

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Test HW stimulátoru pro měření ERP experimentů

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracovala samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 6. května 2014

Tereza Štanglová

Poděkování

Ráda bych poděkovala Ing. Pavlu Mautnerovi, Ph.D. za cenné rady, věcné připomínky a vstřícnost při konzultacích a vypracování bakalářské práce.

Abstract

At the Department of Computer Science and Engineering is being developed an EEG hardware stimulator. The stimulator is used for measurement of evoked potentials (ERP) in neuroinformatics. The main objective of this bachelor thesis is to get acquainted with the functions and options of the stimulator and to create a graphical user interface for this device in order to operate the stimulator using the USB interface of the PC. Program is implemented in Java.

The other objective is to design an appropriate ERP experiment and to record data in order to verify the functionality of the stimulator. Recorded data are analyzed and processed using software called ERPLAB. Data processing is subjected to the predetermined scheme. The outputs of this software are graphs showing the presence of evoked responses. If the evoked response is present, the stimulator is suitable for this type of experiment.

Obsah

1	Úvod	1
2	Teoretická část	2
2.1	Problematika EEG/ERP	2
2.1.1	Elektroencefalografie	2
2.1.2	Evokované potenciály	5
2.1.3	Měření evokovaných potenciálů	6
2.1.4	Elektroencefalograf	7
2.1.5	Rušení signálů	9
2.2	Stimulátor	9
2.2.1	Navázání komunikace se stimulátorem	10
2.2.2	Definice paketu pro USB ovládání	12
2.2.3	Srovnání se softwarem Presentation	13
2.3	EEG data	14
2.3.1	Úprava a zpracování dat	15
2.3.2	Softwarové možnosti	18
3	Realizační část	20
3.1	Program pro nastavování parametrů stimulátoru	20
3.1.1	Použité technologie	20
3.1.2	Možnosti grafického rozhraní	23
3.1.3	Nastavení pro stimulaci evokovaných potenciálů	23
3.1.4	Simulace	25
3.2	Stimulátor	27
3.2.1	Naslouchání	27
3.2.2	Odeslání zpráv	28
3.2.3	Chyby	29
3.2.4	Odpovědi stimulátoru	29
3.3	Měření dat	32
3.3.1	Navržený experiment	33
3.4	Zpracování dat	34

3.4.1	Popis algoritmu	34
3.4.2	Grafy	36
4	Závěr	40
	Literatura	41
	Seznam obrázků	43
A	Příloha – Definice paketu	45
B	Příloha – Schéma	48
C	Příloha – UML diagram tříd	50
D	Příloha – Uživatelská dokumentace	51

1 Úvod

Ve vědním oboru neuroinformatika se diagnostickou metodou elektroencefalografie (EEG) dá zaznamenávat elektrická odpověď mozku, která se později dále zpracovává. Odpovědi mozku na určitý druh stimulu se nazývají evokované potenciály (ERP). Osobu, která je měřena, je potřeba určitým způsobem stimulovat, vizuálně, akusticky nebo kombinací obojího. Používají se softwarové nebo hardwarové stimulatory. Za tímto účelem vznikl na Katedře informatiky a výpočetní techniky Západočeské univerzity v Plzni hardwarový stimulator, který slouží pro měření těchto evokovaných potenciálů. Stimulator se dá ovládat přímo z displeje zařízení nebo přes USB rozhraní.

Cílem práce je vytvořit programové rozhraní pro tento stimulator tak, aby bylo možné ho pohodlně ovládat pomocí počítače. Čtenář bude seznámen s funkcemi stimulatoru, dále s možnostmi nastavení parametrů stimulatoru pomocí vytvořeného uživatelského rozhraní a se způsobem zpracování a zhodnocení naměřených dat pro zvolený experiment.

Hlavní ideou stimulatoru je vytvoření přenosného zařízení, které se bude snadno používat a nebude potřebovat žádná dodatečná zařízení pro ovládání, například externí počítač. Stále probíhá vývoj nových verzí s přidávanými funkcionalitami, proto je možné, že program pro ovládání stimulatoru, který je výsledkem mé práce, nepokrývá veškeré možnosti stimulatoru. Pokrytí většího množství funkcionalit by mohlo být téma pro diplomovou práci.

Teoretická část práce je rozdělena do tří sekcí. Nejprve je popsána problematika EEG, co je elektroencefalografie, kde se dá využít a proč, a problematika evokovaných potenciálů, jejich druhy a způsoby měření. Druhá sekce je věnována hardwarovému stimulatoru, například popisu jeho modulů, možnostem komunikace se zařízením a srovnání se softwarem se stejnou či podobnou funkcností. Třetí sekce se zabývá popisem zpracování EEG dat, které stimulator vrací.

Realizační část se věnuje popisu vytvořeného uživatelského rozhraní pro nastavování parametrů stimulatoru, detailnímu popisu komunikace se stimulatorem, formátu odesílaných a přijímaných dat. Další sekce realizační části slouží k popisu experimentu pro ověření funkčnosti stimulatoru. Naměřená data jsou analyzována a zhodnocena pomocí vhodně zvoleného softwaru.

2 Teoretická část

2.1 Problematika EEG/ERP

V následující části bude objasněno, co je elektroencefalografie a jak pomocí ní probíhá zpracování evokovaných potenciálů (ERP – z ang. Event – Related Potential), jak tyto potenciály vznikají a k čemu se dají využít.

2.1.1 Elektroencefalografie

Elektroencefalografie je diagnostická metoda pro vyšetřování mozkové aktivity. Snímajícími elektrodami může být sledován elektrický potenciál mozku. Tento elektrický potenciál je na povrchu hlavy velmi slabý. Použití elektrod na povrchu hlavy je neinvazivní metoda vyšetření. Invazivní diagnostická metoda je například při umístění elektrod přímo na povrch mozku.



Obrázek 2.1: Ukázka křivky EEG záznamu [3].

Snímaná aktivita je zobrazovaná na displeji nebo vykreslovaná na papír. Záznam má podobu křivky, viz Obrázek 2.1, pro jejíž správné zhodnocení musí mít vyšetřující osoba dostatečné zkušenosti. Z křivek je možné rozeznat určité druhy aktivity. Jedná se o periodicky se opakující vlny nebo vlny, které mají na stejný stimul stejnou odpověď.

Mezi základní typy mozkových aktivit, viz Obrázek 2.2, patří:

- *Gamma vlny* – vlny o frekvenci 30–50 Hz. Amplituda se udává $\approx 10 \mu\text{V}$.

Tyto vlny lze naměřit po celém skalpu. Odráží stav aktivního zpracování informací, mohou také ale souviset se stresem. Jsou pozorovány i během podprahové stimulace. Pokud gamma vlny chybí, signalizuje to neschopnost učit se.

Byly pozorovány až po vynalezení digitálního elektroencefalografu. Analogový elektroencefalograf nebyl schopný identifikovat frekvence vyšší než 25 Hz.

- *Beta vlny* – vlny o frekvenci 14–30 Hz. Amplituda se pohybuje mezi 5–20 μV .

Objevují se přirozeně v bdělém stavu. Jsou spojeny s aktivním přemýšlením a soustředěním. Nastávají také ve stresových stavech nebo stavech napětí. Pozornost je vedena vně a směřuje k problémům spojeným s okolím. Vyskytuje se při zavřených i otevřených očích. Beta rytmus se snižuje při ospalosti.

- *Alfa vlny* – vlny o frekvenci 8–12 Hz. Amplituda se pohybuje mezi 20–200 μV .

Jsou přítomny v bdělém stavu, kdy má subjekt zavřené oči a relaxuje, nezpracovává informace. Po opětovném otevření očí se aktivita snižuje. Klesá i při přemýšlení a soustředění nebo při spánku. Neobjevují se do třetího roku života.

V současnosti využívá navození alfa vln metoda psychoterapie zvaná EEG Biofeedback. Jde o jakési sebeučení mozku pomocí biologické zpětné vazby. Pomocí EEG se sleduje mozková aktivita. Tato metoda se používá u hyperaktivních dětí nebo při překonávání fobií.

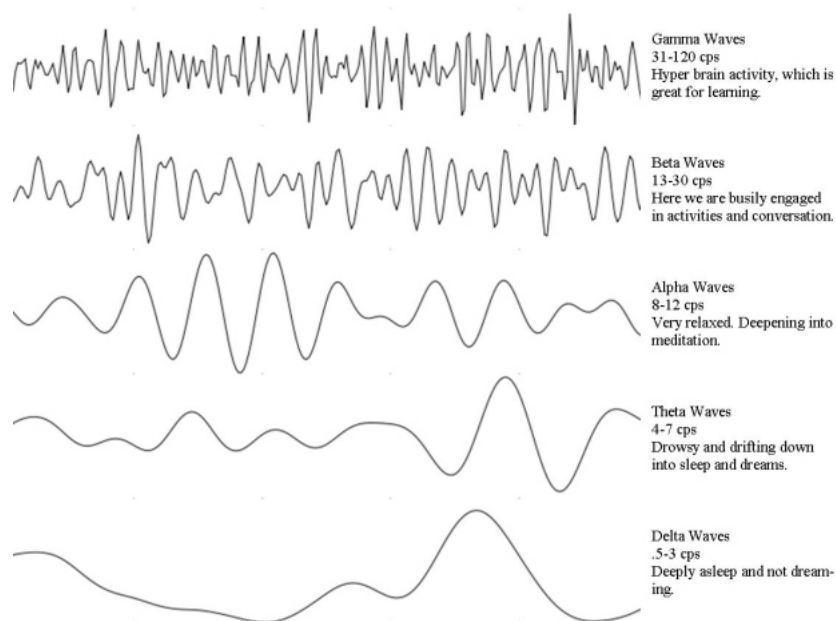
- *Theta vlny* – vlny o frekvenci 4–7 Hz. Amplituda je obvykle menší než 30 μV .

Jsou mezistavem mezi spánkem a bdělostí. Objevují se při meditacích, vnitřním soustředěním nebo ve stavu spánku, doprovázeném sny. Mají souvislost s kreativitou, intuicí, fantazií nebo vzpomínkami.

Jsou běžnou součástí záznamu. Častěji se vyskytují u dětí a adolescentů než u dospělých.

- *Delta vlny* – vlny o frekvenci 0,5–4 Hz. Amplituda je proměnlivá, pohybuje se mezi 10–300 μV .

U dětí do jednoho roku je výskyt normální. U dospělých se vyskytuje v NREM III a IV fázi spánku, fáze hlubokého spánku, s amplitudou 100 μV . Výskyt v bdělém stavu signalizuje poruchy soustředění. Vyskytují se i ve stavu bezvědomí způsobeném nemocí či úrazem.



Obrázek 2.2: Základní typy mozkových vln [9].

Při poškození mozku se dají pomocí EEG sledovat abnormality v mozkové činnosti a lokalizovat oblast, ve které vznikají. Například při vyšetření epilepsie může EEG ukázat místo, kde epilepsie vzniká, a tím i říct, o jaký druh epilepsie se jedná. Pomocí EEG se také kontroluje průběh léčby. Dalším důvodem pro použití EEG jsou spánkové poruchy, kdy je pacient monitorován během spánku, aby se zjistilo, zda jsou přítomné příslušné aktivity. Léze nervového systému lze diagnostikovat evokovanými potenciály, viz dále. Pokud stimulace konkrétního místa nevyvolá očekávanou reakci, lze předpokládat, že léze se nachází právě v tomto místě.

V univerzitním prostředí se EEG využívá například pro interakci člověka s počítačem nebo pro experimentální měření.

2.1.2 Evokované potenciály

Evokované potenciály jsou elektrofyziologické odpovědi centrálního nervového systému na různé druhy stimulů [13]. Tato odpověď je tvořena sledem pozitivních a negativních vln. Evokované potenciály lze rozdělit podle délky latence na krátkolatentní (BEAP – kmenové sluchové evokované potenciály), střednělatentní (VEP – zrakové evokované potenciály) a dlouholatentní (tzv. kognitivní evokované potenciály). Amplituda evokovaných potenciálů je podstatně menší než základní mozková aktivita. Tato aktivita se před vyhodnocováním potenciálů potlačí matematickými úpravami signálu.

Základní druhy evokovaných potenciálů jsou:

- **BEAP** – sluchové evokované potenciály (z angl. Brainstem Acoustic Evoked Potentials).

Vzniká při přenosu vzruchů po nervech sluchové dráhy. Jedná se o *far-field* potenciál, tzv. potenciál vzdáleného pole, kdy je mezi elektrodou a generátorem odpovědi silná vrstva tkáně, mozková hemisféra. Nejčastěji použitým stimulem je cvaknutí, z angličtiny označováno jako „klik“. Tento podnět nemá stálou frekvenci a trvá 0,1 ms nebo méně [12]. Během stimulace je druhé ucho blokováno maskovacím šumem, aby se zabránilo jeho stimulaci kostním vedením. Průběh BEAP je tvořen sérií pozitivních vln, s latencí do 10 ms, označovaných římskými číslicemi. Chybějící vlny jsou známkou postižení sluchu.

- **SSEP** – somatosenzorické evokované potenciály (z angl. Somatosensory Evoked Potentials).

Je vyšetřována periferní nervová soustava, kterou tvoří všechny nervy neležící v centrální nervové soustavě, tj. v mozku a míše. Stimulovanými nervy jsou *n. medianus*¹, nacházející se v oblasti horních končetin, a *n. tibialis*², nacházející se v oblasti dolních končetin. Mohou být stimulovány kdekoliv ve svém průběhu. Nejčastěji na zápěstí a na vnitřním kotníku [12].

- **VEP** – zrakové evokované potenciály (z angl. Visual Evoked Potentials).

Tyto potenciály reprezentují elektrickou aktivitu zrakové kůry. Je testována funkčnost zrakové dráhy optického nervu [13]. Stimulace těchto

¹http://en.wikipedia.org/wiki/Median_nerve

²http://en.wikipedia.org/wiki/Tibial_nerve

potenciálů je možná dvěma způsoby. Lze použít tzv. „záblesky“, z angl. flash, které jsou vhodné pro stimulování novorozenců nebo nespolupracujících osob. Stimulačním podnětem je změna jasu v zorném poli. Druhý způsob je stimulování strukturovanými podněty. Takový podnět mohou být obrazce s tmavými a světlými plochami, např. šachovnice. Stimulovat lze celým polem, polovinou nebo kvadranty a to obě oči samostatně nebo společně. Ve výsledku se hodnotí vlnový komplex skládající se z vln N1(75)-P2(100)-N2(135). Nejdůležitější je vlna P100 [11].

- **MEP** – motorické evokované potenciály (z angl. Motor Evoked Potentials).

Jsou odpovědí na elektrickou nebo magnetickou stimulaci mozku nebo míchy. Odpověď je registrována ze svalu, tzn. je testována funkční integrita motorické dráhy [12]. Vyšetřují se při podezření na postižení motoneuronů, což jsou nervové buňky ve svalových vláknech. Onemocnění se projevuje kolem šedesátého roku života. Začíná postižením specifické skupiny svalů, například svalů končetin nebo dýchacích svalů či postižením svalů jazyka a hlasivek [10].

- **ERP** – kognitivní evokované potenciály (z angl. Event-Related Potentials).

Tyto potenciály jsou vyvolány identifikovatelnými neočekávanými podněty nebo událostmi. Takový podnět či událost generují odpověď, která má výraznější amplitudu než podnět či událost očekávané [12]. Mezi kognitivní evokované potenciály patří například *CNV* (Contingent Negative Variation), *MMN* (Mismatch Negativity) nebo *vlna P300*. Soubor těchto potenciálů je řazen mezi dlouholatentní pomalé mozkové potenciály. Souvisí s šířením informací strukturami mozku zapojenými do daného úkolu [14].

2.1.3 Měření evokovaných potenciálů

Pro správné naměření evokovaných potenciálů je potřeba odpovídající vybavení. Nezbytné součásti měření jsou stimulátor pro interpretaci podnětů, registrační elektrody, diferenciální předzesilovač a přístroj umožňující další zesílení, zprůměrování a zobrazení signálu. Druh stimulátoru odpovídá modalitě evokované odpovědi [12]. Pro eliminování vlivů, které by mohly zapříčinit rušení signálu, by mělo vyšetření probíhat v odhlučněné místnosti, aby nebyl pacient rozptylován okolím.

