak je vyvedený jeden, sú navzájom prepojené) → 7,20,21
- GND (stačí ak je vyvedený jeden, sú navzájom prepojené) → 8,22
- PC0 (vyvedený na všetky registre kde je pripojený na pin 12-Storage register clock)→23
- PC1 (vyvedený na všetky registre kde je pripojený na pin 11-CLOCK)→24
- Piny vyvedené na registre (vstup sériových bitov-14) → 25,26,27,28



Posuvné registre privedené na anódu led diód:

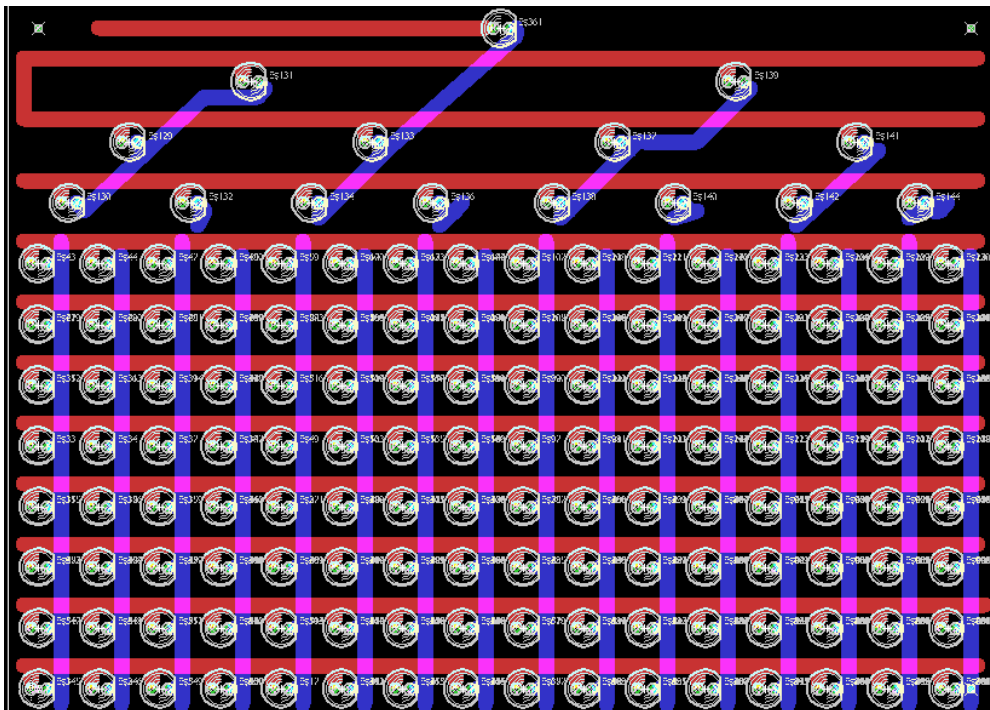
- Piny prepojené s mikroprocesorom Atmega8 →14,11,12
- VCC (napájanie) →10
- GND(zem) →13
- Paralelné výstupy privedené na tranzistory →1,2,3,4,5,6,7,15
- Tranzistory su ovládané posuvnými registrami a vyvedené na pady.



Posuvné registre privedené na katódu led diód:

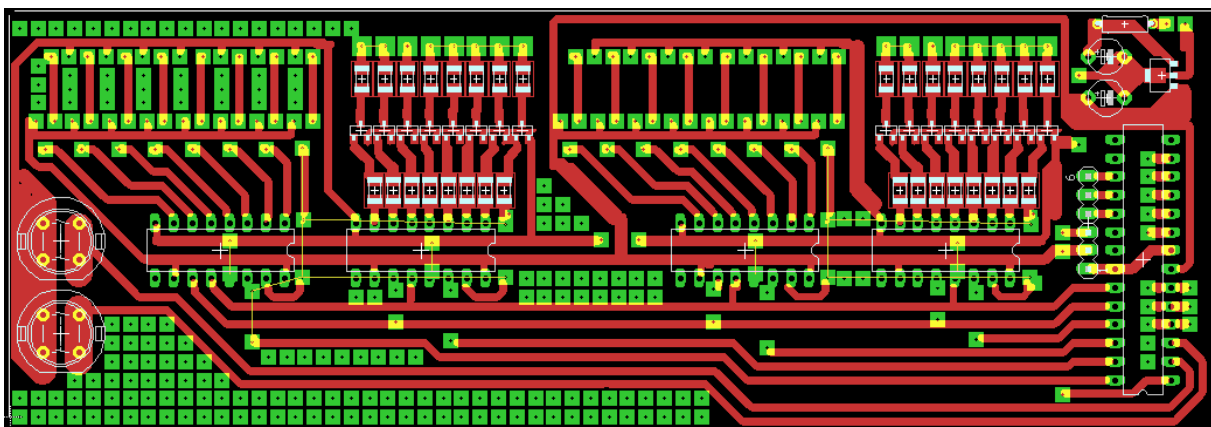
- Piny prepojené s mikroprocesorom Atmega8 →14,11,12
- VCC (napájanie) →10
- GND(zem) →13
- Paralelné výstupy privedené na tranzistory →1,2,3,4,5,6,7,15

