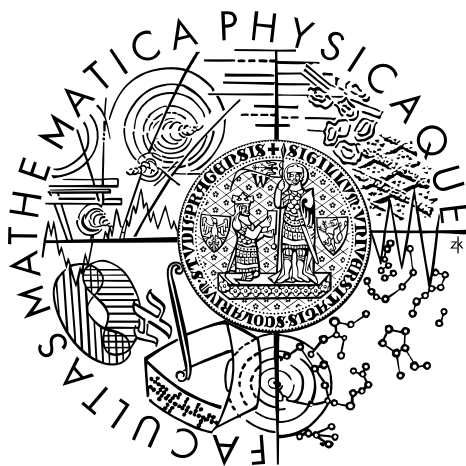


Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Josef Knot

Prezentace detektorů elementárních částic pro veřejnost

Ústav částicové a jaderné fyziky

Vedoucí diplomové práce:
doc. RNDr. Zdeněk Doležal, Dr., ÚČJF MFF UK

Studijní program: Fyzika

Studijní obor: Učitelství fyziky – matematiky pro SŠ

Praha 2011

Zde bych chtěl poděkovat všem lidem, kteří pomohli ke vzniku předkládané diplomové práce. Především bych rád poděkoval vedoucímu práce doc. RNDr. Zdeňkovi Doležalovi, Dr. za jeho trpělivost, kontrolu, připomínky k práci a velmi vstřícné jednání. Dále chci poděkovat RNDr. Peterovi Kodyšovi, CSc. za základní makra zajišťující komunikaci počítače s posuvným stolkem a s čítačem částic a za jeho pomoc při přípravě ovládacího programu k úloze do praktika. Za pomoc při programování bych dále rád poděkoval RNDr. Peterovi Kvasničkovi, Pavlovi Brandejsovi, Pavlovi Jirouškovi a Pavlovi Novotnému. Panu Ing. Petrovi Kubíkovi děkuji za pomoc při přípravě konstrukce k úloze.

Také bych chtěl poděkovat rodině a přátelům, kteří mi vytvořili dobré podmínky pro práci.

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Chebu dne 4.8.2011

Josef Knot

Název práce: Prezentace detektorů elementárních částic pro veřejnost

Autor: Josef Knot

Ústav: Ústav částicové a jaderné fyziky

Vedoucí diplomové práce: doc. RNDr. Zdeněk Doležal, Dr., Ústav částicové a jaderné fyziky MFF UK

Abstrakt: Předkládaná diplomová práce se zabývá detektory elementárních částic a možnostmi jejich využití. V úvodu je čtenář seznámen s pojmy radioaktivita a jaderné záření. Dále se zde dozví, k jakým procesům dochází při průchodu radioaktivního záření látkou a jak se těchto jevů využívá při detekci elementárních částic. Dále je zde rozebrána scintigrafická metoda zobrazování používaná v medicíně. Hlavní částí práce je pak seznámení s přípravou a výsledky připravované úlohy pro středoškolské praktikum. Je zde rozebrán cíl úlohy, postup její přípravy. Dále je zde popsána používaná aparatura, program používaný na ovládání úlohy a popis výsledků získaných při zkušebních měřeních.

Klíčová slova: detektor, radioaktivita, záření

Title: Presentation of particle detectors for public

Author: Josef Knot

Department: Institute of Particle and Nuclear Physics

Supervisor: doc. RNDr. Zdeněk Doležal, Dr., Institute of Particle and Nuclear Physics MFF UK

Abstract: This diploma thesis deals with particle detectors and their use. In the introduction the reader becomes familiar with concepts of radioactivity and radiation. Next part tells the reader how nuclear radiation penetrates through matter and how this is used for particle detection. Next part presents utilizing radiation in nuclear medicine and medical imaging. The main part of this work concentrates on the design and preparation of an original high school experiment. Here the aim of this experiment is described and its design and implementation is documented. Last part brings the description of technical solutions used, steering computer program developed for this experiment and results of tests.

Keywords: detector, radioactivity, radiation

Obsah

Úvod.....	1
1. Teoretická část.....	2
1.1. Atomové jádro a radioaktivita.....	2
1.2. Radioaktivní záření	3
1.2.1. Rozpad α	3
1.2.2. Rozpad β	4
1.2.3. Rozpad γ	5
1.3. Průchod záření látkou.....	5
1.3.1. Průchod těžkých elektricky nabitých částic látkou	5
1.3.2. Průchod elektronů látkou	6
1.3.3. Průchod fotonů látkou	6
1.4. Detekce elementárních částic.....	6
1.4.1. Scintilační detektor.....	7
1.4.2. Polovodičové detektory.....	8
1.5. Využití ionizujícího záření při zobrazování orgánů v medicíně.....	9
2. Experimenty z jaderné fyziky na středních školách.....	11
3. Úloha pro středoškolské praktikum.....	13
3.1. Motivace a popis úlohy	13
3.2. Cíl úlohy.....	13
3.3. Tvorba úlohy	13
3.4. Popis aparatury.....	23
3.5. Postup tvorby programu a popis jednotlivých funkcí	25
3.6. Výsledky měření	28
Závěr.....	33
Seznam obrázků	34
Literatura.....	35
Příloha A: Zachycení průběhu měření v intervalech po 10 minutách	36
Příloha B: Text k úloze do SŠ praktika.....	38
Příloha C: Hlavní makro vytvořené k ovládní úlohy pro SŠ praktikum.....	43

Úvod

V této práci jsem se snažil srozumitelně shrnout základní poznatky o radioaktivitě, radioaktivním záření a jeho detekci. Práce je určena především pro středoškolské studenty, kteří ji mohou využít jako doplněk ke školní výuce.

Na podporu rozšíření povědomí studentů středních škol o vlastnostech radioaktivního záření jsem vytvořil úlohu pro středoškolské praktikum na toto téma. Příprava, popis a výsledky připravené úlohy jsou hlavní náplní předkládané práce.

1. Teoretická část

1.1. Atomové jádro a radioaktivita

Všechny atomy se skládají z atomového jádra a elektronového obalu. Jádro je tvořeno neutrony a protony, které často označujeme souhrnným názvem nukleony. Neutrony jsou elementární částice, které nemají elektrický náboj a jejich klidová hmotnost je 1839krát větší než je klidová hmotnost elektronu. Protony jsou elementární částice, které mají kladný elektrický náboj o velikosti elektrického náboje elektronu. Klidová hmotnost protonu je 1836krát větší než je klidová hmotnost elektronu.

Pro popis složení atomových jader zavádíme protonové, neutronové a nukleonové číslo. Protonové (atomové) číslo Z udává počet protonů a neutronové číslo N udává počet neutronů v jádře atomu. Nukleonové (hmotnostní) číslo A udává počet nukleonů v jádře atomu. Nukleonové číslo je tedy součtem protonového a neutronového čísla.

Každé atomové jádro, takzvaný nuklid, chemického prvku X budeme označovat symbolem ${}^A_Z X$. Existují také nuklidy jednoho chemického prvku, které se liší počtem neutronů, takovým nuklidům říkáme izotopy.

V přírodě se můžeme setkat se dvěma druhy atomových jader. Jádra většiny prvků zůstávají stále stejná, říkáme jim stabilní, ale existují také nestabilní atomová jádra, která se během určité doby samovolně přeměňují (rozpadají) na jádra jiná nebo ztrácejí energii. Atomy, jejichž jádra jsou nestabilní, nazýváme radioaktivní. Pojem radioaktivita tedy značí schopnost atomových jader samovolně se přeměňovat (rozpadat). Při radioaktivní přeměně mohou vzniknout jak stabilní tak i nestabilní jádra, která se mohou nadále přeměňovat a která mohou být v základním nebo excitovaném stavu. Při radioaktivní přeměně vzniká radioaktivní záření. Výsledné záření je složením záření vzniklých při jednotlivých přeměnách. Příspěvky jednotlivých svazků jsou ovlivněny pravděpodobnostmi se kterými dochází k příslušným přeměnám. [1, 2]

Radioaktivní přeměna jader je způsobena jevy, které mají čistě náhodný charakter. Díky velkému počtu přeměn v radioaktivní látce, lze k popisu chování

radioaktivních jevů použít metody matematické statistiky. Pokud tedy budeme mít radioaktivní látku a budeme měřit četnosti počtů vyzářených částic během stejných časových intervalů, získáme rozdělení odpovídající Poissonově statistice. Pokud je počet jevů n naměřených během jednoho časového úseku dostatečně velký, pak můžeme Poissonovo rozdělení nahradit Gaussovým rozdělením, pro které platí, že střední hodnota je rovna rozptylu. Proto odchylka od naměřené hodnoty n je přibližně rovna \sqrt{n} . [5, 6]

1.2. Radioaktivní záření

Zářením většinou rozumíme proud elementárních částic nebo fotonů. Mezi radioaktivní záření zařazujeme kosmické záření, záření vznikající v urychlovačích částic, rentgenové záření a záření vznikající při radioaktivních rozpadech. Zářením, které má dostatečnou energii k ionizaci prostředí, kterým prochází, nazýváme ionizující záření. V další části se blíže podíváme na záření vznikající při radioaktivních rozpadech. Podle typu vyzařovaných částic rozlišujeme tři druhy radioaktivních rozpadů α , β a γ . [1]

1.2.1. Rozpad α

Při radioaktivním rozpadu α je v rozpadajícím se jádře s mnoha nukleony, díky své velké vazebné energii, utvořena částice α (jádro helia ${}^4_2\text{He}$). Částice α je v jádře držena jadernými silami, ale přesto díky takzvanému tunelovému jevu má šanci z jádra atomu uniknout.

Rozpad α lze popsat schématem ${}^A_Z X \rightarrow {}^{A-4}_{Z-2} Y + {}^4_2\text{He}$. Nově vzniklé jádro má tedy o dva protony a dva neutrony méně než mělo původní jádro. Rozpadem α se mohou rozpadat jádra s hmotnostním číslem větším než 150.

Energie vyzářených částic α závisí na nukleonovém čísle mateřského jádra. Přibližně platí, že se kinetická energie vyzářených částic α zvyšuje s rostoucím

nukleonovým číslem. Pro izotopy jednoho prvku to platí naopak. Energie vyzářených částic α se pohybuje mezi jedním až deseti megaelektronvolty¹. [1, 2]

1.2.2. Rozpad β

Tento rozpad je podmíněn přeměnou neutronu na proton. Neutron má o něco vyšší klidovou hmotnost než proton a proto k této přeměně může docházet. Při této přeměně dochází k vyzáření elektronu a elektronového antineutrína $n \rightarrow p + e^- + \bar{\nu}_e$. Zatímco volný neutron má střední dobu života 896 s, u neutronu silně vázaného v jádře k přeměně nedochází. K přeměně neutronů na protony v jádrech atomů tedy dochází jen tehdy, pokud jsou neutrony vázány malou silou. Pro jádra, ve kterých dochází k popsanému rozpadu, platí následující schéma ${}^A_Z X \rightarrow {}^A_{Z+1} Y + e^- + \bar{\nu}_e$. Jádro s A nukleony a Z protony se přemění na jádro s A nukleony a $Z+1$ protony a vyzáří se elektron a elektronové antineutrino. Právě popsaný proces se označuje β^- a probíhá u přírodních radioizotopů, jejichž jádra mají přebytek neutronů.

Existuje ještě druhý typ rozpadu β , který označujeme β^+ . Tento typ probíhá jen u uměle vyrobených radioizotopů. Tyto uměle vytvořené radioizotopy vznikají ostřelováním neradioaktivních atomových jader nabitými částicemi nebo neutrony. Tento druh beta rozpadu je podmíněn přeměnou protonu na neutron, při které je vyzářen pozitron a elektronové neutrino $p \rightarrow n + e^+ + \nu_e$. Pro jádra, ve kterých dochází k rozpadu β^+ , platí následující schéma ${}^A_Z X \rightarrow {}^A_{Z-1} Y + e^+ + \nu_e$. Při tomto rozpadu se jádro s A nukleony a Z protony přemění na jádro s A nukleony a $Z-1$ protony a vyzáří se pozitron a elektronové neutrino. Takto se rozpadají jádra s přebytkem protonů.

Elektrony (pozitrony) vyzařované při rozpadu β mají spojité spektrum energie. To je důsledkem tříčásticového rozpadu. Spojitost spektra energie vyzářených elektronů (pozitronů) lze vysvětlit tím, že energie vzniklá při rozpadu je rozdělena mezi elektrony (pozitrony) a elektronové antineutrino (neutrino). Pokud tedy antineutrino (neutrino) získá velkou část vzniklé energie, pak zbude na elektron (pozitron) energie malá a obráceně. Elektron (pozitron) tedy může být vyzářen

¹ Elektronvolt je fyzikální jednotka pro energii, která není v soustavě SI, ale v jaderné a částicové fyzice se velmi často používá. Mezi elektronvoltem a joulem platí vztah $1 \text{ eV} = 1,6 \cdot 10^{-19} \text{ J}$

s libovolnou energií mezi jeho klidovou energií a téměř maximální energií vzniklé při rozpadu. Energie vyzařovaných elektronů při rozpadu β^- se pohybuje v intervalu od desítek keV až po několik MeV. [1, 2]

1.2.3. Rozpad γ

Zatímco u rozpadů α a β dochází ke změnám protonového nebo nukleonového čísla, u rozpadu γ tomu tak není. K rozpadu γ dochází pouze u atomových jader, která se nacházejí v excitovaném stavu a pomocí vyzáření fotonu s konkrétní energií se dostávají do nižšího energetického stavu nebo přímo do stavu základního. Rozpad γ tedy velmi často doprovází ostatní dva rozpady.

Při tomto rozpadu jsou vyzařovány fotony o energiích pohybujících se v rozmezí od 0,05 MeV do 10 MeV. [1, 2]

1.3. Průchod záření látkou

Pokud radioaktivní záření prochází látkou dochází k interakci částic záření s atomy látky nebo jejich jádry. Při průchodu záření látkou se nejvíce uplatňuje elektromagnetická interakce díky záporně nabitému elektronovému obalu atomů látky a kladně nabitým jádrům. Při interakci záření s látkou může docházet k několika různým jevům podle toho, z jakých částic je záření složeno. [1]

1.3.1. Průchod těžkých elektricky nabitých částic látkou

Těžkými částicemi rozumíme částice s klidovou hmotností srovnatelnou nebo větší než je klidová hmotnost protonu. Tyto částice lehce projdou přes elektronový obal atomů a interagují především s jejich jádry. Při těchto interakcích dochází ke srážkám a rozptylu. Čím vyšší energii mají nalétávající částice tím menší bude úhel, o který se částice po průletu látkou vychýlí. Postupným rozptylováním v látce ztrácejí těžké částice svou energii. Pokud jejich energie dostatečně klesne, začnou se těžké částice nepružně srážet s atomy v látce a tím se ještě urychlí ztráty jejich

energie. Při těchto nepružných srážkách dochází ke zvyšování energie atomů (excitaci), nebo k uvolnění elektronů z atomu (ionizaci). Zatímco při excitaci ztrácejí těžké částice 1 – 10 eV, při ionizaci ztrácejí několik MeV. Těžké částice dosahují maximální specifické ionizace v látce na konci své dráhy, kdy mají velice malou energii. Specifickou ionizací rozumíme počet iontových párů, které těžká částice vytvoří při průletu jednotkové dráhy. Při ionizaci mohou vzniknout takzvané elektrony δ , které mají dostatečnou energii k další ionizaci. [1]

1.3.2. Průchod elektronů látkou

Elektrony procházející látkou interagují s jejími atomy opět především elektromagneticky. Na rozdíl od těžkých částic budou na elektrony mít velký vliv již elektronové obaly atomů látky. Elektrony procházející látkou ztrácejí energii ionizací a vysláním brzděného záření. Ztráty energie brzděným zářením jsou úměrné čtverci zrychlení částice, jsou tedy výraznější pro lehké částice než pro těžké. Elektrony o vysokých energiích ztrácejí svou energii především brzděným zářením a u elektronů s nízkou energií dochází ke ztrátám energie hlavně ionizací. [1]

1.3.3. Průchod fotonů látkou

Při průchodu fotonů látkou dochází k fotoefektu, rozptylování fotonů na elektronech nebo vytváření elektron – pozitronových párů v poli atomových jader. Kvůli těmto jevům se svazek fotonů zeslabuje. [1]

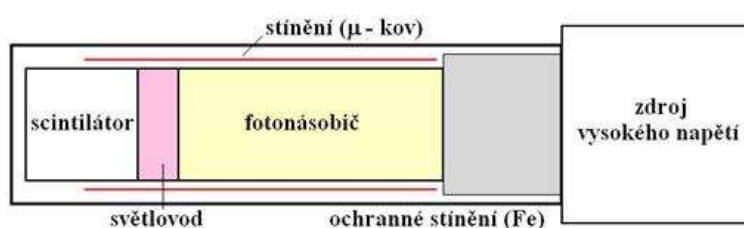
1.4. Detekce elementárních částic

K detekování částic se využívají procesy, které nastávají při průchodu svazku částic látkou. Přístroje určené k zaznamenání přítomnosti elementární částice nazýváme detektory. Některé detektory dokáží kromě přítomnosti částice určit některé její charakteristiky jakými jsou například energie, hybnost nebo elektrický náboj. Kvalitu detektorů můžeme popisovat pomocí několika parametrů jakými jsou

například detekční účinnost, mrtvá doba, prostorové rozlišení, časové rozlišení nebo energetické rozlišení. [1]

1.4.1. Scintilační detektor

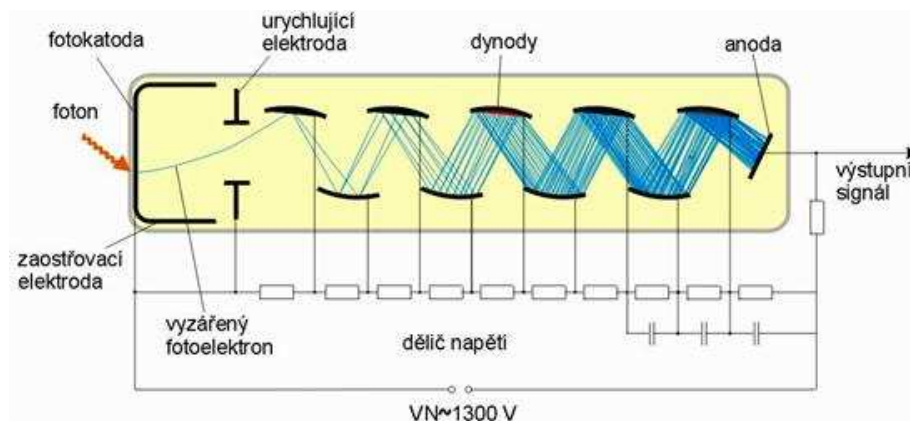
Scintilační detektor využívá excitace látky při průchodu elementárních částic. Skládá se ze scintilátoru, světlovodu a fotonásobiče. Celý detektor je uzavřen ve světlotěsném obalu viz obrázek 1.