2.1.4 Elektroencefalograf

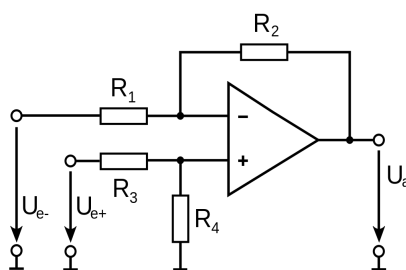
Elektroencefalograf je přístroj, který slouží k záznamu EEG. Skládá ze tří částí [1], které jsou:

- Diferenciální zesilovač
- Registrační elektrody
- Kanály

Diferenciální zesilovač

Signály snímané elektrodami mají velmi nízkou amplitudu. Pro jejich další zpracování se signál musí zesílit pomocí diferenciálního zesilovače. Vstup se skládá z invertujícího a neinvertujícího vstupu. Schéma zapojení zesilovače viz Obrázek 2.3. Zesilovač slouží pro zesílení rozdílu signálů na vstupu, zatímco souhlasný signál, signál stejné polarity, který přenáší rušení, se snaží co nejvíce potlačit.

Na oba vstupy se přivede signál z elektrod a naindukuje se stejné rušivé napětí. Toto rušivé napětí se navzájem na výstupu odečte a na výsledku se neprojeví.



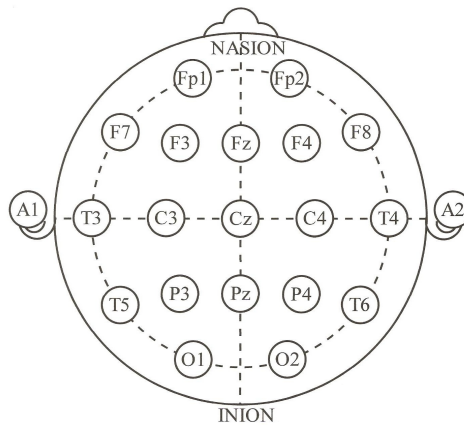
Obrázek 2.3: Schéma diferenciálního zesilovače [6].

Registrační elektrody

Jsou prvním elementem, který zajišťuje převod elektrické aktivity z mozku do zesilovačů EEG přístroje. Elektrická charakteristika elektrod je daná druhem použitého kovu a tekutého média, které společně v klidovém stavu vytvářejí elektrický potenciál, a při průchodu proudem dochází k polarizaci [11].

Podle metody měření, invazivní či neinvazivní, je volen typ elektrod. Pro invazivní typ snímání aktivity mozku jsou to jehlové elektrody, které se většinou využívají na operačních sálech. Pro neinvazivní typ jsou to elektrody plošné, které se používají nejčastěji. Počet povrchových elektrod se může lišit. Nejběžněji používaný systém 10/20, viz Obrázek 2.4, jehož autorem je doktor H. Jasper. V tomto systému je vzdálenost od dvou krajních poloh, jak v podélné, tak v příčné linii, rozdělena na úseky po 10% nebo 20%.

Elektrody jsou popsány čísly podle umístění nad hemisférami. Lichá čísla jsou nad levou mozkovou hemisférou a sudá čísla jsou nad pravou. Jsou označeny také písmeny podle umístění nad mozkovými laloky, A – ušní lalůčky, C – centrum, P – temenní lalok, F – čelní lalok, O – týlní lalok, T – spánkový lalok.



Obrázek 2.4: Systém rozmístění elektrod 10/20 [2].

Elektrody *A1* a *A2* jsou tzv. semicefalické reference, které se připojují na ušní boltce [12].

Elektroencefalogram je tvořen souborem kanálů, jejichž počet odpovídá počtu elektrod.

Kanály

Každá elektroda má svůj kanál, který je výstupem příslušně zapojeného zesilovače. Využívá se zapojení v unipolárním i bipolárním režimu. Signály lze

při postupu kanály různými způsoby upravovat. Mezi nejčastější úpravu patří použití nastavitelných filtrů.

2.1.5 Rušení signálů

EEG signál tvoří rytmické vlny, které mají sinusoidní tvar o různých frekvencích. Je velmi citlivý na jakýkoliv šum. Podle frekvence se rozlišují čtyři základní typy mozkové aktivity, *Alfa*, *Beta*, *Theta* a *Delta* vlny, viz kapitola 2.1.1 *Elektroencefalografie*. Jakákoliv jiná zobrazená vlna je považována za nežádoucí artefakt.

Artefakty

Artefakty se projevují jako potenciály, které negativně ovlivňují sledovanou mozkovou aktivitu. Rozlišujeme dva druhy artefaktů:

- **Biologické artefakty** – Pochází od pacienta. Dochází k nim například při mrkání nebo jiném pohybu pacientových očí. Mohou se projevit při pocení nebo z pohybu svalů generovaném při jeho kontrakci. Artefakty ze svalů jsou nejčastějšími artefakty. Vliv má i srdeční činnost a tep.
- **Interferenční artefakty** – Jedná se o artefakty z prostředí nebo z přístroje. Za artefakt z prostředí je považován například síťový artefakt, který pochází od vlastního EEG nebo od okolních přístrojů. Projeví se jako signál s frekvencí 50 Hz. Tento signál lze odstranit číslicovými filtry, jelikož se v EEG signálu většinou nevyskytuje frekvence vyšší než 45 Hz. Artefakty od přístroje jsou způsobeny indukčním napětím.

2.2 Stimulátor

Programovatelný EEG stimulátor je vyvíjen na KIV ZČU Ing. Jiřím Novotným. Je určen pro měření vizuálních a sluchových ERP experimentů. Stimulátor je modulární, dá se dále rozšiřovat například změnou firmwaru nebo přidáním modulů. Blokové schéma modulů stimulátoru viz Obrázek 2.5.

První sadou jsou vstupně/výstupní moduly. Mezi tyto se řadí vstupní modul pro měření reakčních dob, kdy se zaznamenává doba reakce na odpovídající stimul. Jednotlivé reakční časy jsou ukládány na SD kartu. Další moduly jsou výstupní modul pro synchronizaci a modul pro GPIO, který je tvořen souborem pinů, které se dají softwarově naprogramovat jako vstupní nebo výstupní.

Druhou sadou modulů jsou moduly pro nastavení a sběr dat, mezi které patří modul pro LCD, pomocí něhož lze stimulátor připojit k monitoru, a modul pro komunikaci s USB, který slouží pro připojení zařízení k počítači přes USB port.

Další moduly jsou speciální pro konkrétní typ stimulace, zrakovou či akustickou nebo jejich kombinaci. Zraková stimulace může být provedena LED panely nebo zobrazováním grafických objektů na VGA monitoru nebo na externím LCD displeji. Akustická stimulace se provádí buď zvuky uloženými ve formátu *wav*, nebo krátkými tóny s nastavitelnou frekvencí a délkou trvání.

Data pro nastavení stimulátoru jsou uložena v nevolatilní paměti. Tato paměť není závislá na napájení a její obsah se po odpojení napájení neztratí. Grafické objekty, zvuky a naměřená data jsou uložena na SD kartě stimulátoru.

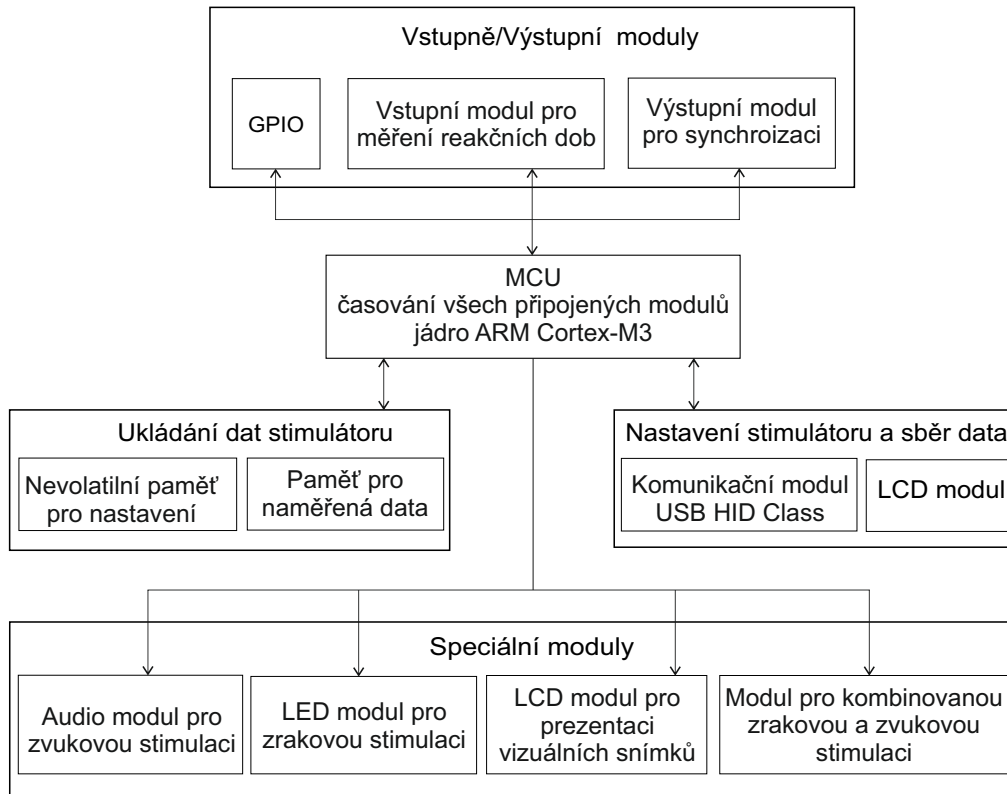
Časování všech připojených modulů řídí kontrolní jednotka s jádrem ARM Cortex-M3.

Stimulátor může být využíván pro nastavování parametrů samostatně, protože jeho součástí je i rezistivní displej. Přes USB port může být připojen k počítači, viz výše moduly pro nastavení a sběr dat, což umožní vytvořenému grafickému rozhraní nastavovat parametry a komunikovat se stimulátorem přes USB rozhraní.

2.2.1 Navázání komunikace se stimulátorem

Komunikace programu se stimulátorem je navázána prostřednictvím Java HIDAPI³, které slouží pro práci se zařízeními standardu HID (Human Interface Devices). Zařízení standardu HID jsou používána pro přímou interakci člověka s počítačem, např. klávesnice nebo myš. Hlavní výhodou použití HID je „driverless“ použití, tzn. do systému není třeba nic instalovat. Způsob komu-

³Webové stránky projektu Java HIDAPI – <http://code.google.com/p/javahidapi/>.



Obrázek 2.5: Blokové schéma stimulatoru.

nikace s příslušným zařízením je dán třídou dle specifikace⁴ HID. Stimulátor patří do třídy *Vendor Specific*⁵. Projekt Java HIDAPI je vytvářen společností **Codeminders**.

HIDAPI je multiplatformní knihovna. Kód knihovny je open-source a smí být použit pod třemi licencemi popsány v příloženém souboru *LICENCE.txt*.

Pro navázání komunikace se zařízením se do projektu v jazyce Java musí nainportovat knihovna *hidapi-1.1.jar*. Tato knihovna obsahuje několik tříd. Tyto třídy jsou:

- **HIDDevices** – Instance této třídy reprezentuje HID zařízení, se kterým má být navázána komunikace.

⁴Třídy HID zařízení – http://www.usb.org/developers/devclass_docs/HID1_11.pdf.

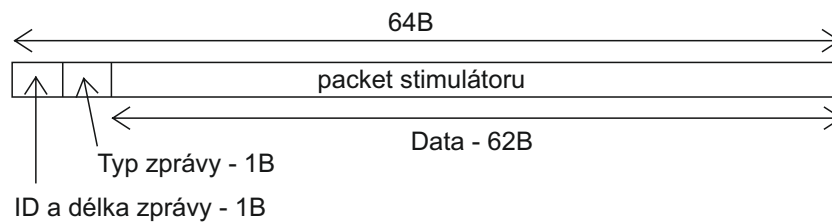
⁵Kódy USB tříd – http://www.usb.org/developers/defined_class.

- `HIDDevicesInfo` – Třída obsahující vlastnosti HID zařízení.
- `HIDDeviceNotFoundException` – Třída vyvolávající výjimku, pokud HID zařízení se zadanými kritérii nelze nalézt.
- `HIDManager` – Třída, která slouží pro nalezení, připojení nebo odpojení HID zařízení.

Podrobnější informace o dokumentaci ke třídám a jejich metodám lze nalézt ve složce *HIDAPI JavaDoc* v souborech přiložených na CD.

2.2.2 Definice paketu pro USB ovládání

Pakety posílané do stimulátoru mají pevnou délku 64B, se strukturou viz Obrázek 2.6.



Obrázek 2.6: Struktura paketu.

1. byte – hlavička

- ID – 2 bity
- LEN – 6 bitů

2. byte – typ zprávy

Zprávy mohou obsahovat informace například o povolení blikání LED nebo jasu LED. Typ zprávy je zapsán například takto:

- `0x01` — Povolení LED – Povolí blikání LED.
- `0x1C` — Jas LED0 – data 1B, hodnota je v %, min. 0 (nesvítí) max. 100

3. a další byty

Tyto byty obsahují data, jejichž délka je určena polem LEN. Formát dat je definován typem zprávy. Nevyužité byty jsou doplněny nulami.

Kompletní popis všech možných typů zpráv paketu lze nalézt v souboru *paket.pdf* v souborech přiložených na CD nebo v příloze A na straně 45.

2.2.3 Srovnání se softwarem Presentation

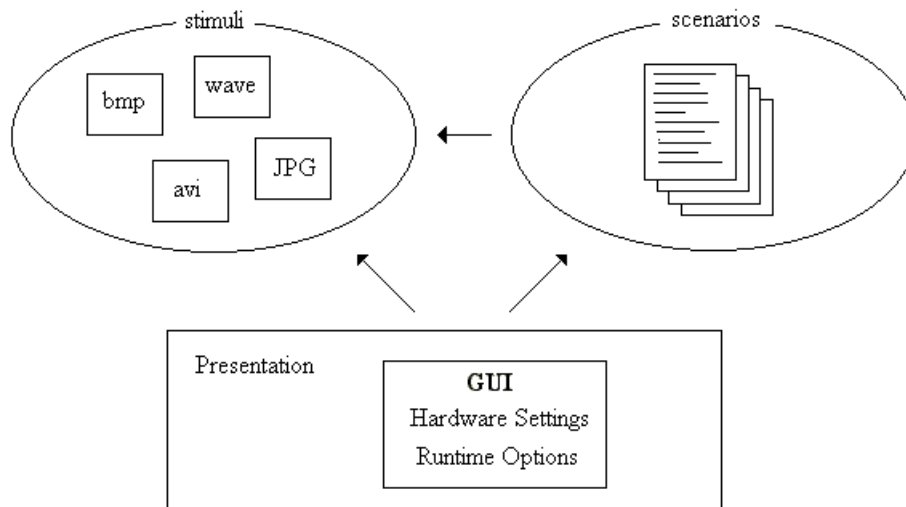
Software Presentation®⁶, který je vyvíjen firmou Neurobehavioral Systems Inc., slouží k zobrazování stimulů a vytváření experimentálních programů pro neurovědy. Podporuje sluchové, vizuální a multimodální podněty. Využívá dva skriptovací programovací jazyky. Prvním je **SDL (Scenario Description Language)**. Slouží k popisu stimulů, které tvoří experiment. Druhým je **PCL (Program Control Language)**. Používá se pro manipulaci s objekty vytvořenými v SDL nebo pro samostatné vytváření stimulů pro prezentaci. Software je využíván pro experimentální účely především na univerzitách.

Vytváření experimentů lze rozdělit do tří kroků [20], viz Obrázek 2.7.

1. Výběr stimulů ve vhodném datovém formátu potřebných k experimentu, viz segment *stimuli*. Software dokáže zpracovat 2D a 3D grafiku, komprimované video a zvuky.
2. Tvorba scénáře pro stimuly ve skriptovacím programovacím jazyce, viz segment *scenarios*.
3. Specifikace konfigurace hardwaru, viz segment *Presentation*.

