Публичный

CM-LYNX 2.10.00
Руководство пользователя

Версия	**2.5.0**
Дата версии	**25.09.2026**
Имя файла	**ML_CM-LYNX_Руководство_пользователя_R_V2.5.0**

ИСТОРИЯ ИЗМЕНЕНИЙ

Версия	Дата	Автор	Описание
2.5.0	25.09.2026	Новосёлов А.В.	Подготовка документа к поставке версии 2.10.00.
2.4.0	05.06.2025	Новосёлов А.В.	Подготовка документа к поставке версии 2.09.00.
2.3.0	29.03.2024	Новосёлов А.В.	Подготовка документа к поставке версии 2.08.00.
2.2.0	23.10.2023	Смирнова Ю.И.	Подготовка документа к поставке версии 2.07.00.
2.1.0	25.04.2023	Смирнова Ю.И.	Подготовка документа к поставке версии 2.06.00. Внесены уточнения в п. 4.10, 5.2, 7.
2.0.6	29.09.2022	Новосёлов А.В.	Подготовка документа к поставке версии 2.05.00. Внесены уточнения в п. 6.
2.0.5	28.04.2022	Смирнова Ю.И.	Добавлено уточнение в п.4.11. Подготовка документа к поставке версии 2.04.00.
2.0.4	28.01.2022	Колесникова Т.В.	Подготовка документа к поставке версии 2.04.00.
2.0.3	06.10.2021	Новосёлов А.В.	Подготовка документа к поставке версии 2.03.00. Внесены уточнения в п. 4.10 и 4.11.
2.0.2	19.03.2021	Смирнова Ю.И.	Подготовка документа к поставке версии 2.02.00.
2.0.1	01.02.2021	Смирнова Ю.И.	Подготовка документа к поставке версии 2.01.02. Внесены уточнения в п. 4.11.
2.0.0	29.10.2020	Смирнова Ю.И.	Подготовка документа к поставке версии 2.01.01.
1.10.0	21.09.2020	Смирнова Ю.И.	Подготовка документа к поставке версии 2.01.00.
1.9.2	19.06.2020	Стрельцова О.В.	Изменено примечание в п. 4.10.
1.9.1	19.06.2020	Смирнова Ю.И.	Скорректировано описание п. 4.11
1.9.0	18.06.2020	Стрельцова О.В.	Подготовка документа к поставке версии 2.00.00.
1.8.0	18.03.2020	Стрельцова О.В.	Подготовка документа к поставке версии 1.09.00.
1.7.0	11.12.2019	Смирнова Ю.И.	Подготовка документа к поставке версии 1.08.00. Уточнен перечень поддерживаемых процессоров и микросборок
1.6.0	29.08.2019	Смирнова Ю.И.	Уточнен состав компонентов. Подготовка документа к поставке версии 1.07.00
1.5.0	15.04.2019	Смирнова Ю.И.	Добавлен п. 4.10. Подготовка документа к поставке версии 1.06.04
1.4.0	11.01.2019	Смирнова Ю.И.	Подготовка документа к поставке версии 1.06.03 Уточнен перечень поддерживаемых ОС, дополнен перечень документации на компоненты системы
1.3.1	23.11.2018	Смирнова Ю.И.	Описание порядка установки CM-LYNX вынесено в отдельную инструкцию.
1.3.0	12.11.2018	Смирнова Ю.И.	Подготовка документа к поставке версии 1.06.02
1.2.1	02.11.2018	Смирнова Ю.И.	Изменены номера версий актуальных документов (п.7)
1.2.0	26.10.2018	Смирнова Ю.И.	Подготовка документа к поставке версии 1.06.01
1.1.1	25.09.2018	Смирнова Ю.И.	Подготовка документа к поставке версии 1.06.00
1.1.0	14.09.2018	Галаев С.В.	Финальная версия
1.0.0	12.09.2018	Галаев С.В.	Первая версия

