

4 Конфигурирование H-Jtag

В этой главе подробно рассказано о том, как конфигурировать и использовать H-Jtag. К тому же для справки приведены простые примеры.

4.1 Детектирование подключения

Перед детектированием, пожалуйста, подключите эмулятор JTAG к USB или параллельному порту, а к эмулятору подключите выбранное устройство. При включении, H-Jtag пытается детектировать подключение автоматически. После запуска H-Jtag, пользователь может кликнуть на кнопку детектирования для детектирования устройства. Если устройство детектируется, ядро ARM и ID чипа отображаются в центре основного окна. Если нет, проверьте конфигурацию и аппаратное подключение.

4.2 Сброс подключения

Пользователь может сбросить устройство через H-Jtag. Для стандартного JTAG интерфейса есть два независимых сигнала сброса: сброс системы (nSRST) и сброс JTAG/TAP (nTRST). С помощью этих сигналов сброса, H-Jtag выполняет сброс системы или сброс JTAG/TAP. Пользователь может выбрать выполнение сброса системы или сброса JTAG/TAP, или обоих вместе. H-Jtag предлагает различные опции. Обратитесь к следующей секции за подробной информацией.

Внимание: Некоторые эмуляторы JTAG на LPT не реализуют сигнал сброса системы, поэтому H-Jtag не может сбросить систему в чипе. Подключенный чип может сбрасываться только вручную. Некоторые эмуляторы JTAG на LPT объединили сигналы сброса системы и сброса JTAG/TAP. В этом случае, любая операция сброса назначается обоим сигналам.

4.3 Автоматическая загрузка во флеш

Некоторые чипы ARM имеют внутреннюю флеш и RAM, так что не требуют внешней памяти. Для такого набора чипов, например чипы серии LPC2000, AT91SAM7, LUMINARY CORTEX-M3 и STM32F, H-Jtag поддерживает автоматическую загрузку во флеш. С помощью автоматической загрузки во флеш, программа может быть прямо загружена или записана во флеш для отладки, как при отладке в RAM/SDRAM. Для использования автоматической загрузки во флеш, пожалуйста, разрешите опцию *auto download*, как показано на рис. 4-1. Одновременно выберите соответствующую флеш в H-Flasher. Во время загрузки, H-Jtag будет определять, какая программа должна быть загружена по какому адресу. При необходимости записи во флеш, H-Jtag вызывает H-Flasher автоматически.



Fig 4-1 Auto Flash Download

Внимание: автоматическая загрузка во флеш поддерживается только для on-chip флеш, такой как LPC2000 и AT91SAM7. Потому что устройство памяти в этих чипах относительно схоже. Для чипов со сложной конфигурацией памяти, такой как

MMU/REMPA, пожалуйста, пишите программу во флеш, используя H-Flasher, до начала отладки.

Внимание: H-Flasher Lite не поддерживает автоматическую загрузку во флеш. Для использования автоматической загрузки во флеш, запустите H-Flasher.

4.4 Скрипт инициализации

Для большинства систем, процессы инициализации должны выполняться после включения питания. Инициализация системы памяти – одна из них. Большинство времени, флеш и on-chip SRAM могут быть доступны прямо после включения питания, но это не относится к внешней SRAM. Внешняя SRAM должна быть инициализирована перед тем, как к ней будет установлен корректный доступ. Для начинающих пользователей, одной из общих проблем является невозможность корректной загрузки программы во внешнюю SDRAM. Это происходит от того, что внешняя SDRAM не была изначально проинициализирована. Существует два способа инициализации подключенной системы. Первый – это записать программу инициализации во флеш. Эта программа будет исполняться сразу при включении питания. Т.о. устройство будет проинициализировано автоматически при включении питания, и будет готово к использованию. Второй метод – это использовать скрипт инициализации. Для продвинутых пользователей, H-Jtag определяет несколько команд скриптов и предлагает автоматическую инициализацию. Для использования автоматической инициализации, пользователю необходимо ввести или загрузить соответствующий скрипт инициализации и разрешить автоматическую инициализацию. Если автоматическая инициализация разрешена, H-Jtag будет исполнять соответствующие скрипты при открытии соединения отладчиком. Редактор скриптов показан на следующем рисунке. Подробная информация о скриптах инициализации в главе 6.