Hlavní nevýhodou tohoto softwaru oproti hardwarovému stimulátoru je především jeho cena, která se pohybuje ve stovkách dolarů a je odvíjena od počtu počítačů, na kterých je software nainstalován, a typu aktivace produktu. Licence je poskytována na časově omezenou dobu, poté musí být prodloužena. Další nevýhodou je omezení v rámci operačního systému. Software lze spustit pouze v OS Windows. Ačkoliv výrobce garantuje časovou

⁶Webové stránky firmy Neurobehavioral Systems Inc. – <http://www.neurobs.com/>.



Obrázek 2.7: Struktura softwaru Presentation [20].

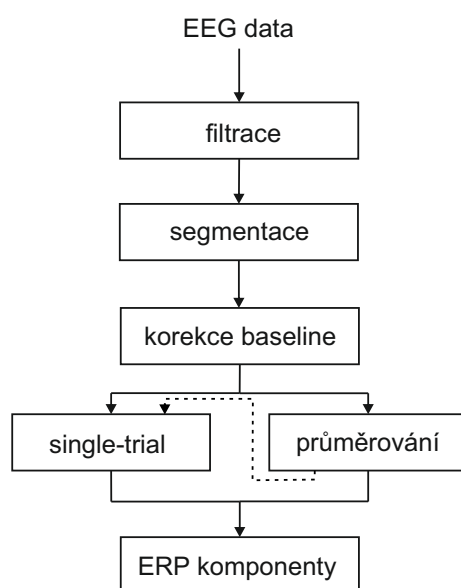
přesnost menší než milisekundu, může docházet ke zpoždění vlivem operačního systému. Pro tvoření experimentů je nutná znalost programování, což může být problém pro doktory. Programovací jazyk není příliš přehledný, protože se jeho syntaxe skládá ze syntaxe několika jiných programovacích jazyků. Problém může nastat při synchronizaci přes paralelní port, který se v dnešní době používá jen zřídka.

Výhodou softwaru je možnost vytvářet univerzální složité experimenty s multimediálními podněty.

2.3 EEG data

Měřený EEG signál vždy obsahuje nežádoucí složky, které musí být před správnou interpretací dat eliminovány. Tyto jevy se dají eliminovat hned při měření signálu, například docílením lepšího kontaktu záznamových elektrod s pokožkou pomocí vodivého gelu, nebo po naměření dat. Signál musí být „očištěn“ od artefaktů, viz kapitola 2.1.5 *Rušení signálů*.

Úprava a zpracování signálu probíhá podle schématu, viz Obrázek 2.8. Čárkovanou čarou je vyznačena volitelná cesta.



Obrázek 2.8: Schéma úpravy a zpracování EEG dat.

Detailnější popis těchto metod viz dále.

Veškerá naměřená data i získané výsledky po zpracování lze zobrazit. Pro vizualizaci dat je možné použít speciální software, také viz dále.

2.3.1 Úprava a zpracování dat

Filtrace dat

Filtry se dělí na digitální a analogové, které se nacházejí například v zesilovačích. Průběh signálu si lze představit jako soubor sinusových vln různé frekvence a fáze. Frekvenčně selektivní filtry jsou aplikovány pro potlačení nebo zachování vybraných frekvencí. Nejčastější typy filtrů jsou: *low-pass* filtr, který tlumí vysoké frekvence, *high-pass* filtr, který tlumí nízké frekvence, *bandpass* filtr, který propouští pouze frekvence daného frekvenčního pásma, a *notch* filtr, který tlumí úzké pásmo frekvencí, vše ostatní projde. Nejčastěji aplikované filtry pro ERP experimenty jsou pro odstranění *beta* aktivity mozku a rušení sítě, 50 Hz.

Aplikace filtrů je důležitá, ale někdy mohou data poškodit. Frekvenčně

selektivní filtry můžeme rozdělit podle typu na FIR nebo IIR filtry [7].

- **FIR** (Finite Impuls Response) – Filtry s konečnou impulzní charakteristikou. Filtry jsou vždy stabilní a lze je navrhnout s lineární fází. Realizují se obvykle nerekurzivním algoritmem. Při zpracování dat může docházet pouze k časovému posuvu, který zpožďuje signál. U EEG dat to může být ale problém z důvodu posunu latence křivky. Křivku nedeformují, protože při lineární fázové charakteristice se jednotlivé frekvenční složky při přepočtení fázového posuvu na časový posunou o stejnou hodnotu, jedná se o tzv. *skupinové zpoždění*.
- **IIR** (Infinite Impuls Response) – Filtry s nekonečnou impulzní charakteristikou. Realizují se vždy rekurzivním algoritmem. Nejsou stabilní. Filtry mají nelineární fází, tj. deformují křivku.

Segmentace ERP signálu

Segmentace je rozdělení signálu na menší úseky, které vykazují stejnou nebo podobnou mozkovou odpověď. Segmentace kontinuálního EEG záznamu se dělí podle délky segmentů na:

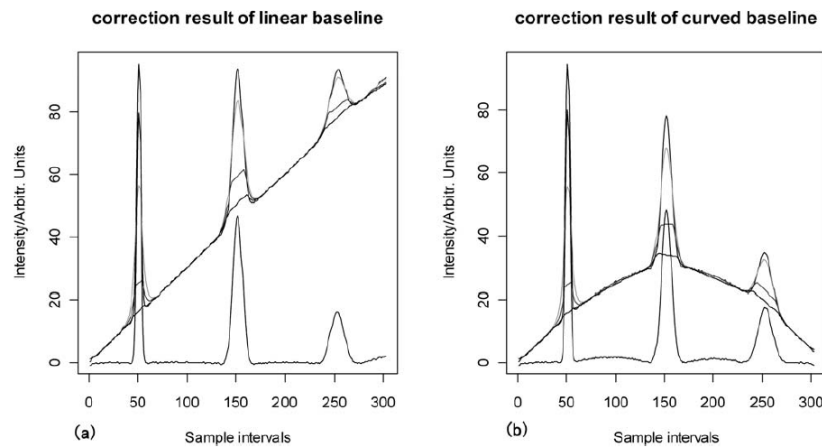
- Konstantní segmentace – Všechny segmenty mají stejnou délku a stejný počet vzorků. Délka intervalu je v rámci několika sekund. Segmenty pak obsahují vlny různých amplitud a frekvencí, které si nejsou podobné.
- Adaptivní segmentace – Délka segmentů je proměnná, závislá na stacionaritě signálu. Signál se rozdělí na stacionární úseky v závislosti na výskytu nestacionarit, například artefakty nebo epileptická aktivita, v signálu.

U segmentů může docházet k posuvu amplitud a při průměrování by mohlo docházet k chybám.

Segmenty mají určité parametry, příznaky, například maximum, minimum, střední hodnota, modus nebo medián hodnot segmentu.

Korekce baseline

Korekce baseline slouží k odstranění artefaktů. Průběh křivky signálu může být ovlivněn například špatně namazanou elektrodou. Průběh signálu se pak může vychýlit, viz Obrázek 2.9, kde je zobrazena původní a požadovaná křivka. Vlevo má původní křivka lineární průběh, vpravo zakřivený. Pro korekci se používá tzv. *pre-stimulus* interval, což je interval většinou -200 až 0 ms před stimulem. Oprava se provádí odečtením tohoto napětí od každého bodu segmentu. Po odečtení jsou odstraněny vysoké hodnoty napětí. Vytvoří se „nulová“ úroveň napětí, viz Obrázek 2.9. Tento proces se provádí pro každý kanál zvlášť. Korekce křivky je manipulace s daty, která může ovlivnit další kroky jejich zpracování [16].

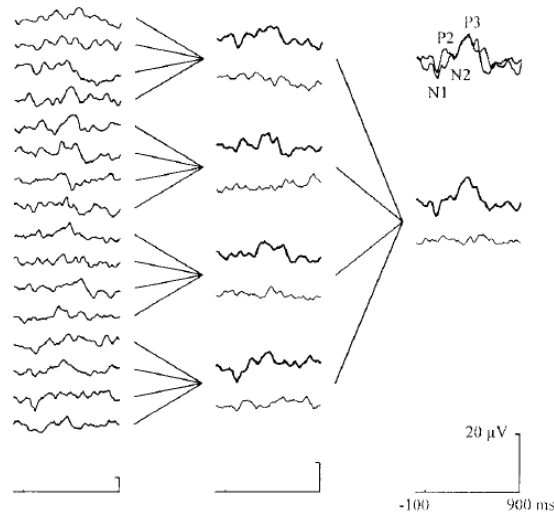


Obrázek 2.9: Korekce baseline [18].

Potlačení základní EEG aktivity

Základní EEG aktivitu lze potlačit více způsoby, například průměrováním signálu nebo „single-trial“ analýzou.

Při průměrování je signál rozdělen do tzv. epoch. Epocha je oblast v okolí synchronizační značky, která reprezentuje výskyt nějaké události. Skládá se z evokovaného potenciálu a základní EEG aktivity, kterou lze také chápat jako náhodný šum. Epochy odpovídají stejnému stimulu. Předpokládá se, že evokovaný potenciál je ve všech epochách skoro stejný, ale šum proměnný, nezávislý na podnětu. Po zprůměrování dostatečného množství epoch zůstane pouze signál evokovaného potenciálu [8], viz Obrázek 2.10.



Obrázek 2.10: Proces průměrování EEG dat [17].

„Single-trial“ analýza je založena na dekompozici signálu a nalezení koeficientů reprezentujících EEG signál. Jejím cílem je extrakce ERP komponent. Jednou z metod této analýzy je waveletová transformace, která poskytuje informace o časové lokalizaci ERP komponent.

Tato transformace analyzuje signál v časové a frekvenční oblasti. Signál se při zpracování rozloží do systému „waveletů“. Wavelet je funkce, která osciluje na okolí lokalizačního bodu. Vstupní signál se rozkládá do řady waveletových koeficientů pomocí dvou ortogonálních filtrů, tzv. otcovský a mateřský wavelet. Otcovský wavelet ukazuje trend signálu – rozklad na škálové koeficienty, mateřský wavelet zachycuje informace o „jemnosti“ na jednotlivých úrovních. Waveletová transformace je okenní operace. Jádro transformace se získá posunutím a roztažením vybrané báze funkce, speciální okenní funkce $\psi(t)$. Okénko je posouváno přes celou šířku signálu a v každém bodě je vypočítána jeho hodnota. Tento proces se několikrát opakuje s užším nebo širším okénkem. Na křivce se tak lokalizují body, které ukazují na místo ERP komponenty [5].

2.3.2 Softwarové možnosti

Pro zpracování a vizualizaci EEG dat je možné použít speciální software.

BrainVision Analyzer

BrainVision Analyzer je software vyvíjený firmou Brain Products GmbH⁷. Software se skládá z několika různých součástí, jako je například část pro čtení EEG dat, část pro transformaci a úpravu dat, export dat nebo jejich zobrazení na obrazovce. Je možné vytvářet další komponenty a tím Analyzer dynamicky rozšiřovat. Rozhraní pro vývoj je volně k dispozici, takže si každý může vytvořit své vlastní komponenty. Výhodou je, že přidané součásti mohou být psány v jakémkoliv jazyce pro .NET aplikace [19].

EEProbe

EEProbe je software vyvíjený firmou ANT Neuro⁸. Jedná se o Unix software, který slouží ke studiu evokovaných potenciálů. Skládá se z několika programů (modulů) pro zpracování dat, například modul pro převody mezi různými formáty dat, modul pro zobrazení nebo zpracování dat. Každý modul je ovládán v příkazovém řádku pomocí příkazů s různými parametry a pomocí konfiguračních souborů. EEProbe není univerzální nástroj pro zpracování signálů. Obsahuje pouze nejběžněji používané operace pro studii ERP. Uživatel by měl umět pracovat v příkazovém řádku unixových operačních systémů a znát základy metodiky ERP [4].

ERPLAB

ERPLAB⁹ je sada nástrojů pro prostředí MATLAB¹⁰, která je volně ke stažení. Vývoj koordinuje Steve Luck a Javier Lopez-Calderon. ERPLAB je úzce spjatý se sadou nástrojů EEGLAB¹¹. Rozšiřuje možnosti EEGLAB pro poskytnutí robustního nástroje pro zpracování, vizualizaci a analýzu ERP. Grafické uživatelské rozhraní umožňuje rychlé osvojení ovládání i začátečníkům, psaní skriptů je zase vhodné pro pokročilé uživatele [15].

⁷Webové stránky firmy Brain Products GmbH – <http://www.brainproducts.com/>.

⁸Webové stránky firmy ANT Neuro – <http://www.ant-neuro.com/>.

⁹Webové stránky – ERPLAB – <http://erpinfo.org/erplab>.

¹⁰Webové stránky – MATLAB – <http://www.mathworks.com/products/matlab/>.

¹¹Webové stránky – EEGLAB – <http://sccn.ucsd.edu/eeglab/>.

3 Realizační část

Realizační část je věnována aplikaci pro nastavení parametrů stimulatoru, komunikaci se stimulatorem a měření a zpracování dat.

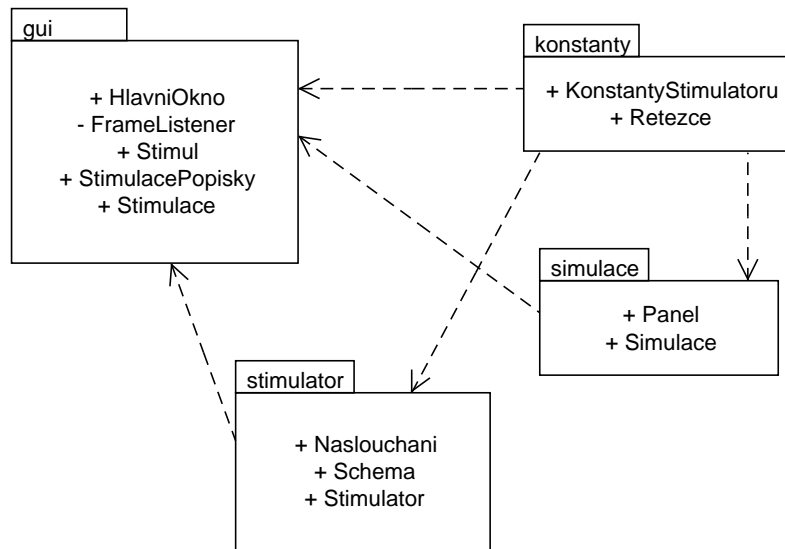
3.1 Program pro nastavování parametrů stimulatoru

Následující část popisuje funkčnost a návrh aplikace pro nastavování parametrů stimulatoru z programátorského hlediska. Návod pro obsluhu aplikace je v dokumentu *Uživatelská dokumentace.pdf* v souborech přiložených na CD nebo v příloze D na straně 51.

Parametry, které je možno zadávat, jsou cíleně určeny pro stimulaci LED diodami. Například pro potenciály SSVEP (z angl. Steady State Visually Evoked Potentials) – ustálené vizuální evokované potenciály. Tyto potenciály jsou speciálním případem VEP.

3.1.1 Použité technologie

Pro realizaci aplikace byl vybrán programovací jazyk Java verze 7, používané prostředí pro vývoj Eclipse. Aplikace se skládá z několika tříd rozdělených tématicky do balíčků. UML diagram balíčků viz Obrázek 3.1. Zobrazení pouze diagramu balíčků bylo záměrné, protože celý UML diagram tříd programu by byl příliš rozsáhlý. UML diagram tříd je v souboru *Diagram tříd.pdf* v souborech přiložených na CD nebo v příloze C na straně 50. Na tomto diagramu můžete vidět u každé třídy seznam nejdůležitějších atributů a metod, které obsahuje. Z důvodu přílišné obsáhlosti nemohou být uvedeny všechny. Detailní popis všech atributů a všech metod lze najít ve vygenerované dokumentaci k programu ve složce *JavaDoc* v souborech přiložených na CD.



Obrázek 3.1: UML diagram balíčků.

Balíček **gui** – Obsahuje třídy pro práci s grafickým uživatelským rozhraním.

