

- 3x32-битная шина АНВ. Шина Cortex®-M7 AXI Master 64-битная, разделенная на 4 мастера по мосту AXI – АНВ.
- 1x64-битная шина АНВ, подключенная к встроенной FLASH
- периферийная шина Cortex® -M7 АНВ
- шина памяти DMA1
- шина памяти DMA2
- периферийная шина DMA2
- Ethernet DMA шина
- USB OTG HS DMA шина
- LCD-контроллер DMA-bus
- Шина памяти Chrom-Art Accelerator™ (DMA2D)
- Восемь подчиненных шин:
 - встроенная Flash на шине АНВ (для доступа к чтению/записи Flash, для выполнения кода и доступа к данным)
 - подчиненный интерфейс Cortex® -M7 АНВS для передачи данных DMA только в оперативной памяти DTCM.
 - основной внутренний SRAM1 (240 КБ)
 - вспомогательный внутренний SRAM2 (16 КБ)
 - периферийные устройства АНВ1, включая мосты АНВ-АРВ и периферийные устройства АРВ
 - периферийные устройства АНВ2, включая мосты АНВ-АРВ и периферийные устройства АРВ
 - FMC
 - Quad SPI

2.1.1 Multi АНВ BusMatrix

Multi АНВ BusMatrix управляет арбитражем доступа между мастерами. Арбитраж использует алгоритм циклического перебора.

Он обеспечивает доступ от главного к подчиненному, обеспечивая параллельный доступ и эффективную работу, даже если несколько высокоскоростных периферийных устройств работают одновременно.

ОЗУ DTCM и ITCM (тесно связанные памяти) не являются частью матрицы шины.

ОЗУ Data TCM доступно через GP-DMA и периферийные DMA через специальную подчиненную шину АНВ ЦПУ.

Инструкция TCM RAM зарезервирована только для процессора. доступ к нему осуществляется с тактовой частотой процессора с 0 состояниями ожидания. Архитектура показана на рисунке 1.

2.1.2 Мосты АНВ / АРВ (АРВ)

Два моста АНВ / АРВ, АРВ1 и АРВ2, обеспечивают полностью синхронные соединения между АНВ и двумя шинами АРВ, обеспечивая гибкий выбор периферийной частоты.

Для получения более подробной информации о максимальных частотах АРВ1 и АРВ2, а также Table 1 для сопоставления адресов периферийных устройств АНВ и АРВ см. Технические характеристики устройства.

После каждого сброса устройства все периферийные часы отключаются (за исключением SRAM, DTCM, RAM ITCM и интерфейса флэш-памяти). Перед использованием периферийного устройства его часы должны быть включены в регистре RCC_AHBxENR или RCC_APBxENR.

Заметка:

Когда 16— или 8-битный доступ выполняется в регистре APB, доступ преобразуется в 32-битный доступ: мост дублирует 16— или 8-битные данные для подачи 32-битного вектора.

2.1.3 Шина AXIM процессора

Эта шина соединяет инструкцию и шину данных Cortex®-M7 с ядром FPU с шиной Matrix-AHB через мост AXI-AHB. Существует 4 доступа к шине AXI:

- CPU AXI доступ к шине 1: целью этой шины AXI является FMC внешней памяти, содержащая код или данные. Для банка NAND, сопоставленного по адресу от 0x8000 0000 до 0x8FFF FFFF, атрибут памяти MPU для этого пространства должен быть переконфигурирован программно для устройства.

- CPU AXI доступ к шине 2: целью этой шины AXI является внешний модуль памяти Quad SPI, содержащий код или данные.

- Доступ к шине AXI ЦПУ 3: целью этой шины AXI являются внутренние SRAM (SRAM1 и SRAM2), содержащие код или данные.

- CPU AXI доступ к шине 4: целью этой шины AXI является встроенная флэш-карта, отображаемая на интерфейсе AXI, содержащем код или данные.

2.1.4 Шина ITCM

Эта шина используется Cortex®-M7 для выборки команд и доступа к данным во встроенной флэш-памяти, отображаемой на интерфейсе ITCM, а выборки команд — только в оперативной памяти ITCM.