Obrázek 1: **Schéma scintilačního detektoru.** Obrázek byl získán z [9].

Při dopadu ionizujícího záření na scintilátor předávají částice záření část své energie atomům nebo molekulám v látce, které se tím dostávají do nestabilního stavu. Dochází k excitaci atomů nebo molekul látky, ze které je scintilátor vyroben (mohou to být například anorganické nebo organické krystaly, plastické a další materiály). Excitované atomy se pak zbavují získané energie pomocí vyzáření elektromagnetického záření a dostávají se opět do základního stavu (dochází k deexcitaci). Materiál pro scintilátor je volen tak, aby z něj při jeho deexcitaci vycházely viditelné světelné záblesky. Vzniklé světelné záblesky jsou vedeny pomocí světlovodu do fotonásobiče.

Fotonásobič slouží k převedení světelného signálu na signál elektrický a jeho zesílení. Fotonásobič je složen z fotokatody, zaostřovací elektrody urychlující elektrody, několika dynod a anody viz obrázek 2.



Obrázek 2: **Schéma fotonásobiče.** Obrázek byl získán z [8].

Fotony ze světelných záblesků jsou přivedeny na fotokatodu fotonásobiče, ze které se díky fotoefektu vyrazí několik elektronů. Získané elektrony jsou v evakuované trubici fotonásobiče urychlovány elektrickým polem a směřovány pomocí zaostřovací elektrody na první dynodu. Po dopadu elektronu na dynodu dojde k sekundární emisi a z dynody se uvolní více elektronů, než kolik jich na dynodu dopadlo. Protože na každé další dynodě je větší kladný potenciál než na předchozí, jsou elektrony vždy směřovány k další dynodě a také ještě více urychlovány. Nakonec se zesílený signál dostane na anodu fotonásobiče, kde je registrován. Počet elektronů dopadajících na anodu může být až milionkrát větší než na první dynodě. Velikost registrovaného elektrického signálu je úměrná energii, kterou částice ztratila při průchodu scintilátorem.

Doba mezi dvěma signály, které je scintilační detektor schopen zaregistrovat, je kratší než jedna mikrosekunda. [1, 8, 9]

1.4.2. Polovodičové detektory

Kromě scintilačních detektorů se pro detekci ionizujícího záření často využívají polovodičové detektory.

Při spojení polovodiče typu p a polovodiče typu n vzniká na jejich hranici takzvaná hradlová vrstva, ve které nejsou volné nosiče náboje. Vzniklá hradlová vrstva je malá, ale pokud připojíme k polovodiči vnější zdroj napětí ve správném směru tj. plus na polovodič typu n a mínus na polovodič typu p, dosáhneme rozšíření hradlové vrstvy. Díky velké šířce hradlové vrstvy má polovodič velký měrný odpor,

který zabraňuje procházení nechtěného proudu polovodičem. Pokud se dostane částice ionizujícího záření do hradlové vrstvy, způsobí tam ionizaci a vznik elektricky nabitých iontů, které jsou odváděny na elektrody a tvoří detekovaný signál.

Detektory tvořené pevným krystalem mají výhodu v tom, že dokáží zabrzdit i částice s velkou energií. Polovodičové detektory se hodně používají pro detekci záření γ . Šířka hradlové vrstvy (vrstvy citlivé na částice) se u bariérových polovodičových detektorů s p – n přechodem pohybuje v rozmezí 0,2 – 0,5 mm. Tento typ polovodičových detektorů se používá na detekci těžkých nabitých částic, jakými jsou například částice α . [1]

1.5. Využití ionizujícího záření při zobrazování orgánů v medicíně

Využíváním radioaktivních látek v lékařství se zabývá nukleární medicína, která se zaměřuje na diagnostiku a terapii. Používají se dva druhy vyšetřovacích metod. Při první je pacientovi do těla přivedena radioaktivní látka a měří se radioaktivita v pacientovi, při druhé se pacientovi pouze odeberou vzorky, které jsou pak zpracovávány pomocí radioaktivní látky a následně je u nich proměřena radioaktivita. Při druhém způsobu tedy pacient s radioaktivní látkou vůbec nepřijde do styku. Podíváme se blíže na první typ vyšetřovacích metod. Tento typ je založen na objektivním měření nebo registraci radioaktivního záření vycházejícího z pacienta po podání radioaktivní látky. Zjišťuje se především kam se tato látka dostává. Při vyšetřeních se používá především záření γ , které je detekováno převážně pomocí scintilačních detektorů.

Pro zobrazení orgánů, ve kterých se nahromadilo dostatečné množství radioaktivní látky se používají takzvané scintigrafické zobrazovací systémy. Orgány mohou být zobrazovány buď plošně nebo tomograficky (trojrozměrně).

Plošné (planární) scintigrafické systémy v současné době používají jako zobrazovací zařízení takzvanou scintilační kameru (gama-kameru). K vytvoření obrazu se používá velkoplošný stacionární scintilační detektor. Plocha scintilátoru, který je většinou tvořen krystalem jodidu sodného, odpovídá ploše, kterou lze najednou zobrazit. Za krystalem scintilátoru je umístěno velké množství fotonásobičů, které tvoří souřadnicovou síť. Polohová rozlišovací schopnost

gama-kamery závisí na počtu fotonásobičů (čím je jich více tím lepší je rozlišovací schopnost). Pokud ve scintilátoru dojde k záblesku, získáme nějaký signál ze všech fotonásobičů. Tyto signály se však liší svou intenzitou podle toho, jak daleko byl daný fotonásobič od místa záblesku. Dva nejsilnější signály určují souřadnice místa záblesku. Získaný bod se vykreslí na obrazovku osciloskopu. Měření trvá tak dlouho dokud se na osciloskopu nezobrazí dobře prokreslený obraz vyšetřované oblasti (dokud nemáme naměřený dostatečný počet bodů). Při tomto měření je důležité správné nastavení polohy detektoru tak, aby docházelo k maximální registraci scintilací ve scintilátoru. K tomu aby byl detektor zaměřen jen na vyšetřovaný orgán se používá olověný kolimátor, který má stejný tvar jako krystal scintilátoru a je tlustý několik cm. V kolimátoru je vytvořený systém otvorů, díky kterým se do krystalu dostanou pouze gama paprsky v kolmém směru. Paprsky nalétávající šikmo jsou pohlceny materiálem kolimátoru. Výsledný obraz získaný gama-kamerou je tedy mozaikou složenou z částí odpovídajících otvorům v kolimátoru. V jednotlivých částech obrazu odpovídá počet zachycených scintilací aktivitě záření. Výsledný obraz je zachycován buď na fotografický materiál nebo zaznamenáván do paměti počítače a uložen pro další analyzování. Tento způsob zobrazování je používán i na sledování rychlých dynamických procesů jako je například studie krevního oběhu. Pak je výstupem sada na sebe navazujících obrazů.

Dnes lze získat i trojrozměrné obrazy orgánů. K tomu se využívají tomografické gama-kamery s jedním nebo třemi detektory, které se po malých úhlových krocích otáčejí kolem pacienta a sbírají velké množství projekcí. Takto posbíraná data jsou pak v počítači zpracována pomocí různých matematických metod a je z nich vytvořen výsledný obraz. Tento způsob sběru dat a zobrazování se používá u jednofotonové emisní tomografie a u pozitronové emisní tomografie. [4]



Obrázek 3: **Pozitronová emisní tomografie PET.** Obrázek byl získán z [11].

2. Experimenty z jaderné fyziky na středních školách

Jaderná fyzika zkoumá jedny z nejmenších částí hmoty, což jsou atomová jádra, a také vlastnosti radioaktivního záření, které je pro člověka ve větší míře nebezpečné. Z těchto důvodů jsou experimenty z této oblasti fyziky celkem náročné a ve středoškolském provedení lze provádět jen omezený počet experimentů zaměřených převážně na studium radioaktivního záření. Na internetu se mi nedařilo nalézt příliš experimentů na téma radioaktivní záření a pokud jsem nějaké našel, tak se velmi podobaly experimentům uvedeným ve středoškolských učebnicích, jejichž seznam je níže.

V učebnici Pokusy z fyziky pro střední školy jsou popsány hlavní experimenty, které lze provádět. K demonstračním pokusům z jaderné fyziky se především využívá souprava GAMABETA, ale existují i některé další možnosti s využitím starších přístrojů.

Popsané experimenty jsou následující:

1. Demonstrace existence a účinků radioaktivního záření Geigerovým – Müllerovým počítačem
2. Demonstrace záření alfa difúzní mlžnou komorou
3. Zvuková a světelná signalizace přítomnosti zdroje záření
4. Stínění detektoru před ionizujícím zářením
5. Stanovení úrovně pozadí ve třídě
6. Statistická povaha radioaktivní přeměny
7. Úbytek záření se vzdalováním radioaktivního zdroje
8. Absorpce záření beta a gama v závislosti na tloušťce vrstvy stínícího materiálu a na jeho typu
9. Rozptyl záření beta
10. Zjišťování dávky záření gama
11. Funkce radionuklidového hlásiče požáru
12. Výstup ionizujícího záření z obrazovky barevného televizoru
13. Poločas rozpadu radionuklidu

V učebnici Fyzika mikrosvěta, podle které se učí především na gymnáziích jsou uvedeny dvě laboratorní práce: zkoumání spekter atomů a zkoumání vlastností radioaktivity. Ke druhé laboratorní práci je opět doporučena souprava GAMABETA a experimenty popsané v návodech u této soupravy.

Kromě soupravy GAMABETA lze v dnešní době použít i další učební pomůcky, jakými jsou například měřicí systém Vernier nebo PASCO. Tyto systémy mají výhodu v tom, že je lze připojit k počítači a naměřené výsledky ihned zpracovávat. Díky jejich mobilnosti lze také měření provádět mimo třídu. Počet typů experimentů z jaderné fyziky, které lze provádět, se příliš nezvýšil.

Pokud máme k dispozici soupravu GAMABETA a systém Vernier, lze jako laboratorní práci zadat například následující zadání:

„Změř úroveň pozadí v místnosti a na louce. Ověř účinek ozáření detektoru od zdroje záření na vzdálenosti, době, tloušťce stínění a materiálu stínění. Ověř zákon radioaktivní přeměny. Urči poločas přeměny baria ^{137m}Ba .“ [15]

Se svou úlohou do středoškolských praktik jsem se přidal k výše zmiňovaným experimentům v tom smyslu, že je to úloha zaměřená na vlastnosti radioaktivního záření. Snažil jsem se však trochu obměnit způsob měření a připodobnit úlohu praktickému využití radioaktivního záření. [12 – 16]

3. Úloha pro středoškolské praktikum

3.1. *Motivace a popis úlohy*

Motivací pro vznik dále popisované úlohy do praktika pro středoškolské studenty bylo vytvoření úlohy, ve které se studenti setkají s radioaktivním zářením, ujasní si některé jeho vlastnosti a možnost jeho využití při zjišťování tvaru nebo i materiálu skrytých objektů pomocí jejich prozařování a následnou detekcí prošlého záření, tomuto postupu budeme v dalším textu říkat skenování. Úkolem studentů je co nejrychleji zjistit jaký nápis vytvořený z číslic nebo písmen je umístěn na zakryté destičce pomocí jejího proskenování radioaktivním zdrojem. Podrobnější popis aparatury a průběhu měření je v následujících kapitolách.

3.2. *Cíl úlohy*

Cílem úlohy je přiblížení radioaktivního záření studentům. Při této úloze by si studenti měli především uvědomit schopnost radioaktivního záření procházet různými materiály a také to, že se intenzita záření při průchodu materiálem snižuje. Další věcí, které by si měli studenti všimnout, je statistické chování radioaktivního rozpadu. V neposlední řadě si zde studenti vyzkouší, že je obtížné vyčíst něco z obrazu, pokud nejsou vyplněné všechny pixely a jak se obraz vylepšuje postupným vyplňováním dalších bodů v rastru. Nakonec by si studenti měli uvědomit souvislost této úlohy s jedním ze skutečných využití radioaktivního záření v praxi, čímž je vytváření obrazů skenováním.

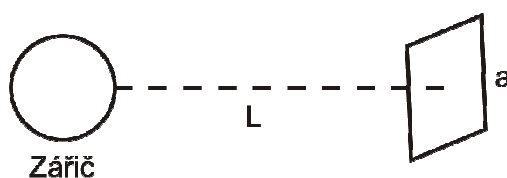
3.3. *Tvorba úlohy*

Základní myšlenkou úlohy je mít nějaké neprůhledné pouzdro, do něhož lze ukrýt nápis, který mají studenti pomocí skenování určit. Aby to šlo provést, musí být dostatečný rozdíl mezi počtem detekovaných částic, které proletí pouze pouzdem a které prolétají i skrz nápis. Prvním krokem tedy bylo zvolit typ radioaktivního

záření, materiál pro nápis a jeho tloušťku. Protože počet částic detekovaných scintilačním detektorem závisí také na velikosti scintilátoru a jeho vzdálenosti od zdroje záření, bylo potřeba odhadnout i tyto parametry.

Při úloze je použito záření β^- , protože záření α by neprošlo přes měřenou destičku s nápisem a při použití záření γ by naopak nebyl vidět velký rozdíl mezi tím, když záření prochází nápisem nebo ne. Jako první materiál pro vytvoření číslic byl vybrán hliník.

Vztah mezi počtem vyzářených a detekovaných částic bez překážky určíme následovně. Máme zdroj záření ve vzdálenosti L od čtvercového detektoru o rozměru a viz obrázek 4.



Obrázek 4: Náčrtek zářiče a detektoru

Protože zářič vyzařuje do všech směrů, bude poměr vyzářených částic a detekovaných částic stejný jako poměr plochy povrchu koule o poloměru L , do které zářič vyzařuje, a plochy detektoru. Platí tedy vztah (1), kde N je počet vyzářených částic, N_D počet detekovaných částic, S plocha, do které zářič vyzařuje a S_Z plocha scintilátoru.

$$\frac{N}{N_D} = \frac{S}{S_Z} \quad (1)$$

Nyní do vztahu (1) dosadíme za povrch koule, za plochu scintilátoru a vyjádříme počet detekovaných částic. Dostaneme přibližný vztah popisující závislost počtu detekovaných částic na geometrii uspořádání

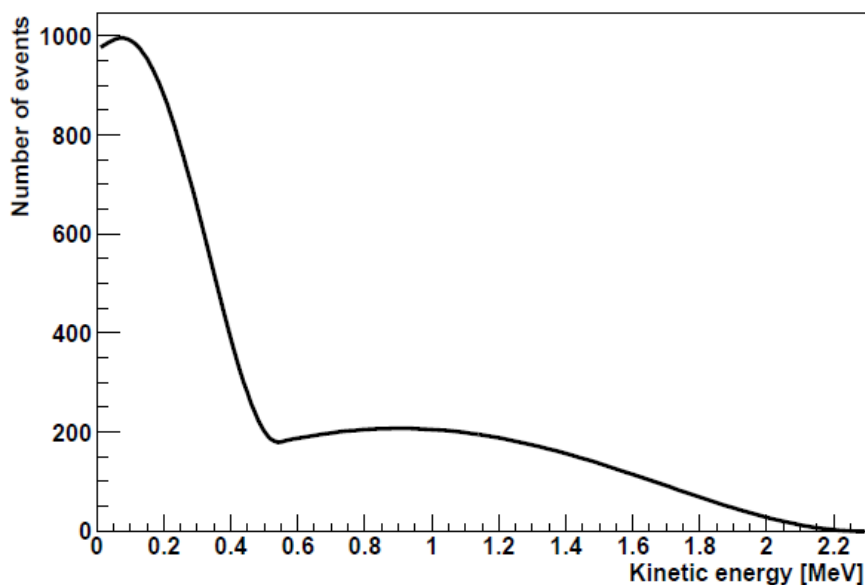
$$N_D = \frac{a^2}{4\pi L^2} N. \quad (2)$$

Dále ještě vyjádříme počet vyzařovaných částic pomocí aktivity zářiče A . Aktivita zářiče vyjadřuje počet vyzářených částic za jednu sekundu. Počet vyzářených částic je tedy součin aktivity a doby měření t . Po dosazení aktivity do vztahu (2) dostaneme

$$N_D = \frac{a^2}{4\pi L^2} At. \quad (3)$$

Pokud víme kolik částic dopadne na plochu scintilátoru, můžeme dále určit kolik částic bude utlumeno při jejich průchodu číslicemi, k tomu potřebujeme znát vlastnosti použitého zdroje záření.