- **HlavniOkno** – Třída slouží pro spuštění programu a vykreslení všech potřebných komponent jako jsou pole pro nastavování parametrů nebo simulace. Třída také slouží pro ukládání a načítání souboru se schématem, viz dále.
- **FrameListener** – Třída slouží pro obsluhu událostí při zavření uživatelského rozhraní ikonou křížku.
- **Stimul** – Třída reprezentuje stimul pro evokované potenciály. Jako její atributy jsou všechny vlastnosti, které mohou být nastaveny, jako například jas, doba aktivity či pauzy svícení nebo frekvence. Jednotlivými metodami lze tyto vlastnosti upravovat, ukládat do proměnných nebo mazat.
- **StimulacePopisky** – Třída slouží spíše jako pomocná pro přehledné seskupení všech popisků, které se týkají jednotlivých LED diod. Metody vrací popisky jako instanci třídy `JLabel`.
- **Stimulace** – Třída slouží pro úpravu a vykreslování prvků pro nastavení hodnot simulace. Mezi prvky patří například panely pro tex-

tová pole, rozbalovací seznam pro nastavení ISI, přepínače pro výběr typu stimulace. Třída také obsluhuje všechny události spojené s těmito prvky. Stará se o komplexní kontrolu dat před odesláním do stimulatoru nebo spuštěním simulace. Součástí je i vykreslování ovládacích prvků simulace.

Balíček **stimulator** – Obsahuje třídy, které slouží pro komunikaci a práci se stimulatorem.

- **Stimulator** – Třída slouží pro přístup a práci se stimulatorem. Pomocí Java HIDAPI, viz dále, se dokáže se stimulatorem přes USB rozhraní spojit a odpojit. Obsahuje všechny potřebné metody pro odesílání zpráv do stimulatoru pro nastavení parametrů LED diod. Přepočítává hodnoty z frekvenční stimulace do časové, aby bylo možné tyto hodnoty ukládat do bytového pole pro schéma.
- **Naslouchani** – Třída slouží pro naslouchání příchozím zprávám od stimulatoru a jejich částečnému zpracování. Je spuštěna v samostatném vlákne.
- **Schema** – Třída slouží jako reprezentace schémat uložených ve stimulatoru nebo na disku uživatele. Zpracovává pole bytů, aby bylo možné je ukládat na disk ve stejném formátu jako jsou data uložena ve stimulatoru. Načítá a ukládá data ze souboru i ze stimulatoru.

Balíček **simulace** – Obsahuje třídy pro vykreslení a ovládání simulace blikání LED diod.

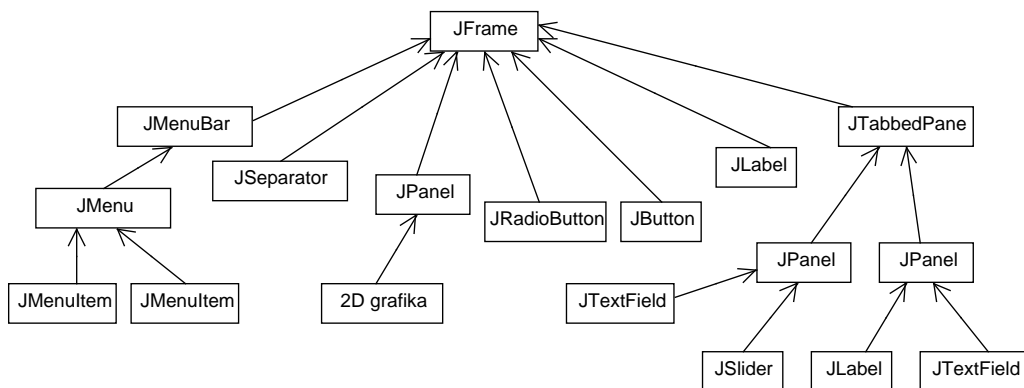
- **Simulace** – Třída slouží pro spuštění simulace, aby si mohl uživatel lépe představit bez nutnosti zapojení stimulatoru, jak bude jeho schéma vypadat. Pokud je nastavena časová stimulace, vytvoří se rovnou promíchané pole stimulů, které budou promítány na plátno. Pokud bude nastavena frekvenční stimulace, provede se před promítáním převod na hodnoty časové stimulace. Třída běží v samostatném vlákne.
- **Panel** – Třída slouží pro vykreslování simulace na plátno. Používá k tomu jednoduché grafické 2D objekty a grafický kontext.

Balíček **konstanty** – Obsahuje dvě pomocné třídy, které jsou naplněny řetězci a konstantami. Veškerý text v programu je řešen pomocí konstant, aby byly jednodušší případné opravy a lokalizace.

- **KonstantyStimulatoru** – Třída obsahuje všechny konstanty, které slouží jako typ zpráv odesílaných do stimulatoru. Třída obsahuje velké množství konstant, takže pro ukázkou je jich v UML diagramu tříd zobrazeno minimum.
- **Retezce** – Třída slouží pro uchování všech řetězců obsažených v každé třídě. Obsahuje velké množství řetězců, takže pro ukázkou je jich v UML diagramu tříd zobrazeno minimum.

3.1.2 Možnosti grafického rozhraní

Grafické rozhraní programu je tvořeno pomocí Java knihovny `javax.swing`. Knihovna nabízí rozhraní pro tvorbu a obsluhu grafických komponent, jako jsou například dialogy, rozbalovací seznamy, rámečky, atp. Tyto komponenty jsou v programu hierarchicky provázány, viz Obrázek 3.2. Pro ilustraci je uvedeno jen omezené množství komponent.

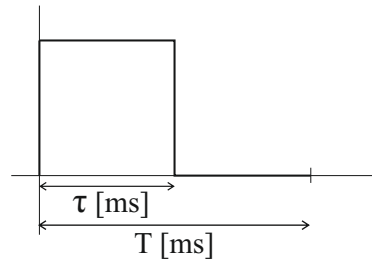


Obrázek 3.2: Hierarchie grafických komponent.

3.1.3 Nastavení pro stimulaci evokovaných potenciálů

Stimulace evokovaných potenciálů se rozdělila dle požadavků na časovou a frekvenční. U časové lze nastavovat procentuální blikání diody, dobu aktivity a pauzy svícení a jas. U frekvenční stimulace lze nastavovat jas, frekvenci a střihu impulsu. Střih je poměr časů, ve kterých je obdélníkový signál v jednotlivých úrovních, viz Obrázek 3.3. Je daná vzorcem $\frac{\tau}{T}$, kde T je perioda a τ

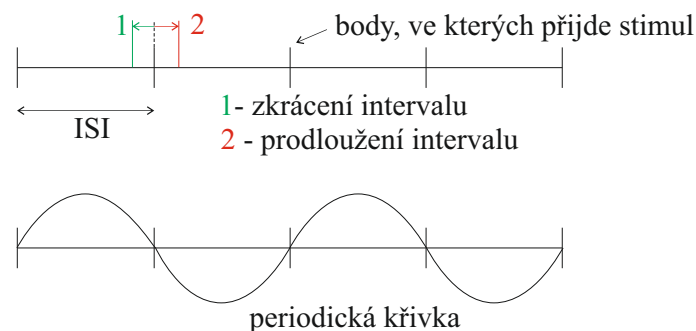
je doba svícení. Nastavení střídý tedy mění poměr doby svícení a nesvícení diody. Čím větší bude hodnota střídý, tím delší bude τ .



Obrázek 3.3: Střída impulsu.

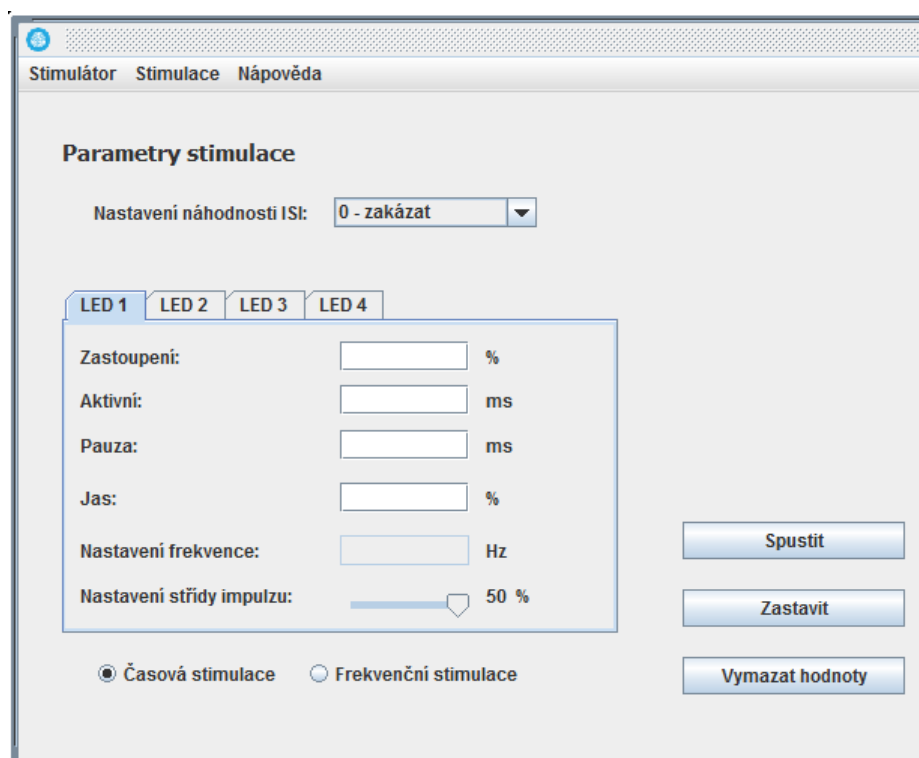
Pomocí prvků `JRadioButton` se lze mezi časovou a frekvenční stimulací přepínat. Pole pro vybranou stimulaci se zaktivní, pole pro nevybranou stimulaci se zneaktivní, aby nešly měnit jejich hodnoty. Na prvku `JTabbedPane` jsou umístěny čtyři panely pro každý diodový panel. Stimulátor obsahuje čtyři USB konektory pro připojení až čtyř diodových panelů. Každý panel v GUI lze nastavit samostatně.

Další nastavitelnou vlastností je nastavení náhodnosti ISI (z angl. Inter-stimulus Interval). Možnosti nastavení tohoto intervalu jsou čtyři – zakázání, kratší interval, delší interval nebo kratší i delší. Používá se, pokud je v EEG přítomna periodická aktivita, příkladem jsou alfa vlny. Stimuly jsou také prezentovány v konstantním čase, takže by se mohlo stát, že stimul bude opakován v určitém cyklu vždy na stejném místě periodické křivky, což by mohlo znehodnotit měření. Aby se zabránilo takovémuto znehodnocení měření, prodlužuje se nebo zkracuje interval doby nesvícení, viz Obrázek 3.4. Tento interval se pohybuje v rozmezí ± 50 ms [16].



Obrázek 3.4: Změna ISI vůči periodické křivce.

Grafické rozhraní obsahuje i další prvky, viz Obrázek 3.5. K prvkům tlačítek `JButton` jsou přiřazeny `listeners`, akce po stisknutí. O akci vypovídá název tlačítka. V horním menu pod výběrem *Stimulátor* jsou jako prvky `JMenuItem` položky menu. Na tyto položky jsou navázány akce pro připojení a odpojení stimulatoru a pro ukládání a načítání schématu. Více o ovládání grafického rozhraní je k dispozici v dokumentu *Uživatelská dokumentace.pdf* v souborech přiložených na CD nebo v příloze D na straně 51.



Obrázek 3.5: Ukázka panelu pro nastavení parametrů pro stimulaci.

3.1.4 Simulace

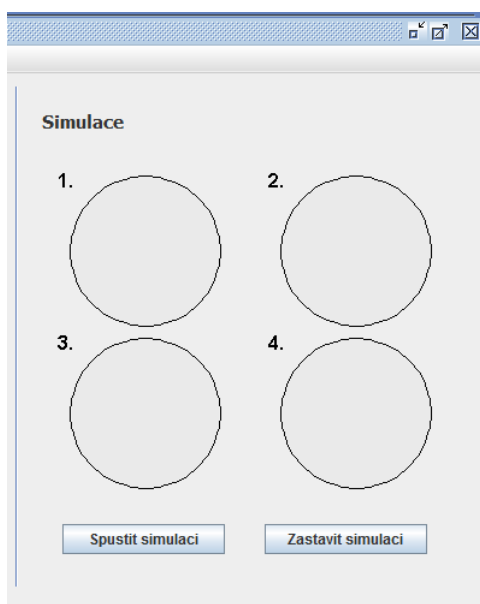
Pro lepší představu o chování nastavené stimulace je do programu zakomponovaná i simulace blikání diod, viz Obrázek 3.6. Skládá se ze dvou tříd, `Simulace` a `Panel`.

Pokud je nastavená výchozí časová stimulace, data se nijak neupravují. Pokud je nastavená frekvenční stimulace, musí se data upravit tak, aby se dala převést na časovou. Vzorec pro převod frekvence na čas je

$$f = \frac{1}{T}[\text{s}]. \quad (3.1)$$

Pro převod na milisekundy se výsledek vynásobí tisícem. Čas daný v milisekundách se podle hodnoty střídy impulzu rozdělí na dobu svícení a pauzu, viz Obrázek 3.3.

Podle hodnoty zastoupení jednotlivých diod se vygeneruje pole náhodně umístěných stimulů. V běhové metodě vlákna se podle hodnot ze stimulace nastavují parametry pro vykreslování blikání na plátno. Podle parametru doby aktivity svícení a pauzy se provede uspání vlákna.



Obrázek 3.6: Ukázka panelu pro simulaci blikání diod.

Pro vykreslení prvků simulace jsou použity třídy z knihovny `java.awt`. Jednou z těchto tříd je třída `Graphics`, která dává k dispozici grafický kontext. Pomocí tohoto kontextu lze přistupovat k základním grafickým prvkům jako jsou čáry, kruhy či čtverce. Pro samotné vykreslení komponent musí být implementována speciální metoda `paint()`, ve které se prvky při změně překreslují. Právě toto překreslení prvků simuluje blikání diod. Jednotlivé kruhy se při simulaci blikání vyplňují různými odstíny žluté.

Pro frekvenční stimulaci při zadaných hodnotách vyšších než cca 15 Hz může docházet ke zkreslení simulace.

3.2 Stimulátor

Komunikace mezi stimulátorem a programem probíhá prostřednictvím tříd Java HIDAPI, viz kapitola 2.2.1 *Navázání komunikace se stimulátorem*. Nejprve musí být získána instance třídy *HIDManager*, pomocí níž se bude moci vyhledávat a manipulovat s HID zařízeními. Vrací také seznam všech připojených HID zařízení, který lze využít pro nalezení stimulátoru. Stimulátor je nalezen podle jednoznačných identifikátorů **Vendor ID** a **Product ID**.

Vendor ID je jedinečné číslo přiřazené hardwaru. Přiřazuje ho firma, ve které byl tento hardware vyroben. Vendor ID stimulátoru je 0x0D28 (NXP dodavatel vývojového kitu). Product ID je identifikační číslo konkrétního produktu. Spravuje ho vlastník VID (Vendor ID) sám, aby nedošlo ke kolizi ovladačů. Product ID stimulátoru je 0x1234.

Stimulátor, pokud je připojen, je reprezentován třídou *HIDDevice*, která slouží pro manipulaci s ním.

