Тринити

РОССИЙСКИЙ ПРИЗВОДИТЕЛЬ ОБОРУДОВАНИЯ

ДЛЯ ПОСТРОЕНИЯ ИТ-ИНФРАСТРУКТУРЫ

ВСТРОЕННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ BIOS RU.ТВЛФ.00001-01

Инструкция по установке экземпляра программного обеспечения

на 11 листах

г. Москва

2025

Содержание

1. Перечень сокращений и обозначений

ОС	Операционная система
ПЗУ	Постоянное запоминающее устройство
ПК	Персональный компьютер
ПО	Программное обеспечение
СТП	Служба технической поддержки
ЭВМ	Электронно-вычислительная машина
API (Application Programming Interface)	Описание способов взаимодействия программного обеспечения с аппаратными и программными компонентами системы
BIOS (Basic Input/Output System)	Базовая система ввода-вывода
UEFI (Unified Extensible Firmware Interface)	Интерфейс между операционной системой и микропрограммами, управляющими оборудованием
Программатор	Аппаратно-программное устройство для чтения и записи информации в ПЗУ, флэш-память или внутреннюю память микроконтроллеров (МК)

2. Общие сведения

Документ содержит информацию по установке экземпляра встроенного программного обеспечения BIOS RU.ТВЛФ.00001-01 (далее – ВПО).

ВПО предназначено для обеспечения взаимодействия между аппаратными и программными компонентами платы материнской Тринити ТВЛФ.469555.001 (далее – материнская плата) в составе серверов Тринити, построенных на базе процессоров Intel Xeon 3-го поколения. ВПО разработано для облегчения процесса начального запуска вычислительной техники и предназначена для инициализации и запуска основных устройств вычислительной техники и ее компонентов. ВПО обеспечивает передачу управления операционной системе в соответствии с предварительно заданными настройками.

ВПО использует технические средства платы материнской Тринити ТВЛФ.469555.001 производства АО «ТРИНИТИ СОЛЮШНС».

3. Назначение и условия применения

3.1. Назначение

ВПО – программа начального запуска, является системной программой низкого уровня, хранящейся в микросхеме ПЗУ на материнской плате, и предоставляет пользователю возможность полного управления системой при загрузке.

ВПО состоит из ряда драйверов, приложений и экранных форм, с помощью которых можно настроить параметры работы системы в соответствии с требованиями пользователя или использовать параметры, заданные по умолчанию.

ВПО предоставляет расширенные функциональные возможности UEFI, унифицированного расширяемого интерфейса микропрограмм для ПО низкого уровня, которое запускается автоматически при старте сервера перед тем, как загрузится операционная система.

Функции ВПО:

- начальная инициализация серверной платформы;
- проверка работоспособности аппаратных компонентов;
- загрузка операционной системы.

ВПО разработано в соответствии со спецификацией UEFI для решения проблемы переносимости встроенного программного обеспечения и расширяемости на будущие платформы, расширения использования различных драйверов, средств разработки, утилит поддержки и загрузочных приложений.

Особенности ВПО:

- поддержка UEFI 2.8, PI 1.7;
- интеграция с Intel Boot Guard;
- Secure Flash для безопасного обновления;
- модульная архитектура.

3.2. Условия применения

Программные и аппаратные требования для работы ВПО:

- процессор: 3th Gen Intel® Xeon® Scalable Processor (Ice Lake);
- память: не менее 8 Гб RAM;
- операционная система: не требуется, но поддерживаются Windows, Linux.

4. Технические средства хранения исходного текста

4.1. Системы контроля версий

Git является основной системой контроля версий. Благодаря распределенной архитектуре Git каждый разработчик имеет полную локальную копию репозитория, что позволяет:

- отслеживать изменения с высокой детализацией;
- управлять ветками, интегрировать новые функции и проводить проверку изменений;

- легко находить предыдущие версии, обеспечивая быстрый возврат при необходимости.

Git позволяет эффективно сотрудничать между командами, даже если речь идет о критически важных системах, где стабильность и воспроизводимость сборок имеют первостепенное значение.