Jako zdroj záření β^- je použit izotop stroncia ^{90}Sr . Tento izotop se rozpadá podle schématu $^{90}\text{Sr} \xrightarrow{\beta^-} ^{90}\text{Y} \xrightarrow{\beta^-} ^{90}\text{Zr}$. Poločas rozpadu ^{90}Sr je 28,5 let a poločas rozpadu ^{90}Y je 68 hodin. Protože je poločas rozpadu stroncia mnohem delší než poločas rozpadu yttria, je počet rozpadů obou izotopů přibližně stejný. Maximální energie vyzářených elektronů při rozpadu stroncia je asi 0,546 MeV a při rozpadu yttria asi 2,283 MeV. Energetické spektrum elektronů vyzařovaných při rozpadu ^{90}Sr je na obrázku 5. ^{90}Sr je hojně využíván v lékařství při radioterapii. [1, 8]



Obrázek 5: Energetické spektrum vyzařované stronciem ^{90}Sr . Obrázek byl získán z [3].

Absorpce elektronů v látce je přibližně exponenciální a platí pro ni tento vztah

$$N'_D = N_D e^{-\mu d}. \quad (4)$$

N'_D je počet elektronů, které projdou, d je tloušťka materiálu a μ je konstanta pro daný zářič a absorbující materiál. Za N_D ještě dosadíme počet částic dopadajících na detektor ze vztahu (3) a dostaneme

$$N'_D = \frac{a^2}{4\pi L^2} A t e^{-\mu d}. \quad (5)$$

Protože je záření z použitého zdroje složeno ze dvou spekter bude vztah pro počet prošlých částic ještě složitější. Konstanta μ se bude pro obě spektra lišit a aktivitu zdroje musíme rozdělit na dvě různě velké části podle toho, ve kterém spektru se více vyznačuje. Konečný vztah má tedy tvar

$$N'_D = \frac{a^2}{4\pi L^2} x_1 A t e^{-\mu_1 d} + \frac{a^2}{4\pi L^2} x_2 A t e^{-\mu_2 d}. \quad (6)$$

Do vztahu (6) jsem dosazoval $x_1 = 0,58$, $x_2 = 0,42$, $\mu_1 = 81 \text{ cm}^{-1}$ a $\mu_2 = 16 \text{ cm}^{-1}$. Tyto hodnoty jsem získal z vysokoškolských praktik. Za aktivitu jsem dosazoval hodnotu 10 000 Bq. [1, 15]

Pro první rozmyšlení, jak by mohly vypadat parametry úlohy jsem vytvořil tabulku obsahující veličiny a , L , t , d , počty částic N_D a N'_D spočítané podle vztahů (3) a (6), poměr $\frac{N'_D}{N_D}$ a rozdíl $N_D - N'_D$. Protože radioaktivní rozpad a tedy i počet detekovaných částic má statistický charakter, přidal jsem do tabulky ještě absolutní a relativní chybu určení rozdílu $N_D - N'_D$. Ve vytvořené tabulce jsem hledal takové nastavení, kde by byly co možná největší hodnoty N_D , N'_D , $N_D - N'_D$ s co nejmenší chybou a co nejmenší poměr $\frac{N'_D}{N_D}$, což je pro úlohu nejdůležitější vlastnost.

Ze získaných hodnot bylo vidět, že pro velký počet detekovaných částic je potřeba mít co největší plochu detektoru, malou vzdálenost mezi zářičem

a detektorem, velký čas a tenký materiál pro písmena. Ale pro chtěný nízký poměr $\frac{N'_D}{N_D}$ je naopak potřeba co nejsilnější materiál. Protože byl k dispozici scintilátor o rozměru 1 cm, snažil jsem se hledat další parametry pouze pro tento rozměr. V tabulce 1 uvádím několik vybraných hodnot s dobrými vlastnostmi.

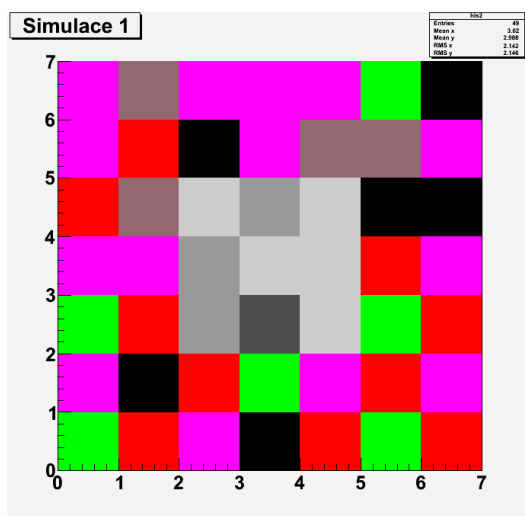
a [cm]	L [cm]	t [s]	d [cm]	N_D	N'_D	N'_D/N_D	$N_D - N'_D$	σ	δ [%]
1	2	3	0,05	597	119	0,20	478	26,75	5,59
1	2	2	0,05	398	79	0,20	319	21,84	6,85
1	2,5	3	0,05	382	76	0,20	306	21,40	6,99
1	3	3	0,05	265	53	0,20	213	17,83	8,39
1	2,5	2	0,05	255	51	0,20	204	17,47	8,56
1	2	2	0,04	398	97	0,24	301	22,25	7,40
1	2,5	3	0,04	382	93	0,24	289	21,80	7,55
1	3	3	0,04	265	65	0,24	200	18,17	9,06
1	2,5	2	0,04	255	62	0,24	192	17,80	9,25
1	2	3	0,04	597	146	0,24	451	27,25	6,04

Tabulka 1: **Parametry úlohy.** V posledních dvou sloupcích je absolutní chyba rozdílu $N_D - N'_D$ σ a relativní chyba δ .

V dalším kroku bylo potřeba zjistit, zda příslušné rozdíly mezi počty detekovaných částic budou dobře patrné i při zobrazování. Také bylo potřeba rozmyslet jakým způsobem bude vytvořená plocha pro skenování. Pro simulaci počtů částic jsem připravil několik maker v programu PSPad Editor, která jsem spouštěl pomocí programu ROOT 5.26.00, který jsem používal i dále ke tvorbě všech potřebných maker a programu na ovládání úlohy. Program ROOT 5.26.00 jsem získal z internetových stránek CERNu [7]. Statistický charakter počtů detekovaných částic je zajištěn pomocí Gaussova rozdělení.

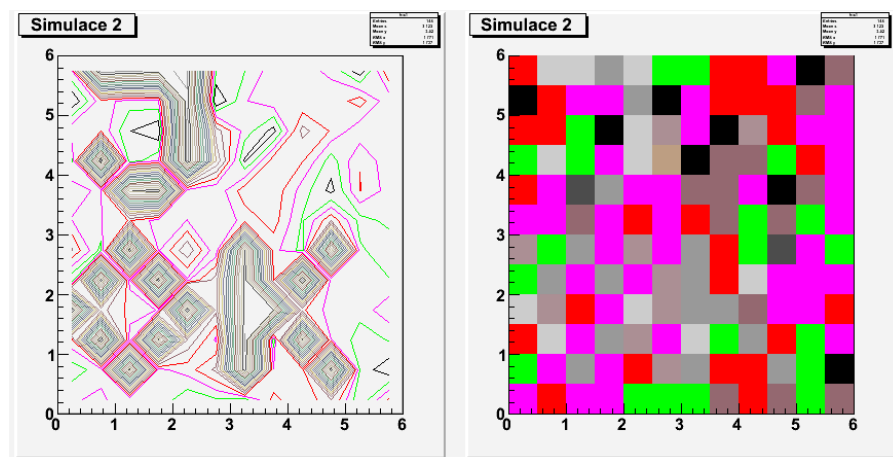
Prvním nápadem bylo vystřížení nápisu z několika vrstev alobalu a jejich umístění mezi další dvě fólie, které by byly orámovány kovovým rámečkem. Na obrázku 6 je histogram, ve kterém je uprostřed nasimulována tloušťka alobalu 1 mm a na okrajích 0,2 mm. Pro výpočet počtů částic je použit vztah 6, který je využíván i ve všech dalších simulacích. Vzdálenost zářiče a detektoru je nastavena na 2 cm, rozměr detektoru je 1 cm a délka měření jednoho políčka v rastru je 3 s. Uvedené parametry úlohy jsem částečně volil na základě výsledků uvedených v tabulce 1. Podle vztahu 6 vychází pro tyto parametry, že po průchodu slabší vrstvou alobalu

detektor zachytí asi 50 částic za tři sekundy a po průchodu tlustší vrstvou detekuje asi 250 částic.



Obrázek 6: Simulace počtů částic prošlých silnější a slabší vrstvou alobalu

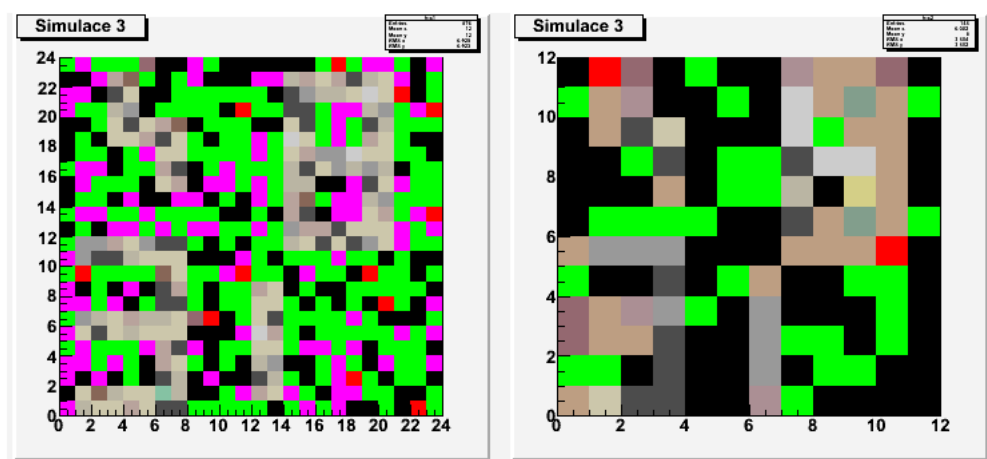
Při druhé simulaci jsem v textovém editoru vytvořil matici o rozměrech 12×12 znázorňující umístění písmen. Na obrázku 7 je zobrazen výsledek pro stejné parametry jako v předchozí situaci. Jsou zde také porovnány dvě z několika možných zobrazení.



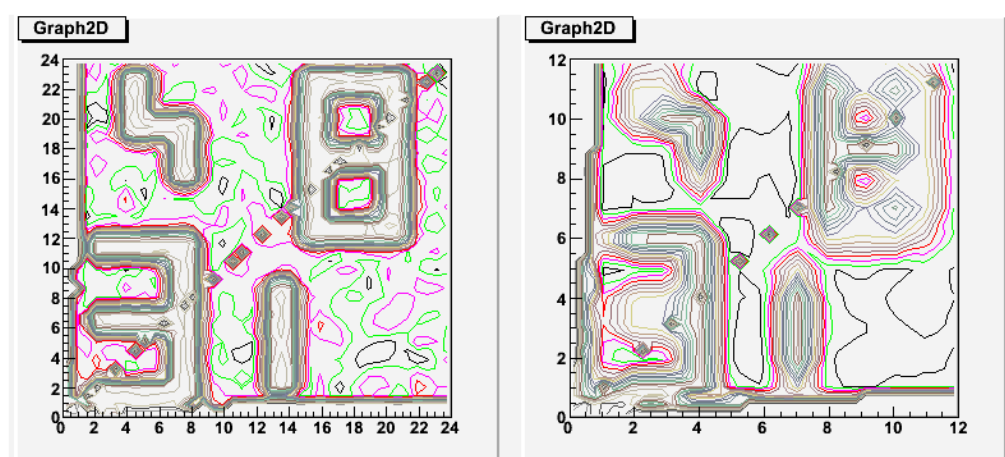
Obrázek 7: Simulace zobrazení písmen J, O a K pomocí histogramu

Z obrázku 7 je vidět, že pro nasimulované parametry budou písmena rozeznatelná, ale ne při všech možných způsobech zobrazování.

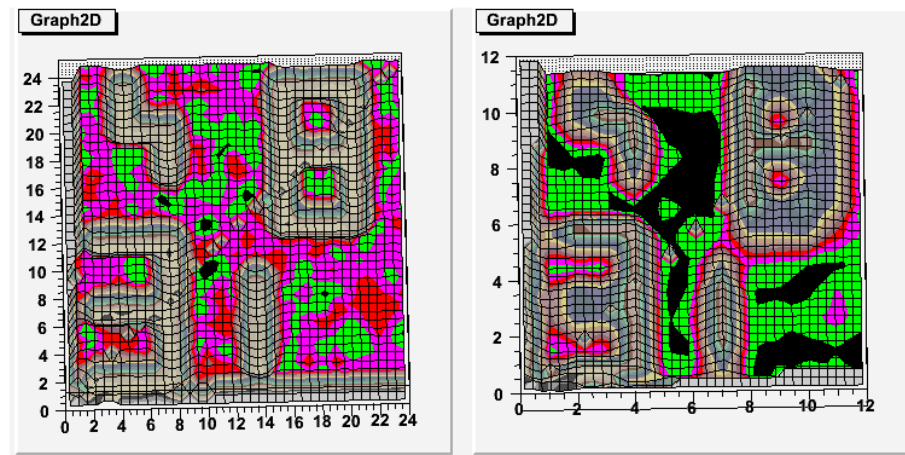
Ve třetí simulaci jsem porovnával vliv počtu proměřovaných bodů plochy na výsledný vzhled získaného obrázku. K tomuto účelu jsem opět v textovém editoru vytvořil matici hodnot o rozměrech 24×24. V případě znázorňujícím podrobné proměřování jsem zobrazil všech 576 změřených hodnot. Pro znázornění méně podrobného proměření jsem zobrazoval vždy jen průměrnou hodnotu ze čtyř nejbližších políček, tím jsem tedy získal jen 144 hodnot. Výsledky této simulace jsou zachyceny při různých způsobech zobrazení a pro stejné parametry jako předešlé simulace na obrázcích 8, 9 a 10. Vlevo je vždy zobrazen obrázek s více proměřenými body. Na obrázku 8 je k zobrazení využit histogram a na obrázcích 9 a 10 je použit graf.



Obrázek 8: Porovnání vzhledu pro různý počet proměřených bodů 1



Obrázek 9: Porovnání vzhledu pro různý počet proměřených bodů 2

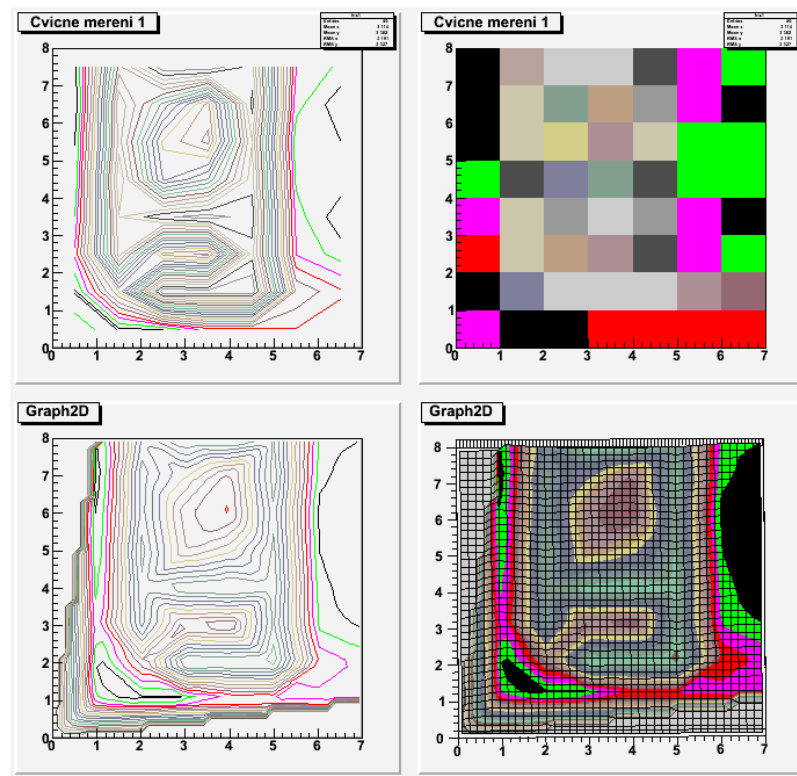


Obrázek 10: Porovnání vzhledu pro různý počet proměřených bodů 3

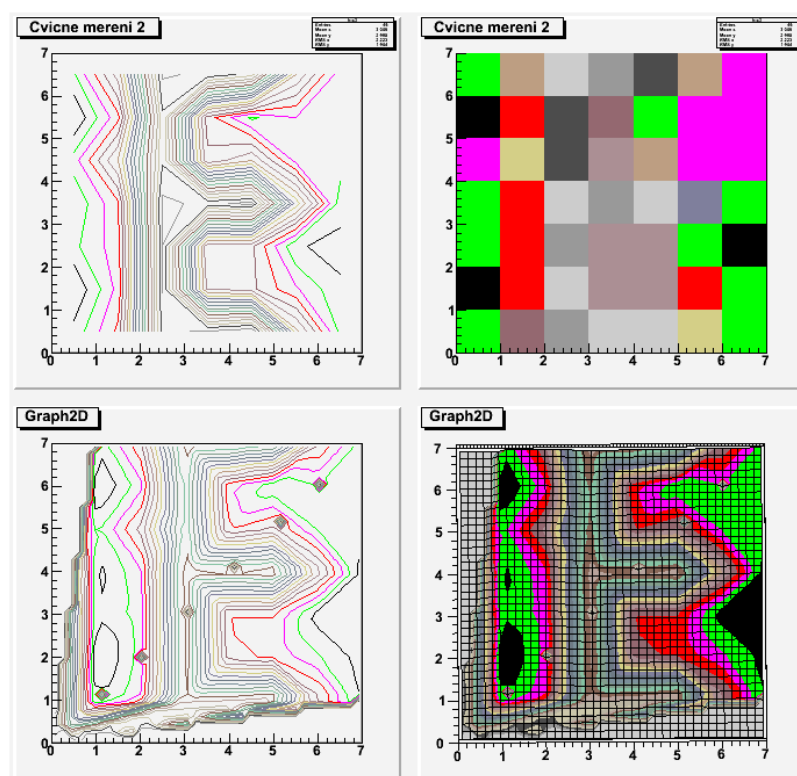
Rozdíl mezi zobrazováním pomocí čtverečkového histogramu nebo grafu je v tom, že histogram zobrazuje jen přímo změřené hodnoty, zatímco graf provádí interpolaci mezi jednotlivými hodnotami. Z předcházejících obrázků je vidět, že graf je pro zobrazování číslic vhodnější.