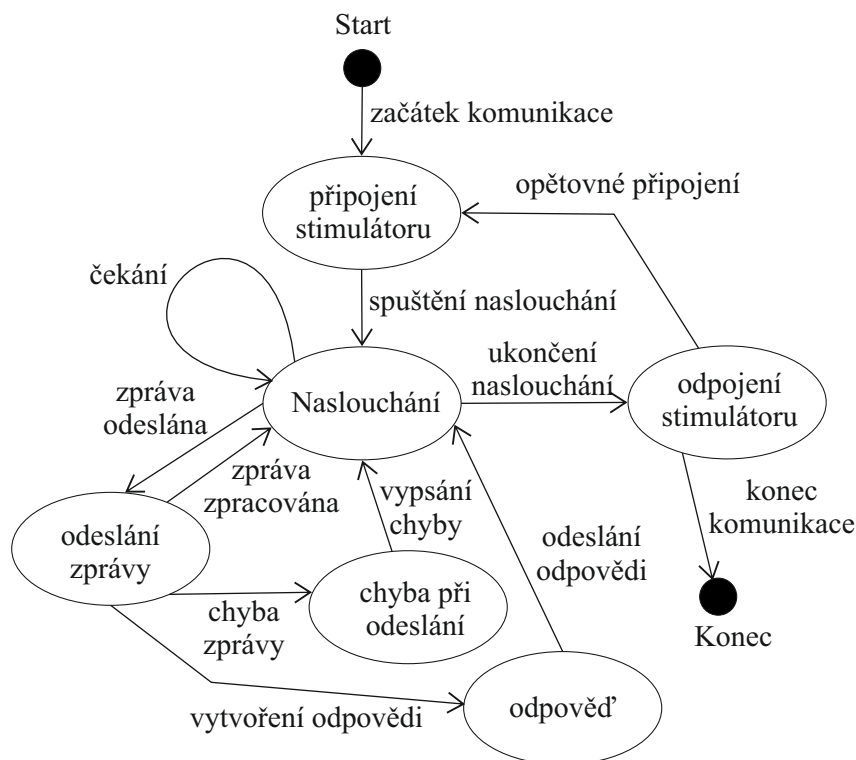
Ukázka kódu pro nalezení a připojení stimulátoru:

```
HIDManager manager = HIDManager.getInstance();
HIDDeviceInfo[] zarizeni = manager.listDevices();
HIDDevice stimulator = zarizeni[i].open();
```

Komunikace programu se stimulátorem je popsána stavovým diagramem, viz Obrázek 3.7. Před začátkem práce se stimulátorem musí být stimulátor připojen. Po připojení se spustí naslouchání, které běží ve smyčce, program čeká na data od stimulátoru. Pro odeslání požadavku do stimulátoru přejde program do stavu odeslání zprávy. Požadavek může buď skončit chybou, být bez odezvy, nebo požádat stimulátor o vrácení dat. Před odpojením stimulátoru se ukončí naslouchání. Pro opětovné zapnutí naslouchání je nutné stimulátor znovu připojit.

3.2.1 Naslouchání

Naslouchání stimulátoru se při jeho připojení spustí v samostatném vlákne. Vlákno je při odpojení stimulátoru ukončeno. V tomto vlákne běží nekonečný cyklus, ve kterém se zachycují příchozí zprávy. Zprávy mají pevnou délku



Obrázek 3.7: Stavový diagram komunikace mezi stimulátorem a programem.

64 bytů. Zprávy jsou tedy zachycovány do pole přesně této délky. Příkladem přijaté zprávy mohou být schémata uložená ve stimulátoru, viz dále.

3.2.2 Odeslání zpráv

Zprávy s daty jsou do stimulátoru posílány metodou `write()` ze třídy `HIDDevice`. Jediným parametrem této metody je bytové pole s potřebnými hodnotami. První byte je rezervován metodou a je vždy `0x00`. Další byty jsou nastavovány podle definice paketu pro USB ovládání. Druhý byte je délka odesílané zprávy. Třetí byte je typ odesílané zprávy. Další byty obsahují data, jejichž délka je definovaná v druhém bytu.

Ukázka kódu pro odeslání zprávy pro povolení blikání LED diod:

```
byte[] povoleniLed = new byte[] { 0x00, 0x00, 0x01, 0x00 };
```

```
stimulator.write(povoleniLed);
```

Poslední byte je nulový, což značí, že nejsou odesílána žádná data, takže bude nulový i druhý byte. Kód *0x01* je konstanta pro povolení blikání LED diod. Všechny konstanty pro ovládání stimulátoru jsou popsány v souboru *paket.pdf* v souborech přiložených na CD nebo v příloze A na straně 45.

Při spojení se stimulátorem nebo při zapisování dat může dojít k chybám, které je potřeba ošetřit.

3.2.3 Chyby

Každá chyba vyvolá v programu výjimku, kterou je nutno zachytit a podle jejího charakteru dále zpracovat. Nejčastější výjimkou je **IOException**, která vzniká při selhání vstupně-výstupní operace. Ve spojení se stimulátorem může být tato výjimka nastat například při selhání zápisu dat metodou **write()**, kdy se program pokouší odeslat data do nepřipojeného zařízení. Pokud taková situace nastane, lze jí vyřešit opětovným zapojením zařízení a připojením v programu.

Další možnou výjimkou je **NumberFormatException**, která vznikne při pokusu o zapsání dat ve špatném formátu. Formát dat je kontrolován před jejich odesláním. Další možnou výjimkou je **NullPointerException**, která vznikne při pokusu o přístoupení k nevytvořené nulové instanci. Zachycení ostatních výjimek je ošetřeno pomocí nadřazené výjimky **Exception**.

3.2.4 Odpovědi stimulátoru

Na určitou posloupnost zpráv reaguje stimulátor odesláním odpovědi zpět. Jediná odpověď stimulátoru, se kterou se v této době pracuje, jsou schémata experimentů uložená na SD kartě. Schémata uložená ve stimulátoru představují soubor parametrů nastavení pro požadovaný typ stimulace. Stimulace se mohou lišit v typu, časová nebo frekvenční, v hodnotách doby aktivity či pauzy svícení diod atd. Ve stimulátoru může být uloženo až dvacet různých schémat. Právě kvůli omezenému počtu uložených schémat byla do programu přidána funkcionality pro uložení a načtení schématu ze souboru, který je uložen na pevném disku uživatele. Ukládání a načítání dat tedy probíhá dvěma způsoby, ze souboru nebo z paměti stimulátoru.

Ukládání a načítání z paměti stimulátoru

Schémat jsou uložena v paměti stimulátoru. Jedno schéma má vždy status aktuální. Hodnoty tohoto schématu představují aktuální nastavení v registrech stimulátoru. Schémata se dají z paměti načíst dvěma způsoby. Prvním způsobem je odeslání zprávy pro čtení celé paměti s kódem zprávy 0x7F a následné vypsání bufferu s kódem zprávy 0x0F. Tento způsob vrátí pole délky 780 bytů, 39 (délka schématu) * 20 (počet). Druhý způsob je postupné načtení všech dvaceti schémat do registrů a vypsání aktuálního schématu. Schémata by se tak vracela po jednom a ne jako dlouhé pole. Sekvence bytových polí, které se odesílají do stimulátoru, viz následující kód.

```
nahrání z paměti = byte[] { 0x00, 0x00, 0x71, 0x00, i};  
vrácení položek = byte[] { 0x00, 0x00, 0x5B, 0x00};  
čtení bufferu = byte[] { 0x00, 0x00, 0x0F, 0x00};
```

Zpráva s typem 0x71 říká, že mají být do zařízení načtena data ze stránky zadané indexem *i*, který představuje index schématu. Zpráva s typem 0x5B říká, že mají být vráceny položky aktuálního schématu, a zpráva s typem 0x0F říká, že má být přečten buffer zařízení.

Při ukládání schématu se nejprve odešlou všechny potřebné hodnoty stimulace z aplikace do stimulátoru a následně se odešle zpráva s typem 0x70, která říká, že se má současné nastavení uložit do paměti. Ke zprávě je přidáván další byte, který určuje index schématu, které bude upraveno.

Schémat stimulátoru mají přesně danou strukturu. Jedná se o pole bytů délky 39. Každý byte nese určitou informaci. Mezi tyto informace patří například:

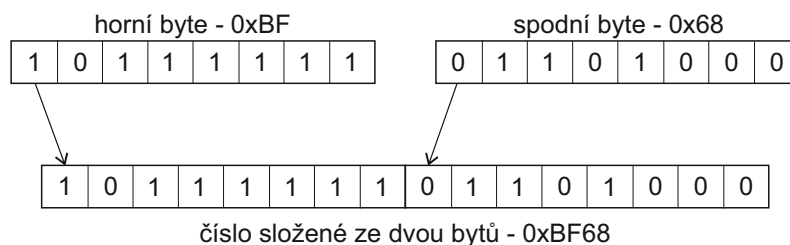
- `schema[0]` – Délka pole.
- `schema[1]` – Kód odpovědi, například 0x5B pro aktuální schéma.
- `schema[2]` – Index schématu. Říká, na jakém indexu 0–19 se přijaté schéma nachází.
- `schema[3]` – Horní byte doby aktivity LED 0.
- `schema[4]` – Dolní byte doby aktivity LED 0.

- `schema[5]` – Horní byte doby pauzy LED 0.
- `schema[6]` – Dolní byte doby zapnutí LED 0.
- `schema[7]` – Zastoupení LED 0.
- `schema[8]` – Jas LED 0.

Analogicky pokračuje nastavení ostatních tří diod. Další doplňkové informace jsou k dispozici v příloženém souboru *schéma.pdf* nebo v příloze B na straně 48.

Jak je vidět z příkladů informací ze schématu, hodnoty doby aktivity a pauzy diody jsou rozděleny na dva byty. Pro zjištění těchto dvou hodnot musí být byty spojeny v jedno číslo. To samé platí i pro ukládání. Velké číslo musí být rozděleno na dva byty.

Dvě osmibitová čísla se spojují v jedno šestnáctibitové, viz Obrázek 3.8, aby bylo dosaženo většího rozsahu. Pro ukázkou jsou použity čísla `0xBF`, v binární soustavě `10111111`, pro horní byte a `0x68`, v binární soustavě `01101000`, pro dolní byte.



Obrázek 3.8: Spojení dvou osmibitových čísel.

Výsledné dvoubytové číslo typu `short` musí být převedeno na číslo typu `int`, neboť v Javě, ve které je aplikace napsaná, nelze pracovat s čísly bez znaménka (unsigned number), takže první bit je brán jako znaménkový. Kód pro spojení čísel je následující:

```
byte a, b;  
short s = (a << 8) | (b & 0xff);  
int i = s & 0xffff;
```

Pro uložení hodnot doby aktivity a pauzy diod z textových polí aplikace musí být čísla na dva byty naopak rozdělena. Kód pro rozdělení čísel je následující:

```
byte a = (cislo >> 8) & 0xff;  
byte b = cislo & 0xff;
```

Proměnná `cislo` je číslo, které má být rozděleno. Do proměnné `a` je vloženo horních osm bitů pomocí bitového posuvu a do proměnné `b` dolních osm bitů.

Ukládání a načítání ze souboru

Schéma, které je ve formě bytového pole a má stejnou strukturu jako schéma stimulátoru, je do binárního souboru uloženo jako proud dat pomocí třídy `BufferedOutputStream`, aby se každý byte nemusel zapisovat zvlášť. Typ stimulace, časová nebo frekvenční, je určen prvním bytem v datech, kdy 0 je pro časovou stimulaci a 1 pro frekvenční. Data jsou před uložením kontrolována, jestli mají čísla správný formát a jestli sedí do rozsahu možných hodnot. Po uložení vznikne soubor s příponou `bin`, ze kterého se dá schéma opět načíst. Při otvírání souboru se o typu stimulace rozhodne podle prvního bytu a podle toho se příslušně vyplní hodnoty polí.

Při opětovném načítání schémat může dojít ke změně informace. Při časové stimulaci se data ukládají rovnou do pole bytů, ale při frekvenční stimulaci, dané parametry frekvence a střída impulsu, se musí data přepočítat, aby seděla na časovou stimulaci, danou parametry doba aktivity a pauzy. Při této konverzi může docházet vlivem zaokrouhlovacích chyb k drobným odchylkám, takže je dobré si hodnoty schématu po načtení do aplikace překontrolovat.

3.3 Měření dat

Měření probíhá na Katedře informatiky a výpočetní techniky v učebně UU403. K měření je potřeba několik věcí:

- Stimulátor s diodovými panely.
- Program pro nahrávání dat.

- Počítač pro spuštění programu.
- Zvukotěsná kabina.
- Potřeby pro přípravu subjektu – elektrody, vodivý gel, čepice pro EEG.

Osobu, která bude měřena, je potřeba před začátkem připravit. Referenční, zemní a registrační elektrody musí být správně namazány pomocí vodivého gelu, aby byl zajištěn co nejlepší kontakt s pokožkou hlavy. Jestli jsou elektrody správně připraveny lze zjistit s využitím programu BrainVision Recorder, který současně slouží i jako program pro nahrávání dat. Subjekt je umístěn do zvukotěsné kabiny. Na externím počítači je tedy spuštěný program pro nastavení parametrů stimulátoru a program pro nahrávání dat. Po spuštění vybrané stimulace se rozblíkají diody podle aktuálního nastavení. V BrainVision Recorderu je spuštěno nahrávání dat. Po vypnutí stimulace vrátí BrainVision Recorder tři soubory ve formátech **.eeg*, **.vhdr* a **.vmrk*.

3.3.1 Navržený experiment

Pro detekci vizuální P3 vlny byl pro stimulátor navržen jednoduchý experiment. Cílem experimentu je ověřit, jestli je stimulátor vhodný pro měření tohoto druhu experimentů.

Pro vyvolání vizuální P3 (P300) vlny, což je viditelná odpověď na neočekávaný podnět, u měřeného subjektu bylo použito tří LED diod. Každá dioda má jinou barvu, zelenou, červenou a žlutou, a každá bliká s jinou pravděpodobností. Barva diody se nastavuje hardwarově. Dioda blikající s pravděpodobností 85% je non-target stimul. Dioda blikající s pravděpodobností 5% je target stimul, vyvolává *P3a* komponentu P3 vlny. Dioda blikající s pravděpodobností 15% je také target stimul, vyvolává *P3b* komponentu P3 vlny. Výskyt vlny P3 se v EEG signálu projeví markerem, značkou, v místě výskytu. Podle těchto markerů se EEG signál rozděluje na epochy a dále zpracovává, viz Obrázek 2.8.

3.4 Zpracování dat

Pro zpracování dat byl vybrán software ERPLAB¹, open-source sada nástrojů pro Matlab, ve verzi 4.0.2.3. Před použitím ERPLABu je nutné mít nainstalovaný Matlab a EEGLAB, se kterým je ERPLAB úzce spojen.

EEGLAB se přidá do seznamu pluginů a spustí příkazem `>> eeglab`. Složka s ERPLABem je k dispozici na stránkách softwaru. Složka se jednoduše přepokopíruje do složky `plugins` EEGLABu. Při opětovném spuštění EEGLABu je již nabídka s ERPLABem zobrazena.

Pro zpracování dat pro experiment pro detekci P3 vlny byl vytvořen skript s názvem `zpracovaniP3.m`. Ve skriptu je napsaná krátká nápověda, kterou lze v Matlabu zobrazit příkazem

```
>> help cesta_k_souboru\zpracovaniP3.m.
```

3.4.1 Popis algoritmu

Data jsou načítána ze souboru, vytvořeným softwarem BrainVision Recorder, s příponou `.vhdr` příkazem:

```
EEG = pop_loadbv(cesta, soubor, [], [])
```

Soubor `.vhdr` je hlavičkový soubor, který obsahuje například informace o jednotlivých kanálech, nastavení zesilovače nebo seznam aplikovaných filtrů při měření. V tomto souboru je také uvedeno, jak se jmenuje soubor s naměřeným EEG (přípona `.eeg`) a soubor pro popis markerů (přípona `.vmrk`). Načtením souboru je vytvořena nová množina dat, tzv. dataset. Argumenty funkce `pop_loadbv` jsou cesta k souboru, jeho název, interval, který bude načten, a kanály. Kanály představují množinu snímacích elektrod. Zde je načten celý záznam a všechny kanály. Lokace kanálů v datasetu musí být upřesněna.

```
EEG = pop_chanedit(EEG, 'lookup',  
'cesta_k_souboru\standard-10-5-cap385.elp')
```

Argumenty funkce `pop_chanedit` jsou aktuální data set, EEG, a soubor, který najde kanály podle určitých standardů. Před zpracováním je vhodné data přefiltrovat alespoň základním filtrem.

¹Webové stránky – ERPLAB – <http://erpinfo.org/erplab>

```
EEG = pop_basicfilter( EEG, 1:19, 'Cutoff', [0.1 20],  
'Order', 2, 'Design', 'butter', 'Filter', 'bandpass' )
```

Ve funkci `pop_basicfilter` je nejdůležitější argument `Design`, jehož hodnota *butter* říká, že bude aplikován IIR filtr, viz kapitola 2.3.1 *Úprava a zpracování dat* odstavec *Filtrace dat*.