ОГЛАВЛЕНИЕ

1. Введение 4

2. Термины и сокращения 5

3. Назначение и функции CM-LYNX 7

3.1. Функциональное назначение программы 7

3.2. Эксплуатационное назначение программы 7

4. Состав компонентов 8

4.1. Графическая оболочка 8

4.2. Кросс-компилятор CLANG 8

4.3. Ассемблер MCA-TSH 8

4.4. Компоновщик MCLINK-TSH 8

4.5. Отладчик LLDB 9

4.6. Утилита сборки Make 9

4.7. Инструмент для записи флеш-памяти 9

4.8. Генератор образа встраиваемого программного обеспечения 10

4.9. Разделитель сегментов DXE-файла 10

4.10. Набор программных библиотек 11

4.11. Симулятор 11

5. Условия выполнения программы 13

5.1. Минимальный состав аппаратных средств 13

5.2. Минимальный состав программных средств 13

5.3. Поддерживаемые процессоры и микросборки 13

5.4. Требования к персоналу (пользователю) 13

6. Взаимосвязь компонентов 14

7. Состав документации компонентов среды 15

1. Введение

Данный документ содержит общую информацию об интегрированной среде разработке CM-LYNX, структуре компонентов с кратким описанием каждого из них, а также список поддерживаемых процессоров и отладочных комплектов. Дополнительно указан список актуальной документации для каждого компонента среды с их местонахождением.

Для компонентов, имеющих открытый исходный код (Open source), документация выполнена в сокращенном объеме с необходимыми пояснительными ссылками на сайты производителей конкретных компонентов среды.

2. Термины и сокращения

Сокращение	Описание
CDT	C/C++ Development Tools – модуль среды разработки Eclipse для языков C и C++.
DLB	Расширение библиотечных файлов.
DOJ	Расширение объектных файлов после обработки утилитой ассемблера.
DXE	Расширение исполняемых бинарных файлов, генерируемых компоновщиком.
ELF	Executable and Linkable Format — формат исполняемых и компонуемых файлов.
GDB	GNU debugger (отладчик GNU).
GNU Make	Версия программы make, распространяемая Фондом Свободного Программного Обеспечения (Free Software Foundation - FSF) в рамках проекта GNU (www.gnu.org).
GUI	Graphical User Interface – графический пользовательский интерфейс.
HAL	Hardware Abstraction Layer – набор программных функций, предназначенный для взаимодействия с периферийными устройствами процессора.
HASP	Hardware Against Software Piracy (аппаратная защита от пиратского использования программ) – это мультиплатформенная аппаратно-программная система защиты программ и данных от незаконного использования и несанкционированного распространения.
LDF	Linker Description File –файл сценария для компоновщика.
LST	Listing files – файлы листинга ассемблера.
OVL	Расширение файла оверлеев.
SM	Расширение файла разделяемой памяти.
USB	Universal Serial Bus (универсальная последовательная шина) – последовательный интерфейс для подключения периферийных устройств к вычислительной технике.
ИСР	Интегрированная среда разработки, IDE
Оверлей (Overlay)	Фрагмент программы, который будет загружен в память.
Отладчик GDB-MI	Отладчик является построчным, машинно-ориентированным текстовым интерфейсом к GDB.
Программный продукт CM-LYNX, CM-LYNX, Продукт	Набор средств разработки, который включает в себя такие компоненты, как среда разработки, ассемблер, компилятор, компоновщик, отладчик, утилиты для формирования загрузочного образа и встраиваемого программного обеспечения ПЗУ, набор программных библиотек.
Среда CM-LYNX	Интегрированная среда разработки Milandr CodeMaster-Lynx (CM-Lynx).
Тулчейн (toolchain)	Набор программ (компилятор, ассемблер, компоновщик, библиотеки), необходимых для компиляции и генерации выполняемого кода из исходных текстов.
АО	Акционерное общество.

Сокращение	Описание
ОЗУ	Оперативное запоминающее устройство — энергозависимая часть системы компьютерной памяти, в которой во время работы компьютера хранится выполняемый машинный код (программы), а также входные, выходные и промежуточные данные, обрабатываемые процессором.
ОС	Операционная система.
ПЗУ	Постоянное запоминающее устройство.
ПК	Персональный компьютер.
ПО	Программное обеспечение.