Z obrázků 8, 9 a 10 je jasně vidět, že výsledný obrázek je výrazně lepší pokud je změřeno více bodů. Tato vlastnost je důležitá, protože je to jedna z hlavních věcí, které má úloha ukazovat.

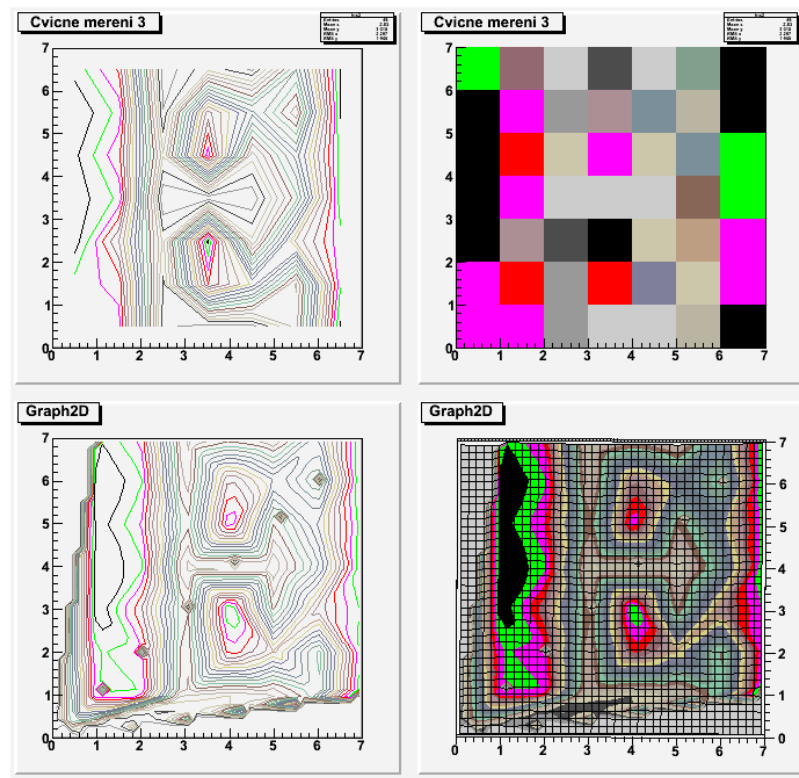
Po dokončení simulování jsem provedl tři cvičná měření pro různě silné vrstvy. Protože se první nápad s číslicemi mezi dvěma vrstvami alobalu ukázal jako ne moc praktický, použil jsem pro měření jako podložku hladší část obalu na CD, na který jsem dal neprůhledný papír, na něj dal měřenou číslici a přes ni neprůhlednou fólii. Velikost obalu na CD však ovlivňuje možnou velikost proměřované plochy, na kterou je možno dát zkoumaný nápis. Velikost použitelné části obalu na CD je 12×12 cm. K měření jsem použil scintilátor o rozměru 1×1 centimetr a zářič, který měl 1.1.2008 aktivitu 57,34 kBq. Vzdálenost mezi zářičem a scintilátorem byla 3 cm a doba měření jednoho bodu byla 5 s. Na následujících obrázcích jsou získané výsledky ukázané různými způsoby zobrazení. V horní části obrázků jsou to histogramy a v dolní části jsou to grafy. V prvním cvičném měření je měřeno 56 bodů a ve druhých dvou cvičných měřeních je to 49 bodů. Jednotlivé měřené body jsou velké 1 cm².



Obrázek 11: **Cvičné měření 1.** Tloušťka alobalu je 0,9 mm a tvar číslice je osmička.

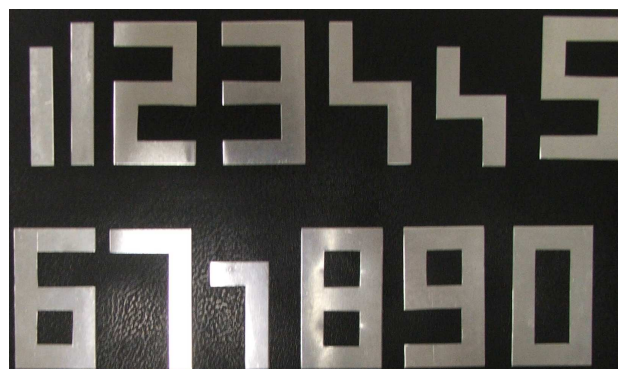


Obrázek 12: **Cvičné měření 2.** Tloušťka hliníkového plechu je 1 mm a jeho tvar je E.



Obrázek 13: **Cvičné měření 3.** Tloušťka alobalu je 1,3 mm a tvar čísla je osmička.

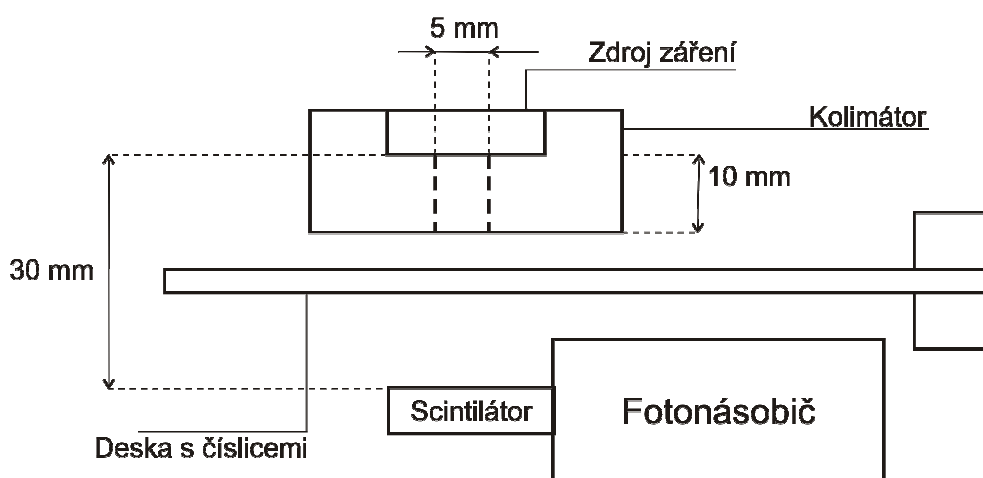
Tato cvičná měření ukazují, že použité nastavení je možné pro úlohu použít. Jako ukrytý nápis jsem zvolil digitální číslice z hliníkového plechu silného 1 mm. Číslice jsou vysoké 6 cm a široké 3,5 cm. Číslice s jednodušším tvarem jsou vytvořeny i v menší velikosti, protože je jednodušší je při skenování rozpoznat. Číslice vytvořené v menší velikosti jsou 0, 1, 4 a 7. 0, 1 a 4 jsou vysoké 5 cm a 7 je vysoká 4,5 cm. Šířky číslic 0, 4 a 7 jsou postupně 3,5, 3, 2,5 cm. Všechny číslice jsou na obrázku 14. Pro zobrazování jsem zvolil způsob ukázaný na obrázcích 11, 12 a 13 v pravé dolní části.



Obrázek 14: **Číslice z hliníkového plechu**

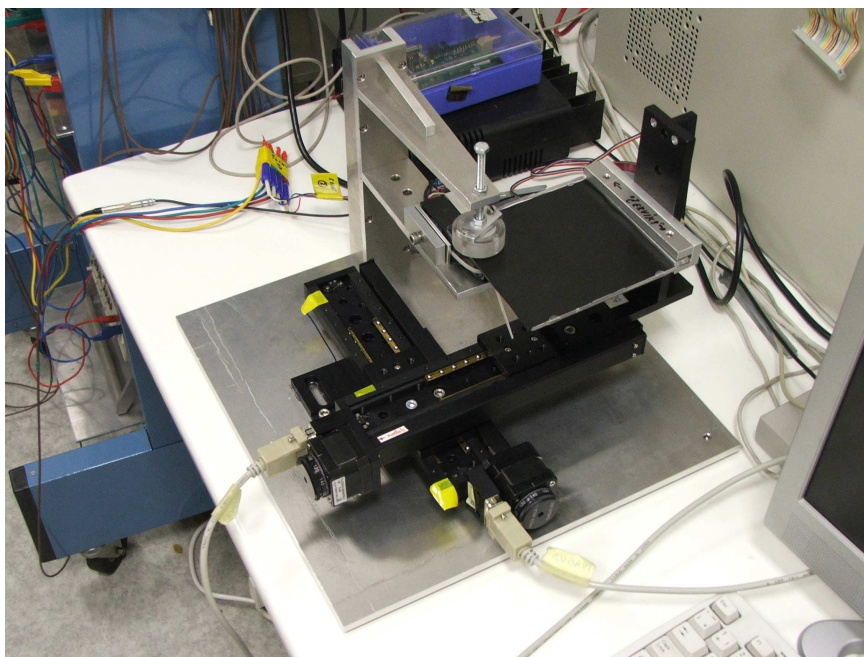
3.4. Popis aparatury

Měřicí aparatura se skládá ze zdroje radioaktivního záření β^- kruhového tvaru o průměru 5 mm, kterým je Stroncium ^{90}Sr . Abychom dostali alespoň trochu rovnoběžný svazek elektronů, je zdroj záření umístěn v kolimátoru o výšce 10 mm s otvorem o průměru 5 mm. Pro detekci záření je použit scintilační detektor. Scintilátor má plochu 1 cm^2 . Vzdálenost zdroje záření a scintilátoru lze měnit, ale je výhodné ji nastavit na nejmenší možnou hodnotu, což je 30 mm. Popsané části jsou znázorněné na obrázku 15.



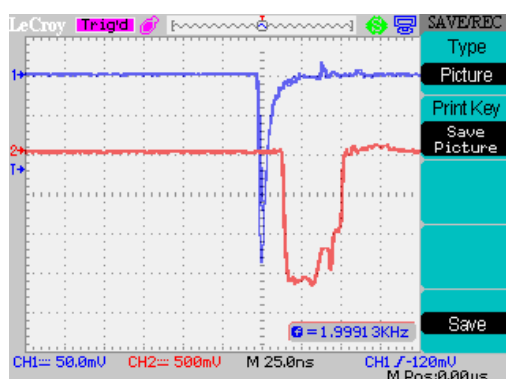
Obrázek 15: Schéma sestavení radioaktivního zdroje a scintilačního detektoru

K posouvání destičky jsou použity dva posuvné stolky, jejichž rozsah posuvu je 15 cm, což pro potřeby úlohy plně dostačuje. Celou sestavu drží pohromadě ocelová konstrukce viz obrázek 16. Abych mohl kolimátor přiblížit co nejbližší k proměřované destičce, musel jsem omezit x-ový rozměr proměřované oblasti na 10 cm. Kolimátor pak totiž naráží na uchycení destičky s čísly. Ypsilonový rozměr jsem nechal 12 cm.

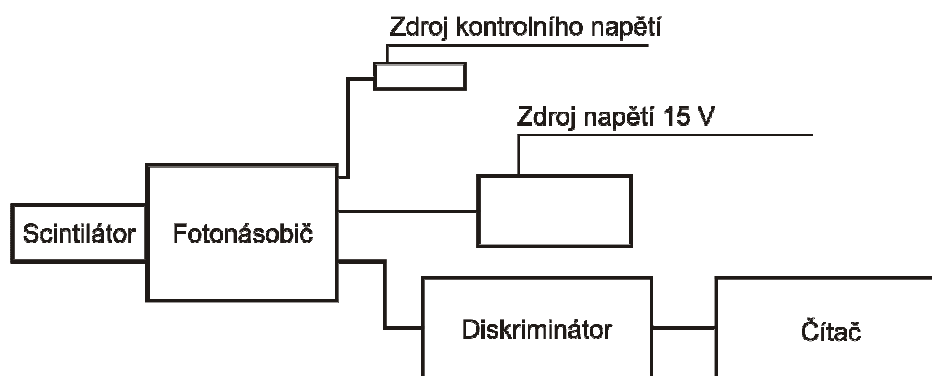


Obrázek 16: Celá aparatura

K fotonásobiči je připojen zdroj napětí nastavený přibližně na 15 V, který je potřeba k jeho napájení. Dále je k fotonásobiči připojeno řídicí napětí, kterým lze ovlivňovat urychlovací napětí ve fotonásobiči. Zvýšením řídicího napětí lze tedy zvýšit výtěžek. Výtěžek z fotonásobiče je lineární funkcí řídicího napětí. Řídicí napětí lze nastavit v rozmezí od 0,25 do 0,90 V. Protože v našem experimentu je potřeba mít co největší výtěžek, je potřeba nastavit řídicí napětí na hodnotu 0,90 V. Signály z fotonásobiče jsou vedeny do diskriminátoru. Diskriminátor porovnává absolutní hodnotu amplitudy přicházejících signálů s předem nastavenou hodnotou napětí (prahovým napětím). Pokud je absolutní hodnota amplitudy přicházejícího signálu větší než prahové napětí, vyšle diskriminátor logický signál. Tvary signálů z fotonásobiče a z diskriminátoru jsou na obrázku 17. Pokud je přicházející signál menší než prahová hodnota, diskriminátor již nic nevysílá. Diskriminátor se tedy především využívá na odstranění nechtěných malých signálů a šumu. Z diskriminátoru jdou signály do čítače částic Dual Counter/Timer Model 512 od firmy Canberra Nuclear. Tento čítač částic tedy sbírá a zobrazuje počet signálů z diskriminátoru, který odpovídá počtu detekovaných částic ve scintilátoru. Na používaném čítači lze nastavit čas, po který se mají sbírat signály a po jehož uplynutí se měření zastaví. Druhou mnou používanou funkcí je automatické opakované měření po určitý nastavený čas, které se zastaví po proběhnutí předem daného počtu měření nebo po ručním zastavení. Celé zapojení je na obrázku 18. [3]



Obrázek 17: Tvary signálů z fotonásobiče a z diskriminátoru. Signál z fotonásobiče je modrý a z diskriminátoru červený.

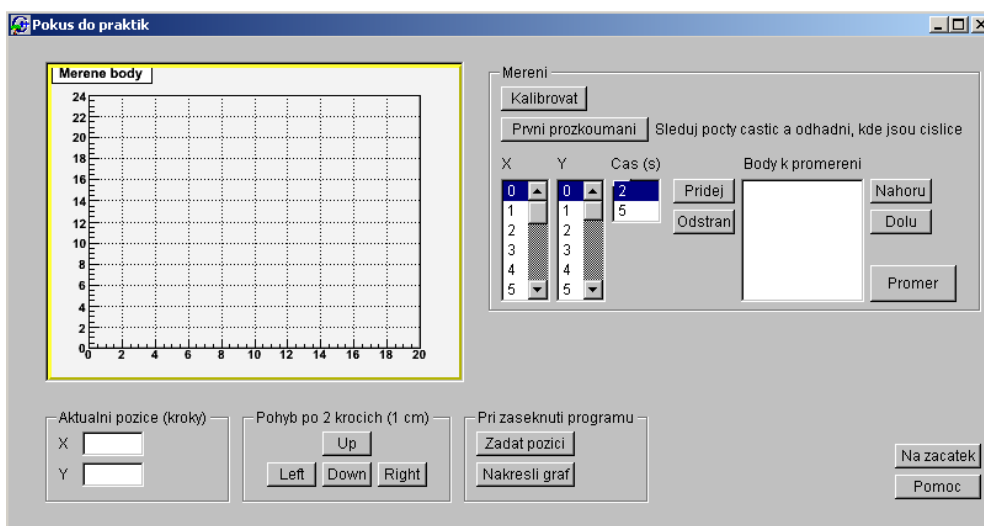


Obrázek 18: Schéma zapojení jednotlivých měřících modulů

3.5. Postup tvorby programu a popis jednotlivých funkcí

Při sestavování programu jsem použil odladěné programové části na čtení čítače a ovládání stolků, které připravili Peter Kodyš, Pavel Novotný, Pavel Jiroušek a Jan Brandejs. Mým hlavním úkolem pak bylo tyto jednotlivé části propojit a uzpůsobit potřebám úlohy. Dále bylo potřeba vytvořit grafické rozhraní a vymyslet způsob zobrazování naměřených výsledků, se kterým mi pomáhal Peter Kvasnička.

