

Комуникационна среда на човеко-машинен интерфейс със слушалки MindSet

Георги Кръстев

Communication Medium of Human-machine Interface with headphones: The paper presents the Communication Medium of Human-machine Interface with headphones MindSet. The MindSet Communication Protocol has been described in Detail. An Algorithm and a particular Pattern for Processing the incoming Information has been presented.

Key words: Brain-Computer Interaction Interface (BCII), electroencephalography (EEG), NeuroSky, MindSet.

ВЪВЕДЕНИЕ

Интерфейсът човек-компютър чрез анализ на активността на мозъка (Brain-Computer Interaction Interface - BCII) е създаден през последните години. Става дума за уникалното взаимодействие между човек и компютър, което се осъществява след регистриране и обработка на сигнали с произход от човешкия мозък (най-често това са електроенцефалографските сигнали - ЕЕГ).

С повишаването на интереса към BCII, възниква въпросът с достъпността на технологиите нужни за имплементацията на този интерфейс.

В статията е представена комуникационната среда на сензор за неинвазивен BCII на базата на слушалки MindSet.

КОМУНИКАЦИОННА СРЕДА НА СЛУШАЛКИ MINDSET

На фиг.1 е представена схема на комуникацията между сензора и персонален компютър. ThingGear Communication Driver, представлява DLL библиотека. Тя съдържа нужните функции за обработка на данните, от серийният порт, по дефиниран протокол, както и достъп до API функционалността на драйвера. Достъпът до API функциите, при програмиране на C# става посредством InteropServices (виж Пример).



Фиг. 1 – Схема на комуникация

Производителят предоставя, както DLL библиотека, така и хедър файлове за програмиране на C/C++. Разликата между двете е в това, че вместо да се налага да се обръщаме към драйвер, при програмира на C/C++, четенето на данните става директно от данновият поток идващ от COM порта. Данните биват изпращани по специален протокол, разработен от производителят.

Пример: Функция използвана за свързване към слушалките

```
[DllImport ("ThinkGear")] public static extern int
TG_Connect( int connectionId, string serialPortName, int serialBaudrate,
int serialDataFormat );
```

Bluetooth Interface

Слушалките MindSet предават ThinkGear Data Values, кодирани в ThinkGear пакети, като сериен байтов поток, чрез Bluetooth сериен порт (SPP).

ThinkGear Data Values

- POOR_SIGNAL Quality (Качество на сигнала) – този тип данни представляват цяло число, без знак (unsigned int), което указва нивото на сигнала измерено от драйвера. Числото е в интервала от 0 до 200. 0 оказва, че няма смущения, а всяко различно число от 0 или 200 оказва, че са налични смущения. Ако числото е 200, това означава, че сензорите на слушалките не са в допир с кожата на потребителя. Стойността бива извеждана веднъж на всяка секунда.

Забележка: Състоянията „Съсредоточеност“ и „Медитация“ не се отчитат при показание на POOR_SIGNAL различно от 0.

- eSense™ измервания – Всички eSense измервания („Съсредоточеност“ и „Медитация“) се измерват на относителна скала от 1 до 100.
- Attention eSense („Съсредоточеност“) – без знакова еднобайтова стойност отчитаща нивото на внимание, съсредоточеност, на потребителя. Стойността е в границите от 1 до 100 и се обновява всяка секунда.
- Meditation eSense („Медитация“) – без знакова стойност с големина един байт, отчитаща нивото на умственото спокойствие, релаксация на потребителят. Стойността е в границите от 1 до 100 и се извежда веднъж в секунда
- RAW Wave („Сурови“ ЕЕГ данни) – състои се от два байта, представляващи единична проба на „сурови“ данни. Стойността на тези данни представлява 16-битово цяло число със знак, представено в допълнителен код. Интервалът е от -2048 до 2047. Стойностите се извеждат по 512 пъти в секунда.
- ASIC_EEG_POWER – представлява информацията на 8 известни вида мозъчни вълни. Стойността се извежда като серия от 3 байтови цели числа, без знак, във формат „Little-endian“. Осемте вълни се извеждат в следната последователност: делта (0.5 - 2.75Hz), тета (3.5 - 6.75Hz), алфа-ниски (7.5 - 9.25Hz), алфа-високи (10 - 11.75Hz), бета-ниски (13 - 16.75Hz), бета-високи (18 - 29.75Hz), гама-ниски (31 - 39.75Hz) и гама-високи (41 - 49.75Hz). Тези стойности нямат мерни единици и затова са носят информация, ако се сравняват едни с други или със самите себе си, за определяне на относително количество или временни колебания.
- Честота на премигване – едно байтова без знакова стойност, с интервал от 1 до 255, отчита интензитета на премигване с очи. Извежда се когато се регистрира премигване. Няма мерна единица.

