

Karen OS

Teacher manual

Introduction:

This Operative System was developed for education purposes. It is a very basic non GUI OS that displays a terminal to issue commands. This OS was built for the Raspberry Pi A first generation. Trying to use this OS with any modern Raspberry Pi will not work due to the change in the modern computer architecture. Due to this old model, not many keyboards work for this OS due to complications with the USB driver. The Logitech classic keyboard 200, perixx periboard-203, Saitek Cyborg v1 are some keyboard the OS USB driver is compatible with.

API:

Github repository: <https://github.com/semylevy/rPiOS>

Next, we list all the functions developed for Karen OS, which can be used freely to implement new functions.

Drawing:

- SetForeColour:
- SetGraphicsAddress:
- DrawPixel:
- DrawLine:
- DrawCharacter:
- DrawString:

FrameBuffer:

- InitialiseFrameBuffer:

GPIO:

- GetGpioAddress:
- SetGpioFunction:
- SetGpio:

Keyboard:

- KeyboardUpdate:
- KeyWasDown:
- KeyboardGetChar:

Mailbox:

- GetMailboxBase:
- MailboxWrite:
- MailboxRead:

Math&Random:

- DivideU32:
- Random:

SystemTimer:

- GetTimerAddress:
- GetTimeStamp:

TimerFunction:
Tags:
FindTag:
Text:
ReverseString:
UnsignedString:
SignedString:
FormatString:

Technical details:

The project was developed in ARMv6 Assembly language, and compiled with the GNU Assembler.

A simple font file is included in the “source” folder, which is used to display text on screen. The font is image-based, as opposed to vector-based, which is a planned feature for Karen OS.

The project consists of several .s source files, which are placed under the “source” folder. Provided is a Makefile, which compiles all the source files into .o files inside the “build” folder. It also compiles the C library CSUD (<https://github.com/Chadderz121/csud>) and links it to the assembly files (using the “kernel.ld” file).

In order to change compiler, or any other compilation settings, the Makefile should be edited.

The output file “kernel.img” is used to boot the Raspberry Pi with Karen OS.

Scrum documentation:

The progress and documentation for this project can be found on a Trello board available here:

<https://trello.com/b/guol4Mph/backlog>

The project was divided into 12 short sprints, each 1 day long, and with a deliverable ready by the end of the day. We had a daily meeting to assess the progress of the team, and to correct mistakes and improve our workflow.

In the “Appendix” folder are included photos and videos of the project progress, for every step of the development process. Future updates are planned and will be added to the Scrum workflow to assure efficiency.

Team:

Christian Aguilar	A01024157
Salomon Levy	A01023530