### 3.6 Návrh dosky:



*Matica led diod*

Na obrázku môžeme vidieť DPS led matice a príkladné zapojenie led diód v riadkoch a stĺpoch pročom sa prepájajú s ovládacou doskou.



Riadiaca doska



### 3.7 Program a jeho opis:

```

#include <avr/io.h>                //pridavanie potrebných knižníc
#include <avr/interrupt.h>
#include <util/delay.h>

//          pc5      pc4      pc3      pc2      pc1  pc0
//          28      27      26      25      24  23
// 14- 595  s4      s3      s2      s1      11  12
//          lavystlpec dolnyrad pravystlpec hornyrad clk  res
//          zapina 0v  zapina 5v zapina 0v  zapina 5v
//horne 7 8
uint16_t riadok7;                //určovanie premenných použitých v programe
uint16_t riadok8;
//dolne 16x8
uint8_t stlpce[16]; //stlpce
uint8_t posch[2];
uint16_t riadok;
uint8_t posun =0;
uint8_t maska =0;
uint8_t temp1 =1;
uint8_t temp2 =1;
uint8_t i=0;
uint8_t k=0;
uint8_t sek =0;
uint8_t min =0;
uint8_t hod =0;
uint8_t pocet =0;
uint8_t funkcia =0;
uint8_t v =1;
//;bitove
#define ledleft 0b00000010        //definovanie vstupov a výstupov
#define ledrigh 0b00000001
#define tl1 0b00000001 // 1 pd0
#define tl2 0b00000010 // 2 pd1
// pripojenie 74hc595
#define dats4 0b00000101 // 28 pc5 riadky horne
#define dats3 0b00000100 // 27 pc4 stlpce vľavo
#define dats2 0b00000011 // 26 pc3 riadky dolne
#define dats1 0b00000010 // 25 pc2 stlpce vpravo
#define sift 0b00000001 // 24 pc1 posuv
#define stor 0b00000000 // 23 pc0 zapísať

int main()                        //
{
    DDRB = 0b11111111;
    PORTB = 0b00000000;
    DDRC = 0b11111111;
    PORTC = 0b11111111;
    DDRD = 0b11111100;
    PORTD = 0b00000011;
    nastav();
    _delay_ms (500);
    while(1)
    {
        dolna();
        _delay_ms (5);
        key();
    }
}

```

```

int gaus() //riešenie grafického znázornenia gausovej vety
{
    pocet=0;
    nuluj();
    while(pocet <30)
    {
        pocet++;
        _delay_ms (10);
        int opakuj()
        {
funkcia = rand() % 2;
        if(funkcia == 0)
        {
            v=2;
        }
        if(funkcia == 1)
        {
            v=3;
        }
    }
    if(v == 2)
    {
funkcia = rand() % 2;
        if(funkcia == 0)
        {
            v=4;
        }
        if(funkcia == 1)
        {
            v=5;
        }
    }
    if(v == 3)
    {
funkcia = rand() % 2;
        if(funkcia == 0)
        {
            v=5;
        }
        if(funkcia == 1)
        {
            v=6;
        }
    }
    . //v dôsledku veľkosti programu bolo v dokumentácií skrátené
    . riešenie určovania cesty pre jednu ledku, v programe sú zapísané príkazy
    . if iba zo začiatku určovania cesty (kde v= 2 alebo 3) a konca
    . (kde v= 77 alebo 78). V programe sú vypísane všetky možnosti.
    if(v == 66)
    {
funkcia = rand() % 2;
        if(funkcia == 0)
        {
            v=77;
        }
        if(funkcia == 1)
        {
            v=78;
        }
    }
}