2.1.5 DTCM шина

Эта шина используется Cortex®-M7 для доступа к данным в оперативной памяти DTCM. Это также может быть использовано для получения инструкций.

2.1.6 Шина процессора AHBS

Эта шина соединяет ведомую шину AHB Cortex®-M7 с BusMatrix. Эта шина используется DMA и периферийными DMA для передачи данных только в оперативной памяти DTCM.

Шина ITCM недоступна в AHBS. Поэтому передача данных DMA в/из ОЗУ ITCM не поддерживается. Для передачи DMA в/из Flash через интерфейс ITCM все передачи выполняются через шину AHB.

2.1.7 AHB периферийная шина

Эта шина соединяет периферийную шину AHB Cortex®-M7 с BusMatrix. Эта шина используется ядром для выполнения всех операций доступа к данным периферийным устройствам. Цель этой шины — периферия AHB1, включая периферию APB и периферию AHB2.

2.1.8 Шина памяти DMA

Эта шина соединяет главный интерфейс шины памяти DMA с BusMatrix. Он используется DMA для передачи в / из памяти. Цели этой шины — память данных: внутренняя память SRAM1, SRAM2 и DTCM (через шину AHBS от Cortex®-M7), внутренняя флэш-память и внешняя память через FMC или Quad SPI.

2.1.9 DMA периферийная шина

Эта шина соединяет интерфейс главной шины периферийного устройства DMA с BusMatrix. Эта шина используется DMA для доступа к периферийным устройствам АНВ или для передачи из памяти в память. Целевыми объектами этой шины являются периферийные устройства АНВ и APB, а также память для хранения данных: внутренняя память SRAM1, SRAM2 и DTCM (через шину AHBS Cortex®-M7), внутренняя флэш-память и внешняя память через FMC или Quad SPI.

2.1.10 Шина Ethernet DMA

Эта шина соединяет главный интерфейс Ethernet DMA с BusMatrix. Эта шина используется Ethernet DMA для загрузки/сохранения данных в памяти. Цели этой шины — это память данных: внутренняя память SRAM1, SRAM2 и DTCM (через шину AHBS Cortex®-M7), внутренняя флэш-память и внешняя память через FMC или Quad SPI.

2.1.11 Шина USB OTG HS DMA

Эта шина соединяет основной интерфейс USB OTG HS DMA с BusMatrix. Эта шина используется USB OTG DMA для загрузки/сохранения данных в памяти. Цели этой шины: память данных: внутренняя память SRAM1, SRAM2 и DTCM (через шину AHBS Cortex®-M7), внутренняя флэш-память и внешняя память через FMC или Quad SPI.

2.1.12 LCD-TFT контроллер шины DMA

Эта шина соединяет основной интерфейс DMA контроллера LCD с BusMatrix. Используется LCD-TFT DMA для загрузки данных из памяти. Цели этой шины: память данных: внутренняя память SRAM1, SRAM2 и DTCM (через шину AHBS Cortex®-M7), внешняя память через FMC или Quad SPI и внутренняя флэш-память.

2.1.13 Шина DMA2D

Эта шина соединяет главный интерфейс DMA2D с BusMatrix. Эта шина используется графическим ускорителем DMA2D для загрузки / сохранения данных в памяти. Цели этой шины: память данных: внутренняя память SRAM1, SRAM2 и DTCM (через шину AHBS Cortex®-M7), внешняя память через FMC или Quad SPI и внутренняя флэш-память.