3. Назначение и функции CM-LYNX

3.1. Функциональное назначение программы

Продукт **Интегрированная среда разработки CM-LYNX** (CM-LYNX) позволяет вести разработку встраиваемого ПО для высокопроизводительных процессоров цифровой обработки сигнала серии **1967ВН0ХХ** и предоставляет следующие функциональные возможности:

- Управление проектами и конфигурациями разрабатываемого ПО;
- Синтаксически ориентированное редактирование исходных текстов на языках C++, C и ассемблер для **1967ВН0ХХ**;
- Автоматическая проверка синтаксической корректности исходных текстов в процессе редактирования;
- Отображение подсказок, автоматическое дополнение и рефакторинг на основе структурного анализа исходного текста;
- Поддержка процесса сборки проекта с использованием утилит командной строки (компилятор C/C++, ассемблер, компоновщик и др.);
- Структурированное отображение диагностических сообщений по результатам сборки проекта (ошибок, предупреждений) с привязкой к позициям в исходном тексте;
- Обеспечение процесса интерактивной (высокоуровневой и низкоуровневой) отладки ПО;
- Поддержка режима отладки с использованием аппаратного отладчика и симулятора;
- Автоматическая загрузка программы в ОЗУ целевого устройства в момент начала отладки;
- Подготовка бинарных образов встраиваемого ПО для целевых устройств;
- Запись образа ПЗУ во флеш-память целевого устройства;
- Отображение и редактирование данных, содержащихся в памяти и регистрах общего и специального назначения **1967ВН0ХХ**.

3.2. Эксплуатационное назначение программы

Продукт CM-LYNX предназначен для обеспечения процессов разработки и отладки встраиваемого программного обеспечения для высокопроизводительных процессоров цифровой обработки сигнала серии **1967ВН0ХХ**.

4. Состав компонентов

4.1. Графическая оболочка

Графическая оболочка CM-LYNX базируется на функциональности, предоставляемой Eclipse, и предназначена для обеспечения работы с исходным кодом, запуска программы на исполнение, расстановки точек останова, просмотра содержимого ячеек памяти и выполнения прочих действий в рамках процесса разработки и отладки программного обеспечения для высокопроизводительных процессоров цифровой обработки сигнала серии **1967BH0XX**.

4.2. Кросс-компилятор CLANG

В работе CM-LYNX используется кросс-компилятор C/C++ на базе Clang/LLVM для микропроцессоров серии **1967BH0XX** (далее – Компилятор Clang), предназначенный для компиляции программ на языке высокого уровня (C/C++) в ассемблерное представление целевой архитектуры. Применяется для выполнения оптимизации программы как без привязки к архитектуре, так и под целевую архитектуру, с учетом её особенностей.

Компилятор автоматически запускается средой CM-LYNX в момент начала сборки пользователем проекта целиком либо отдельных файлов проекта. Возможен ручной запуск компилятора пользователем из командной строки операционной системы как обычной консольной утилиты.

Результатом работы компилятора является ассемблерный файл, который далее может использоваться прочими утилитами тулчейна (ассемблером и далее компоновщиком) для сборки итогового бинарного файла.

Компилятор Clang разработан на базе проектов с открытым исходным кодом Clang и LLVM версии 4.0.1.

4.3. Ассемблер MCA-TSH

Ассемблер MCA-TSH для микропроцессоров **1967BH0XX**, входящий в состав продукта CM-LYNX, предназначен для трансляции кода, написанного на языке ассемблера целевой платформы, в объектные файлы в формате ELF (промышленном стандарте для объектных файлов). Возможна генерация файла списка (.lst), который показывает соответствие между двоичным кодом и кодом на языке ассемблер.

4.4. Компоновщик MCLINK-TSH

Компоновщик MCLINK-TSH для микропроцессоров серии **1967BH0XX** обрабатывает объектные файлы, получаемые от утилит ассемблера и компилятора, и формирует на их основе исполняемый файл для целевой аппаратной платформы.

Результатом работы компоновщика является исполняемый файл, который далее может использоваться прочими утилитами для создания выходного файла, используемого на целевом процессоре.