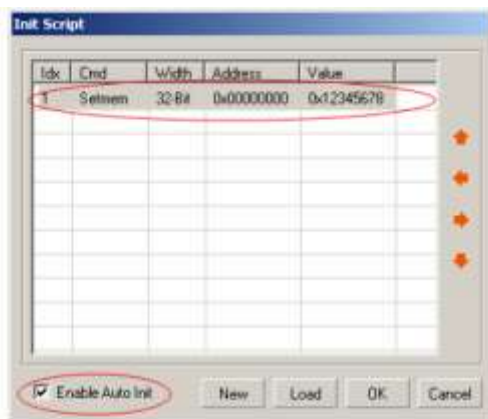


Fig 4-2 Script Editor

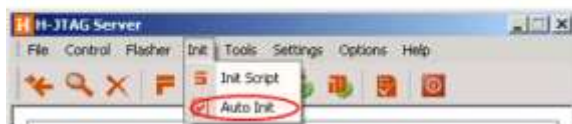


Fig 4-3 Auto Init

Внимание: Если разрешена автоматическая инициализация в H-Jtag сервере, то будут предлагаться скрипты инициализации. Иначе, H-Jtag предупредит пользователя сообщением об ошибке: "Не могу открыть заданный скрипт инициализации".

Внимание: Подробная информация о скриптах инициализации в главе 6.

4.5 Выбор интерфейса USB/LPT

H-Jtag поддерживает эмуляторы JTAG на LPT и H-JTAG на USB. Пользователь может выбрать соответствующее аппаратное средство.



Fig 4-4 USB/LPT Interface Selection

4.6 Конфигурация JTAG

В этой секции рассказывается о назначении JTAG интерфейса, подключении к порту USB/LPT и конфигурации JTAG.

4.6.1 JTAG сигналы

JTAG – это тестовый стандарт, предлагаемый IEEE. Для отладки ARM, JTAG используется в качестве интерфейса. Интерфейс ARM JTAG назначает 7 сигналов: TMS, TCK, TDI, TDO, RTCK, nSRST и nTRST. Для отладки ARM7 и ARM9 сигналы: TMS, TCK, TDI и TDO - обязательны, а RTCK, nSRST и nTRST – опциональны.

Внимание: Для XScale оба сигнала nSRST и nTRST – обязательны. Эти сигналы должны быть разделены. Иначе отладка невозможна.

4.6.2 Подключение JTAG

Типовое подключение JTAG отображено на рис. 4-5. Эмулятор JTAG подключён к USB/LPT и к выбранному устройству. H-JTAG генерирует сигналы JTAG в эмуляторе JTAG для контроля над выбранным устройством. Интерфейс JTAG между эмулятором JTAG и выбранным ARM устройством забирает 14 выводов из 20 стандартных. Сервер H-Jtag и эмулятор JTAG связываются друг с другом через USB порт или LPT порт. Когда используется LPT, пользователь должен обеспечить точную конфигурацию JTAG и удостовериться, как H-JTAG и JTAG эмулятор подключены к LPT. Подробная информация о конфигурации JTAG в следующих секциях.

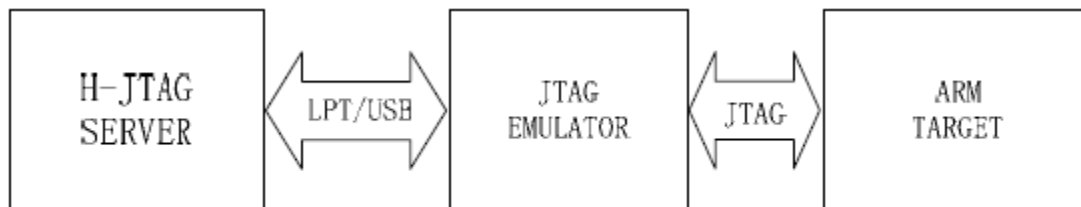


Fig 4-5 JTAG Connection

4.6.3 Настройки USB JTAG

Эмулятор H-Jtag USB поддерживает различные скорости TCK (25 кГц – 15 МГц). Пользователь может выбрать соответствующую скорость в диалоге USB JTAG setting (рис. 4-6). Когда выбирается AUTO TCK, H-Jtag выбирает скорость TCK автоматически в момент тестирования.