Grafický vzhled okna jsem vytvořil pomocí nástroje GuiBuilder, který je součástí programu ROOT. Vzhled okna programu je na obrázku 19.



Obrázek 19: Okno programu

Celý další postup pak spočíval v připojování jednotlivých funkcí potřebných k měření k jednotlivým tlačítkům grafického rozhraní.

První důležitou funkcí je kalibrace, po jejíž provedení program ví, že je na počátku souřadnic, od kterého se pak odvíjí souřadnicová soustava. Přesun na zadané souřadnice pak probíhá tak, že se provede rozdíl mezi souřadnicemi, kam se chceme dostat a stávajícími souřadnicemi. O vzniklý rozdíl se pak posouvají stolky.

Další funkcí mělo být hrubé proskenování měřené plochy, například po dvou nebo třech centimetrech, které by naznačilo kde jsou hledané číslice. Toto první proskenování by však zabralo hodně času, proto jsem zvolil následující postup. Z celé destičky projede detektor se zářičem jen tři vodorovné řádky vzdálené od sebe tři centimetry. Při tomto projíždění měří čítač částic souvisle a z displeje lze odečítat hodnoty každou sekundu. Studenti tedy mají sledovat naměřené hodnoty a mohou odhadnout polohu číslic. Tento způsob je rychlejší než původně zamýšlené hrubé proskenování.

Pro zadávání souřadnic jsem napřed vytvořil dvě políčka pro zadávání. Tímto způsobem by však šlo měřit jen po jednom bodu. Aby šlo zadat více bodů najednou, vytvořil jsem výběrová políčka a funkci, která přidává označené souřadnice do seznamu. Políčka původně určená pro zadávání souřadnic k proměření nyní slouží pro zobrazení souřadnic, na kterých se detektor nachází po doměření. Kromě přidávání jednotlivých bodů k proměření do seznamu jsou ještě tyto body zobrazovány v histogramu, aby bylo vidět, které části plochy jsou již proměřeny. V tomto histogramu je také vidět, kdybychom chtěli omylem měřit bod, který již

máme, takový bod je možné ze seznamu odstranit. Jednotlivé body v seznamu lze také prohazovat, kdybychom chtěli změnit pořadí jejich proměřování. Změna pořadí jednotlivých bodů může být třeba pro zefektivnění projíždění daného místa, aby se posuvné stolky zbytečně nevracely, to je dobré pro ušetření času. Jednotlivé body v histogramu se měli zobrazovat až při proměřování daného bodu, ale často se stávalo, že se jednotlivé body zobrazily až po proměření všech bodů ze seznamu. Tento problém se mi nepodařilo vyřešit, a proto jsem tedy nechal body do histogramu přidávat ještě před měřením. Ke každému bodu lze pomocí třetího výběrového políčka zvolit délku proměření. Z vytvořeného seznamu, kam lze přidat libovolné množství bodů, funkce proměř přečte jednotlivé souřadnice a čas, spustí posuv stolků, po jejich dojetí spustí čítač na danou dobu. Po naměření počtu částic jsou do textového souboru data.txt zapsány souřadnice měřeného bodu a číslo $150 - \frac{N}{t}$, kde N je počet naměřených částic a t je délka měření jednoho bodu.

Vznikající soubor je tedy po každém proměřeném bodu doplňován. Po zapsání souřadnic a počtu částic je textový soubor přečten a hodnoty jsou zakresleny do vznikajícího grafu, který je vždy před každou změnou smazán. Tento postup tvorby výsledného grafu jsem zvolil poté co jsem nedokázal po každém přidání bodu graf aktualizovat. Zápis naměřených hodnot do grafu je výhodný také proto, že je lze vždy znovu použít. To se vyplatí například při zaseknutí programu při měření. Pokud dojde k zaseknutí a musí se program vypnout, lze se podívat do souboru na poslední změřené souřadnice a ty zadat do programu, kvůli znovu nastavení souřadnic. Pak už jen stačí znovu zobrazit graf a pokračovat v proměřování. Protože při dalším měření by se hodnoty do souboru data.txt pouze přidávaly, bylo potřeba udělat místo pro nové hodnoty a na čas chovat ty z předešlého měření. Práce se soubory je tedy vyřešena tak, že na začátku měření je smazán nejstarší soubor, soubor data.txt z minulého měření je přejmenován na zaloha.txt a při měření vzniká nový soubor data.txt.

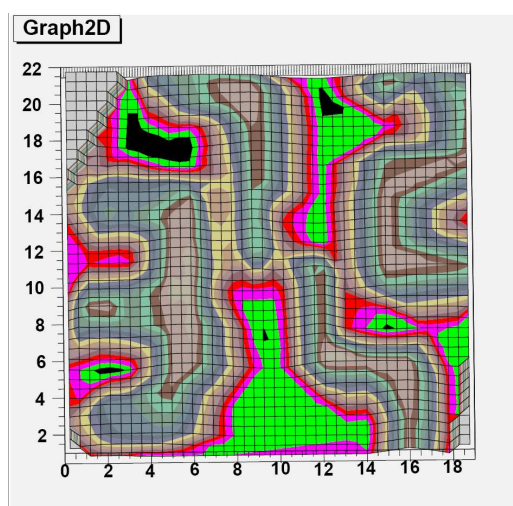
Po skončení měření je vhodné s detektorem dojet opět na počátek souřadnicové soustavy, odkud bude začínat nové měření, protože ještě funguje kalibrace, k tomu je také připravena jedna funkce. Pokud se program vypne dříve, musí se na počátek dojet pomocí směrových tlačítek, které posouvají stolkem o jeden krok, což je půl centimetru.

Poslední funkcí je možnost otevřít textový soubor obsahující doporučený postup měření

3.6. Výsledky měření

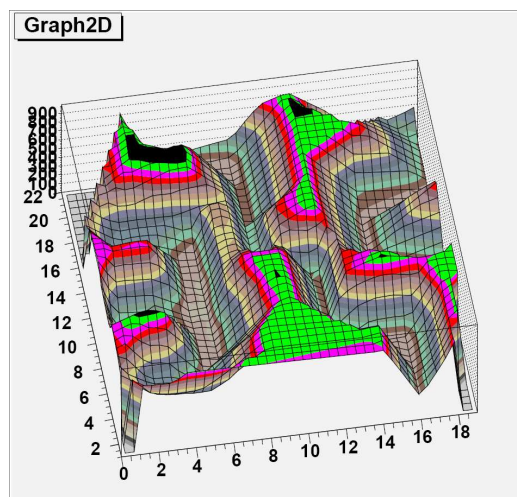
Po dokončení příprav a dokončení aparatury jsem provedl několik kompletních měření. Konečná proměřovaná plocha má rozměry 10×12 cm a jsou na ní čtyři číslice z 1 mm silného hliníkového plechu. Celá plocha je rozdělena na půl centimetrové dílky. Celkový počet bodů, které je možné proměřovat je 480. Každý bod je možné měřit buď 2 nebo 5 s. Získané výsledky jsou v následujících odstavcích.

Při prvním měření jsem proměřil 158 bodů po pěti sekundách. Celé měření trvalo asi 75 minut. Získaný výsledek je na obrázku 20.



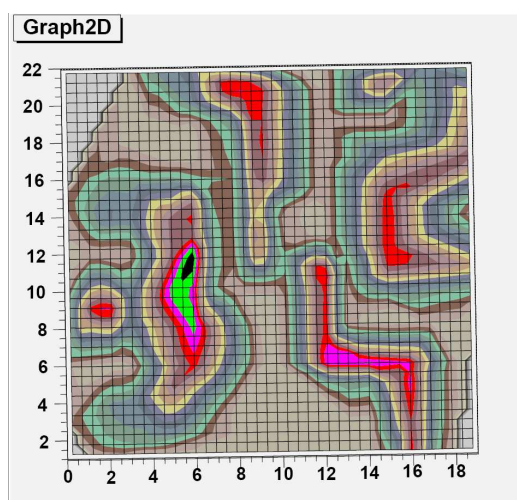
Obrázek 20: Výsledek prvního měření 1. Pohled shora

Při použitím způsobu zobrazení, kdy na místě číslic jsou malé hodnoty a na místech kde číslice nejsou jsou velké hodnoty, jsou číslice trochu zakryté, jak je vidět z obrázku 21.

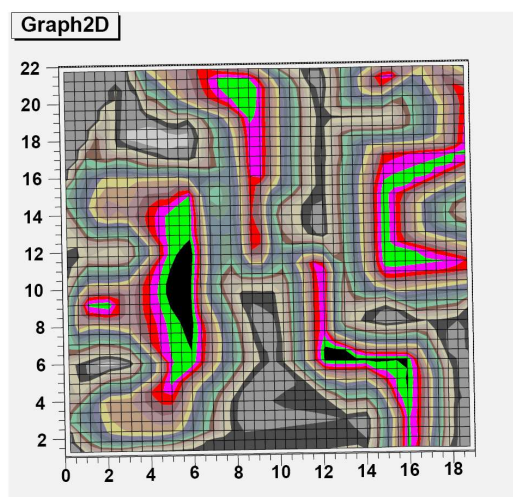


Obrázek 21: Výsledek prvního měření 2. Tak jak vzniká bez natáčení

Z tohoto důvodu jsem zkusil místo zobrazování přímo naměřených počtů částic zobrazovat jejich převrácené hodnoty viz obrázek 22. Na obrázku 23 je zobrazováno číslo $c - N$, kde c je přibližný počet částic naměřených mimo místa s číslicemi a N je naměřený počet částic.



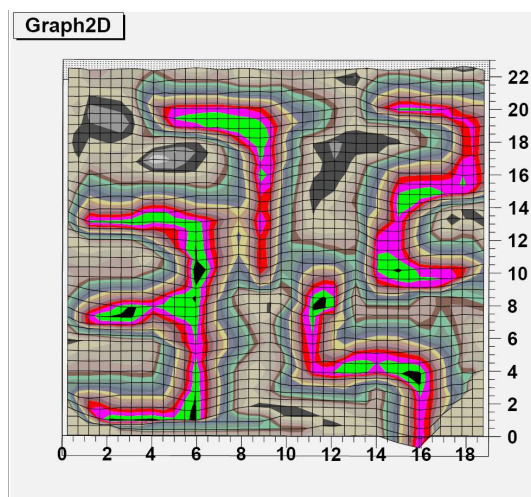
Obrázek 22: Výsledek prvního měření 3. Zobrazeny jsou převrácené hodnoty.



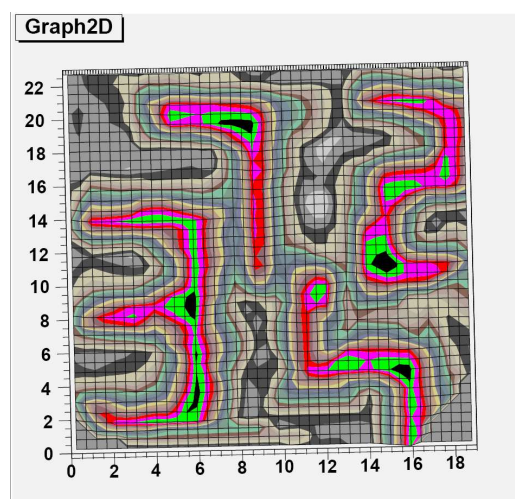
Obrázek 23: **Výsledek prvního měření 4.** Zobrazeny jsou hodnoty $850 - N$.

V prvním měření byly pro zobrazení použity počty částic získané během celých pěti sekund měření každého bodu. Protože je v úloze možné měřit různé body různě dlouho, je lepší zobrazovat vždy hodnoty vydělené délkou měření. V dalších měřeních jsou tedy zobrazovány počty částic za jednu sekundu. Pro všechna další měření jsem zvolil způsob zobrazování jako je na obrázku 23, protože se mi zdá ze všech zkoušených způsobů nejlepší.

Při dalším měření jsem proměřil 254 bodů po dvou sekundách. Celé měření trvalo 85 minut. Získaný výsledek je na obrázku 24 a v příloze A je sled obrázků získaných v intervalech po 10 minutách. Pro porovnání jsem stejné body proměřil i po pěti sekundách viz obrázek 25. Měřením pro různé doby proměřování jednoho políčka jsem chtěl ověřit, že rozptyl měřených počtů částic, vzniklý statistickým charakterem radioaktivního rozpadu, bude menší pro větší počty detekovaných částic.



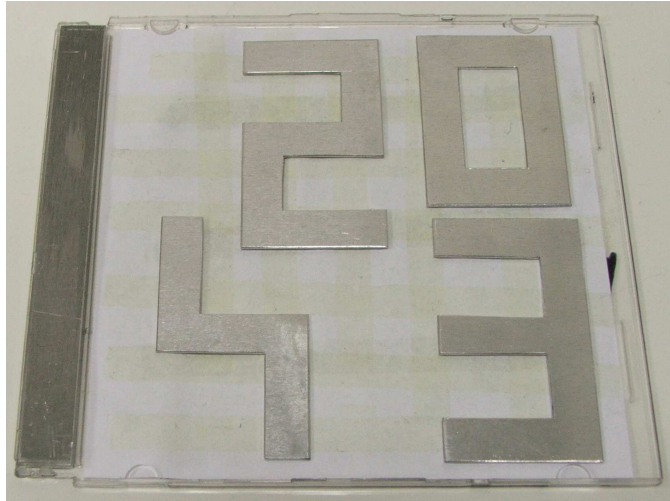
Obrázek 24: Výsledek druhého měření pro 2 sekundy



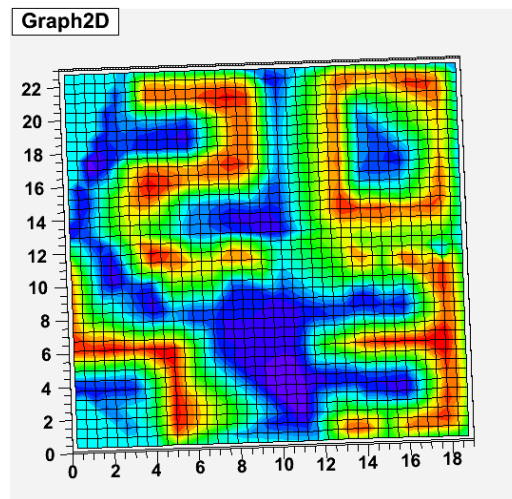
Obrázek 25: Výsledek druhého měření pro 5 sekund

Z obrázků 24 a 25 je vidět, že rozdíl mezi měřeními po dvou nebo pěti sekundách není vůbec patrný a tedy tímto způsobem nelze ověřovat zmenšování rozptylu naměřených počtů částic v závislosti na délce měření jednotlivých políček. Velmi podobný vzhled výsledků měření pro dvě a pět sekund je pravděpodobně způsoben tím, že je pro zobrazování zvolen graf, který interpoluje mezi měřenými body a tak celý výsledný obrázek vyhlazuje.

Při posledním měření jsem již jen zkusil jiné barevné odstíny při zobrazování. Měřil jsem 231 bodů po dvou sekundách. Celé měření trvalo 78 minut. Na obrázku 26 jsou měřené číslice a na obrázku 27 je získaný výsledek.



Obrázek 26: Měřené číslice



Obrázek 27: Výsledek třetího měření

Závěr

V rámci této diplomové práce byla vytvořena úloha pro středoškolské praktikum zaměřená na zkoumání vlastností radioaktivního záření a přiblížení jednoho ze způsobů jeho využití. Byla zhotovena měřicí aparatura, byl zhotoven ovládací a zobrazovací program a byly nalezeny co nejlepší experimentální podmínky pro řešení úlohy.

Dále bylo vypsáno několik dosavadních experimentů prováděných ve středních školách.

V teoretické části byly shrnuty základní poznatky o radioaktivním záření a některých způsobech jeho využití.