Процесс компоновки: в соответствии с файлом описания компоновщика (.ldf), командной строкой и параметрами диалогового окна (параметры проекта CM-LYNX), утилита компоновщика принимает на вход объектные (.doj) и библиотечные (.dlb) файлы для создания исполняемых файлов (.dxe), которые могут быть загружены на симулятор или целевой процессор.

4.5. Отладчик LLDB

Отладчик LLDB проекта LLVM, портированный на платформу **1967BH0XX** (далее – Отладчик LLDB), предназначен для выполнения функций отладки программ для микропроцессоров серии **1967BH0XX** с JTAG-эмулятором и программатором JEM-LYNX в одно- и многопроцессорном режиме.

Пользователь может запустить отладчик двумя способами:

- через консольный интерфейс с помощью команд LLDB;
- через графический интерфейс CM-LYNX (основной способ работы).

Контекст процессора представляется в виде отдельной группы потоков (thread group) с одним входящим в группу потоком. В симметричном многопроцессорном режиме сессия работы с отладчиком LLDB включает несколько групп с потоками, которые соответствуют контекстам нескольких процессоров.

Отладка через LLDB поддерживает возможность многопроцессорной отладки группы процессоров, соединенных в JTAG цепочку через один JEM-LYNX, в одном общем сеансе отладки.

Поддержка многопроцессорной отладки обеспечивает:

- выполнение отдельных отладочных сессий для независимых процессоров (AMP);
- выполнение групповых отладочных сессий для симметричных групп процессоров (SMP).

4.6. Утилита сборки Make

Make – утилита, автоматизирующая процесс преобразования файлов из одной формы в другую. Наиболее частым случаем использования является компиляция исходного кода в объектные файлы и последующая компоновка в исполняемые файлы или библиотеки.

Утилита использует специальные make-файлы, в которых указаны зависимости файлов друг от друга и правила для их выполнения. На основе информации о времени последнего изменения каждого файла make определяет и запускает необходимые программы.

В текущей версии среды CM-LYNX используется утилита Make, основанная на GNU make версии 4.2.

4.7. Инструмент для записи флеш-памяти

Для записи программного кода проекта в ПЗУ целевого устройства в CM-LYNX используется утилита Flasher. Инструмент для записи может использоваться через консоль и через пользовательский интерфейс (GUI). Пользовательский интерфейс позволяет выбрать тип микросхемы FLASH, указать ее параметры и опции, указать программу–программатор FLASH и выполнить нужные манипуляции с флеш-памятью в ручном режиме: стереть, прочитать, записать. Дополнительно пользователь может задать опции частичной или полной очистки микросхемы перед программированием либо опции слияния новых данных с существующими. В последнем случае обновляются только указанные адреса флеш-памяти.

Консольное приложение принимает список аргументов и непосредственно выполняет переданные ему операции в пакетном режиме.

Образ для записи должен быть получен посредством утилиты Loader, генератора образа встраиваемого ПО.

4.8. Генератор образа встраиваемого программного обеспечения

Для генерации образов встраиваемого программного обеспечения ПЗУ процессоров серии **1967ВН0XX** в интегрированной среде разработки CM-LYNX используется утилита Loader. Она может использоваться как в консольном режиме, так и при помощи GUI. Данная утилита позволяет на основе входных файлов формата **.dxe**, **.hex**, **.srec** и/или **.bin**, а также правил для них получать образы следующих форматов: **.bin**, **.hex** или **.srec**.

Входными данными утилиты являются файлы исполняемого кода, файл загрузчика, а также произвольные данные (например, мультимедийные ресурсы). Утилита формирует образ ПЗУ на основании входных файлов и аргументов командной строки. Образ ПЗУ пригоден для непосредственного встраивания ПО в ПЗУ.

Пользователь может сформировать требуемый **.dxe**-файл в процессе разработки и сборки проекта в графическом интерфейсе ИСР CM-LYNX, а затем проверить его работоспособность средствами отладчика GDB-MI, после чего передать на вход утилите Loader для формирования итогового встраивания ПО в устройство.

Сгенерированный образ может быть загружен в ПЗУ целевого устройства с помощью инструмента для встраивания ПО флеш-памяти Flasher.

