
Название проекта: «Андроид-Мышь»

Введение

Проект *Андроид-Мышь* позволяет использовать ваш телефон, под управлением ОС Андроид, в качестве беспроводной блютуз-мышь. Этот документ описывает структуру проекта и объясняет методы работы его реализации.

Проект примечателен тем, что не требует установки стороннего программного обеспечения на компьютере с которым будет взаимодействовать, то есть на компьютер не надо устанавливать каких либо программ для правильной работы проекта. Это означает что телефон с установленной программой «Андроид-Мышь» будет взаимодействовать с любой ОС, что делает его универсальным решением для дистанционного позиционирования мыши. Также нужно отметить удобство управления курсором, для перемещения которого требуется одна рука.

- Скачать исходный код можно по этой ссылке:

<https://github.com/Arseny-N/Android2Mouse/archive/master.zip> .

- Узнать больше о процессе установки можно пройдя по этой ссылке:

<http://arseny-n.github.io/Android2Mouse/doc/howTo.html> .

- Увидеть пример применения данного проекта на практике можно, пройдя по этой ссылке:

<http://arseny-n.github.io/Android2Mouse/doc/index.html> .

- Просмотреть код можно пройдя по этой ссылке:

<https://github.com/Arseny-N/Android2Mouse> .

Оглавление.

1. Цель проекта.
2. Реализация проекта.
 - 2.1 Принципы работы технологии Bluetooth
 - 2.1.1 HIDP.
 - 2.1.2 SDP.
 - 2.1.3 Код класса.
 - 2.1.4 L2CAP.
 - 2.2 Метод реализации проекта
 - 2.2.1 hidp_clientd.
 - 2.2.2 Графический интерфейс.
 - 2.2.3 Attribute Changer.
 - 2.2.4 Сеанс работы.
 - 2.3 Процесс разработки, установки и компиляции.
 - 2.3.1 Используемые программы.
 - 2.3.2 Процесс разработки.
 - 2.3.3 Процесс компиляции.
 - 2.3.4 Процесс установки.
 - 2.4 Описание файлов проекта.
3. Результат проекта.
4. Литература.

1 Цель проекта: сделать альтернативный манипулятор, заменяющий классическую компьютерную мышь, которая при некоторых обстоятельствах не удобна.

Данный манипулятор должен отвечать следующим критериям:

- Не требовать поверхности для передвижения курсора.
- Умещаться в одной руке и быть полностью управляемым ею.
- Не иметь привязок к какой либо ОС.

2.1 Принципы работы технологии Bluetooth

Чтобы полнее понять то, как *Андроид-мышь* выполняет свою работу, читатель должен быть ознакомлен с принципами технологии Bluetooth. Эта часть документа посещена введению читателя в эту среду. Напомним, что документ ставит перед собой цель описать структуру проекта *Андроид-Мышь*, а не принципы работы *Bluetooth*, поэтому многое будет упрощено или упущено в силу своей малой значимости в данном контексте. На протяжении всего документа автор будет отсылать читателя к этой секции, т.к. технологии и термины, описанные здесь, являются ключевыми в работе программы.

2.1.1 HIDP

Любое *периферийное* устройство, которое использует в виде канала связи *Bluetooth*, должно отвечать спецификации *HIDP (Human Interface Device Profile)*. Данная спецификация описывает протоколы, которые должно использовать устройство, и вводит некоторые критерии. Спецификация была создана с опорой на схожий стандарт *USB*, также направленный на периферийные устройства. В нашем случае, мы будем рассматривать критерии, выдвигаемые к *Bluetooth* мыши.

Чтобы устройство распознавалось как мышь и могло выполнять роль таковой оно должно:

- Иметь соответствующий *SDP* рекорд.
- Иметь ID класса соответствующий нужному устройству.
- Должно уметь установить *L2CAP* связь с *хостом**.

* Далее мы будем использовать термин *хост* в качестве обозначения компьютера, с которым взаимодействует *Андроид-мышь*, такое обозначение встречается в разных спецификациях Bluetooth и удобно в силу своей лаконичности.

Первый и последний пункты являются ключевыми, без их выполнения устройство не сможет

выполнять поставленные перед ним задачи. Выполнение второго пункта зависит от *хоста*, но в общем случае его выполнением пренебрегать не стоит. Далее мы подробно рассмотрим каждый пункт.