```

```

if(v == 67)//ledky sú spúšťané od 3. po 14. (v programe od 2. do 13.)
    stĺpca, t.j. 3. a 14.(2. a 13.)stĺpec sú rozsvietené
{
    stlpce[2] =stlpce[2]>>1;
    stlpce[2] =stlpce[2] | 0b10000000;
}
if(v == 68)
{
    stlpce[3] =stlpce[3]>>1;
    stlpce[3] =stlpce[3] | 0b10000000;
}
. //v dôsledku veľkosti programu bolo v dokumentácií skrátené
. zapisovanie bitov do stĺpcov, v programe sú zapísané príkazy
. if s podmienkami pri ktorých v= od 67 do 78.
if(v == 78)
{
    stlpce[13] =stlpce[13]>>1;
    stlpce[13] =stlpce[13] | 0b10000000;
}
for (u=0;u<30;u++)
{
    opakuj();
}
}
}

int nuluj() //nastavenie každého stĺpca na hodnotu 0 (všetky ledky budú vypnuté)
{
    for(i=0;i<16;i++)
    {
        stlpce[i]=0;
    }
}

int nastav() //nastavenie každej ledky ľubovoľne podľa bytov v stĺpcoch
{
    stlpce[0]=0b00000010; //pravá strana
    stlpce[1]=0b00000010;
    stlpce[2]=0b00000010;
    stlpce[3]=0b00000010;
    stlpce[4]=0b00000010;
    stlpce[5]=0b00000010;
    stlpce[6]=0b00000010;
    stlpce[7]=0b00000010;
    stlpce[8]= 0b00100000; //ľavá strana
    stlpce[9]= 0b00100000;
    stlpce[10]=0b00100000;
    stlpce[11]=0b00100000;
    stlpce[12]=0b00100000;
    stlpce[13]=0b00100000;
    stlpce[14]=0b00100000;
    stlpce[15]=0b00100000;

    riadok7 =0b00000001;
    riadok8 =0b00000010;
}

```

```

void key() //po stlačení tlačidla(t11 alebo t12) sa program presunie na začiatok
           podprogramu gaus alebo v prípade stlačenia t12 na začiatok podprogramu nastav
{
    if((PIND & t11) == 0)
    {
        gaus();
        while((PIND & t11) == 0){}
    }

    if((PIND & t12) == 0)
    {
        nastav();
        while((PIND & t12) == 0){}
    }
}

int pridaj() //v tejto časti programu sa zapisuje čas
{
    sek++;
    if(sek == 60) {sek = 0; min++;}
    if(min == 60) {min = 0; hod++;}
    if(hod == 24) {hod = 0;}
}

int dolna() //tu nastáva aktivácia častí dosky
{
    riadok = 0b00000001;
    for(i=0; i<8; i++)
    {
        maska = 0b10000000;
        for(k=8; k>0; k--)
        {
            if((stlpce[k+7] & riadok)==0) //register ovládajúci ľavú
            {PORTC &= ~(1<<dats4);}      stranu lediek
            else{ PORTC |= (1<<dats4);}

            if((stlpce[k-1] & riadok)==0) //register ovládajúci pravú
            {PORTC &= ~(1<<dats2);}      stranu lediek
            else{ PORTC |= (1<<dats2);}

            if((riadok & maska)==0) //register ovládajúci
            {PORTC &= ~(1<<dats1);}      spodné riadky
            else{ PORTC |= (1<<dats1);}

            PORTC |= (1<<sift); //čas požitý pri zapisovaní
            PORTC &= ~(1<<sift);
            maska = maska >>1;
        }
        PORTC |= (1<<stor); //zapisovanie do registrov
        _delay_us(1);
        PORTC &= ~(1<<stor);
        _delay_us(1);
        _delay_ms(1);
        riadok = riadok<<1;
    }
}

```

## 4 VÝSLEDKY A DISKUSIA

Výsledkom mojej práce je funkčné zobrazenie Galtonovej rovnice a kompletný pracovný postup na zostrojenie tohto zariadenia. Najmenšie problémy som mal pri vyhotovovaní schémy horšie to bolo už pri vytváraní riadiacej DPS aby boli všetky piny na jednej strane pričom aby bola riadiaca doska čo najmenšia a s čo najmenej prepojami, myslím že sa mi to v celku podarilo. Ak by som mal porovnať prvú DPS a druhú (finálnu) ktorú som zmenšil aspoň o 50% tak som s ňou spokojný. Náklady na vyhotovenie tohto zariadenia sa pohybujú do rozmedzia 30€ pričom najväčšiu časť robia LED diódy(8mm). LED diódy robia sumu 20€ pričom ostatné súčiastky sa zmestia do sumy necelých 9€. Ak by som mal porovnať mechanický princíp zobrazenia Galtonovej rovnice a moj Elektronické zobrazenie Galtonovej rovnice tak si trúfam povedať že to je porovnateľné.