4.9. Разделитель сегментов DXE-файла

Для извлечения секций из **.dxe**-файлов целевой системы (в формате ELF) используется утилита Splitter. Извлеченные секции получаются в виде файлов кода или данных, которые далее могут быть встроены во внешнюю память и при необходимости запущены на целевом устройстве.

Утилита Splitter принимает на вход **.dxe**-файл и выгружает запрошенные пользователем секции в отдельный файл или файлы, при необходимости разрезая слова. По умолчанию предполагается, что разрезать данные не нужно, в этом случае Splitter работает как конвертер **.dxe** в заданный формат. При указании этой опции к именам результирующих файлов присоединяется суффикс, означающий порядковый номер записи данных в слове (0 – самый младший, 1 – следующий и так далее, максимум до 3).

4.10. Набор программных библиотек

В состав CM-LYNX входит набор программных библиотек:

- Библиотека стандартных функций clib, включая функции ввода-вывода;
- Библиотека функций цифровой обработки сигналов libdsp;
- Библиотеки стандартных функций C++[1];
- Библиотека HAL для микропроцессоров 1967ВН028/1967ВН044/1967ВН058.

Каждая библиотека представляет собой набор функций, вызываемых из C/C++ программ. Описание библиотек clib и libdsp представлено в документе «ML_Описание_функций_библиотек». Описание библиотек HAL для каждого периферийного устройства процессоров 1967ВН028/1967ВН044 представлено в документах «hal_.. .docx», которые расположены по пути «toolchain\1967VN0XX\HAL\doc» относительно корневого каталога среды.

4.11. Симулятор

В отсутствие отладочной платы возможна отладка с использованием программного симулятора, который эмулирует работу микропроцессоров серии **1967ВН0XX**, а также многопроцессорных систем на их основе (допустимо объединение до 8 микропроцессоров 1967ВН028 в многопроцессорную систему). Пользователь может задать конфигурацию многопроцессорной системы (при необходимости) и вызвать симулятор через графический интерфейс CM-LYNX.

Начиная с версии 2.01.02, добавлена экспериментальная поддержка симулятора для 1967ВН058. Симулятор 1967ВН058 включает:

- поддержку системы команд 1967ВН058, включая все новые расширения;
- поддержку работы MMU;
- поддержку кэшей команд и данных 1-го уровня;
- контроллер прерываний и таймер в периферии 2-го уровня.
- поддержку до 16 ядер (8 процессоров по 2 ядра);
- поддержку памяти DDR, NOR, NAND и SRAM, с возможностью конфигурирования объёма;
- поддержку межъядерного адресного пространства;
- отправку прерывания Kernel на соседнее ядро по инструкции SEND_EVENT.

Примечание: Есть ограничения в поддержке регистров периферии 1967ВН058 в среде.

Подробнее о работе симулятора 1967ВН058 см. документ Инструкция по работе с симулятором 1967ВН058.

Для процессора 1967ВН044 разработан отдельный симулятор. Симулятор 1967ВН044 разработан основе симулятора для 1967ВН028 со следующими отличиями:

- внутренняя память уменьшена в 2 раза;
- внешняя память поддерживает байтовый обмен;
- удалены все многопроцессорные функции ВН28;

[1] Поддержка стандартных библиотек C++ ограничена. Начиная с версии 1 08.00 добавлена поддержка библиотеки libunwind, libcxxabi, libcxx, compiler-rt в объеме, необходимом для возможности использования механизма исключений C++ при разработке приложений. Начиная с версии 2 00.00 поддерживается почти весь функционал библиотеки libcxx для CHAR32 + DOUBLE64 (функции, доступные до версии 2.00.00 поддерживаются для любой комбинации разрядности char и double), кроме генерации исключений внутри библиотеки, региональных настроек. Начиная с версии 2.03.00 также добавлены библиотеки C++ без поддержки механизма исключений.

- вместо HOST-пространства добавлен блок памяти для эмуляции записи-чтения области периферийных устройств.

5. Условия выполнения программы

5.1. Минимальный состав аппаратных средств

Для работы CM-LYNX необходим ПК, удовлетворяющий следующим требованиям:

- Совместимость с ОС Microsoft Windows 7 (64-bit)/ Windows 10 (64-bit);
- Свободное место на диске не менее 1 Гбайт;
- ОЗУ не менее 4 Гбайт.

