Description for SpaceChain OS Code Open Source Online

OS Online Description and Subsequent Development Plan

Category	Content
Key words	OS online description Development plan
Abstract	Online Description\Development plan

Revision History

Vision	Date	Reason
V1.00	2018/04/20	Document creation

Contents

1. Open source online description

On March 14, 2018, SpaceChain Foundation officially submitted and released the code of SpaceChain OS. This open code is a complete project that includes Blockchain applications.

This document describes the basic process of SpaceChain OS open source online and subsequent development plan. For more descriptions, please refer to other relevant manuals.

SpaceChain OS is an embedded real-time operating system that can be run only when deployed to hardware. Currently, we support multiple hardware with different CPU architecture, including the classic zynq7000 series with ARM architecture and PPC, MIPS, x86, Sparc, etc.

Development of Blockchain-based DAPPs does not require special development tools.

The SpaceChain OS top script application can be developed directly with scripting tools such as Python and Lua. Any application developed can be uploaded to the target board via FTP, which means that the application can be run directly.

If the source code for the kernel and related system files needs to be modified and recompiled, the corresponding compiler is required. For details, see the IDE manual.

At present, the final development of SpaceChain OS development tools has not been completed. For similar needs, please use RealEvo-IDE temporarily. Please visit the SpaceChain official website if you wish to obtain realevo-IDE and the license. The application will be launched at the end of March. You can contact info@ SpaceChain.com before the website is open.

1.1 Code instruction

Code name	Description
libboost	A portable, open source C++ library that encapsulates a large number of algorithms and containers. The boost library was initiated by members of the C++ Standards Committee Library Working Group, and some of the content in this library is expected to become the content of the next-generation C++ standard library.
libdb	Berkeley DB (database). Berkeley DB is an open source embedded database that provides applications with a scalable, high-performance, transaction-protected data management service. Berkeley DB provides a set of concise functions for data access and management to call API interfaces.

libevent	A lightweight, high-performance open source event notification library which is written in C. Libevent with a lightweight event-driven architecture has the features of high execution efficiency and cross-platform.
qtum	Qtum full nodes, including Qtum client, RPC-based command-line tool, and qt-based graphics tool for Qtum wallet. A lot of porting and rewriting work has been done so that it can run on SpaceChain OS.

The code organization is as follows:

- spacechainos-base----SpaceChain OS
- spacechainos -base/ libSpaceChainos----Kernel
- spacechainos -base/ libcextern---- C library
- spacechainos -base/ libexpat ---- library for parsing XML
- spacechainos -base/ liblua---- support library for lua script
- spacechainos-base/ libluaplugin----set of lua plug-ins
- spacechainos-base/ libpcap---- pcap library providing bottom interface support for network packet analysis tool
- spacechainos-base/ libreadline---- providing interactive text editing
- spacechainos-base/ libsalsa---- support library for lightweight ALSA (advanced Linux sound architecture)
- spacechainos-base/ libsqlite3---- sqlite database
- spacechainos-base/ libVxWorks---- VxWorks compatibility layer
- spacechainos-base/ libzmodem---- support library for zmodem communication protocol
- spacechainos-base/ openssl---- password library for openssl secure sockets layer
- spacechainos-base/ pciutils---- PCI bus toolset
- spacechainos-base/ unfsd---- nfs server

2. Subsequent Development Plan

2.1 Simulator

It is expected that the x86-based virtual machine—Simulator will be available later. It can not only simulate running SpaceChain OS on x86 computers, but also fully run APP and DAPP. If you only need to run a DAPP or basic application, Simulator can be helpful to develop and run without hardware. If the product eventually needs to be deployed to the hardware, Simulator can also provide great convenience for the R&D and testing before launch of formal product.

2.2 SpaceChain IDE

A beta version of the development tool. All functions of the entire lifecycle software, such as code design, development, debugging, compilation, testing, and deployment will be implemented by using the IDE. The license can be obtained via SpaceChain official website. This version is expected to be released about one month after the OS goes online.

2.3 SpaceChain IDE SPC

Based on the above IDE, we will launch a development environment that can exchange usage rights via SPC, and it is also the final version we hope to provide for enthusiasts in the future. All future technological evolutions and upgrades will also be implemented in this version.

This version is expected to be released about 4 months after the OS goes online. Stay tuned.

3. Document description

We will gradually update more documents later. At present, the document update plan is as follows. Please note that this list may be updated at any time.

3.1 Development documents

Document name	Description
QT Development Guide	This document describes how to use QT for development. Experienced QT engineers can ignore this document.
Development Guide for SpaceChain OS Device Driver	This document describes how to develop a device driver. Bottom System Engineers who need to adapt to a new hardware platform can refer to this document.
SpaceChain OS APP Development Manual	This document describes how to develop a SpaceChain OS-based APP. For a modular satellite attitude control algorithm or a combination function, please refer to this manual.
SpaceChain OS DAPP Development Manual	This document describes how to use the SpaceChain OS blockchain. Please refer to this manual to develop a blockchain DAPP. At present, we have supported Qtum. You can also refer to the Qtum manual for development or run the developed applications.
SpaceChain Lib Development and Use Manual	This document describes the common code base for the spacecraft we currently provide. Specific hardware modules and functions can be quickly adapted and supported by using these codes to minimize duplication and meaningless work. This document also presents to code contributors how to develop these codes.

3.2 Use manual

Document name	Description
SpaceChain OS Blockchain Primer	This document is a primer that explains how to get started and install IDE, and how to implement simple functions such as transfer.
SpaceChain OS White Paper	This document provides a detailed description of OS.
SpaceChain IDE Use Manual	This document describes how to use a series of functional modules of development tools, such as development, deployment, testing, simulation, etc.
SpaceChain Simulator Use Manual	This document describes how to use our virtual machine to run SpaceChain OS, blockchain and other functions via virtual software platform without a hardware platform.

3.3 Test document

Document name	Description
SpaceChain OS real-time test report	Real-time is the most important indicator of a real-time operating system. This document details our test environment, test code, and test result.