Fig 4-6 USB JTAG Setting

Внимание: Скорость TCK прямо влияет на характеристики отладки. Различные системы подключенных устройств имеют различные максимальные скорости TCK. Максимальная TCK также зависит от конфигурации системного такта в выбранной системе. Нужная скорость TCK может быть разработана. Превышение скорости TCK приводит к непредсказуемым последствиям. В некоторых ситуациях AUTO TCK может не работать. В этом случае, разработайте вручную более низкую скорость TCK.

4.6.4 Настройки LPT JTAG

Контроллер JTAG LPT не имеет фиксированной разводки, даже для WIGGLER и SDT-JTAG. Некоторые JTAG эмуляторы поставляются с nSRST, тогда как другие не имеют его. Некоторые JTAG эмуляторы предлагают отдельные nSRST и nTRST, а другие объединяют их. Для поддержки различных JTAG LPT эмуляторов, H-JTAG предлагает гибкий интерфейс конфигурации. Он необходим пользователям, чтобы сделать H-Jtag точно таким же, как и эмулятор JTAG, подключенный к LPT.

LPT имеет 8 бит данных, D0 - D7, на выходе и несколько статусных битов на входе. Биты данных могут использоваться в качестве выходных сигналов JTAG: TMS, TCK, TDI, nSRST и nTRST. Один любой статусный бит может быть использован как вход для приёма TDO. Конфигурация JTAG определяет, как подключаются сигналы JTAG к битам данных и статуса. В таком эмуляторе JTAG, сигналы nSRST и nTRST инвертированы. Это также нужно задать в конфигурации JTAG.

Посмотрите на иллюстрированный пример. Разводка эмулятора JTAG, используемая в этом примере, показана на рис. 4-7. Соединение между LPT и JTAG осуществляется по схеме, приведённой в следующей таблице. Обратите внимание на то, что сигнал nTRST инвертирован, а сигнала nSRST нет.

TMS	➡	LPT D1 (PIN3)
TCK	➡	LPT D2 (PIN4)
TDI	➡	LPT D3 (PIN5)
TDO	➡	LPT BUSY (PIN11)
nTRST	⬅	LPT D0 (PIN2) INVERTED
nSRST	✗	NOT AVAILABLE