5.2. Минимальный состав программных средств

Для работы CM-LYNX необходим следующий минимальный состав программных средств:

- ОС Microsoft Windows 7 (64-bit)/ Windows 10 (64-bit);
- Java Runtime Environment не ниже версии 1.8.x;
- Microsoft Visual C++ 2015 Redistributable (или комплект, включающий его).

5.3. Поддерживаемые процессоры и микросборки

Среда CM-LYNX поддерживает работу со следующими интегральными микросхемами и микросборками производства АО «ПКК Миландр»:

- Сигнальный процессор со статической суперскалярной архитектурой 1967ВН028, К1967ВН028, К1967ВН028К, 1967ВН02Н4, К1967ВН02Н4;
- Микросхема высокопроизводительного процессора цифровой обработки сигнала с суперскалярной архитектурой 1967ВН044, К1967ВН044, К1967ВН044К, 1967ВН04Н4, К1967ВН04Н4;
- Микросборка «Флип-Чип» (с перовой приемкой наименование – «СОД», с пятой приемкой наименование – «МВМ-03») – состав: 4 кристалла процессора 1967ВН028, память 1636РР2У (на текущий момент заменен на 1636РР4У);
- Микросборка «Осведомленность» (с перовой приемкой наименование – «СЗД», с пятой приемкой наименование – «9018ВК016») – состав: процессор 1967ВН044, память 1636РР4У, 2 кристалла АЦП 5101НВ015, 2 кристалла 1309ЕР1.8Т, один кристалл микросхемы 1310ПН1У.

5.4. Требования к персоналу (пользователю)

Для эксплуатации среды CM-LYNX от пользователя требуются общие навыки работы с персональным компьютером, командной строкой ОС, знания в области разработки программного обеспечения для микропроцессорной техники.

6. Взаимосвязь компонентов

Взаимосвязь компонентов CM-LYNX показана на схеме ниже (см. Схема 1).

