

Руководство программиста

Абдуллин Азат

28 мая 2017 г.

Оглавление

1	Назначение и условия применения программы	2
2	Обращение к программе	2
3	Структура программы	3
3.1	Подсистема считывания конфигурации	3
3.2	Подсистема «Оконный менеджер»	4
4	Установка и настройка программы	7
4.1	Состав установочного комплекта	7
4.2	Установка программы	7
4.3	Деинсталляция программы	8
5	Проверка программы	8

1 Назначение и условия применения программы

Данная программа предназначена для запуска графических пользовательских приложений в ОС Linux на платформе Raspberry Pi Zero. Для запуска программы в системе должны быть установлены пакеты Wayland и WLC.

2 Обращение к программе

Программа имеет один входной параметр — путь к конфигурационному файлу. При отсутствии данного аргумента программа открывает файл по-умолчанию `/.config/xxwm`. При запуске оконного менеджера автоматически запускаются системные приложения строки состояния и рабочего стола. Имеется возможность указания исполняемых файлов системных приложений в конфигурационном файле. Формат конфигурационного файла приведен в листинге 1

Листинг 1: Формат конфигурационного файла

```
1 [statusbar]
2 exe=*path to executable*
3
4 [desktop]
5 exe=*path to executable*
```

При запуске программы на экране отображается строка состояния и рабочий стол (рис. 1).

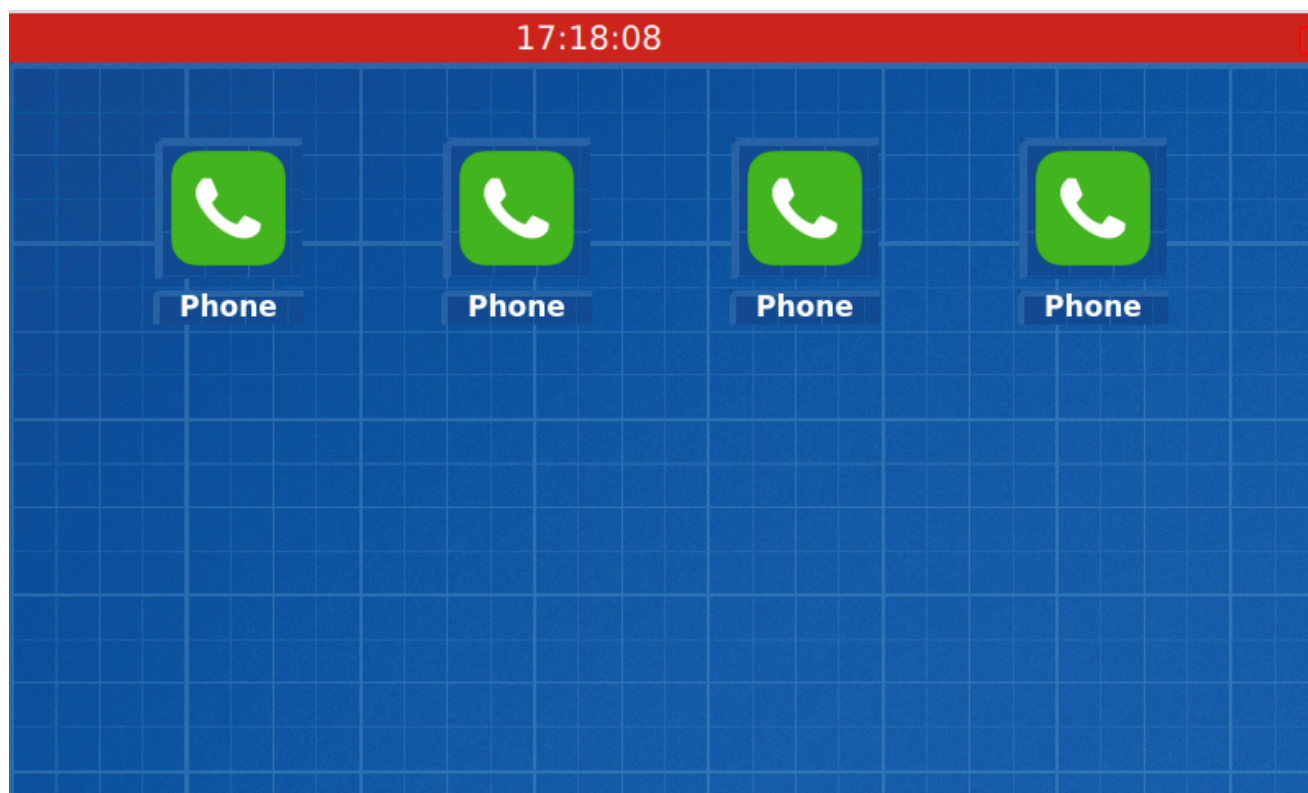


Рис. 1: Стартовый вид программы

При нажатии на одну из иконок на рабочем столе запустится соответствующее приложение (рис. 2). Программа поддерживает следующие комбинации клавиш:

- CTRL+Enter — запуск терминала (рис. 3)
- CTRL+q — закрытие приложения

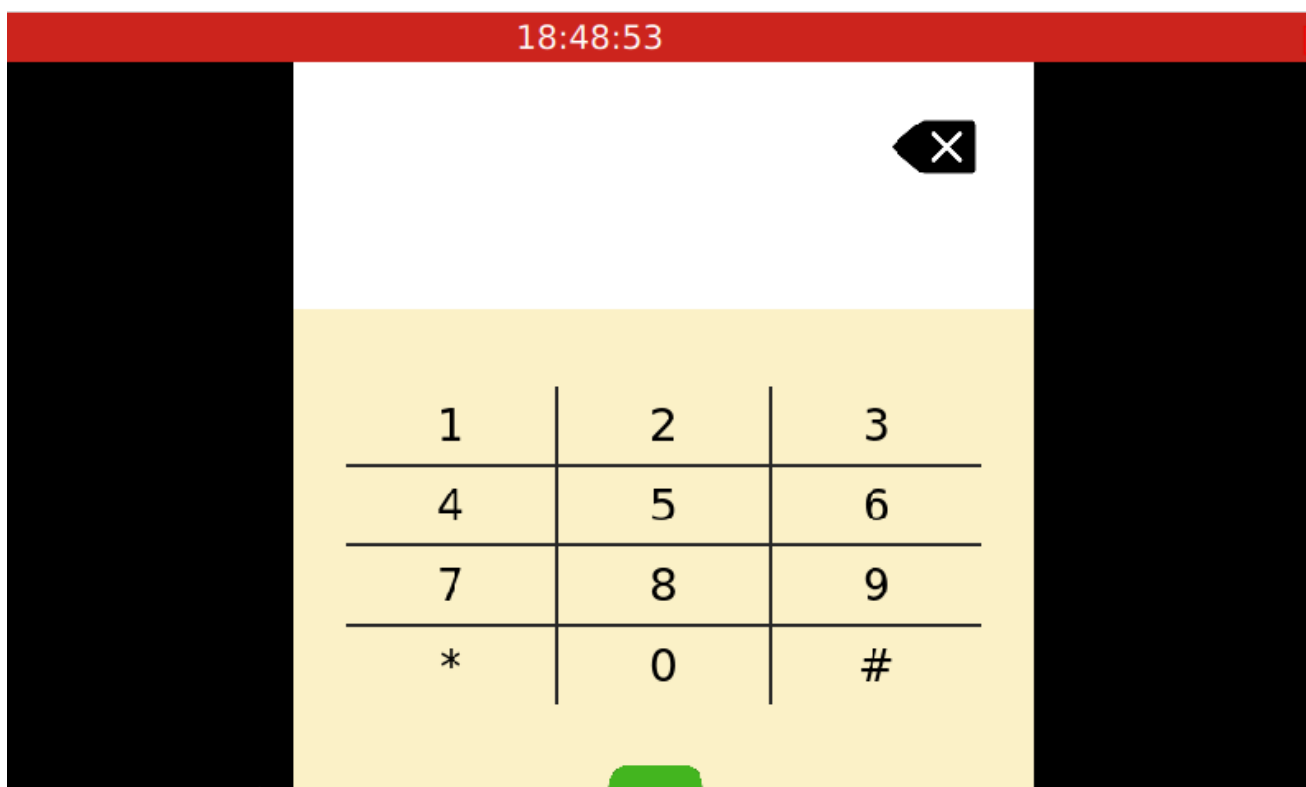


Рис. 2: Запуск приложения с рабочего стола

- CTRL+стрелка вниз — переключение окон
- CTRL+Esc — завершение работы программы

Программа так же поддерживает управление окнами с помощью мыши (рис. 4):

- CTRL+ЛКМ — перемещение окна
- CTRL+ПКМ — изменение размеров окна

3 Структура программы

Структурно программный комплекс делится на две подсистемы: подсистема считывания конфигурации и оконный менеджер.