4.2. Хранилища кода. Локальные и облачные решения

Для хранения исходного кода используется Git Server, развернутый на виртуальной машине под управлением Ubuntu 24.04, работающей в среде виртуализации ESXi. Такой подход обеспечивает следующие преимущества:

- локальный контроль: возможность управления инфраструктурой хранения, настройка точечных политик безопасности и контроля доступа;
- масштабируемость: легкость добавления новых репозиториев или узлов для повышения отказоустойчивости;
- облачное хранение: при необходимости можно интегрировать облачные решения для дополнительной резервной копии и глобальной доступности исходного кода.

4.3. Форматы хранения, безопасность и резервное копирование

4.3.1. Форматы хранения

В проектах с низкоуровневым программированием и разработкой BIOS используется широкий спектр файловых форматов, что отражает многообразие исходного материала. Основные форматы (расширения) файлов:

- языки программирования и скрипты:

.c, .asm, .py, .sh, .bat, .cmd

- системные и объектные файлы:

.dll, .exe, .lib, .bin, .rom

- файлы конфигурации и метаданные:

.json, .xml, .ini, .yaml, .cfg

- специфичные для BIOS/UEFI:

.efi, .EFI, .asl, .ASL, .aslc, а также специализированные расширения, например, .BiosGuardVkeyExp

- документация и ресурсы:

.pdf, .md, .html, .chm, .bmp, .gif

Такой разнообразный перечень позволяет хранить исходный код, документацию, скрипты сборки, конфигурационные файлы и вспомогательные ресурсы в едином хранилище.

4.3.2. Меры безопасности

Для защиты исходного кода применяются следующие меры:

- контроль доступа: жесткая настройка прав пользователей и групп, что предотвращает несанкционированные изменения;
- шифрование: использование сертификатов и криптографических подписей (например, файлы .pem) для обеспечения целостности и аутентификации данных;
- аудит изменений: автоматизированный мониторинг коммитов и использование встроенных возможностей Git для отслеживания истории изменений.

4.3.3. Резервное копирование

Надежность хранения достигается за счет автоматизированных процедур резервного копирования:

- Git Hooks: позволяют автоматически запускать скрипты резервного копирования при каждом коммите или слиянии веток. Это гарантирует, что любые изменения будут мгновенно скопированы в резервное хранилище;
- Rsync: используется для синхронизации репозиториев между серверами, что минимизирует риск потери данных при сбоях оборудования или аварийных ситуациях.

Такая комбинация инструментов обеспечивает высокую доступность и целостность исходного кода, позволяя быстро восстановить рабочее состояние системы в случае непредвиденных инцидентов.

Использование Git в качестве системы контроля версий в сочетании с локальными и облачными хранилищами, поддержка множества форматов исходного текста, строгие меры безопасности и автоматизированное резервное копирование через Git Hooks и Rsync создают надежную и масштабируемую инфраструктуру для управления исходным кодом, особенно в критичных для системы низкоуровневых разработках.

5. Подготовка к установке ВПО

Установка ВПО осуществляется методом аппаратной прошивки в энергонезависимую память SPI Flash на этапе производства. Для записи прошивки используется специальное оборудование – программаторы Xgecu Pro T48 в сочетании с программным обеспечением «Xgpro».

Минимальная конфигурация рабочего места для записи ВПО в энергонезависимую память материнской платы:

- материнская плата Тринити ТВЛФ.469555.001 с флеш-памятью Macronix MX25L51245;
- технологический ПК с ОС «Windows» и установленным ПО «Xgpro»;
- программатор Xgecu Pro T48 с необходимыми переходниками.