2.2 Организация памяти

2.2.1 Введение

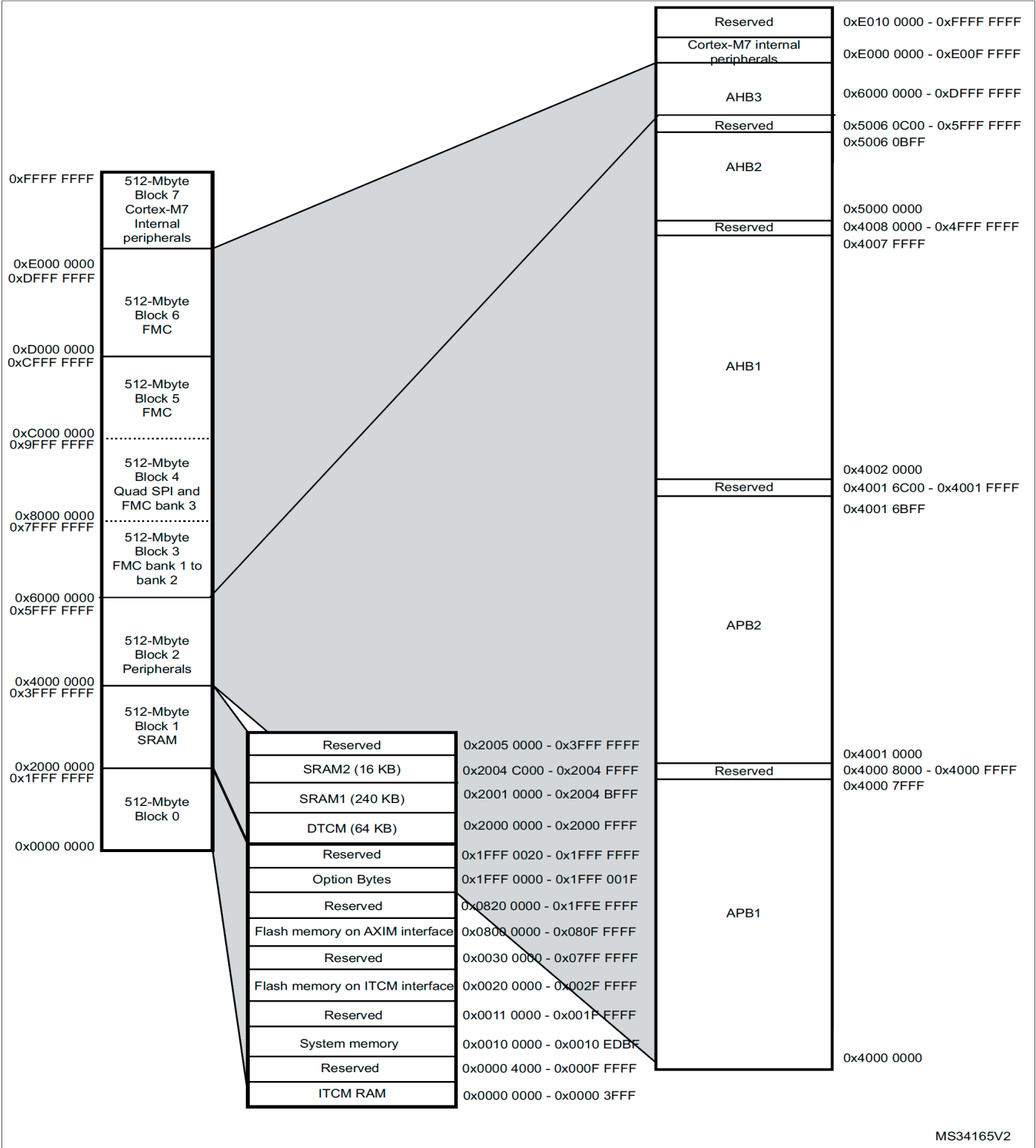
Память программ, память данных, регистры и порты ввода/вывода организованы в одном линейном 4-гигабайтном адресном пространстве.

Байты кодируются в памяти в формате Little Endian. Байт с наименьшим номером в слове считается наименее значимым байтом слова, а байт с наибольшим номером самым значимым.

Адресная область памяти разделена на восемь основных блоков по 512 Мбайт каждый.