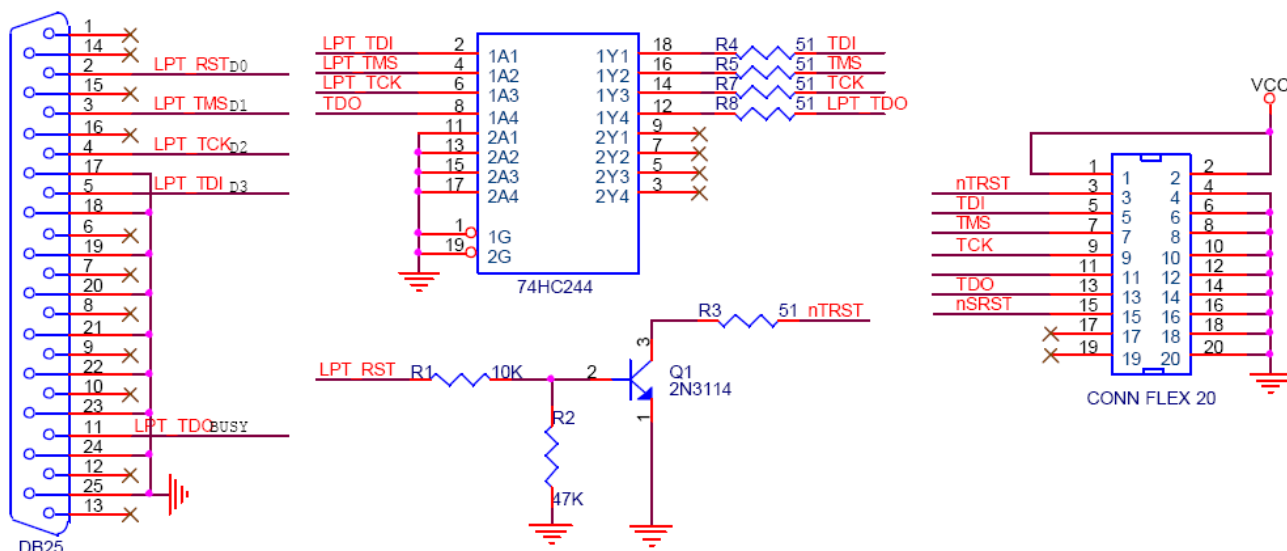


Fig 4-7 Example Schematic of JTAG Emulator

Основываясь на схеме, представленной на рис. 4-7, и вышеописанном анализе, будут использованы некоторые следующие настройки. Обе настройки, приведённые на рис. 4-8, задают конфигурацию соединения LPT с интерфейсом JTAG посредством эмулятора JTAG. Приведённый пример дан для справки. Опираясь на него, вы сможете сконфигурировать собственный эмулятор JTAG, основываясь на своей схеме.

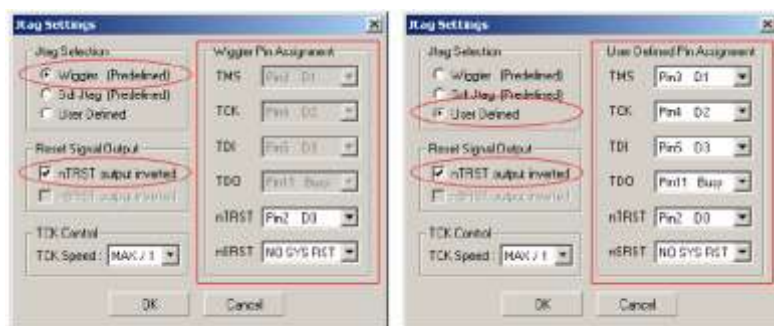


Fig 4-8 Example JTAG Settings

В диалоге настроек LPT JTAG, пользователь также может выбрать различные скорости TCK. Выбираемая скорость определяется между MAX и MAX/8. Известно, что LPT порт – низкоскоростной интерфейс. Обычно пользователь использует MAX/1 в качестве скорости по умолчанию, для достижения максимальных характеристик.

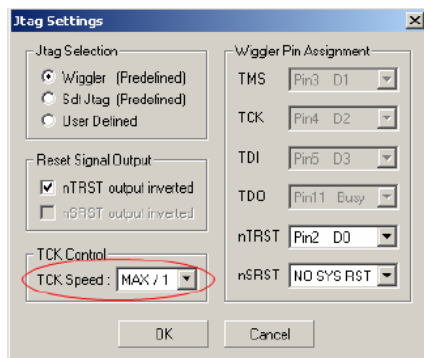


Fig 4-9 Selection of LPT TCK Speed

4.7 Настройки LPT порта

Для большинства компьютеров, адрес LPT порта по умолчанию 0x378, но есть некоторые исключения. В H-Jtag можно задать другой адрес порта. Диалог настроек порта приведён на рис. 4-10. В диалоге есть кнопка тестирования, которая используется для простого теста чтения/записи в порт.

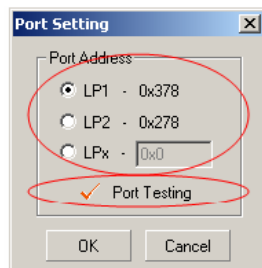


Fig 4-10 LPT Port Setting

4.8 Тестирование подключенного устройства

H-Jtag читает ID через JTAG и определяет тип ядра ARM. H-Jtag может различать большинство известных чипов. Для подключенных устройств, которые не могут определиться, пользователь указывает тип ядра ARM в настройках устройства. Диалог настроек устройства показан на рис. 4-11.

Большинство ARM чипов поддерживают прямой (little endian) и обратный (big endian) порядок байтов. В различных порядках байтов, сохранение данных и инструкций сильно отличается. Если порядок байтов не задан корректно, отладка будет неверной. Пользователь может задать порядок байтов в настройках устройства (см. ниже).