2.1.2 SDP

SDP протокол даёт возможность приложениям выяснять, какие сервисы устройства доступны и их характеристики. Мы не будем подробно описывать *SDP* протокол, ограничимся только кратким изложением его сути.

Любое устройство, предоставляющее какие либо сервисы *Bluetooth* должно иметь *SDP* сервер, который по запросу другого устройства будет отправлять списки сервисов и их характеристики запрашивающему клиенту. По сути, все обыденные функции, выполняемые *Bluetooth* устройством, описываются такими записями.

SDP запись устройства, соответствующего спецификации *HIDP*, довольно сложна, упомянем тут только о специальном поле, именуемое в спецификации как *HID* запись.

Схожесть в названии не случайна, как было сказано ранее *HIDP* берёт свои корни из *USB*, там запись играет ту же роль, что и в *Bluetooth*, а именно описывает устройство и структуру пакета посылаемого устройством при появлении информации.

Если объяснить на примере *HID* рекорд говорит следующее: «Это устройство - мышь, у неё три кнопки, в первых двух байтах будет информация о её нахождении, а в последнем информация о состоянии её кнопок».

2.1.3 Код класса

Любое *Bluetooth* устройство имеет некоторые атрибуты, которые считываются при его обнаружении, одним из таких является код (тут речь идёт о более низком уровне, нежели *SDP*, путать эти атрибуты с атрибутами в *SDP* рекорда не стоит). Его предназначение — дать возможность ОС быстро выяснять тип обнаруженного устройства.

2.1.4 L2CAP

L2CAP протокол предоставляет услуги по работе с данными, в *Bluetooth* он является основой для более высокоуровневых протоколов типа *HIDP*.

Если привести аналогию с TCP: «IP адресом» устройства является *MAC*, а «портами» являются каналы. У каждого канала есть свой номер или *PMS*. Периферийное устройство желающие установить связь с неким хостом и начать трансляцию пакетов должно, согласно спецификации *HIDP*, открыть два канала с *PSM* равными 0x13 и 0x12.

2.2 Метод реализации проекта.

В этой секции документа читатель будет ознакомлен основными частями проекта и с причинами такого деления. Также в конце для большей ясности мы опишем сеанс работы с данным проектом.

“Андроид-мышь” можно разделить на три основные части.

- *hidp_clientd* или демон (расшифровку термина см. ниже).
- Графический интерфейс.
- Attribute Changer.

Чтобы более полно объяснить тонкости работы программы далее будут использоваться следующие термины:

- **Исполняемый файл** или **бинарник**- файл, содержащий программу в виде, в котором она может быть (после загрузки в память и настройки по месту) исполнена компьютером. В данном контексте мы не будем относить этот термин к java бинарникам во избежание путаниц.
- **Графическое приложение** или **приложение** - приложение, отображающее информацию на экране телефона в понятном для пользователя виде.
- **Демон** - это специальная программа, работающая без контролирующего терминала. По иному, программа, выполняемая в фоновом режиме. В нашем контексте не принципиально различать демона и обычную программу, но для правильности мы всё-таки будем.

2.2.1 *hidp_clientd*

hidp_clientd это – демон, устанавливающий *L2CAP* соединение и отсылающий поступающую к нему информацию (Подробнее о *L2CAP* см. 2.1.4), он хранится в виде бинарника в памяти телефона и запускается графическим интерфейсом при надобности. Установление соединения осуществляется при помощи открытия двух сокетов *L2CAP*, с соответствующими *PSM*.

Открытие сокета – операция требующая прав супер-пользователя, следовательно телефон на котором запущен *hidp_clientd* должен иметь софт, позволяющий запускать приложения с правами супер-пользователя, или по иному быть “рутнутым”.

hidp_clientd отдельный бинарник, потому что стандартная программная среда Андроид не

позволяет ни запускать приложения с правами супер-пользователя, ни создавать сокеты.

2.2.2 Графический интерфейс.

Графический интерфейс – это приложение, которое считывает прикосновения пользователя и интерпретирует их, предварительно запустив демон. Это делается при помощи специальной java библиотеки.

Далее данные записываются в специальный файл – *fifo* , откуда их позже считывает демон. При завершении сеанса демону отсылается специальный код, и он тоже прекращает работу.