2.2.2 Memory map and register boundary addresses

Figure 2. Memory map



2.3 Встроенная SRAM

Функции STM32F75xxx и STM32F74xxx:

- Системная SRAM до 320 Кбайт, включая оперативную память TCM 64 Кбайт
- Память ОЗУ (ITCM-RAM) 16 Кбайт.
- 4 Кбайт резервной SRAM (см. Раздел 5.1.2: Домен резервного копирования

батареи)

Встроенная SRAM разделена на четыре блока:

- SRAM системы:
 - SRAM1 отображается по адресу 0x2001 0000 и доступен для всех мастеров АНВ из матрицы шины АНВ.
 - SRAM2 отображается по адресу 0x2004 C000 и доступен всем мастерам АНВ из шины АНВ Matrix.
 - DTCM-RAM на интерфейсе TCM (интерфейс Tightly Coupled Memory), отображенный по адресу 0x2000 0000 и доступный для всех мастеров АНВ из матрицы шины АНВ, но через определенную подчиненную шину АНВ ЦПУ.
- SRAM команд:
 - Память с инструкциями (ITCM-RAM) отображается по адресу 0x0000 0000 и доступна только для CPU.

SRAM1 и SRAM2 могут быть доступны в виде байтов, полуслов (16 бит) или полных слов (32 бита). Доступ к ОЗУ DTCM и ITCM осуществляется в виде байтов, полуслов (16 бит), полных слов (32 бита) или двойных слов (64 бита). Эти запоминающие устройства могут быть адресованы на максимальной системной тактовой частоте без состояния ожидания.

Мастера АНВ поддерживают одновременный доступ к SRAM (из Ethernet или USB OTG HS): например, Ethernet MAC может считывать/записывать из/в SRAM2, в то время как ЦП читает/записывает из/в SRAM1.

2.4 Обзор флэш-памяти

Интерфейс флэш-памяти управляет доступом CPU AXI и TCM к флэш-памяти. Он реализует операции стирания и программирования флэш-памяти, а также механизмы защиты от чтения и записи. Это ускоряет выполнение кода с помощью ART на интерфейсе TCM или L1-Cache на интерфейсе AXIM.

Флэш-память организована следующим образом:

- Блок основной памяти, разделенный на сектора.
- Информационный блок:
 - Системная память, с которой устройство загружается в режиме загрузки системной памяти
 - 1024 байта ОТР (одноразового программирования) для пользовательских данных.
 - Опциональные байты для настройки защиты от чтения и записи, уровня BOR, контроля программного/аппаратного обеспечения, базового адреса загрузочной памяти и сброса, когда устройство находится в режиме ожидания или остановки.

Обратитесь к Разделу 3: Встроенная флэш-память (FLASH) для получения более подробной информации.

2.5 Конфигурация загрузки

В STM32F75xxx и STM32F74xxx две разные загрузочные области могут быть выбраны через вывод BOOT и базовый адрес загрузки, запрограммированные в байтах опций BOOT_ADD0 и BOOT_ADD1, как показано в Таблице 2

Таблица 2. Режимы загрузки

Выбор режима загрузки		Загрузочная зона
BOOT	Байт параметра адреса загрузки	
0	BOOT_ADD0[15:0]	Загрузочный адрес определяется байтом пользовательской опции BOOT_ADD0 ST [15: 0:] запрограммированное ST значение Flash на ITCM в 0x0020 0000
1	BOOT_ADD1[15:0]	Загрузочный адрес определяется байтом пользовательской опции BOOT_ADD1 [15: 0:] запрограммированное ST значение системный загрузчик в 0x0010 0000

Значения на выводе BOOT фиксируются на 4-м переднем фронте SYSCLK после сброса сброса. Пользователь может установить контакт BOOT после сброса.

Контакт BOOT также пересэмплируется, когда устройство выходит из режима ожидания. Следовательно, они должны храниться в требуемой конфигурации режима загрузки, когда устройство находится в режиме ожидания.

После задержки запуска выбор области загрузки выполняется перед сбросом перезагрузки процессора.

Байты опций адреса BOOT_ADD0 и BOOT_ADD1 позволяют запрограммировать любой адрес загрузочной памяти от 0x0000 0000 до 0x3FFF FFFF, который включает в себя:

- Все адресное пространство Flash отображается на интерфейсе ITCM или AXIM.
- Все адресное пространство RAM: ITCM, DTCM RAM и SRAM, отображенные на интерфейсе AXIM
- Системный загрузчик памяти

Байты опции BOOT_ADD0 / BOOT_ADD1 могут быть изменены после сброса для загрузки с любого другого загрузочного адреса после следующего сброса.