ThinkGear пакети

Слушалките предоставят данните под формата на асинхронен сериен поток от данни. Потокът трябва да се обработи и интерпретира като ThinkGear пакети за да може правилно да се извлекат описани стойности. Един ThinkGear пакет се състои от 3 части:

1. Header
2. Payload
3. Checksum

Структура на пакета

Пакетите биват изпратени от слушалките като сериен поток от данни. Средата за транспорт може да бъде USB, COM, Bluetooth, file или всяка друга с възможност

за пренос на потока.

В началото на всеки пакет се намира Header (вид заглавна част), след него следва сегментът Data Payload (съдържа стойностите, на описаните в горната глава данни) и завършва с Checksum (байт потвърждаващ интегритета на данните в Payload), както е показано на фиг. 2.



Фиг. 2 – Структура на ThinkGear пакет

Големината на секцията [PAYLOAD] е разрешено да е максимално 169 байта, а секциите [SYNC], [PLENGHT] и [CHKSUM] са по един байт всяка. Според което един цял, валиден пакет може да има минимална големина от 4 байта в случай, че [PAYLOAD] е 0 (т.е. празен) и максимална големина от 173 байта.

- **Header** фрагмент – той се състои от три байта: 2 байта за синхронизация (0xAA, 0xAA) последвани от [PLENGHT] байт (дължината на фрагмента Payload). Поставени са два байта за синхронизация в случай, че фрагментът Payload започва със стойност 0xAA. [PLENGHT] байтът се използва за индикация на дължината на Payload фрагментът и стойността му е в границите от 0 до 169, по-висока стойност е индикатор за грешка. Да се има предвид, че [PLENGHT] съдържа дължината на Payload, а не на целият пакет. Дължината на целият пакет винаги ще бъде [PLENGHT] + 4.
- **Payload** фрагмент – този фрагмент на пакета представлява серия от данни. Броят на байтовете данни е обозначен в [PLENGHT] байта на Header фрагментът. Интерпретацията на данните може да се разгледа в *Error! Reference source not found.* по-надолу.
- **Checksum** – [CHKSUM] байтът се използва за верификация на интегритетът на Payload фрагментът. Checksum фрагментът на пакетът се дефинира по следният начин:
 - Сумират се всички байтове от Payload.
 - Вземат се последните 8 бита от сумата.
 - Прави се инверсия на битовете.

Получаващата страна трябва да следва тези стъпки за да изчисли стойността на Checksum, за получения Payload и след това да сравни резултата с [CHKSUM] от получения пакет. Ако двете стойности не съвпадат, пакета трябва да се смята за невалиден, ако те съвпадат, може да се продължи към обработка на данните от Payload

Ако полученият пакет е валиден, данните от Payload фрагмента му трябва да се извлекат. Фрагментът съдържа разгледаните Data Values и самите им стойности. Начинът за определяне е следният: за по-лесна обработка, разглеждаме два пакета, извлечени в шестнайсетична бройна система от COM порта, на който са свързани слушалките.

Пакет 1

AA AA 04 80 02 FF EB 93
[1] [2] [3] [4] [5] [6] [7] [8]

Пакет 2

AA AA 20 02 00 83 18 0D 22 56 02 EE 8D 01 9A 1C 00 30 45 00 3D FD
[1] [2] [3] [4] [5] [6] [7] [8] [9] [10][11][12][13][14][15][16][17][18][19][20][21][22]
00 59 FD 00 20 93 01 EA 7D 04 04 05 2C 50
[23][24][25][26][27][28][29][30][31][32][33][34][35][36]

Алгоритъм за обработка

1. Преглеждат се първите два байта на пакета (трябва да са AA AA)
2. Ако стъпка 1 е изпълнена коректно, се преглежда третият байт – указващ дължината на Payload фрагмента. В Пакет 1 той е 0x04 байта, в Пакет 2 е 0x20 байта, 4 и 32 съответно.
3. Преглежда се следващия байт от пакета [4] (началото на Payload фрагмента). Стойността на този байт е обозначаващ, като значението и е посочено в табл. 1. Ако стойността на този байт е по-голяма от 0x7F, това означава, че стойността на типа данни е съхранен в повече от един байт. А дължината на серията се определя в следващия байт.