3.1 Подсистема считывания конфигурации

Данная подсистема необходима для считывания конфигурационного файла и определения параметров запуска оконного менеджера. Подсистема состоит из двух файлов:

- config.h — заголовочный файл с определениями функций
- config.c — файл с реализациями функций

Система считывания конфигурации содержит следующие функции:

- init_config — функция инициализация конфигурации. В качестве входного параметра необходимо передать имя файла конфигурации в виде строки. Выходные параметры отсутствуют.

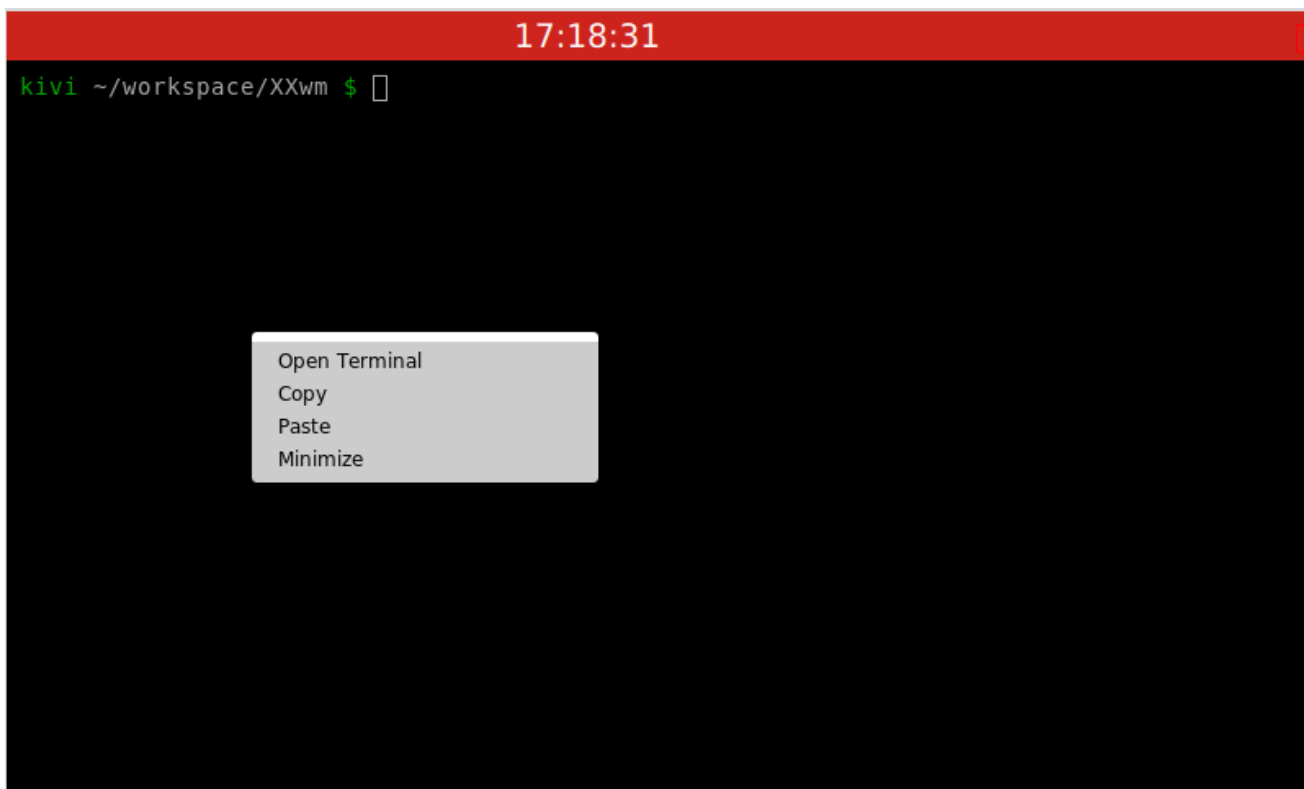


Рис. 3: Запуск терминала

- `get_statusbar` — функция считывает имя исполняемого файла строки состояния. Перед вызовом конфигурация должна быть проинициализирована вызовом функции `init_config`. Входные параметры отсутствуют. Выходной параметр — имя исполняемого файла строки состояния в виде строки.
- `get_desktop` — функция считывает имя исполняемого файла рабочего стола. Перед вызовом конфигурация должна быть проинициализирована вызовом функции `init_config`. Входные параметры отсутствуют. Выходной параметр — имя исполняемого файла рабочего стола в виде строки.

Так же содержится переменная `configuration` — структура, которая хранит считанную конфигурацию. Устанавливается в функции `init_config`.