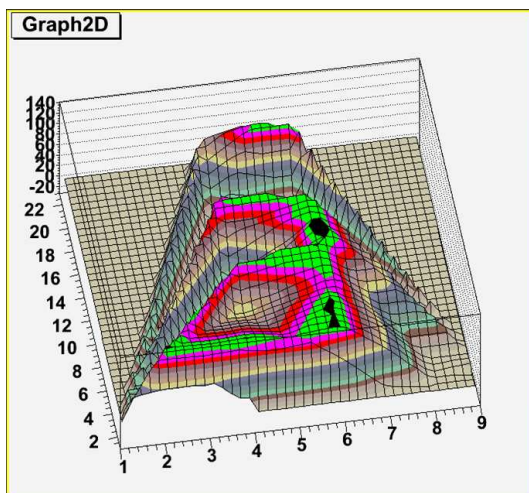
Seznam obrázků

Obrázek 1: Schéma scintilačního detektoru	7
Obrázek 2: Schéma fotonásobiče	8
Obrázek 3: Pozitronová emisní tomografie PET	11
Obrázek 4: Náčrtek zářiče a detektoru	15
Obrázek 5: Energetické spektrum vyzařované stronciem ^{90}Sr	16
Obrázek 6: Simulace počtů částic prošlých silnější a slabší vrstvou alobalu	19
Obrázek 7: Simulace zobrazení písmen J, O a K pomocí histogramu	19
Obrázek 8: Porovnání vzhledu pro různý počet proměřených bodů 1	20
Obrázek 9: Porovnání vzhledu pro různý počet proměřených bodů 2	20
Obrázek 10: Porovnání vzhledu pro různý počet proměřených bodů 3	21
Obrázek 11: Cvičné měření 1	22
Obrázek 12: Cvičné měření 2	22
Obrázek 13: Cvičné měření 3	23
Obrázek 14: Číslice z hliníkového plechu	24
Obrázek 15: Schéma sestavení radioaktivního zdroje a scintilačního detektoru	24
Obrázek 16: Celá aparatura	25
Obrázek 17: Tvary signálů z fotonásobiče a z diskriminátoru	26
Obrázek 18: Schéma zapojení jednotlivých měřících modulů	26
Obrázek 19: Okno programu	27
Obrázek 20: Výsledek prvního měření 1	29
Obrázek 21: Výsledek prvního měření 2	30
Obrázek 22: Výsledek prvního měření 3	30
Obrázek 23: Výsledek prvního měření 4	31
Obrázek 24: Výsledek druhého měření pro 2 sekundy	32
Obrázek 25: Výsledek druhého měření pro 5 sekund	32
Obrázek 26: Měřené číslice	33
Obrázek 27: Výsledek třetího měření	33

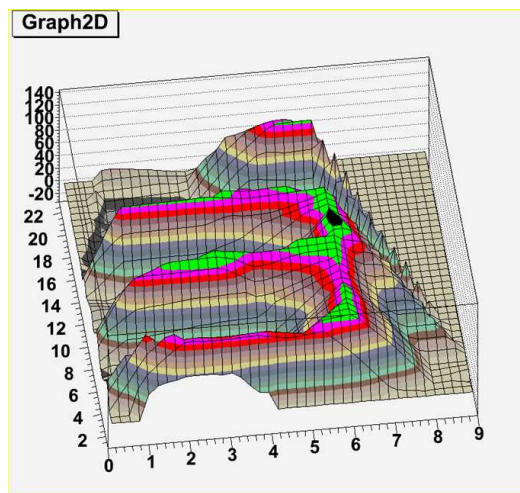
Literatura

- [1] Úlehla I., Suk M., Trka Z. (1990): *Atomy, jádra, částice*. Academia, Praha.
- [2] Reichl J., Všetická M. (2006 – 2011): *Encyklopedie fyziky*;
<http://fyzika.jreichl.com/index.php?sekce=browse&page=796>; [cit. 26.6.2011]
- [3] Jindra T. (2010): *Development of semiconductor detectors for high energy physics experiments*. UK MFF, bakalářská práce, Praha, 2010
- [4] Navrátil L., Rosina J. a kol. (2005): *Medicínská biofyzika*. Grada Publishing, a.s., Praha.
- [5] Ústav fyzikálního inženýrství – VUT Brno (2003): *Statistické vlastnosti radioaktivního zářiče*;
<http://physics.fme.vutbr.cz/~mcerny/vyuka/labiny/10a.pdf>; [cit. 30.6.2011]
- [6] Institut fyziky VŠB – TU Ostrava (2007): *Měření parametrů jaderného záření*;
http://if.vsb.cz/Veda/foto1/jaderne_zareni.pdf; [cit. 30.6.2011]
- [7] The ROOT Team (1995 – 2011): *Downloads*;
<http://root.cern.ch/drupal/content/downloading-root>; [cit. 10.11.2010]
- [8] Reichl J., Všetická M. (2006 – 2011): *Encyklopedie fyziky*;
<http://fyzika.jreichl.com/index.php?sekce=browse&page=747> [cit. 1.8.2011]
- [9] Reichl J., Všetická M. (2006 – 2011): *Encyklopedie fyziky*;
<http://fyzika.jreichl.com/index.php?sekce=browse&page=854> [cit. 1.8.2011]
- [10] Hrubý V.: *Absorpce beta záření*;
<http://www.praktikum.brejlovec.net/ref/ref-408-Eliska.pdf>; [cit. 20.10.2010]
- [11] Wikipedia (2011): *Pozitronová emisní tomografie*;
http://cs.wikipedia.org/wiki/Pozitronov%C3%A1_emisn%C3%AD_tomografie;
[cit. 30.6.2011]
- [12] Štoll I. (2002): *Fyzika mikrosvěta*. Prometheus, Praha.
- [13] Svoboda E. a kol. (2001): *Pokusy z fyziky na střední škole*. Prométheus, Praha.
- [14] Jonášová L. (2005): *Fyzika mikrosvěta na vysoké a střední škole*;
http://is.muni.cz/th/53369/prif_m/Diplomka.pdf; [cit. 28.7.2011]
- [15] Exploratorium fyziky (2011): *Radioaktivita a ochrana před zářením*;
<http://mojewiki.cz/exploratoriumfyziky/doku.php>; [cit. 29.6.2011]
- [16] PASCO (2011): *Pokusy s radioaktivitou*;
http://www.pasco.cz/images/stories/ceniky%20plakaty/PASCOpro_atom.pdf;
[cit. 29.6.2011]

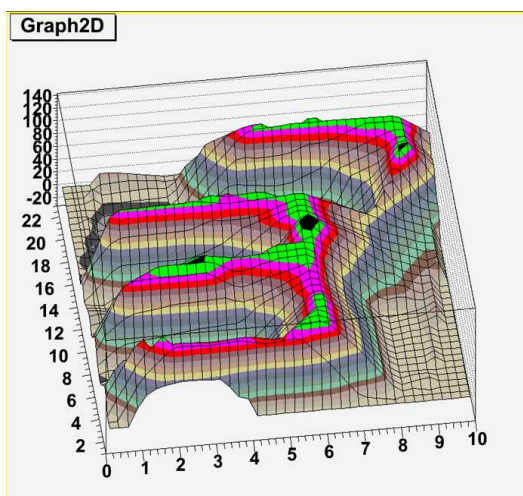
Příloha A: Zachycení průběhu měření v intervalech po 10 minutách



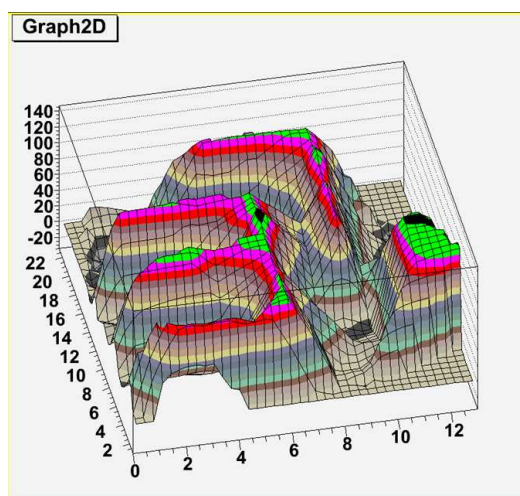
Obrázek A1: Měření po 10 minutách



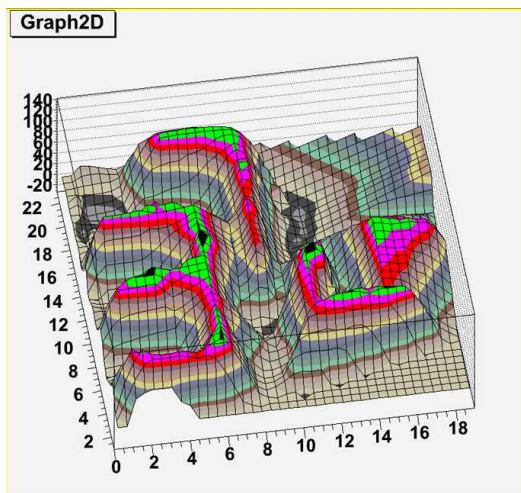
Obrázek A2: Měření po 20 minutách



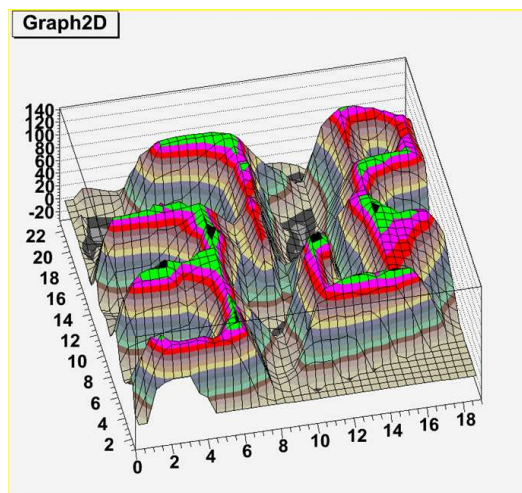
Obrázek A3: Měření po 30 minutách



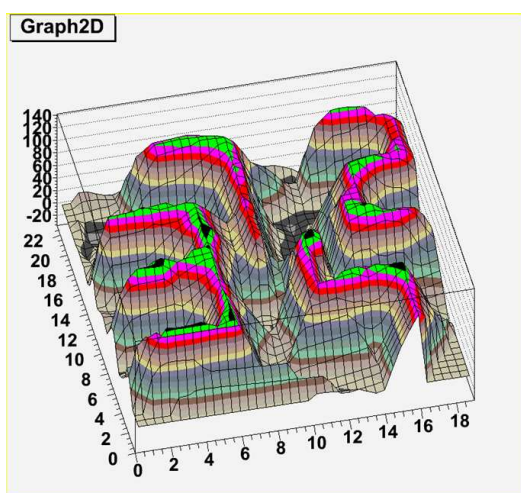
Obrázek A4: Měření po 40 minutách



Obrázek A5: Měření po 50 minutách



Obrázek A6: Měření po 60 minutách



Obrázek A7: Měření po 70 minutách

Příloha B: Text k úloze do SŠ praktika

V této úloze se budeme zabývat radioaktivním zářením, jeho pohlcováním v látce, způsobem jeho detekce a jeho využitím k zobrazování.

Radioaktivní záření je proud hmotných částic (elektronů, protonů, jader helia a dalších) nebo fotonů. Mezi radioaktivní záření patří například proud částic dopadající na Zemi z vesmíru nebo svazek částic urychlovaný v urychlovačích. Do radioaktivního záření zařazujeme také brzdné rentgenové záření, které vzniká při průletu elektronů nějakým materiálem. Radioaktivní záření vzniká také při rozpadu nestabilních atomových jader. Podle toho, jaké částice jsou při rozpadech vyzařovány, rozlišujeme několik druhů rozpadů. Při rozpadu α jsou vyzařována jádra helia, při β^+ jsou to pozitrony, při β^- elektrony a při γ fotony. V naší úloze je použito záření β^- . Záření β^- vzniká při přeměně neutronu na proton. Při této přeměně je vyzářen elektron a elektronové antineutrino. Rozpad β^- nastává u atomových jader, které mají přebytek neutronů. Při tomto rozpadu tedy vznikají nová jádra, která mají méně neutronů a více protonů.

Pokud dáme proudu elektronů do cesty nějaký materiál jako překážku, budou elektrony při průchodu danou překážkou ztrácet energii a zpomalovat se. K tomu dojde proto že elektrony ze svazku předávají část své energie elektronům v atomech látky a zvyšují jim tak jejich energii (dochází k excitaci) nebo dokonce dojde k odtržení elektronu z atomu (atomy se ionizují). K dalším ztrátám energie nalétávajících elektronů dochází proto, že se vlivem Coulombovské síly odpuzují s elektrony v obalech atomů, tím se brzdí a vyzařují takzvané brzdné rentgenové záření. Elektrony o vysokých energiích ztrácejí svou energii především brzdným zářením a u elektronů s nízkou energií dochází ke ztrátám energie hlavně ionizací. Pokud je látka, kterou záření prochází, dostatečně silná neprojde přes ní žádné záření. Slabší překážka záření pouze utlumí.

Díky schopnosti elektronů záření excitovat elektrony atomů v látce lze tyto elektrony zaznamenat pomocí takzvaných scintilačních detektorů. Pokud dopadne svazek elektronů záření na scintilátor, což bývá například nějaký krystal nebo skleněný materiál, dojde k jeho excitaci a poté ke slabému záblesku světla. Světlo

z tohoto záblesku je pak světlovodem přivedeno do fotonásobiče, kde jsou světelné signály převedeny na signály elektrické, které jsou zaznamenávány.

Využití radioaktivního záření k zobrazování

Radioaktivní záření se hojně používá v medicíně k diagnostice a terapii. Podívejme se na jeden ze způsobů zobrazování orgánů. Pacientovi je podána radioaktivní látka vyzařující většinou záření γ , která se shromažďuje v orgánu, který chceme zobrazovat. K zobrazování se využívá takzvaná scintilační kamera (gama-kamera). K vytvoření obrazu se používá velkoplošný stacionární scintilační detektor. Plocha scintilátoru odpovídá ploše, kterou lze najednou zobrazit. Za krystalem scintilátoru je umístěno velké množství fotonásobičů, které tvoří souřadnicovou síť. Polohová rozlišovací schopnost gama-kamery závisí na počtu fotonásobičů (čím je jich více tím lepší je rozlišovací schopnost). Pokud ve scintilátoru dojde k záblesku, získáme nějaký signál ze všech fotonásobičů. Tyto signály se však liší svou intenzitou podle toho, jak daleko byl daný fotonásobič od místa záblesku. Dva nejsilnější signály určují souřadnice místa záblesku. Získaný bod se vykreslí na obrazovku osciloskopu. Měření trvá tak dlouho dokud se na osciloskopu nezobrazí dobře prokreslený obraz vyšetřované oblasti (dokud nemáme naměřený dostatečný počet bodů). Při tomto měření je důležité správné nastavení polohy detektoru tak, aby docházelo k maximální registraci scintilací ve scintilátoru. K tomu aby byl detektor zaměřen jen na vyšetřovaný orgán se používá olověný kolimátor, který má stejný tvar jako krystal scintilátoru a je tlustý několik cm. V kolimátoru je vytvořený systém otvorů, díky kterým se do krystalu dostanou pouze gama paprsky v kolmém směru. Paprsky nalétávající šikmo jsou pohlceny materiálem kolimátoru. Výsledný obraz získaný gama-kamerou je tedy mozaikou složenou z částí odpovídajících otvorům v kolimátoru. V jednotlivých částech obrazu odpovídá počet zachycených záblesků síle záření. Výsledný obraz je zachycován buď na fotografický materiál nebo zaznamenáván do paměti počítače a uložen pro další analyzování. Tento způsob zobrazování je používán i na sledování rychlých dynamických procesů jako je například studie krevního oběhu. Pak je výstupem sada na sebe navazujících obrazů. Dnes lze získat i trojrozměrné obrazy orgánů. K tomu se využívají tomografické gama-kamery s jedním nebo třemi detektory, které se po

malých úhlových krocích otáčejí kolem pacienta a sbírají velké množství projekcí. Takto posbíraná data jsou pak v počítači zpracována pomocí různých matematických metod a je z nich vytvořen výsledný obraz. Tento způsob sběru dat a zobrazování se používá u jednofotonové emisní tomografie a u pozitronové emisní tomografie.



Obrázek B1: **Pozitronová emisní tomografie PET.** Obrázek byl získán z [11]

Radioaktivní záření se dále používá například při vyhledávání skrytých vad materiálů (defektoskopie) nebo při kontrole jakosti svarů. Rentgenové záření (patří taky mezi radioaktivní záření) se používá například pro zobrazování obsahu různých zavazadel nebo balíků, ke kontrole kamionů na hraničních přechodech nebo ke zjišťování nastavení vnitřních mechanismů u nástražných výbušných zařízení. Podobně jako rentgenové záření lze využívat i záření gama. Oproti rentgenu při využívání záření gama není potřeba výkonný zdroj elektrické energie a celá aparatura je menší. [4, 11]

Úkol

Pomocí proměrování jednotlivých částí destičky zjistíte, jaké čtyři digitální číslice jsou na ní schované. Číslice se snažte určit v co nejrychlejším čase s proměřením co nejmenšího počtu bodů.

Postup měření

Na začátku měření je potřeba zkontrolovat, zda je detektor na souřadnicích [0, 0]. Pokud je detektor na počátku, zadejte programu souřadnice [0, 0] pomocí tlačítka Kalibrovat. Pokud detektor není na počátku souřadnic, přesuňte ho pomocí směrových tlačítek Up, Down, Left, Right a pak dejte Kalibrovat.