Табл. 1 – Структура на Payload фрагмента. Дефиниции за CODE

	[CODE]	(byte) [Length]	Тип Данни (ThinkGear Data Values)
Едно байтови [CODE]	0x02	–	POOR_SIGNAL
	0x04	–	Съсредоточеност
	0x05	–	Медитация
	0x16	–	Сила на мигане
Много байтови [CODE]	0x80	2	RAW Wave
	0x83	24	ASIC_EEG_POWER
	0xAA	–	Запазено за [SYNC]

Ако разгледаме Пакет 2 се вижда, че след двата синхронизиращи байта [1] и [2] в байт [3] е записана дължината на Payload фрагментът (0x20 което е 32 байта). Байт [4] указва (според Таблица 5 0x02 е POOR_SIGNAL), че следващият [5] байт съдържа стойността на POOR_SIGNAL, която в дадения пакет е 0. Байт [6] е със стойност 0x83, което е по-голямо от 0x7F, следователно следващия байт [7] указва дължината на серията. В случая серията е ASIC_EEG_POWER и има дължина от 0x18 (24 байта). Вече е известно, че серията ASIC_EEG_POWER съдържа стойностите на 8-те вида мозъчни вълни, като стойността на всяка се разполага в три байта. Следователно байтове [8], [9] и [10] съдържат стойността на Делта, байтове [11], [12] и [13] съдържат стойността на Тета и т.н. Байт [32] указва, че следващата стойност (байт [33]) съдържа стойността на „Съсредоточеност“, в случая 0x04. Байт [34] обозначава следващия тип данни – „Медитация“, чиято стойност се намира в байт [35].

4. След като данните са извлечени стойностите на всички байтове от фрагментът Payload следва да се сумират. От полученият резултат се вземат последните 8 бита и се прилага инверсия върху тях. Полученото число се сравнява с последният байт (CHKSUM фрагментът) от пакета. При еднаквост пакета се запазва, в противен случай, трябва да бъде изхвърлен

Ако разгледаме точка 4 и вземем Пакет 1, байтове на Payload са [4], [5], [6] и [7]. При сумирането им се получава 026C(16) (0010 0110 1100(2)). След като приложим инверсия върху последните 8 бита се получава следното число 1001 0011(2) = 93(16), което съпада с последният байт [8] на Пакет 1.

ЗАКЛЮЧЕНИЕ

В статията е разгледана комуникационната среда на човеко-машинен интерфейс със слушалки MindSet. Подробно е описан MindSet Communication Protocol. Представен е алгоритъм и конкретен пример за обработка на постъпващата информация.

Важно е да се отбележи, че ограниченията наложени от слушалките водят до ниската честота на извличане на данни, опресняването на данни се извършва по

веднъж на секунда за всички мозъчни вълни и eSense състояния. Възможна е и имплементацията на изобразяване на „суровите“ EEG данни. За съжаление производителят не предоставя информация тази възможност.

ЛИТЕРАТУРА

[1] Karray, F et al. Human-Computer Interaction: Overview on State of the Art. International Journal On Smart Sensing and Intelligent Systems, Vol. 1, No.1, March 2008.

[2] Preece, J., Rogers, Y. & Sharp, H. (2002) Interaction Design: Beyond Human-Computer Interaction. New York, NY: John Wiley & Sons.

[3] Preece, J.; Rogers, Y.; Sharp, H.; Benyon, D.; Holland, S. & Carey, T. Human-Computer Interaction. Wokingham (GB): Addison-Wesley.

[4] <http://www.emotiv.com/>

[5] <http://www.neurosky.com/>

За контакти:

доц. д.н. Георги Кръстев, Катедра “Компютърни системи и технологии”, Русенски университет “Ангел Кънчев”, тел.: 082-888 672, e-mail: gkrastev@ecs.uni-ruse.bg

Докладът е рецензиран.