Fig 4-11 Target Setting

4.9 Менеджер подключенного устройства

В предыдущих секциях было сказано, что H-Jtag определяет ядро ARM по ID чипа. Для чипов, которые невозможно идентифицировать, пользователь может задать ядро ARM вручную. Пользователь также может добавить ID чипа в список чипов. После обновления списка чипов, H-Jtag сможет детектировать и распознавать новые чипы автоматически. Всё это можно сделать в менеджере подключенного устройства. Для добавления нового ID устройства, пользователю необходимо ввести ID и задать соответствующее ядро ARM. В менеджере подключенного устройства пользователь может удалить существующие ID чипов. Менеджер подключенного устройства показан на рис. 4-12.



Fig 4-12 Target Manager

Совет: В соответствии со стандартом IEEE-1149, ID чипа – 32 бита, и младший бит должен быть 1. Пользователь может выбрать, следовать ему или нет. Если у вас есть чип, который не распознаётся H-Jtag, пожалуйста, пришлите нам его ID и тип ядра ARM. Мы обновим список чипов для следующей версии.

4.10 Конфигурация TAP

В большинстве чипов ARM, цепи сканирования JTAG разделены. Для этих чипов, конфигурация TAP (тест-порта) может быть адаптирована, как показано на рис. 4-13. На рисунке показано, что нет никаких других последовательных цепей сканирования перед и после ядра ARM.

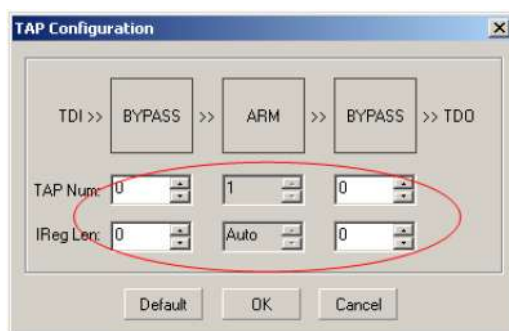


Fig 4-13 Default TAP Configuration

Некоторые чипы включают в себя другие цепи сканирования, не относящиеся к ядру ARM. Для таких чипов, пользователю нужно соответствующим образом сконфигурировать TAP. STR91xF – это чип ST. Этот чип включает в себя несколько внутренних цепей сканирования, как показано на рис. 4-14, - STR91xF имеет 3 тест-порта (TAP): TAP#1, TAP#2 и TAP#3. Из всех этих тест-портов, только TAP#2 предназначен для ARM отладки. TAP#1 и TAP#3 последовательно подключены до и после TAP#2. IR (регистр инструкций) для TAP#1 и TAP#3 имеет 5-битовую и 8-битовую длину соответственно. Для этого чипа, конфигурация TAP должна выглядеть как на рис. 4-15. На этом рисунке видно, что имеется цепь сканирования перед ядром ARM, и её IR длина – 5 бит. Дополнительно, есть другая цепь сканирования после ядра ARM, и её IR длина – 8 бит. Основываясь на этой конфигурации, H-Jtag знает, как получить доступ к ядру ARM, для отладки.