Po kalibraci spusťte projíždění tří řádků, při kterém sledujte počty částic na čítači. Podle zjištěných počtů částic si rozmyslete, kde přibližně jsou umístěné hledané číslice. Po skončení tohoto projíždění začněte proměřovat jednotlivé body. V části měření si v prvním a druhém výběrovém políčku zadejte souřadnice bodu, který chcete proměřit, ve třetím výběrovém poli zvolte délku měření tohoto bodu. Tlačítkem Přidej přidáte zvolený bod do políčka Body k Proměření. Tímto způsobem lze přidávat libovolný počet bodů, které jsou zároveň vyznačovány v histogramu Měřené body. Pokud přidáte bod, který proměřovat nechcete, lze ho ze seznamu odstranit tlačítkem Odstraň. Pomocí tlačítek Nahoru a Dolu lze měnit pořadí bodů v seznamu. Snažte se aby měřené body byly co nejlépe srovnané vzhledem k dráze projíždění. Počet bodů v seznamu nelze měnit během měření, proto volte rozumný počet bodů, které budete proměřovat najednou (vhodné je volit počet od jednoho do deseti bodů). Tlačítkem Proměř spustíte proměrování bodů v seznamu. Během tohoto proměrování sledujte graf a rozmyšlejte, které další body měřit nebo jaké vidíte číslice. Postupně proměřujte další body dokud si nebudete jisti, jaké číslice jsou na destičce schovány.

Pokud se program při měření zasekne, je potřeba ho ukončit, znovu spustit a postupně kliknout na Kalibrovat, Zadat pozici a Nakresli graf. Poté je možno pokračovat v měření tam, kde jste přestali.

Seznam tlačítek

- Kalibrovat – pokud je detektor na souřadnicích [0, 0], toto tlačítko to řekne programu
- První prozkoumání – spustí projíždění plochy po třech řádcích
- Přidej – přidá označené souřadnice a čas do pole Body k proměření
- Odstraň – odstraní označený řádek z pole Body k proměření
- Nahoru, Dolu – složí ke změně pořadí bodů, které se budou proměřovat
- Proměř – spustí měření bodů v políčku Body k proměření

- Up, Down, Left, Right – slouží k posunu detektoru o dva kroky
- Zadat pozici – informuje program, že detektor je na souřadnicích zadaných v polích Aktuální pozice
- Nakresli graf – nakreslí graf z bodů v souboru Data.txt
- Na začátek – přesune detektor na souřadnice [0, 0]
- Pomoc – otevře soubor s postupem měření

Příloha C: Hlavní makro vytvořené k ovládání úlohy pro SŠ praktikum

```
// Mainframe macro generated from application: C:\cygwin\root\bin\root.exe  
// By ROOT version 5.20/00 on 2011-03-24 16:24:37
```

```
#include "TGDockableFrame.h"  
#include "TGMenu.h"  
#include "TGMdiDecorFrame.h"  
#include "TG3DLine.h"  
#include "TGMdiFrame.h"  
#include "TGMdiMainFrame.h"  
#include "TGuiBldHintsButton.h"  
#include "TGMdiMenu.h"  
#include "TGListBox.h"  
#include "TGNumberEntry.h"  
#include "TGScrollBar.h"  
#include "TGComboBox.h"  
#include "TGuiBldHintsEditor.h"  
#include "TGFrame.h"  
#include "TGFileDialog.h"  
#include "TGShutter.h"  
#include "TGButtonGroup.h"  
#include "TGCanvas.h"  
#include "TGFSContainer.h"  
#include "TGFontDialog.h"  
#include "TGuiBldEditor.h"  
#include "TGColorSelect.h"  
#include "TGTextEdit.h"
```



```
#include "TGButton.h"
#include "TGFSComboBox.h"
#include "TGLabel.h"
#include "TGView.h"
#include "TGMsgBox.h"
#include "TRootGuiBuilder.h"
#include "TGTab.h"
#include "TGListView.h"
#include "TGSplitter.h"
#include "TGStatusBar.h"
#include "TGToolTip.h"
#include "TGToolBar.h"
#include "TRootEmbeddedCanvas.h"
#include "TCanvas.h"
#include "TGuiBldDragManager.h"
#include "TGTextEntry.h"
#include "TGNumberEntry.h"
#include "TGuiBldNameFrame.h"
#include "TGListTree.h"
#include "TGuiBldGeometryFrame.h"
#include <TGraph2D.h>
#include <TRandom3.h>
#include <TStyle.h>
#include <TString.h>
#include <TMath.h>
#include "Riostream.h"
#include "sstream"
#include "move2.c"
#include "RQ_OBJECT.h"
#include "TH2.h"
```

```
#include "TH1.h"
#include <TRandom.h>
#include "canberra.cpp"
#include <string.h>

RESULTS vystup;

TGMainFrame *fMainFrame1862;
TRootEmbeddedCanvas *fRootEmbeddedCanvas515;
TGTextButton *fTextButton566;
TGTextButton *fTextButton1064;
TGTextButton *fTextButton1068;
TGTextButton *fTextButton1108;
TGTextButton *fTextButton1383;
TGTextButton *fTextButton1343;
TGTextButton *Start;
TGTextButton *Kalibrovat;
TGTextButton *Promer;
TGTextButton *Pridej;
TGTextButton *Zacatek;
TGTextButton *Pomoc;
TGListBox *fListBox1;
TGListBox *fListBox2;
TGListBox *fListBox3;
TGListBox *fListBox4;
TGComboBox *fComboBox560;
TGTextEntry *fTextEntry1197;
TGTextEntry *fTextEntry1226;
TGLabel *fLabel1255;
TGLabel *fLabel1258;
```

```
TGLabel *fLabel1;
TGLabel *fLabel2;
TGLabel *fLabel3;
TGLabel *NazevPoleProCas;
TGLabel *PopisStartu;
TGGroupFrame *PohybSipkami;
TGGroupFrame *MereniRamec;
TGGroupFrame *PriZaseknuti;
TCanvas *c1;
```

```
static int x_position; // x poloha stolku v krocich po 5 mm
static int y_position; // y poloha stolku v krocich po 5 mm
static int pocet_bodu=0; // pocet bodu v listboxu
```

```
//-----
// PLOTTER.
//-----
```

```
// Kresleni grafu.
// Je vhodne oddelit uzivatelske rozhrani od funkcnich casti programu.
// Proto nechceme kreslit graf primo v metodach objektu Functions, ty at slouzi
// pouze jako dispecer.
```

```
class Platno {
public:
```

```

/** Konstruktor.
 * Objektu musime rict, kam se bude graf kreslit.
 */
Platno(TCanvas *d): fCanvas(d), fHist(new TH2F("fHist", "Merene body", 20,0,20,24,0,24)) {;}

/** Destruktor.
 * Pri zruseni objektu je potreba zrusit graf, Canvas vlastni frame uzivatelskeho rozhrani.
 */
~Platno()
{ if (fHist) delete fHist; }

/** Pridat bod do grafu.
 * Cislo bodu si budeme strezit v objektu.
 */
int AddPoint1(double u, double v, double z){

    fHist->Fill(u, v, z);
}

void Draw1(){

    fCanvas->SetGrid();
    fCanvas->cd(); fHist->Draw("Col"); fCanvas->Update();

    TStyle* style = new TStyle();
    style->cd();

```

```

style->SetCanvasPreferGL(kTRUE);

//nastaveni palety barev
style->SetPalette(1); //std. paleta: cervena -> fialova

//vypne zobrazeni statistik & parametry fitu
style->SetOptStat(0);
style->SetOptFit(0);

//nastavi bila pozadi vseh ramecku + odstrani nektere osklivosti
style->SetHistFillColor(0);
style->SetCanvasColor(0);
style->SetCanvasBorderMode(0);
style->SetPadBorderMode(0);
style->SetStatColor(0);
style->SetFrameBorderMode(0);
style->SetTitleFillColor(0);

//nastavi ramecky kolem titulku a statistik
//(0=zadny ramecek,1=jednoduchy ramecek,2=ramecek se stinem)
style->SetStatBorderSize(1);
style->SetTitleBorderSize(1);

}

void Reset1()
{

```

```
        fHist->Reset();  
    }
```

```
private:
```

```
    TCanvas *fCanvas;  
    TH2F *fHist;
```

```
};
```

```
//-----  
// PLOTTER.  
//-----
```

```
// Kresleni grafu.  
// Je vhodné oddelit uzivatelske rozhrani od funkcnich casti programu.  
// Proto nechceme kreslit graf primo v metodach objektu Functions, ty at slouzi  
// pouze jako dispecer.
```

```
class Plotter {
```

```
public:
```

```
/** Konstruktor.
```

```
* Objektu musime rict, kam se bude graf kreslit.
```

```
*/
```

```
Plotter(TCanvas *c1): fNPoints(0), fCanvas(c1), fGraph(new TGraph2D()) {;}
```

```
/** Destruktor.
```

```
* Pri zruseni objektu je potreba zrusit graf, Canvas vlastni frame uzivatelskeho rozhrani.
```

```
*/
```

```
~Plotter()
```

```
{ if (fGraph) delete fGraph; }
```

```
/** Pridat bod do grafu.
```

```
* Cislo bodu si budeme strezit v objektu.
```

```
*/
```

```
int AddPoint(double u, double v, double z)
```

```
{
```

```
    fGraph->SetPoint(fNPoints++, u, v, z);
```

```
    return fNPoints;
```

```
}
```

```
void Draw()
```

```
{
```

```
    fCanvas->cd(); fGraph->Draw("Surf1"); fCanvas->Update();
```

```
}
```

```
void Reset()  
{  
    for (int iPoint = 0; iPoint < fNPoints; ++iPoint)  
        fGraph->RemovePoint(iPoint);  
    fNPoints = 0;  
}
```

```
private:
```

```
    int fNPoints;  
    TCanvas *fCanvas;  
    TGraph2D *fGraph;
```

```
};
```

```
//-----  
// MERICI PRISTROJ  
//-----
```

```
/** Trida reprezentujici merici pristroj.
```



```

*/

/*Predelava int na stream*/

string convertInt(int number)
{
    stringstream ss;//create a stringstream
    ss << number;//add number to the stream
    return ss.str();//return a string with the contents of the stream
}

class MericiPristroj {
public:

    /** Nastaveni plotteru, na ktery ma pristroj kreslit. */
    static void NastavPlotter(Plotter* plotter);
    static void NastavPlotter1(Platno* platno);

    static double start();
    static double zadatpozici();
    static void promer();
    static void kalibrovat();
    static void up();
    static void down();
    static void left();
    static void right();
    static void pridej();
}

```

```
static void naZacatek();
static void odstran();
static void posun_up();
static void posun_down();
static void nakresliGraf();
static void nakresliGraf1();
static void Reset();
static void Pomoc();
```

```
private:
```

```
static Plotter* fPlotter;
static Platno* fPlatno;
```

```
};
```

```
bool zkalibrovano=0;
int MericiPristroj::x_position = 0; // staticka promenna, nastavujeme na 0.
int MericiPristroj::y_position = 0; // -,-,-
```

```
Plotter* MericiPristroj::fPlotter = 0;
Platno* MericiPristroj::fPlatno = 0;
```

```
//graf
void MericiPristroj::NastavPlotter(Plotter* plotter){
```

```

    fPlotter = plotter;
}

//histogram
void MericiPristroj::NastavPlotter1(Platno* platno) {

    fPlatno = platno;
    fPlatno->Draw1();

}

void MericiPristroj::up(){

    if(zkalibrovano){
        if(y_position+1>23){

            move2(0,115-5*y_position);
            y_position = 23;
            fTextEntry1226->SetText("23");
        }
        else{

            move2(0,5);
            y_position=y_position+1;
            TString pozice(convertInt(y_position));
            fTextEntry1226->SetText(pozice.Data());
        }
    }
}

```

```

        else{
            move2(0,5);
        }
    }

void MericiPristroj::down(){

    if(zkalibrovano){
        if(y_position-1<0){

            move2(0,0-5*y_position);
            y_position=0;
            fTextEntry1226->SetText("0");

        }
        else{
            move2(0,-5);
            y_position=y_position-1;
            TString pozice(convertInt(y_position));
            fTextEntry1226->SetText(pozice.Data());
        }
    }
    else{
        move2(0,-5);
    }
}

void MericiPristroj::left(){

```

```

if(zkalibrovano){
    if(x_position-1<0){

        move2(0-5*x_position,0);
        x_position=0;
        fTextEntry1197->SetText("0");

    }
    else{

        move2(-5,0);
        x_position=x_position-1;
        TString pozice(convertInt(x_position));
        fTextEntry1197->SetText(pozice.Data());

    }
}
else{
    move2(-5,0);
}
}

```

```

void MericiPristroj::right(){

    if(zkalibrovano){
        if(x_position+1>19){

```

```

        move2(95-5*x_position,0);
        x_position=19;
        fTextEntry1197->SetText("19");
    }
    else{
        move2(5,0);
        x_position=x_position+1;
        TString pozice(convertInt(x_position));
        fTextEntry1197->SetText(pozice.Data());
    }
}
else{
    move2(5,0);
}
}

```

```

double MericiPristroj::zadatpozici(){
    TString s1(fTextEntry1197->GetText());
    TString s2(fTextEntry1226->GetText());

    x_position = s1.Atoi();
    y_position = s2.Atoi();
}

double MericiPristroj::start()

```

```

{
    if (zkalibrovano){

        //mereni po sekundach
        canberra_initialization(0, 9, 1.00, -0.25, -0.25, 1.00, 999);
        vystup = canberra_run();
        move2(0,30);
        move2(90,0);
        move2(0,30);
        move2(-90,0);
        move2(0,30);
        move2(90,0);

        x_position=18;
        y_position=18;

        fTextEntry1197->SetText("18");
        fTextEntry1226->SetText("18");

        canberra_exit();
    }
}

void MericiPristroj::nakresliGraf()
{

    MericiPristroj::Reset();
}

```

```

//cteni ze souboru

int a,b,c;

FILE *f;
f=fopen("data.txt", "r");

while(fscanf(f, "%d, %d, %d", &a, &b, &c)!=EOF) {

    fPlotter->AddPoint(a, b, c);

}

fPlotter->Draw();

fclose(f);
}

void MericiPristroj::nakresliGraf1()
{

    MericiPristroj::Reset();

    //cteni ze souboru

    int a,b,c;

    FILE *f;

```



```

f=fopen("data.txt", "r");

while(fscanf(f, "%d, %d, %d", &a, &b, &c)!=EOF) {

    fPlotter->AddPoint(a, b, c);
    fPlatno->AddPoint1(a, b, 30);

}

fPlotter->Draw();
fPlatno->Draw1();

fclose(f);
}

```

```

void MericiPristroj::promer(){

    if (zkalibrovano){

        int x_dif, y_dif, castic;
        float casticzasekundu;
        FILE *f;

        for(int i=0;i<pocet_bodu;i++)
        {
            //souradnice
            TGTextLBEntry *tee2 = fListBox4->GetEntry(i);

```

```

TString s(tee2->GetTitle());
TString x(strtok(s.Data(),""));
TString y(strtok(NULL,"-"));
TString t(strtok(NULL,"s"));

//jezdeni
x_dif = (x.Atoi() - x_position)*5;
y_dif = (y.Atoi() - y_position)*5;

move2(x_dif,y_dif);

x_position = x.Atoi();
y_position = y.Atoi();

//mereni
canberra_initialization(0, 1, t.Atof(), -0.25, -0.25, 1.00, 0);
vystup = canberra_run();
castic=vystup.cl;

                fTextEntry1197->SetText(x);
                fTextEntry1226->SetText(y);

casticzasekundu=castic/t.Atof();

//zapis dat do txt
f=fopen("data.txt","a");
fprintf(f,"%d, %d, %d \n", x_position, y_position, 150-casticzasekundu);

```

```

fclose(f);

canberra_exit();
MericiPristroj::nakresliGraf();
}

        for (int j=0;j<=pocet_bodu;j++){

                fListBox4->RemoveEntry(j);
                fListBox4->MoveResize(205,95,100,100);

        }
pocet_bodu=0;
}
}

void MericiPristroj::kalibrovat(){

MericiPristroj::Reset();

        if(!zkalibrovano){
                x_position=0;
                y_position=0;
                fTextEntry1197->SetText("0");
                fTextEntry1226->SetText("0");
                zkalibrovano=1;
        }
}

```

```

    gSystem->Exec("del zaloha.txt");
    gSystem->Exec("rename data.txt zaloha.txt");
}

void MericiPristroj::pridej(){

int x, y;

if(fListBox1->GetSelected() != -1 && fListBox2->GetSelected() != -1 && fListBox3->GetSelected() != -1
)
{

    TGTextLBEntry *tee1 = fListBox1->GetSelectedEntry();
    TGTextLBEntry *tee2 = fListBox2->GetSelectedEntry();
    TGTextLBEntry *tee35 = fListBox3->GetSelectedEntry();

    TString s1(tee1->GetTitle());
    TString s2(tee2->GetTitle());
    TString s35(tee35->GetTitle());

    s1.Append(", ");
    s1.Append(s2.Data());
    s1.Append(" - ");
    s1.Append(s35.Data());
    s1.Append("s");

    x=s1.Atoi();
    y=s2.Atoi();

```

```

        fPlatno->AddPoint1(x, y, 30);
fPlatno->Draw1();

        fListBox4->AddEntrySort(s1.Data(), pocet_bodu);
fListBox4->Select(pocet_bodu);
pocet_bodu++;

        fListBox4->MoveResize(205,95,100,100);
    }
}

//presune detektor na [0,0]

void MericiPristroj::naZacatek(){

    move2(-x_position*5, -y_position*5);

    x_position=0;
    y_position=0;

    fTextEntry1197->SetText("0");
    fTextEntry1226->SetText("0");

}

void MericiPristroj::odstran(){

int x_souradnice, y_souradnice;