Aby bylo jednodušší EEG signál upravovat, je vytvořen tzv. *eventlist*. V této struktuře jsou zaznamenány kódy pro události, které nesou stejnou informaci. Každá událost je označena markerem. V tomto případě je jedním z markerů marker označující událost pro *non-target stimula*, který má v počtu markerů majoritní, několikanásobně větší, zastoupení. Dalšími markery jsou markery událostí, kde se předpokládá výskyt *P3a* a *P3b* vlny.

Pokud jsou v signálu zaneseny markery, může se provést rozdělení signálu na epochy.

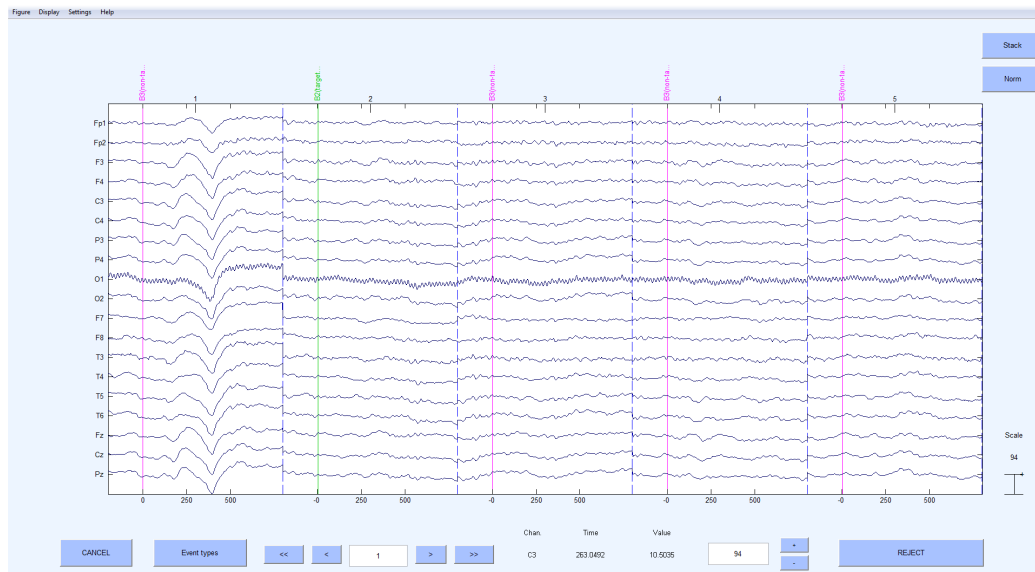
```
EEG = pop_epochbin( EEG , [-200.0 800.0], [-50 0])
```

Argumenty funkce `pop_epochbin` jsou aktuální dataset, interval pro epochu kolem markeru v milisekundách. Epocha tudíž začíná 200 ms před markerem a končí 800 ms po markeru, viz Obrázek 3.9, kde je vidět rozdělený signál a označené markery. Marker události pro *non-target stimula* je označen růžovou barvou, marker události pro *target stimula* zelenou. Druhý interval je pro korekci *baseline*, viz kapitola 2.3.1 *Úprava a zpracování dat* odstavec *Korekce baseline*, aby se signál více přimknul k ose.

Další úpravou signálu je detekce artefaktů.

```
EEG = pop_artextval( EEG, 'Channel', 1:19, 'Flag', 1:3,  
'Threshold', [ -50 80], 'Twindow', [ -200 800])
```

Důležité argumenty funkce `pop_artextval` jsou *Twindow*, jehož hodnota je interval, na kterém se budou artefakty hledat a který odpovídá velikosti epochy, a *Threshold*, jehož hodnota je interval, práh v μV . Epochy, ve kterých se artefakt vyskytne, jsou označeny, viz Obrázek 3.10. Poslední úpravou dat před samotným zobrazením ERP komponent je průměrování signálu.



Obrázek 3.9: Signál rozdělený na epochy s vyznačenými markery.

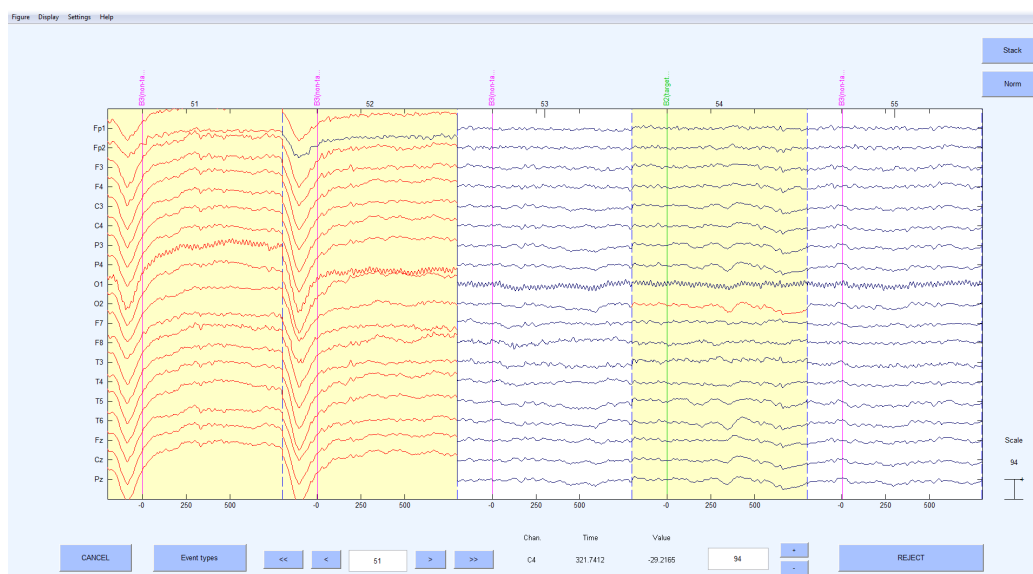
```
ERP = pop_averager( EEG , 'Criterion', 'good',
'ExcludeBoundary', 'on', 'SEM', 'on')
```

Argumentem funkce `pop_averager` *Criterion* s hodnotou *good* je nastaveno, že se nebudou průměrovat epochy s artefakty. Argumentem *ExcludeBoundary* je nastaveno, že se nebudou průměrovat epochy obsahující hraniční markery. Argument *SEM* říká, že se počítá se standardními chybami při průměrování. Posledním příkazem před samotným vykreslením výsledků je vytvoření a uložení ERP komponent funkcí `pop_savemyerp`.

```
ERP = pop_savemyerp(ERP, 'erpname', erpname,
'filename', [erpname, '.erp'], 'filepath', cesta)
```

3.4.2 Grafy

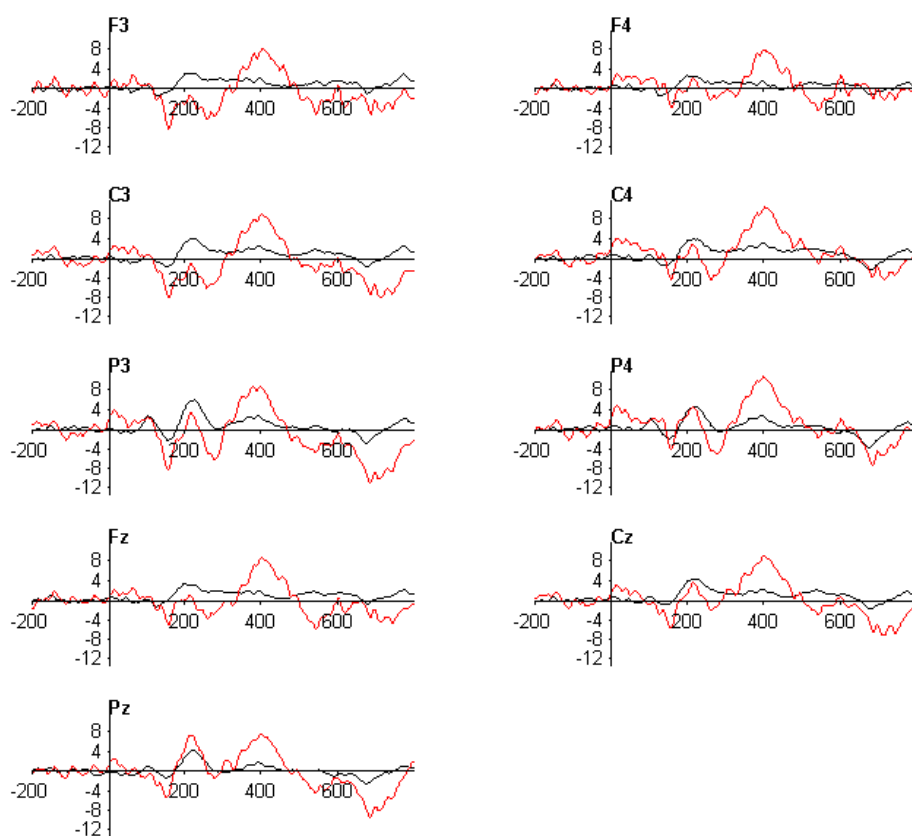
Výsledky skriptu pro zpracování EEG signálu a extrahování ERP komponent jsou vykresleny ve formě grafů, viz Obrázek 3.11. Na tomto obrázku je černě vykreslena non-targetová aktivita a červeně je vykreslena kýžená vlna P3b. Zobrazeno je jen devět z devatenácti elektrod, u kterých je výsledek nejvíce patrný. Jména elektrod jsou shodná se systémem 10/20, viz Obrázek 2.4.



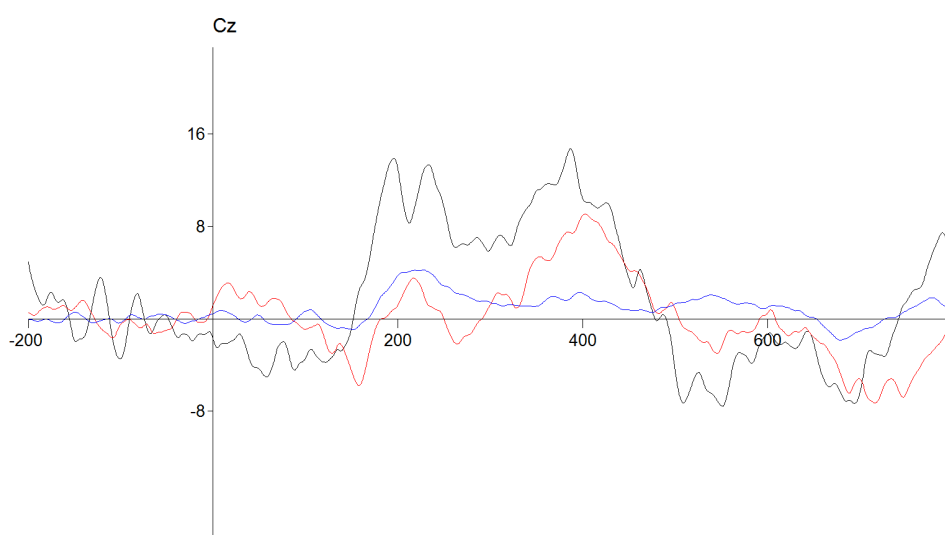
Obrázek 3.10: Signál s vyznačenými epochami s artefakty.

Z obrázku je vidět, že vlna P3b přichází kolem 400 ms po stimulu. Soubor těchto grafů pochází z jednoho měření.

Jen pro zajímavost, v návrhu experimentu bylo zmíněno, že z 5% bliká dioda pro vyvolání vlny P3a. Tato vlna se objevuje těsně přes vlnou P3b a je o něco vyšší, viz Obrázek 3.12. Vlna P3a vzniká ve frontocentrálních oblastech, zatímco vlna P3b vzniká v zadní oblasti mozku. Vlna P3a je v grafu označena černou barvou, vlna P3b je označena červeně a základní aktivita je označena modře.



Obrázek 3.11: Vlna P3b (červeně).



Obrázek 3.12: Vlna P3b (červeně) a P3a (černě).

4 Závěr

Během mojí bakalářské práce jsem se seznámila s problematikou elektroencefalografie a evokovaných potenciálů. Získané poznatky jsem shrnula v teoretické části. Dále jsem se seznámila s funkcemi a rozhraním hardwarového stimulatoru, který slouží pro měření ERP experimentů.

Prvotním cílem bylo vytvoření grafického uživatelského rozhraní pro hardwarový stimulator. K dispozici jsem dostala seznam kódů pro zprávy, které je možno stimulatoru odesílat, konzolovou aplikaci pro kontrolu dat a popis struktury, jak jsou jednotlivá schémata experimentů uložena v paměti stimulatoru. Vytvořené grafické rozhraní slouží pro pohodlnější nastavování parametrů stimulace pro vybrané ERP experimenty. Prozatím je implementována pouze funkčnost pro stimulaci LED diodami, tj. pro zrakovou stimulaci.

Vývoj stimulatoru ještě není ukončen. Pořád se přidávají nové a upravují staré funkcionality, což znamená, že v době čtení práce již nemusí vše odpovídat popisovaným skutečnostem. Tvorbu programu mi ztěžovalo právě upravování funkcionalit, například změny konstant pro odesílané zprávy a další. V průběhu psaní programu jsem tudíž vytvořila spoustu kódu, který byl ve výsledku nepoužitelný nebo musel být optimalizován. Jelikož jsem byla první, kdo stimulator tímto způsobem testoval, bylo vlastně odhalení chyb pro pokrok žádoucí.

Po vytvoření programu bylo mým dalším úkolem zhodnotit naměřené ERP experimenty. Experiment byl zaměřen právě při stimulaci LED diodami, kdy byl subjekt stimulován blikáním tří diodových panelů různých barev. Experiment měl prokázat, zda je u subjektů v kontinuálním EEG záznamu přítomna P3 komponenta, což je odezva mozku na neočekávaný stimul. Po úpravě a zpracování naměřených dat bylo zjištěno, že v signálu je tato komponenta přítomna. Z toho vyplývá, že je stimulator vhodný pro tento druh experimentů.

Do budoucna by bylo možné rozšířit program dále podle toho, jaké funkcionality budou přidávány do stimulatoru, například ovládání pro sluchovou stimulaci.

Literatura

- [1] Elektroencefalogram. Dostupné na <http://cs.wikipedia.org/wiki/Elektroencefalografie>. Naposledy navštíveno 1. prosince 2013.
- [2] 10 – 20 system (EEG). Dostupné na [http://en.wikipedia.org/wiki/10-20_system_\(EEG\)](http://en.wikipedia.org/wiki/10-20_system_(EEG)). Naposledy navštíveno 23. prosince 2013.
- [3] What Is Electroencephalography (EEG). Dostupné na <http://kb4brainfunction.com/what-is-electroencephalography-eeg/>, 28. ledna 2013. Naposledy navštíveno 2. ledna 2014.
- [4] ANT Neuro. *EEProbe Reference Manual*, 2000–2003.
- [5] C. Valens. A Really Friendly Guide to Wavelets. Dostupné na <http://polyvalens.pagesperso-orange.fr/clemens/wavelets/wavelets.html>, 1999–2010. Naposledy navštíveno 1. února 2014.
- [6] Daniel Braun. Differential amplifier. Dostupné na http://commons.wikimedia.org/wiki/File:Differential_Amplifier.svg, 3. června 2007. Naposledy navštíveno 2. ledna 2014.
- [7] Daniel Schwarz. *Lineární a adaptivní zpracování dat*. Akademické nakladatelství CERM, s.r.o., 2012. ISBN: 978-80-7204-779-6.
- [8] Jindřich Ciniburk. Možnosti využití Hilbert-Huangovy transformace pro detekci evokovaných potenciálů. Dostupné na <http://dai.fmph.uniba.sk/events/kuz2009/prispevky-pdf/ciniburk.pdf>, 2009. Naposledy navštíveno 8. ledna 2014.
- [9] Li-Fen Chen, MD. MEG signal processing and localization. Dostupné na http://maki.bme.ntu.edu.tw/wp-content/uploads/2013/05/%E9%99%B3%E9%BA%97%E8%8A%AC_slides.pdf, 2013. Naposledy navštíveno 19. prosince 2013.