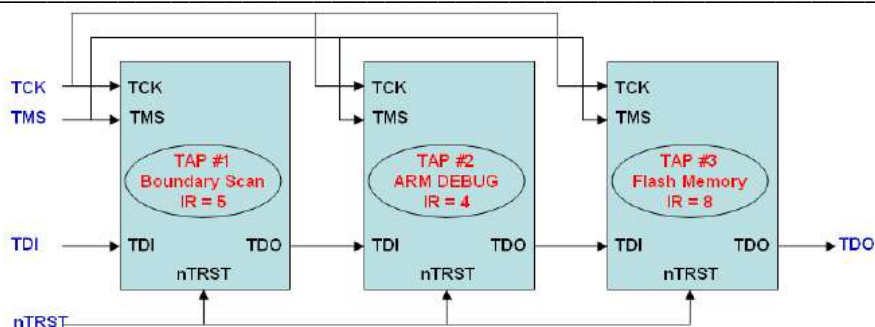


Fig 4-14 STR91xF Scan Chains

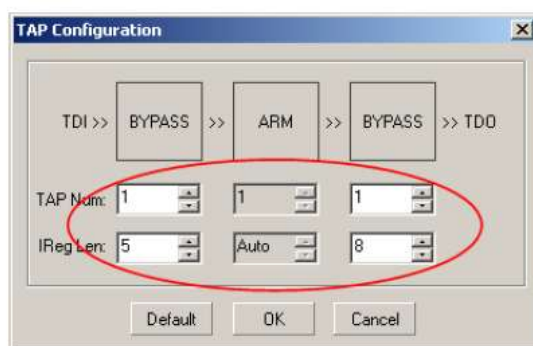


Fig 4-15 TAP Configuration (STR91xF)

4.11 Опции H-Jtag

H-Jtag предлагает следующие основные опции. Пользователь может сделать выбор, используя основное окно опций или меню опций. Основное окно опций показано на рис. 4-16.

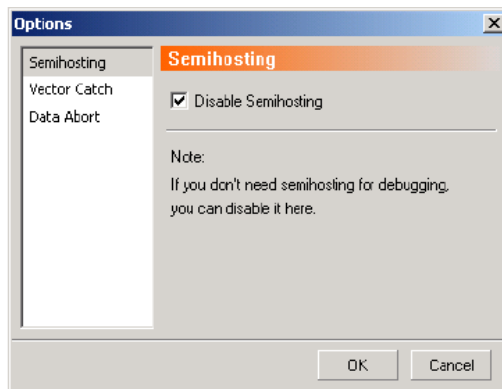


Fig 4-16 Main Window of Options

■ CONNECTION MODE

Различные режимы соединений могут быть определены в H-Jtag. В списке режимов соединений пользователь сможет выбрать нужный тип соединений.

■ VECTOR CATCH CONFIG

H-Jtag сервер управляет конфигурацией вектора захвата. В диалоге конфигурации (рис. 4-17) пользователь может выбрать, какие исключения/прерывания будут захватываться во время отладки. Когда все флаги сброшены, вектор захвата запрещается автоматически. Иначе, если разрешена опция глобального запрета вектора захвата, то автоматически игнорируются назначения в этом диалоге.



Fig 4-17 Vector Catch Configuration

- **DISABLE SEMIHOSTING**

Semihosting – это механизм отладки, который используется для связи между хостом и подключенным устройством. Semihosting нуждается в поддержке эмулятором м программой, запущенной в подключенном устройстве. К тому же, Semihosting может использоваться только в целях отладки и не может использоваться в готовом продукте. Semihosting потребляет ресурсы точек останова. Мы рекомендуем запрещать Semihosting.

- **DISABLE VECTOR CATCH**

Вектор захвата используется для захвата исключений. Когда вектор захвата разрешён в H-Jtag и в отладчике, то они извещаются о событиях исключений. Внимание, вектор захвата потребляет ресурсы точек останова. Мы рекомендуем запрещать вектор захвата.

- **REPORT DATA ABORT**

Во время отладки, всегда при остановке процессора, отладчику необходим доступ к памяти подключенного устройства. Прерываемые данные могут оказаться в неопределённой области или получить доступ к защищённой области. Если разрешён отчет от прерываемых данных, H-Jtag извещает пользователя, когда данные прерваны. Иначе, H-Jtag обрабатывает прерванные данные внутри, без извещения пользователя.

- **DISABLE IAR BKPT @ 0x8**

В IAR точка останова задана на 0x8 во время отладки. Эта точка останова невидима в списке точек останова. Когда отладка ведётся внутри флеш, пользователь может запретить невидимую точку останова, чтобы высвободить дополнительную единицу точки останова.

4.12 Инструменты H-Jtag

H-Jtag предлагает некоторые инструменты в меню Tools. В будущем, мы добавим больше инструментов в это меню, необходимых пользователю.

4.13 Проверка обновлений

Пользователь может проверять обновления, используя отметку в меню. Если доступна свежая версия, H-Jtag известит о ней пользователя. Пользователь также может посетить [домашнюю страницу H-Jtag](#) за дополнительной информацией.