```

```

TGTextLbEntry *tee11 = fListBox4->GetSelectedEntry();

        TString s(tee11->GetTitle());
        TString x(strtok(s.Data(),","));
        TString y(strtok(NULL,"-"));
        TString t(strtok(NULL,"s"));

        x_souradnice=x.Atoi();
        y_souradnice=y.Atoi();

        printf("%d, %d", x_souradnice, y_souradnice);

        fPlatno->AddPoint1(x_souradnice, y_souradnice, -30);
        fPlatno->Draw1();

if(fListBox4->GetSelected() != -1){

    int vybrano=fListBox4->GetSelected();

    if(vybrano!=pocet_bodu-1){

        for(int i = vybrano; i < pocet_bodu-1; i++){

            TGTextLbEntry *tee2 = fListBox4->GetEntry(i+1);
            TGTextLbEntry *tee3 = fListBox4->GetEntry(i);
            tee3->SetTitle(tee2->GetTitle());
        }
    }
}

```

```

        fListBox4->Select(vybrano);
    }

        fListBox4->RemoveEntry(pocet_bodu-1);
    pocet_bodu--;

        fListBox4->MoveResize(205,95,100,100);
    }

}

void MericiPristroj::posun_up(){
    if(fListBox4->GetSelected()!= -1){
        if(fListBox4->GetSelected()!=0)
        {
            TGTextLBEntry *tee2 = fListBox4->GetSelectedEntry();
            TString s(tee2->GetTitle());
            TGTextLBEntry *tee3 = fListBox4->GetEntry(fListBox4->GetSelected()-1);
            tee2->SetTitle(tee3->GetTitle());
            tee3->SetTitle(s.Data());
            fListBox4->MoveResize(205,95,100,100);
            fListBox4->Select(fListBox4->GetSelected()-1);
        }
    }
}

```

```

}
void MericiPristroj::posun_down(){
    if(fListBox4->GetSelected() != -1){
        if(fListBox4->GetSelected() != pocet_bodu-1)
        {
            TGTextLBEntry *tee2 = fListBox4->GetSelectedEntry();
            TString s(tee2->GetTitle());
            TGTextLBEntry *tee3 = fListBox4->GetEntry(fListBox4->GetSelected()+1);
            tee2->SetTitle(tee3->GetTitle());
            tee3->SetTitle(s.Data());
            fListBox4->MoveResize(205,95,100,100);
            fListBox4->Select(fListBox4->GetSelected()+1);
        }
    }
}

void MericiPristroj::Reset()
{
    fPlotter->Reset();
}

void MericiPristroj::pomoc()
{
    gSystem->Exec("start napoveda.doc");
}

class Functions{

```



```

        RQ_OBJECT("Functions")

private:

public:

    void Up();
    void Down();
    void Left();
    void Right();
    void Zadatpozici();
    void Start();
    void Kalibrovat();
    void Promer();
    void Pridej();
    void NaZacatek();
    void NakresliGraf();
    void NakresliGraf1();
    void Posun_Up();
    void Posun_Down();
    void Odstran();
    void Pomoc();

};

void Functions::Odstran()
{
    MericiPristroj::odstran();
}

```

```
void Functions::NakresliGraf()  
{  
    MericiPristroj::nakresliGraf();  
}  
  
void Functions::NakresliGraf1()  
{  
    MericiPristroj::nakresliGraf1();  
}  
  
void Functions::NaZacatek()  
{  
    MericiPristroj::naZacatek();  
}  
  
void Functions::Kalibrovat()  
{  
    MericiPristroj::kalibrovat();  
}  
  
void Functions::Up()  
{  
    MericiPristroj::up();  
}  
  
void Functions::Down()  
{  
    MericiPristroj::down();  
}
```

```
void Functions::Left()
{
    MericiPristroj::left();
}

void Functions::Right()
{
    MericiPristroj::right();
}

void Functions::Start()
{
    MericiPristroj::start();
}

void Functions::Zadatpozici()
{
    MericiPristroj::zadatpozici();
}

void Functions::Promer()
{
    MericiPristroj::promer();
}

void Functions::Pridej()
{
    MericiPristroj::pridej();
}
```

```

}

void Functions::Posun_Up()
{
    MericiPristroj::posun_up();
}

void Functions::Posun_Down()
{
    MericiPristroj::posun_down();
}

void Functions::Pomoc()
{
    MericiPristroj::pomoc();
}

void MyMainFrame()
{
    // hlavni ramec
    fMainFrame1862 = new TGMainFrame(gClient->GetRoot(),800,400,kMainFrame | kVerticalFrame);
    fMainFrame1862->SetLayoutBroken(kTRUE);
    fMainFrame1862->SetMWMHints(kMWMDecorAll, kMWMFuncAll, kMWMInputModeless);
    fMainFrame1862->Resize(800,400);

    // okno

```

```

fRootEmbeddedCanvas515 = new TRootEmbeddedCanvas ("RootEmbeddedCanvas515", fMainFrame1862, 340,
260);

    fMainFrame1862->AddFrame(fRootEmbeddedCanvas515,          new          TGLayoutHints(kLHintsExpandX |
kLHintsExpandY, 100, 100, 2, 2));
    fRootEmbeddedCanvas515->MoveResize(30,20,340,260);
    Int_t wfRootEmbeddedCanvas515 = fRootEmbeddedCanvas515->GetCanvasWindowId();

// Ted si uz muzeme vytvorit plotter a inicializovat merici pristroj.
c1 = new TCanvas("c1","Graf",0,0,550,550);
MericiPristroj::NastavPlotter(new Plotter(c1->GetCanvas()));

MericiPristroj::NastavPlotter1(new Platno(fRootEmbeddedCanvas515->GetCanvas()));

// Mereni - ramec
MereniRamec = new TGGroupFrame(fMainFrame1862,"Mereni");
MereniRamec->SetLayoutBroken(kTRUE);
MereniRamec->MoveResize(390,20,400,210);
fMainFrame1862->AddFrame(MereniRamec, new TGLayoutHints(kLHintsLeft | kLHintsTop,2,2,2,2));

//Tlacitko Kalibrovat

Kalibrovat = new TGTextButton(MereniRamec,"Kalibrovat");

```

```

Kalibrovat->SetTextJustify(36);
Kalibrovat->MoveResize(10,20,70,20);
Kalibrovat->Connect("Clicked()", "Functions", "MereniRamec", "Kalibrovat()");

//Tlacitko Start

Start = new TGTextButton(MereniRamec,"Prvni prozkoumani");
Start->SetTextJustify(36);
Start->MoveResize(10,45,120,20);
Start->Connect("Clicked()", "Functions", "MereniRamec", "Start()");

// Vzkaz u tlacitka prvni prozkoumani

PopisStartu = new TGLLabel(MereniRamec,"Sleduj pocty castic a odhadni, kde jsou cislice");
PopisStartu->SetTextJustify(36);
PopisStartu->MoveResize(135,45,250,20);

//popis listboxu 1
fLabel2 = new TGLLabel(MereniRamec,"X");
fLabel2->SetTextJustify(36);
fLabel2->MoveResize(10,75,10,15);

//ListBox x_position
fListBox1 = new TGLListBox(MereniRamec);
fListBox1->SetName("fListBox1");
fListBox1->AddEntrySort("0",0);
fListBox1->AddEntrySort("1",1);

```

```
fListBox1->AddEntrySort("2",2);
fListBox1->AddEntrySort("3",3);
fListBox1->AddEntrySort("4",4);
fListBox1->AddEntrySort("5",5);
fListBox1->AddEntrySort("6",6);
fListBox1->AddEntrySort("7",7);
fListBox1->AddEntrySort("8",8);
fListBox1->AddEntrySort("9",9);
fListBox1->AddEntrySort("10",10);
fListBox1->AddEntrySort("11",11);
fListBox1->AddEntrySort("12",12);
fListBox1->AddEntrySort("13",13);
fListBox1->AddEntrySort("14",14);
fListBox1->AddEntrySort("15",15);
fListBox1->AddEntrySort("16",16);
fListBox1->AddEntrySort("17",17);
fListBox1->AddEntrySort("18",18);
fListBox1->AddEntrySort("19",19);

fListBox1->Select(0);
fListBox1->MoveResize(10,95,40,100);

//popis listboxu 2
fLabel3 = new TGLLabel(MereniRamec,"Y");
fLabel3->SetTextJustify(36);
fLabel3->MoveResize(55,75,10,15);
```

```

        //ListBox2 y_position

        int FirstEntry = 0, LastEntry = 24;
        char tmp[20];
        fListBox2 = new TGLListBox(MereniRamec);
        for (int i = FirstEntry; i < LastEntry; i++) {
            sprintf(tmp, "%i", i);
            fListBox2->AddEntry(tmp, i);
        }

        fListBox2->Select(0);
        fListBox2->MoveResize(55,95,40,100);

// Popis vyberu casu
        NazevPoleProCas = new TGLLabel(MereniRamec, "Cas (s)");
        NazevPoleProCas->SetTextJustify(36);
        NazevPoleProCas->MoveResize(100,75,40,15);

// Pole pro vyber casu (delky mereni)

        fListBox3 = new TGLListBox(MereniRamec);
        fListBox3->SetName("fListBox3");

        fListBox3->AddEntrySort("2",0);
        fListBox3->AddEntrySort("5",1);
        fListBox3->Select(0);

```



```

fListBox3->MoveResize(100,95,40,50);

//Tlacitko Pridej

Pridej = new TGTextButton(MereniRamec,"Pridej");
Pridej->SetTextJustify(36);
Pridej->MoveResize(150,95,50,20);
Pridej->Connect("Clicked()", "Functions", "MereniRamec", "Pridej()");

    //Tlacitko Odstran

Odstran = new TGTextButton(MereniRamec,"Odstran");
Odstran->SetTextJustify(36);
Odstran->MoveResize(150,120,50,20);
Odstran->Connect("Clicked()", "Functions", "MereniRamec", "Odstran()");

// Popis merenych bodu
BodyPromereni = new TGLLabel(MereniRamec,"Body k promereni");
BodyPromereni->SetTextJustify(36);
BodyPromereni->MoveResize(205,75,100,15);

// Seznam bodů k promereni
fListBox4 = new TGLListBox(MereniRamec);
fListBox4->SetName("fListBox4");
fListBox4->MoveResize(205,95,100,100);

//Tlacitko Nahoru
fTextButton1234 = new TGTextButton(MereniRamec,"Nahoru");
fTextButton1234->SetTextJustify(36);

```

```

fTextButton1234->MoveResize(310,95,50,20);
fTextButton1234->Connect("Clicked()", "Functions", "MereniRamec", "Posun_Up()");

//Tlacitko Dolu

fTextButton54674 = new TGTextButton(MereniRamec,"Dolu");
fTextButton54674->SetTextJustify(36);
fTextButton54674->MoveResize(310,120,50,20);
fTextButton54674->Connect("Clicked()", "Functions", "MereniRamec", "Posun_Down()");

//Tlacitko Promer

Promer = new TGTextButton(MereniRamec,"Promer");
Promer->SetTextJustify(36);
Promer->MoveResize(310,165,70,30);
Promer->Connect("Clicked()", "Functions", "MereniRamec", "Promer()");

// Vlastnosti vkladacich policek
TGFont *ufont; // will reflect user font changes
ufont = gClient->GetFont("-*-helvetica-medium-r-*-*-*-12-*-*-*-*-*iso8859-1");

TGGC *uGC; // will reflect user GC changes
// graphics context changes
GCValues_t valEntry1197;
valEntry1197.fMask = kGCForeground | kGCBackground | kGCFillStyle | kGCFont |
kGCGraphicsExposures;
gClient->GetColorByName("#000000",valEntry1197.fForeground);
gClient->GetColorByName("#c0c0c0",valEntry1197.fBackground);

```

```

valEntry1197.fFillStyle = kFillSolid;
valEntry1197.fFont = ufont->GetFontHandle();
valEntry1197.fGraphicsExposures = kFALSE;
uGC = gClient->GetGC(&valEntry1197, kTRUE);

// Aktualni pozice - vnitri ramec
TGGroupFrame *fGroupFrame1152 = new TGGroupFrame(fMainFrame1862,"Aktualni pozice (kroky)");
fGroupFrame1152->SetLayoutBroken(kTRUE);
fGroupFrame1152->MoveResize(30,300,150,85);
fMainFrame1862->AddFrame(fGroupFrame1152, new TGLayoutHints(kLHintsLeft |
kLHintsTop,2,2,2,2));

//Popis policka
fLabel1255 = new TGLabel(fGroupFrame1152,"X");
fLabel1255->SetTextJustify(36);
fLabel1255->MoveResize(10,20,10,20);
fGroupFrame1152->AddFrame(fLabel1255, new TGLayoutHints(kLHintsLeft | kLHintsTop,2,2,2,2));

//Policko pro zadani textu
fTextEntry1197 = new TGTextEntry(fGroupFrame1152, new TGTextBuffer(15),-1,uGC-
>GetGC(),ufont->GetFontStruct(),kSunkenFrame | kDoubleBorder | kOwnBackground);
fTextEntry1197->SetMaxLength(10);
fTextEntry1197->SetAlignment(kTextLeft);
fTextEntry1197->MoveResize(30,20,50,20);

//popis druheho policka
fLabel1258 = new TGLabel(fGroupFrame1152,"Y");
fLabel1258->SetTextJustify(36);

```

```

fLabel1258->MoveResize(10,45,10,20);

//Druhe policko pro text
fTextEntry1226 = new TGTextEntry(fGroupFrame1152, new TGTextBuffer(15),-1,uGC-
>GetGC(),ufont->GetFontStruct(),kSunkenFrame | kDoubleBorder | kOwnBackground);
fTextEntry1226->SetMaxLength(10);
fTextEntry1226->SetAlignment(kTextLeft);
fTextEntry1226->MoveResize(30,45,50,20);

// Pohyb sipkami - ramec
PohybSipkami = new TGGroupFrame(fMainFrame1862,"Pohyb po 2 krocich (1 cm)");
PohybSipkami->SetLayoutBroken(kTRUE);
PohybSipkami->MoveResize(190,300,170,85);
fMainFrame1862->AddFrame(PohybSipkami, new TGLayoutHints(kLHintsLeft | kLHintsTop,2,2,2,2));

//Tlacitko Up

fTextButton566 = new TGTextButton(PohybSipkami,"Up");
fTextButton566->SetTextJustify(36); //36 pozice nápisu (asi uprostřed)
fTextButton566->Connect("Clicked()", "Functions", "PohybSipkami", "Up()");
fTextButton566->MoveResize(65,20,40,20);

//Tlacitko Right

fTextButton1064 = new TGTextButton(PohybSipkami,"Right");
fTextButton1064->SetTextJustify(36);
fTextButton1064->MoveResize(110,45,40,20);
fTextButton1064->Connect("Clicked()", "Functions", "PohybSipkami", "Right()");

```

```

//Tlacitko Left

fTextButton1068 = new TGTextButton(PohybSipkami,"Left");
fTextButton1068->SetTextJustify(36);
fTextButton1068->MoveResize(20,45,40,20);
fTextButton1068->Connect("Clicked()", "Functions", "PohybSipkami", "Left()");

//Tlacitko Down

fTextButton1108 = new TGTextButton(PohybSipkami,"Down");
fTextButton1108->SetTextJustify(36);
fTextButton1108->MoveResize(65,45,40,20);
fTextButton1108->Connect("Clicked()", "Functions", "PohybSipkami", "Down()");

// Pri zaseknuti programu - ramec
PriZaseknuti = new TGGroupFrame(fMainFrame1862,"Pri zaseknuti programu");
PriZaseknuti -> SetLayoutBroken(kTRUE);
PriZaseknuti->MoveResize(370,300,150,85);
fMainFrame1862->AddFrame(PriZaseknuti, new TGLayoutHints(kLHintsLeft | kLHintsTop,2,2,2,2));

// Tlacitko Zadat pozici
fTextButton1343 = new TGTextButton(PriZaseknuti,"Zadat pozici");
fTextButton1343->SetTextJustify(36);
fTextButton1343->MoveResize(10,20,80,20);
fTextButton1343->Connect("Clicked()", "Functions", "PriZaseknuti", "Zadatpozici()");

```

```

        //Tlacitko Nakrasli Graf
fTextButton1383 = new TGTextButton(PriZaseknuti,"Nakresli graf");
fTextButton1383->SetTextJustify(36);
fTextButton1383->MoveResize(10,45,80,20);
fTextButton1383->Connect("Clicked()", "Functions", "PriZaseknuti", "NakresliGraf1()");

        //Tlacitko Na Zacatek

Zacatek = new TGTextButton(fMainFrame1862,"Na zacatek");
Zacatek->SetTextJustify(36);
Zacatek->MoveResize(720,330,70,20);
Zacatek->Connect("Clicked()", "Functions", "fMainFrame1862", "NaZacatek()");

        //Tlacitko Pomoc

Pomoc = new TGTextButton(fMainFrame1862,"Pomoc");
Pomoc->SetTextJustify(36);
Pomoc->MoveResize(720,355,70,20);
Pomoc->Connect("Clicked()", "Functions", "fMainFrame1862", "Pomoc()");

        fMainFrame1862->SetWindowName("Pokus do praktik");
fMainFrame1862->MapSubwindows();
fMainFrame1862->MapWindow();

}

```