3.2 Подсистема «Оконный менеджер»

Данная реализует функции оконного менеджера и отвечает за его запуск. Подсистема состоит из файла `main.c`. Оконный менеджер содержит следующие функции:

- `start_interactive_action` — функция которая начинает интерактивное действие. Входные параметры:
 - идентификатор окна, на котором начинается действие
 - точка, в которой начато действие

Выходные параметры: `true` если действие было начато, иначе `false`.

- `start_interactive_move` — функция, которая начинает интерактивное действие передвижения окна. Входные параметры:
 - идентификатор окна, на котором начинается действие
 - точка, в которой начато действие

Выходные параметры отсутствуют.

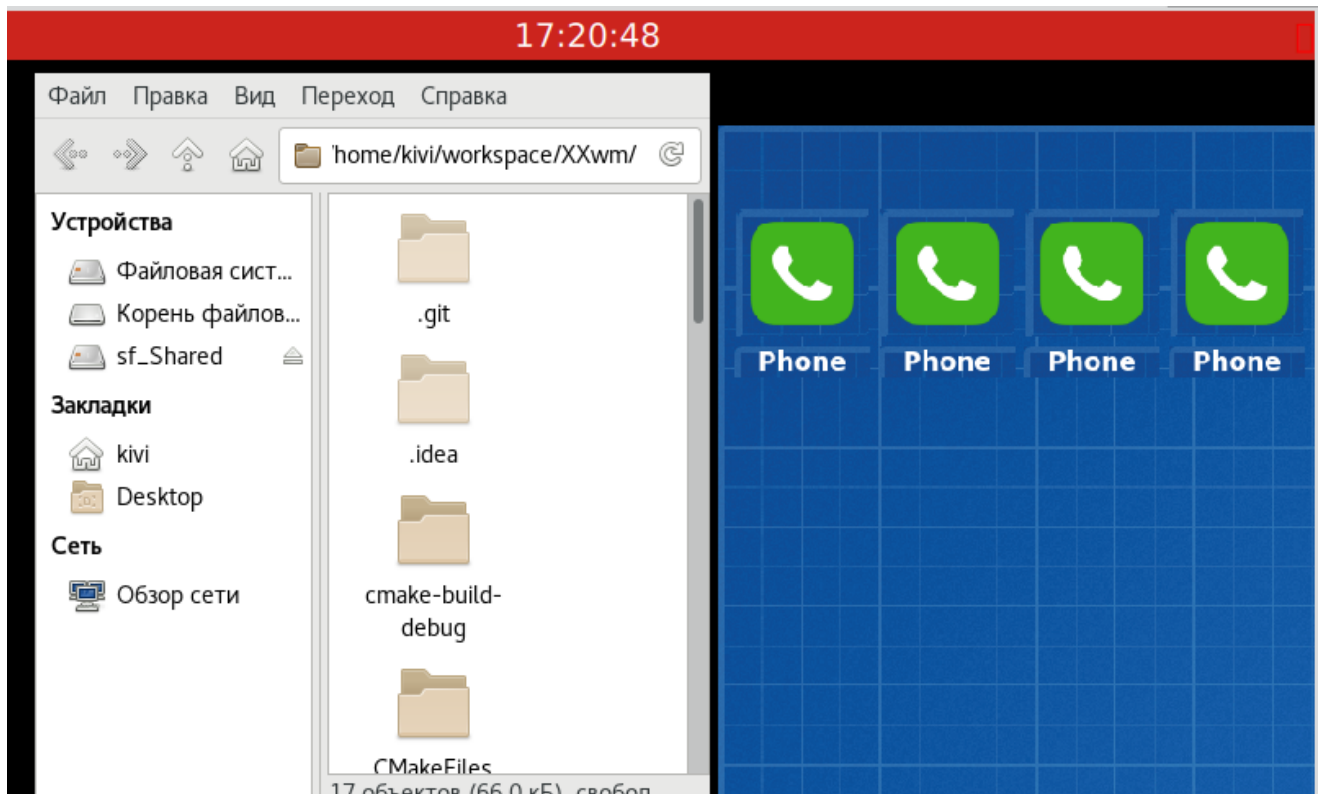


Рис. 4: Управление окнами с помощью мыши

- **start_interactive_resize** — функция, которая начинает интерактивное действие изменения размеров окна. Входные параметры:

- идентификатор окна, на котором начинается действие
- грань окна, которая должна быть передвинута
- точка, в которой начато действие

Выходные параметры отсутствуют.