- [10] Mgr. Jiří Široký. Nemoc motorického neuronu. Dostupné na http://www.medicina.cz/verejne/clanek.dss?s_id=5590, 4.6.2003. Naposledy navštíveno 28. prosince 2013.
- [11] Mudr. Miroslav Moráň. *Praktická elektroencefalografie*. Institut pro další vzdělávání pracovníků ve zdravotnictví, Brno, 1995. ISBN: 80-7013-203-5.
- [12] Mudr. Petr Kaňovský, CSc., Mudr. Jaroslav Dufek. *Evokované potenciály v klinické praxi*. NCO NZO, Brno, 2000. ISBN: 80-7013-306-6.
- [13] Prof. Miroslav Kuba, M.D., Ph.D. *Motion-onset Visual Evoked Potentials and their Diagnostic Applications*. Nucleus HK, Praha, 2006. ISBN: 80-86225-89-5.
- [14] prof. MUDr. Martin Bareš, Ph.D. Kognitivní evokované potenciály. Dostupné na <http://www.csmn.eu/ceska-slovenska-neurologie-clanek/kognitivni-evokovane-potencialy-36052>, 2011. Naposledy navštíveno 28. prosince 2013.
- [15] Steve Luck, Candace Markley, Javier Lopez-Calderon, Eric Foo, Jason Arit. *ERPLAB Toolbox Manual*, 2013.
- [16] Steven J. Luck. *An Introduction to the Event-Related Potential Technique*. A Bradford Book, 2005. ISBN-10: 0262621967, ISBN-13: 978-0262621960.
- [17] Terence W. Picton, Otavio G. Lins, Michael Scherg. The recording and analysis of event-related potentials. Dostupné na <http://www.sfu.ca/~jmcd/courses/925/papers/Picton%20%281995%29.pdf>, 1995. Naposledy navštíveno 12. ledna 2014.
- [18] Zhi-Min Zhang, Shan Chen, Yi-Zeng Liang. Baseline correction using adaptive iteratively reweighted penalized least squares. Dostupné na http://www.bio.umass.edu/biology/kunkel/LabWiki/images/c/c6/Zhang_Baselin-Analy2010.pdf, 2010. Naposledy navštíveno 12. ledna 2014.
- [19] Brain Products GmbH. *BrainVision Analyzer 2.0 User Manual*, 2008.
- [20] Neurobehavioral Systems Inc. *Dokumentace k softwaru Presentation*, 2003–2010. Dostupné na http://www.neurobs.com/presentation/docs/index_html.

Seznam obrázků

2.1	Ukázka křivky EEG záznamu.	2
2.2	Základní typy mozkových vln.	4
2.3	Schéma diferenciálního zesilovače	7
2.4	Systém rozmístění elektrod 10/20.	8
2.5	Blokové schéma stimulatoru.	11
2.6	Struktura balíčku.	12
2.7	Struktura softwaru Presentation.	14
2.8	Schéma úpravy a zpracování EEG dat.	15
2.9	Korekce baseline.	17
2.10	Proces průměrování EEG dat.	18
3.1	UML diagram balíčků.	21
3.2	Hierarchie grafických komponent.	23
3.3	Strída impulsu.	24
3.4	Změna ISI vůči periodické křivce.	24
3.5	Ukázka panelu pro nastavení parametrů pro stimulaci.	25
3.6	Ukázka panelu pro simulaci blikání diod.	26

3.7	Stavový diagram komunikace mezi stimulátorem a programem.	28
3.8	Spojení dvou osmibitových čísel.	31
3.9	Signál rozdělený na epochy s vyznačenými markery.	36
3.10	Signál s vyznačenými epochami s artefakty.	37
3.11	Vlna P3b (červeně).	38
3.12	Vlna P3b (červeně) a P3a (černě).	39

A Příloha – Definice paketu

Stimulační zařízení pro kognitivní experimenty

Definice paketu pro USB ovládání rev. 0.5

Jednotlivé pakety mají pevnou délku 64B s následující strukturou:

1B ID + LEN
1B MSG TYPE
62B DATA

1. byte – Hlavička

ID 2bity

Rezerva pro další implementaci, v této verzi ponechávat 00

LEN 6bitů

0-62 délka dat v bytech, 1 paket

63 fragment paketu, poslední část fragmentu má délku $\leq 62B$

2. byte – Typ zprávy

0x00 – 0x0F Základní ovládání – nemá žádné parametry (data)

0x00 – Obnovení displeje (REFRESH) – překreslí aktuální hodnoty proměnných na display

0x01 – Povolení LED – povolí blikání LED

0x02 – Zakázání LED – zakáže blikání LED

0x03 – Stav LED – vrátí stav LED

0x04 – Změna stavu náhodnosti – náhodnost mezi intervaly (doba nesvícení je náhodně protažena nebo zkrácena) režim 0 – vypnuto, 1 – zkrácena, 2- protažena, 3 - oboji

0x05 – Stav náhodnosti – vrátí stav náhodnosti

0x06 – Automatický režim - start

0x07 – Automatický režim - stop

0x08 - 0x0E – Rezerva

0x0F – Čtení USB bufferu

0x10 – 0x3F Nastavení stimulátoru

0x10 – Doba svícení LED0 – data 2B, hodnota je v 0.1ms

0x11 – Pauza LED0 – data 2B, hodnota je v 0.1ms

0x12 – Doba svícení LED1 – data 2B, hodnota je v 0.1ms

0x13 – Pauza LED1 – data 2B, hodnota je v 0.1ms

0x14 – Doba svícení LED2 – data 2B, hodnota je v 0.1ms

0x16 – Pauza LED2 – data 2B, hodnota je v 0.1ms

0x16 – Doba svícení LED3 – data 2B, hodnota je v 0.1ms

0x17 – Pauza LED3 – data 2B, hodnota je v 0.1ms

0x18 – Zastoupení LED0 – data 1B, hodnota je v %, min 0 max 100, součet přes LEDx = 100

0x19 – Zastoupení LED1 – data 1B, hodnota je v %, min 0 max 100, součet přes LEDx = 100

0x1A – Zastoupení LED2 – data 1B, hodnota je v %, min 0 max 100, součet přes LEDx = 100

0x1B – Zastoupení LED3 – data 1B, hodnota je v %, min 0 max 100, součet přes LEDx = 100

0x1C – Jas LED0 – data 1B, hodnota je v %, min 0 (nesvítí) max 100

0x1D – Jas LED1 – data 1B, hodnota je v %, min 0 (nesvítí) max 100

0x1E – Jas LED2 – data 1B, hodnota je v %, min 0 (nesvítí) max 100

0x1F – Jas LED3 – data 1B, hodnota je v %, min 0 (nesvítí) max 100

0x20 – Doba sync pulsu – data 2B, hodnota je v ms, max zobrazení na displeji 9999

0x21 – Hrana synchronizačního pulsu – 0 náběžná, 1 sestupná

0x22 – Nastavení frekvence LED0 – data 1B, frekvence periody svítí-nesvítí v Hz

0x23 – Nastavení středu periody LED0 – data 1B, určuje podíl svítí-nesvítí (výchozí 50%)

0x24 – Nastavení frekvence LED1 – data 1B, frekvence periody svítí-nesvítí v Hz
0x25 – Nastavení středu periody LED1 – data 1B, určuje podíl svítí-nesvítí (výchozí 50%)
0x26 – Nastavení frekvence LED2 – data 1B, frekvence periody svítí-nesvítí v Hz
0x27 – Nastavení středu periody LED2 – data 1B, určuje podíl svítí-nesvítí (výchozí 50%)
0x28 – Nastavení frekvence LED3 – data 1B, frekvence periody svítí-nesvítí v Hz
0x29 – Nastavení středu periody LED3 – data 1B, určuje podíl svítí-nesvítí (výchozí 50%)
0x2A – Přehrání zvuku z SD karty – data 8 bytů, název zvuk. souboru bez přípony
0x2B – Nastaví všechny položky aktuálního schématu – data 33B v definovaném formátu
0x2C – Zapnutí LED, data 1B, číslo LED 0 - 3
0x2D – Vypnutí LED, data 1B, číslo LED 0 - 3
0x2E - 0x3F – Rezerva

0x40 – 0x6F Kontrola stimulátoru

0x40 – 0x51 – Vrací nastavenou hodnotu viz Nastavení stimulátoru

0x52 – Velikost synchronizační pauzy, vrací data 2B

0x52 – Procentuální změna doby nesvícení v náhodném režimu, vrací data 1B

0x52 - 0x59 – Rezerva

0x5A – Vrací seznam zvuků (*.WAV) na SD kartě

0x5B – Vrací všechny položky aktuálního schématu – data 33B v definovaném formátu

0x5C - 0x6F – Rezerva

0x70 – 0x7F Práce s pamětí – parametrem je číslo schématu (pointer na stránku v paměti) nebo data v předepsaném formátu

0x70 – Uložení současného nastavení do paměti – uloží data na zadanou stránku

0x71 – Nahrání nastavení z paměti – nahraje data ze zadané stránky do zařízení

0x72 – Uložení dat do paměti – zapíše do paměti celé schéma. 1B dat udává stránku, zbytek dat je samotný obsah v předem definovaném formátu

0x73 – Čtení dat z paměti – přečte z paměti celé schéma dané stránkou a odešle přes USB.

0x74 - 0x7E – Rezerva

0x7F – Čtení celé paměti – přečte obsah celé paměti a odešle přes USB

0x80 - 0xEF – Rezerva

0xF0 – 0xFF DEBUG

Zprávy pro přímý přístup na systémové sběrnice UART, SPI, IIC

Určeno pouze pro účely ladění stimulátoru, není určeno pro další využití a bude odstraněno ve verzi RELEASE.

3. a další byty

Data, typ je definován typem zprávy uvedeným výše. Platná data se rozlišují polem LEN (*délka paketu*), nevyužití byty jsou automaticky doplněny nulami.

Rezervované kódy mohou být použity v dalších verzích jednotky stimulátoru.

B Příloha – Schéma

schema[0] – délka pole
schema[1] – kód odpovědi
schema[2] – index schématu

LED0

schema[3] – horní byte doby zapnutí
schema[4] – dolní byte doby zapnutí
schema[5] – horní byte doby vypnutí
schema[6] – dolní byte doby vypnutí
schema[7] – zastoupení diody
schema[8] – jas diody

LED1

schema[9] – horní byte doby zapnutí
schema[10] – dolní byte doby zapnutí
schema[11] – horní byte doby vypnutí
schema[12] – dolní byte doby vypnutí
schema[13] – zastoupení diody
schema[14] – jas diody

LED2

schema[15] – horní byte doby zapnutí
schema[16] – dolní byte doby zapnutí
schema[17] – horní byte doby vypnutí
schema[18] – dolní byte doby vypnutí
schema[19] – zastoupení diody
schema[20] – jas diody

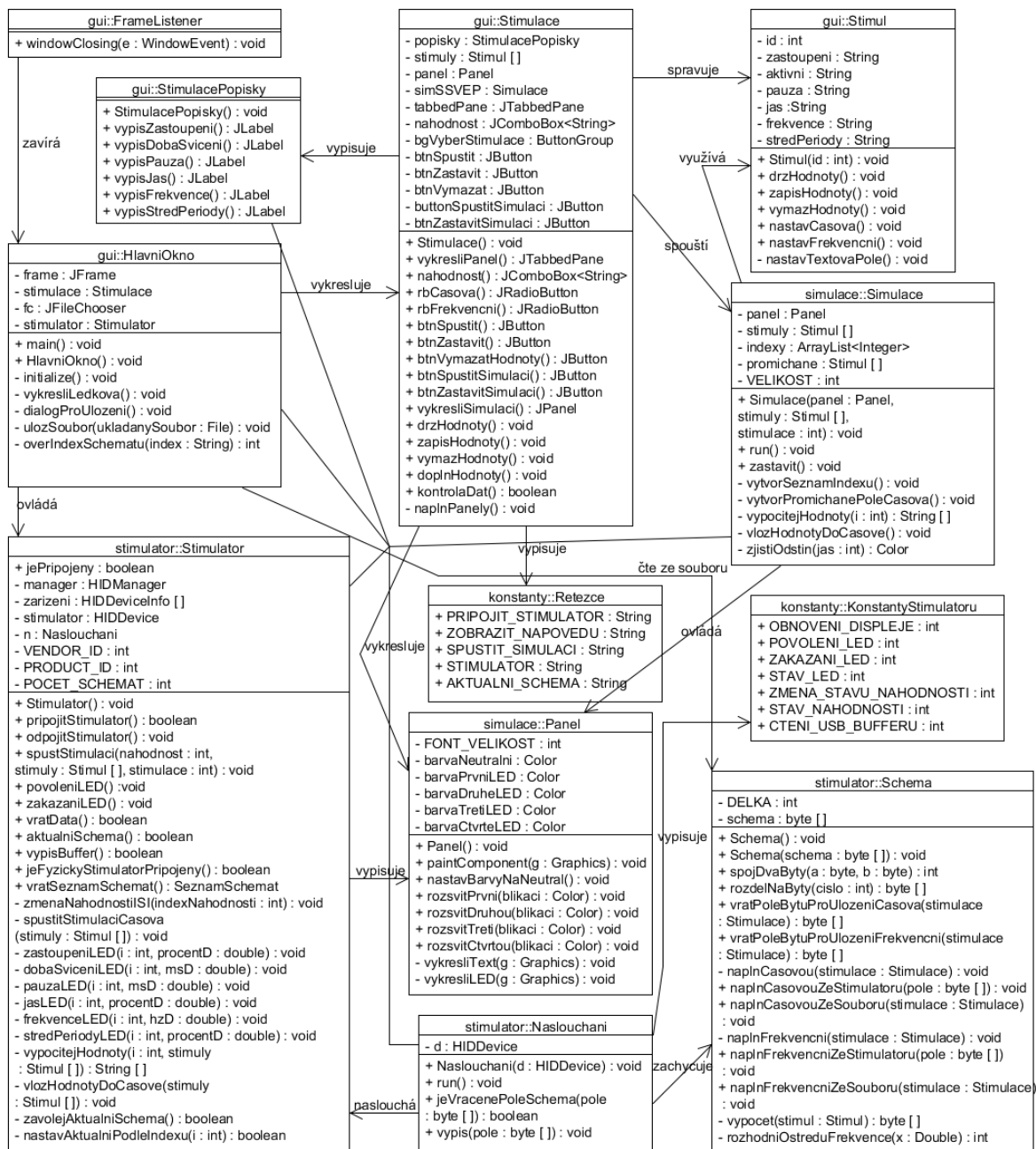
LED3

schema[21] – horní byte doby zapnutí
schema[22] – dolní byte doby zapnutí
schema[23] – horní byte doby vypnutí
schema[24] – dolní byte doby vypnutí
schema[25] – zastoupení diody
schema[26] – jas diody

schema[27] – horní byte délky synchronizačního pulsu
schema[28] – dolní byte délky synchronizačního pulsu
schema[29] – hrana pulsu (nabývá hodnot 0 nebo 1)
schema[30] – rezervováno
schema[31] – závislosti LED0

schema[32] – závislosti LED1
schema[33] – závislosti LED2
schema[34] – závislosti LED3
schema[35] – horní byte pauzy mezi synchronizačními pulsy
schema[36] – dolní byte pauzy mezi synchronizačními pulsy
schema[37] – náhodnost v pauze vypnutí LED, hodnoty: 0 – nic (zakázáno),
1 – kratší, 2 – delší, 3 – kratší i delší
schema[38] – náhodnost v pauze vypnutí LED, velikost přidané/ubrané doby
k pauze

C Příloha – UML diagram tříd



D Příloha – Uživatelská dokumentace

Úvod

Tento uživatelský manuál slouží jako příručka k programu pro nastavení parametrů hardwarového EEG stimulátoru a jako doplňkový materiál k mé bakalářské práci *Test HW stimulátoru pro měření ERP experimentů*.

Spuštění aplikace

Aplikace je psaná v programovacím jazyce Java ve verzi 7, takže pro správné spuštění je potřeba mít Javu nainstalovanou. Ke stažení je k dispozici na stránce <https://www.java.com/en/download>.