2.2.3 Attribute Changer

Эта часть программы состоит из двух частей: *бинарника*, который при запуске либо сменяет класс устройства и добавляет соответствующую *SDP* запись, либо удаляет запись и возвращает нужный класс и приложения, которое показывает пользователю нынешний класс устройства и предлагает его сменить (Подробнее о *SDP* см. 2.1.2).

2.2.4 Сеанс работы

Сеанс работы с *Андроид-мышью* выглядит следующим образом

1. Пользователь запускает Attribute Changer и меняет их так, что бы телефон стал мышью.
2. В стандартном меню *Bluetooth* создаёт пару с желаемым устройством.
3. Запускает основное приложение и в списке спаренных устройств выбирает нужное.
4. Использует программу.
5. При помощи стрелки “назад” завершает сеанс.

2.3 Описание процесса разработки, установки и компиляции.

- **Процесс сборки или компиляции** - это процесс преобразования исходного кода в бинарный код, понятый компьютеру.
- **Процесс установки** - это процесс перенесения уже скомпилированных файлов в каталог, откуда они будут считаны и интерпретированы.

2.3.1 Используемые программы

Мы не будем входить в подробности работы программ использованных для разработки, установки и сборки проекта, ограничимся только их кратким описанием.

- *arm-linux-gnueabi-gcc* — кросс компилятор пригодный для компиляции кода для *arm* процессоров (именно на *arm* работает большинство телефонов), использовался для компиляции *hidp_clientd* и *attributeChanger*.
- Пакет программ *Android SDK* от Google использовался для разработки графических приложений на Андроид.
- *EclibseIDE** — графическая программа, позволяющая удобно писать, компилировать и устанавливать код. Использовалась специальная версия *Eclipse IDE* со встроенными программами разработки под Андроид, далее она будет называться просто *Eclipse IDE*. Процесс сборки приложения под Андроид это процесс компиляции некоторых файлов и перенесения их в файл с расширением *.apk*.
- Adb (Android Debug Bridge) – утилита, позволяющая переносить файлы на устройство с Андроидом и устанавливать и собирать файл с расширением *.apk*. Это утилита командной строки, что позволяет использовать её в скриптах.

2.3.2 Процесс разработки

Графические приложения были написаны и скомпилированы в *Eclibse IDE*.

Программы, хранящиеся в бинарном виде, были написаны в текстовом редакторе.

2.3.3 Процесс компиляции.

adb компилирует графические приложения, а *arm-linux-gnueabi-gcc* компилирует программы, хранящиеся в бинарном.

2.3.4 Процесс установки

adb устанавливает графические приложения, и переносит программы, хранящиеся в бинарном виде.

2.4 Описание файлов проекта

В этой секции документа описана «файловая структура» проекта.

Для большей ясности ниже после названия папки будет стоять знак деления (*имя-папки/*).

- *build/* - в этой директории хранятся скомпилированные файлы готовые к установке на устройство.
- *src/* - в этой директории хранится исходный код, из которого создаются файлы в директории *build/*.

-
- *BecomeAMouse/*, *BluetoothMouse/* — каталоги с кодом графических приложений.
 - *bin/* — каталог с кодом бинарных приложений.
 - *tools/* - в этой директории хранятся файлы нужные для сборки файлов в каталоге *build/*.
 - *tmp/* - эта директория нужна для временных файлов, создаваемых во время процесса установки.
 - *doc/* - в этой директории хранится документация к проекту.

 - *install.bat*, *install.sh*, *Makefile* - файлы нужные для компиляции и установки проекта.

3 Результат

Результатом данного проекта является набор программ, отвечающий всем поставленным требованиям.

С демонстрационным видео-роликом о работе *Андроид-Мыши* можно ознакомиться, пройдя по ссылке <http://arseny-n.github.io/Android2Mouse/doc/index.html>

4 Ссылки на использованную литературу

При разработке данного проекта, использовались следующие интернет — ресурсы:

- Спецификации Bluetooth:

<https://www.bluetooth.org/>

<http://www.palowireless.com/infotooth>

- Спецификации HID:

<http://www.usb.org/developers/hidpage/>

- Документация с официального портала Андроид:

<http://developer.android.com/>

- Википедия:

<https://en.wikipedia.org>