## ZÁVERY PRÁCE

Hlavný cieľ ktorý bol zhotoviť funkčnú učebnú pomôšku pre nasimulovanie modulu matematickej rovnice o pravdepodobnosti určitých kombinácii si myslím že sme dosiahli ale podme si to rozobrať postupne. Postupné dosiahnutie čiastkových cieľov sa nám podarilo pričom výsledné zariadenia funguje tak ako bolo navrhnuté. Výsledky práce si myslím že si nadmieru pozitívne. Zobrazenie Galtonovej rovnice som ešte nevidel v elektrickej forme preto sme sa ho rozhodli preukázať aj takýmto spôsobom čo si myslím že že je aj prínos pre prepojenie elektrotechniky a matematiky. Veľký prínos môže mať naše zariadenie aj ako učebná pomôckaže nejde úplne predpovedať náhodnosť aj keď vieme, že je väčšia pravdepodobnosť rozsvietenia stĺpcov. Najväčšie problémy boli pri navrhovaní schémy a navrhnutí riadiacej DPS. Musel som sa bližšie zoznámiť s mikroprocesorom Atmega8 a posuvnými registrami aby bola riadiaca doska plne funkčná a pripravená na programovanie.

## ZHRNUTIE

Cieľom práce bolo za pomoci číslicovej techniky zobrazit' simuláciu gaussovej vety. Pomocou programu, do ktorého boli vložené matematické princípy gaussovej vety boli na matici LED diód zobrazené výsledky náhodného javu gaussovej vety a pravdepodobnosti Pascalovho trojuholníka. Výsledné zobrazenie na svetelnej matici názorne zobrazuje náhodný prechod pólom lediek, ktorého výsledok je zaznamenaný do spodnej matice zobrazovača a ukazuje výsledok gaussovej vety a pascalovho trojuholníka. Touto prácou sme eliminovali potvrdenie gaussovej vety za pomoci drevenej dosky pri výrobe ktorej mohli vzniknúť nepresnosti ktoré mohli ovplyvniť výsledok experimentu. Program je spoľahlivý a výsledky nie sú ovplyvnené vonkajšími fyzikálnymi vplyvmi.

## RESUME

The aim of this thesis was to show the simulation of Gaussian sentence using digital techniques. Using a program into which the mathematical principles of the Gauss sentence were inserted, the results of the random Gaussian phenomenon and the probability of the Pascal triangle were displayed on the LED matrix. The resulting display on the light matrix illustrates a random passage of LED matrix, the result of which is recorded in the lower matrix of the viewer, and shows the result of the Gaussian sentence and the Pascal triangle. This work eliminated the confirmation of a Gaussian sentence with the help of a wooden board that could produce inaccuracies that could affect the result of the experiment. The program is reliable and the results are not influenced by external physical effects.



## ZOZNAM BIBLIOGRAFICKÝCH ODKAZOV

[https://en.wikipedia.org/wiki/Atmel\\_AVR](https://en.wikipedia.org/wiki/Atmel_AVR) → základy o firme Atmel

<http://www.circuitstoday.com/avr-atmega8-microcontroller-an-introduction> → základy mikroprocesora Atmega8

[https://en.wikipedia.org/wiki/EAGLE\\_\(program\)](https://en.wikipedia.org/wiki/EAGLE_(program)) → základy o programe EAGLE

[https://sk.wikipedia.org/wiki/C\\_\(programovac%C3%AD\\_jazyk\)](https://sk.wikipedia.org/wiki/C_(programovac%C3%AD_jazyk)) → základy o jazyku C

<https://sites.google.com/site/arduinosllovakia/74hc595-posuvny-register> → základy o posuvných registroch

<http://www.alldatasheet.com/datasheet-pdf/pdf/16109/PHILIPS/BC817.html> → informácie o tranzistore BC817

<https://www.silicon-ark.co.uk/bc170-silicon-npn-transistor> → informácie o tranzistore BC170