Pokud je Java správně nainstalovaná a nastavená, půjde program spustit buď dvojklikem na ikonu nebo přes příkazový řádek příkazem

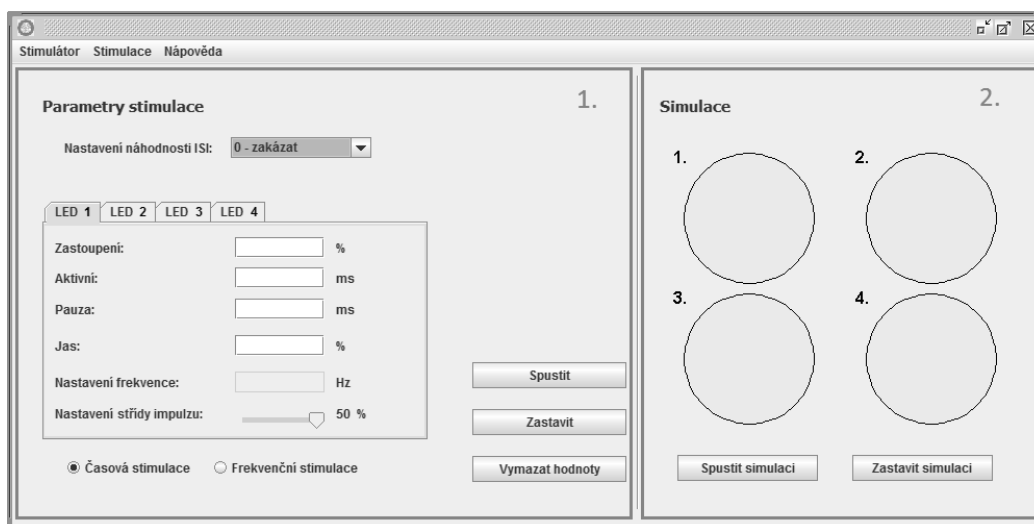
```
java -jar nazev.jar
```

Připojení stimulátoru

Součástí programovatelného stimulátoru je několik ledkových panelů a síťový zdroj. Stimulátor má USB vstup na připojení čtyř panelů. Před použitím je nutné stimulátor připojit do elektrické sítě. Kabel zdroje je zakončen rozdvojenou koncovkou. Jedna koncovka je klasická adaptérová a druhá koncovka je USB B konektor. Pro práci se stimulátorem bez uživatelského rozhraní se zapojují obě tyto koncovky. Pro práci s uživatelským rozhráním se připojí pouze adaptérová koncovka a pro spojení počítače se stimulátorem se použije A-B USB kabel, klasický kabel pro připojení například tiskáren. Že je stimulátor připojen správně, poznáme podle rozsvíceného displeje se základní nabídkou.

Popis GUI

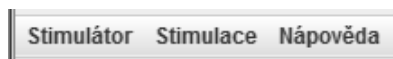
Po spuštění aplikace se zobrazí grafické rozhraní, viz Obrázek D.1. Toto rozhraní je rozděleno na dvě části. První část slouží k zadávání parametrů pro stimulaci a druhá pro simulaci běhu podle těchto parametrů.



Obrázek D.1: Grafické rozhraní programu.

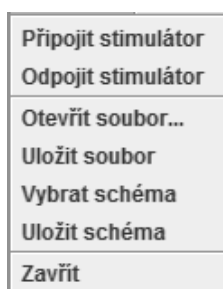
Menu

Horní lišta aplikace obsahuje menu pro akce, které nejsou spojeny s nastavováním parametrů stimulatoru, viz Obrázek D.2. Menu obsahuje záložku pro zobrazení nápovědy. Po kliknutí na záložku se zobrazí samostatné okno s krátkou nápovědou. Další záložka *Stimulace* slouží pro výběr stimulace, která bude použita. V současné době obsahuje pouze výběr *Ledková*.



Obrázek D.2: Menu.

Poslední záložka *Stimulátor* obsahuje submenu pro práci se stimulatorem, viz Obrázek D.3. Jsou zde výběry pro připojení a odpojení stimulatoru. Další sekce obsahuje položky pro práci se schémata, která lze načítat ze stimulatoru nebo ze souboru. Poslední položkou je zavření programu. Zavřít program lze



Obrázek D.3: Submenu.

i v horním pravém rohu křížkem. Při zavření se automaticky ukončí spojení se stimulátorem, o čemž je uživatel informován.

Otevírání a ukládání souboru

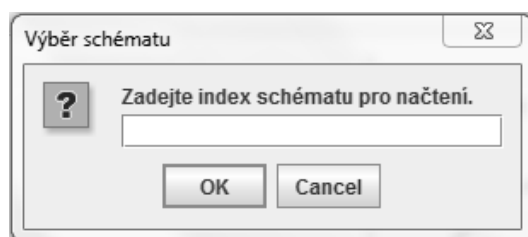
Při kliknutí na položku *Otevřít soubor...* se zobrazí dialog pro otevření souboru. Je možné otevřít pouze soubor s příponou `.bin`. Pro každý soubor v jiném formátu se zobrazí chybová hláška. Pokud je vybrán správný formát souboru, vyplní se data do příslušných polí. Při ukládání se do souboru uloží i informace, o jaký typ stimulace se bude jednat, časová nebo frekvenční. Při vyplňování polí pro frekvenční stimulaci dochází k přepočítávání hodnot z časové. V rámci těchto výpočtů může docházet vlivem zaokrouhlovacích chyb k malým chybám v hodnotách, proto je doporučováno, aby si uživatel data zkontroloval.

Pokud chce uživatel svá data uložit, klikne na položku *Uložit soubor*. Zobrazí se podobný dialog jako při otvírání, ale zde uživatel zadá jméno souboru. Může, nebo nemusí, zadávat i příponu. Pokud nebude zadána přípona, doplní se automaticky. Doporučuje se pojmenovávat soubory co nejpřesněji.

Otvírání a ukládání schématu

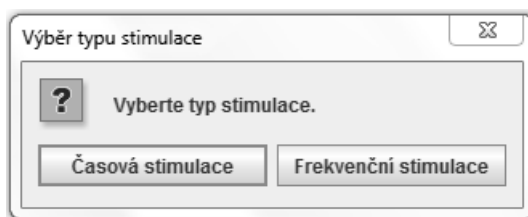
Při kliknutí na položku *Vybrat schéma* se zobrazí dialog, viz Obrázek D.4, kde uživatel zadá index schématu, které chce ze stimulátoru načíst. Schéma si lze vybrat na displeji stimulátoru, kde se dají všechna schémata libovolně procházet. Aby bylo zpět odesláno správné schéma, je potřeba ho v nabídce vybrat na displeji stimulátoru.

Index musí být v rozmezí 1–20, protože do paměti stimulátoru je možné uložit dvacet schémat. Pokud se uživatel pokusí zadat neplatný index, bude



Obrázek D.4: Dialog pro zadání indexu.

upozorněn chybovou hláškou. Po zadání indexu se do stimulatoru odešle požadavek. Pokud bude tento požadavek správně zpracován, vrátí stimulator data z požadovaného indexu. Pokud budou k dispozici nějaká data, bude uživatel vyzván dialogem, viz Obrázek D.5, k výběru typu stimulace. Přijatá data budou ve vhodném formátu vložena do textových polí.



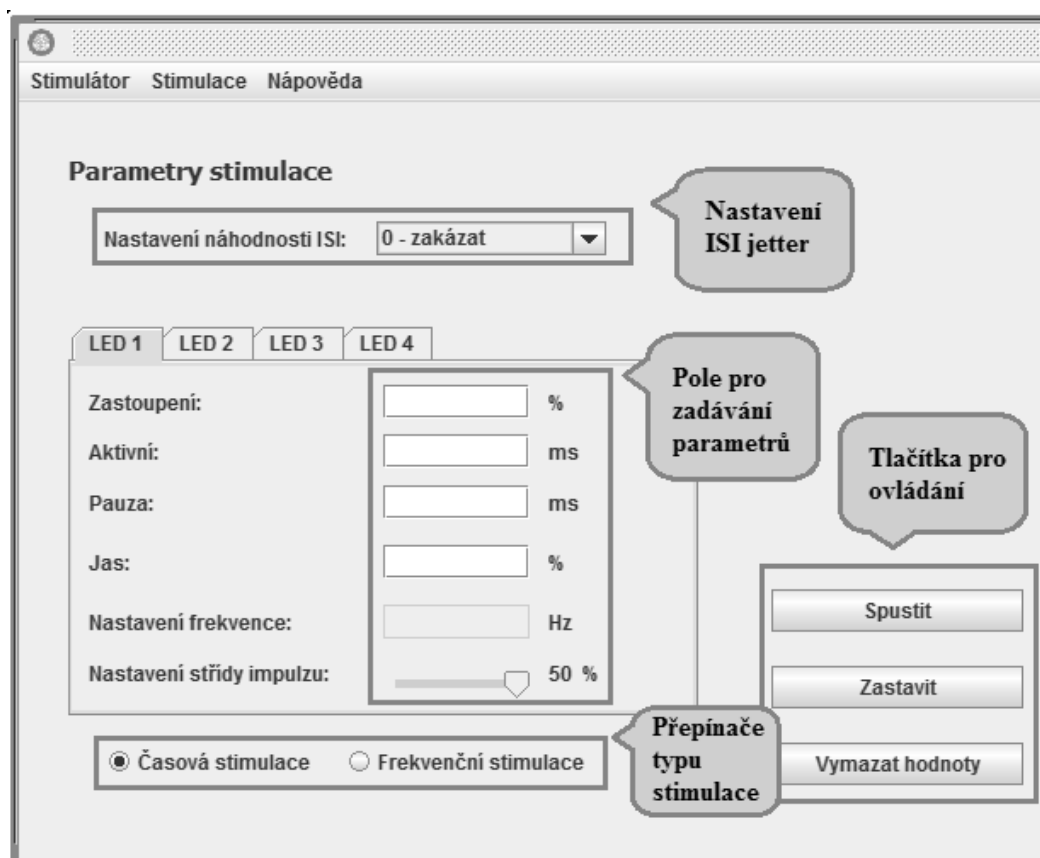
Obrázek D.5: Dialog pro výběr stimulace.

Pro ukládání hodnot z programu do stimulatoru vybere uživatel z menu položku *Uložit schéma*. Následně bude vyzván, aby zadal index, na který se schéma ve stimulatoru uloží. Předpokládá se, že si uživatel prošel nabídku schémat na displeji stimulatoru a ví, kam přesně chce svá data uložit, aby nedošlo k přepsání uložených dat jiného uživatele.

Nastavení parametrů

Při nastavování parametrů je k dispozici několik možností, viz Obrázek D.6.

První nastavení se týká intervalu ISI jitter. Možnosti nastavení tohoto intervalu jsou čtyři – zakázání, kratší interval, delší interval nebo kratší i delší. Používá se, pokud na vstupu přichází periodický signál, příkladem jsou alfa vlny. Stimuly jsou také prezentovány v konstantním čase, takže by se mohlo stát, že stimul bude opakován v určitém cyklu vždy na stejném místě periodické křivky, což by mohlo znehodnotit měření. Aby se zabránilo takovému



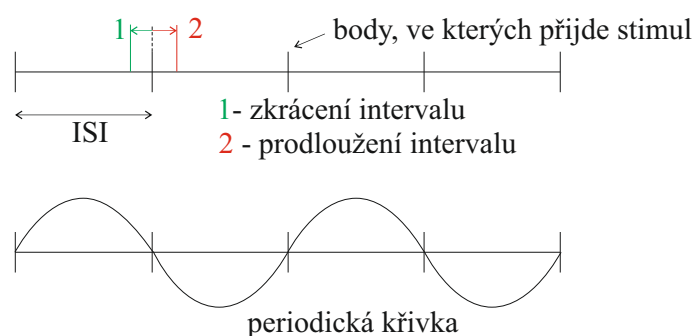
Obrázek D.6: Část pro nastavování parametrů stimulace.

znehodnocení měření, tak se prodlužuje nebo zkracuje interval doby nesvícení, viz Obrázek D.7. Tento interval se pohybuje v rozmezí ± 50 ms.

Druhým nastavením je výběr stimulace, viz Obrázek D.6 dole. Možnosti jsou časová nebo frekvenční.

Stimulátor má čtyři USB porty pro připojení až čtyř diodových panelů. Každý panel se dá nastavit samostatně, proto jsou vytvořeny odpovídající záložky pro LED 1 až LED 4. Podle výběru typu stimulace se zpřístupní pouze ta textová pole, která jsou pro tento druh stimulace určena. Pro časovou stimulaci lze zadávat zastoupení diod, dobu aktivity a pauzy svícení a jas diod. Pro frekvenční stimulaci lze zadávat jas diod, frekvenci a střidu impulzu. Pro hodnoty každého textového pole existují omezení:

- Zastoupení – Součet zastoupení všech čtyř diod dohromady musí být



Obrázek D.7: Změna ISI vůči periodické křivce.

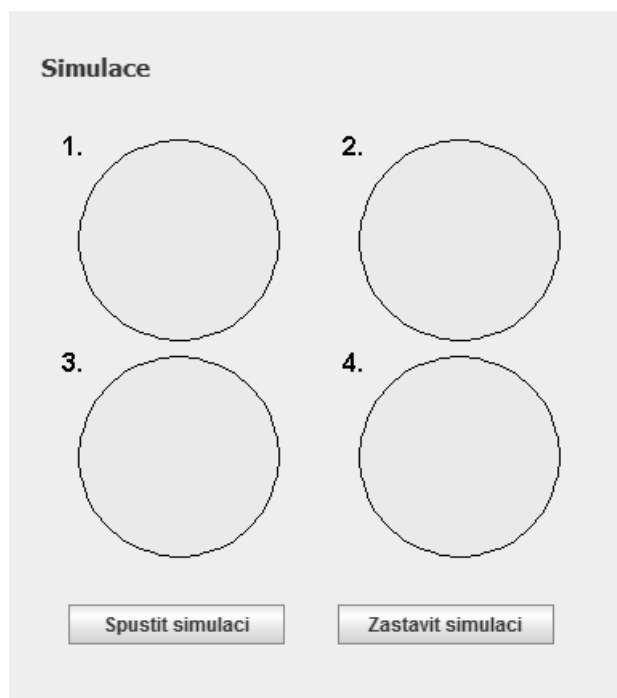
rovno 100. Musí být zadáno. Pokud není, počítá se automaticky nulové zastoupení.

- Doba aktivity – Musí být zadána, pokud je zastoupení nenulové.
- Doba pauzy – Musí být zadána, pokud je zastoupení nenulové.
- Jas – Musí být v rozsahu 0–100. Pokud není zadán a zastoupení je nenulové, doplní se automaticky na hodnotu 50.
- Frekvence – Musí být v rozsahu 0–30. Pokud není zadána, počítá se automaticky nulová frekvence, při které nebude docházet k žádnému blikání.
- Střída impulzu – Hodnota je daná hodnotou na posuvníku, který je v rozmezí 5–50.

Po nastavení všech parametrů podle potřeby je možné tlačítkem *Spustit* odeslat tyto parametry do stimulátoru. Před odesláním je nutné stimulátor připojit. Pokud k tomu ještě nedošlo, je uživatel upozorněn informační hláškou. Stimulátor se připojuje v submenu položkou *Připojit stimulátor*, viz Obrázek D.3. Data jsou před odesláním podle daných podmínek testována. Pokud nevyhoví, zobrazí se varovná hláška. Pokud vyhoví podmínkám, jsou odeslána do stimulátoru, který následně data zpracuje a interpretuje. Zastavení blikání je možné tlačítkem *Zastavit*. Poslední tlačítko *Vymazat* slouží k vymazání všech hodnot ve všech polích, tj. k nastavení výchozích hodnot rozhraní.

Simulace

Pro představu, jak bude simulace po odeslání do stimulatoru probíhat, slouží panel *Simulace*, viz Obrázek D.8. Tento panel je rozdělen na oblast, která imituje jednotlivé diody (diodové panely), a tlačítka pro spuštění a zastavení simulace. Simulace se snaží co nejvěrněji napodobit skutečný průběh blikání diod. Existují zde ale časová omezení na hodnotu frekvence. Při frekvenci vyšší než cca 15 Hz může simulace probíhat zkresleně, trhaně.



Obrázek D.8: Panel simulace.