Подготовку к установке ВПО выполнять в следующем порядке:

1) включить технологический ПК, дождаться загрузки ОС;
2) установить на технологический ПК ПО «Xgpro»;
3) приготовить бинарный файл прошивки ВПО или выполнить компиляцию файла прошивки из исходного кода согласно руководству по компиляции и сборке исходного кода RU.ТВЛФ.00001-01 91 01;
4) извлечь микросхему SPI Flash из материнской платы, затем вставить ее в адаптер, установленный в программатор. Адаптер должен соответствовать типу корпуса микросхемы (в комплект программатора могут входить адаптеры для разных корпусов);
5) если микросхему SPI Flash памяти извлечь невозможно (распаяна непосредственно на материнской плате), то для подключения программатора следует использовать адаптер типа «клипса» с гибким кабелем. Число и шаг контактов адаптера должны соответствовать типу корпуса микросхемы.
6) подключить программатор к технологическому ПК, проверить соединение.

6. Установка ВПО

Для записи ВПО в микросхему SPI Flash памяти необходимо выполнить следующие действия:

- на технологическом ПК, подключенном к программатору, открыть программу «Xgpro»;
- в настройках программы «Xgpro» выбрать микросхему Macronix MX25L51245;
- в программе «Xgpro» указать путь к бинарному файлу прошивки ВПО RU.ТВЛФ.00001-01;
- запустить процесс программирования. Далее процесс программирования продолжается автоматически;
- дождаться окончания процесса программирования;
- проверить контрольную сумму и запись прошивки в микросхему;
- проверить записанной информации путем чтения содержимого SPI Flash;
- вынуть микросхему из адаптера программатора и установить в материнскую плату (если изымалась), или отсоединить адаптер типа «клипса» с гибким кабелем (если микросхема припаяна к плате);
- установить материнскую плату в технологический сервер и выполнить функциональное тестирование для проверки корректной работы BIOS.

После успешной прошивки серверная платформа проходит дополнительные тесты, включая проверку POST и загрузку в режим UEFI Setup Utility для контроля установленных параметров.

Этот процесс гарантирует, что BIOS корректно функционирует и готов к эксплуатации без необходимости дополнительной настройки со стороны пользователя.

7. Работа ВПО

7.1. Установка и настройка ВПО

Встроенное программное обеспечение BIOS RU.ТВЛФ.00001-01 записывается в энергонезависимую память материнской платы на этапе производства. Запись выполняется с использованием специального оборудования – программаторов SPI Flash.

На момент поставки серверной платформы BIOS уже установлен, а его базовые настройки соответствуют требованиям производителя. Пользователь не нуждается в дополнительной установке или настройке.

В случае необходимости внесения изменений, например, обновления микрокода процессора или активации специфичных функций, можно использовать BMC Web-интерфейс.

7.2. Установка обновлений

Обновление ВПО выполняется квалифицированными специалистами и требует строгого соблюдения процедуры во избежание возможных сбоев.

Методы обновления:

1) Через Web-интерфейс BMC:

- войти в IPMI/BMC Web UI через веб-браузер;
- загрузить новую версию ВПО в раздел Firmware Update;
- проверить контрольные суммы и совместимость прошивки;
- запустить процесс обновления;
- дождаться автоматической перезагрузки сервера после завершения.

2) Через BIOS Recovery Mode (в случае повреждения прошивки):

- использование резервной копии BIOS, хранящейся в энергонезависимой памяти;
- автоматическое восстановление при обнаружении сбоя загрузки.

После обновления ВПО все настройки могут быть сброшены к заводским, поэтому рекомендуется заранее сохранить текущую конфигурацию.

7.3. Штатное функционирование

После подачи питания на сервер ВПО загружается автоматически и выполняет инициализацию системы.

Способы запуска:

- физическое включение сервера – нажатием кнопки Power на передней панели;

- удаленный запуск через BMC – с помощью IPMI/Web-интерфейса администратора.

Основные этапы штатного функционирования:

1) POST (Power-On Self-Test) – проверка процессора, памяти, контроллеров и периферийных устройств;

2) загрузка микропрограмм чипсетов и инициализация периферии;

3) определение загрузочных устройств (NVMe, SATA, USB, PXE);

4) запуск операционной системы с выбранного носителя.

ВПО также поддерживает удаленное управление, мониторинг температуры и управление питанием через BMC/IPMI, что делает возможным диагностику и контроль работы сервера без физического доступа к оборудованию.