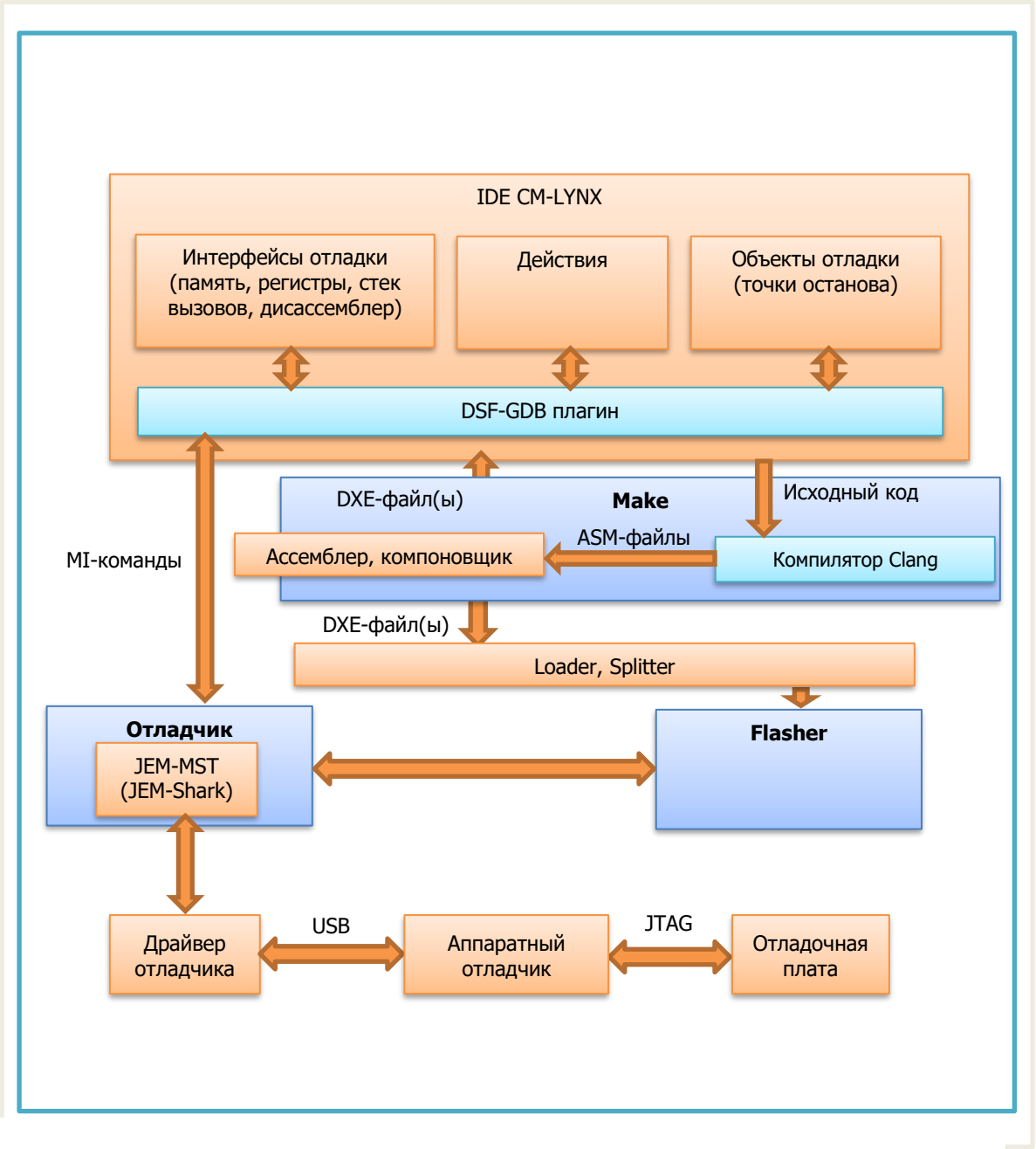


Схема 1. Взаимосвязь компонентов CM-LYNX

Среда CM-LYNX позволяет обрабатывать пользовательские файлы с исходным кодом и преобразовывать их в исполняемые объектные файлы DXE. Для этого среда формирует временный Make-файл и запускает утилиту Make. В ходе своей работы Make запускает утилиты тулчейна: компилятор, ассемблер, компоновщик. Полученные в результате последовательной работы этих утилит DXE-файлы могут быть переданы утилитам Loader, Splitter и впоследствии утилите Flasher, либо напрямую использованы отладчиком для загрузки на целевое устройство. Среда CM-LYNX позволяет пользователю прозрачно работать с отладчиком для запуска программ и пошаговой отладки. Утилиты Loader и Flasher могут использоваться пользователем для работы с DXE-файлом напрямую из командной строки операционной системы или запускаться с графическим интерфейсом, в том числе из среды CM-LYNX. Утилита Splitter доступна только из командной строки операционной системы. Библиотека JEM-Shark используется только при работе со старым драйвером отладчика («Phyton USB Device»), но не используется при работе с новым драйвером отладчика («AngioScan JEM-HS»).

7. Состав документации компонентов среды

В таблице ниже (см. Таблица 1) представлены документы компонентов среды.

Таблица 1. Состав документации компонентов среды

Компонент	Документ
Графическая оболочка	ML_GUI _Руководство_пользователя
Компилятор Clang	ML_Clang_Руководство_пользователя
Ассемблер MCA-TSH	ML_MCA-TSH_Руководство_пользователя
Компоновщик MCLINK-TSH	ML_MCLINK-TSH Руководство_пользователя
Отладчик LLDB	ML_LLDB Руководство_пользователя
Утилита Flasher	ML_Flasher_Руководство_пользователя
Утилита Loader	ML_Loader_Руководство_пользователя
Утилита Splitter	ML_Splitter_Руководство_пользователя
Набор программных библиотек	ML_Описание_функций_библиотек
ОСРВ МАКС Lynx[2]	ML_ОСРВ_МАКС_Lynx_Руководство_пользователя
МАКС Профилирование. Многопроцессорные системы	ML_МАКС_Профилирование_Lynx_Руководство_пользователя

[2] Компонент добавлен в состав среды CM-LYNX для ознакомления. Разработка и техническая поддержка компонента осуществляется компанией ООО «МИР».