- **stop_interactive_action** — функция, которая завершает интерактивное действие. Входные параметры отсутствуют. Выходные параметры отсутствуют.
- **get_topmost** — функция получения окна по смещению. Входные параметры:

- идентификатор отображаемого экрана
- смещение

Выходной параметр — идентификатор окна.

- **relayout** — функция перерисовывания экрана. Входные параметры:

- идентификатор отображаемого экрана

Выходные параметры отсутствуют.

- **output_resolution** — функция смены разрешения экрана. Входные параметры:

- идентификатор отображаемого экрана
- старое разрешение
- новое разрешение

Выходные параметры отсутствуют.

- **view_created** — функция обработки отображения нового окна. Входные параметры:
 - идентификатор окна

Выходной параметр: **true** если окно обработано, **false** если окно необходимо уничтожить.

- **view_destroyed** — функция обработки уничтожения окна. Входные параметры:
 - идентификатор окна

Выходные параметры отсутствуют.

- **view_focus** — функция установки активного окна. Входные параметры:
 - идентификатор окна
 - **true** если окно необходимо сделать активным

Выходные параметры отсутствуют.

- **view_request_move** — функция запроса перемещения окна. Входные параметры:
 - идентификатор окна
 - координаты начального положения окна

Выходные параметры отсутствуют.

- **view_request_resize** — функция запроса изменения размера окна. Входные параметры:
 - идентификатор окна
 - грань окна, которая должна быть передвинута
 - координаты начального положения окна

Выходные параметры отсутствуют.

- **view_request_geometry** — функция запроса изменения параметров отображения окна. Входные параметры:
 - идентификатор окна
 - новые параметры отображения окна

Выходные параметры отсутствуют.

- **keyboard_key** — функция обработки клавиш клавиатуры. Входные параметры:
 - идентификатор окна
 - время нажатия
 - флаги модификаторов клавиатуры
 - код клавиши
 - состояние клавиши

Выходной параметр: **false** если событие необходимо передать окну.

- **pointer_button** — функция обработки кнопок мыши. Входные параметры:
 - идентификатор окна
 - время нажатия
 - флаги модификаторов клавиатуры
 - код кнопки
 - состояние кнопки
 - позиция указателя мыши

Выходной параметр: `false` если событие необходимо передать окну.

- `pointer_motion` — функция обработки перемещения указателя мыши. Входные параметры:
 - идентификатор окна
 - время перемещения
 - позиция указателя мыши

Выходной параметр: `false` если событие необходимо передать окну.

- `cb_log` — функция логирования. Входные параметры:
 - тип сообщения
 - сообщение

Выходные параметры отсутствуют.

Так же содержится переменная `compositor` — структура, которая хранит информацию об интерактивном действии. Используется в функциях обработки интерактивных действий.

4 Установка и настройка программы

4.1 Состав установочного комплекта

- `main.c` — основной файл оконного менеджера
- `config.h` — заголовочный файл с объявлениями функций считывания конфигурации
- `config.c` — файл с определениями функций считывания конфигурации
- `CMakeLists.txt` — файл сборки программы
- `config.ini` — файл конфигурации
- `xxon.desktop` — файл ярлыка для экранного менеджера

4.2 Установка программы

Первым шагом при подготовке оконного менеджера к работе является компиляция исходных кодов программы в бинарные файлы. Для компиляции необходимо воспользоваться системой сборки CMake, версии, не ниже 3.7. Так же в системе должны быть установлены компилятор GCC и система сборки Make. Компиляция осуществляется следующими командами:

```
cmake .  
make
```

В результате, в рабочей директории появится исполняемый файл оконного менеджера `xxwm`.

Для установки программы необходимо иметь права администратора. Для установки необходимо:

- Поместить исполняемый файл оконного менеджера в каталог исполняемых файлов:

```
cp xxwm /usr/bin/
```

- Поместить стандартный конфигурационный файл в директорию хранения конфигураций:

```
cp config.ini ~/.config/xxwm
```

- Поместить файл ярлыка для экранного менеджера в директорию ярлыков оконных менеджеров:

```
cp xxon.desktop /usr/share/wayland-sessions/
```


4.3 Настройка программы

Для настройки программы необходимо в конфигурационном файле `/.config/xxwm` указать пути к исполняемым файлам строки состояния и рабочего стола

4.4 Деинсталляция программы

Для деинсталляции программы необходимо удалить ранее скопированные файлы:

```
rm /usr/bin/xxwm  
rm ~/.config/xxwm  
rm /usr/share/wayland-sessions/xxon.desktop
```

5 Проверка программы

Методики проверки работоспособности программы описаны в документе "Программа и методика испытаний".