Если запрограммированный адрес загрузочной памяти находится за пределами области отображения памяти или зарезервированной области, адрес загрузки по умолчанию запрограммирован следующим образом:

- Адрес загрузки 0: ITCM-FLASH в 0x0020 0000
- Адрес загрузки 1: ITCM-RAM на 0x0000 0000. Когда включена защита уровня 2 флэш-памяти, будет доступна только загрузка с флэш-памяти (на интерфейсе ITCM или AXIM) или системный загрузчик. Если уже запрограммированный загрузочный адрес в байтах опций BOOT_ADD0 и/или BOOT_ADD1 находится вне диапазона памяти или адреса ОЗУ (в ITCM или AXIM), выборка по умолчанию будет принудительно вызвана из Flash на интерфейсе ITCM по адресу 0x00200000.

Встроенный загрузчик

Код встроенного загрузчика находится в системной памяти. Программируется ST во время производства. Для получения полной информации см. Примечание по применению (AN2606) Режим загрузки системной памяти микроконтроллера STM32.

По умолчанию при выборе загрузки из системного загрузчика код выполняется из интерфейса TCM. Это может быть выполнено из интерфейса AXIM путем перепрограммирования байтов опций адреса BOOT_ADDx в 0x1FF0 0000.

3 Встроенная флэш-память (FLASH)

3.1 Введение

Интерфейс флэш-памяти управляет доступом Cortex®-M7 AXI и TCM к флэш-памяти. Он реализует операции стирания и программирования флэш-памяти, а также механизмы защиты от чтения и записи.

Интерфейс флэш-памяти ускоряет выполнение кода благодаря системе предварительной выборки команд и строкам кэша на интерфейсе ITCM (ART Accelerator™).

3.2 Основные функции Flash

- Операции чтения флэш-памяти
- Программа флэш-памяти/операции стирания
- Защита от чтения/записи
- 64 строки кэша по 256 бит на интерфейсе ITCM (ART Accelerator™)
- Предварительная выборка в коде команды TCM. На рисунке 3 показано подключение интерфейса флэш-памяти в архитектуре системы.

3.3 Описание функционала Flash

3.3.1 Организация флэш-памяти

Флэш-память имеет следующие основные функции:

- Емкость до 1 МБ на устройствах STM32F756xx и STM32F74xx и до 64 КБ на устройствах STM32F750xx
 - 256 бит считывания данных
 - Байт, половина слова, слово и двойное слово записи
 - сектор и массовое стирание

Флэш-память организована следующим образом:

- На устройствах STM32F756xx и STM32F74xx блок основной памяти разделен на 4 сектора по 32 КБ, 1 сектор по 128 КБ и 3 сектора по 256 КБ.

- На устройствах STM32F750xx блок основной памяти разделен на 2 сектора по 32 Кбайт.

- Информационные блоки, содержащие:

— Системная память, с которой устройство загружается в режиме загрузки системной памяти

— 1024 ОТР (одноразовое программирование) для пользовательских данных

— Область ОТР содержит 16 дополнительных байтов, используемых для блокировки соответствующего блока данных ОТР.

— Опциональные байты для настройки защиты от чтения и записи, уровня BOR, контроля программного/аппаратного обеспечения, базового адреса загрузочной памяти и сброса, когда устройство находится в режиме ожидания или остановки.

Встроенная флэш-память имеет три основных интерфейса:

- 64-битный интерфейс ITCM:
 - Он подключен к шине ITCM Cortex-M7 и используется для выполнения инструкций и доступа к чтению данных.
 - Этот доступ не поддерживается на интерфейсе ITCM
 - Поддерживает унифицированные 64 строки кэша по 256 бит (ускоритель ART) • 64-битный интерфейс АНВ:
 - Он подключен к шине AXI Cortex-M7 через матрицу шины АНВ и используется для выполнения кода, доступа для чтения и записи.
 - DMA и периферийные устройства Передача данных прямого доступа к памяти по Flash выполняется через интерфейс АНВ независимо от адресации интерфейса флэш-памяти TCM или АНВ.
 - 32-битный интерфейс регистра АНВ:
 - Используется для контроля и доступа к регистру состояния.

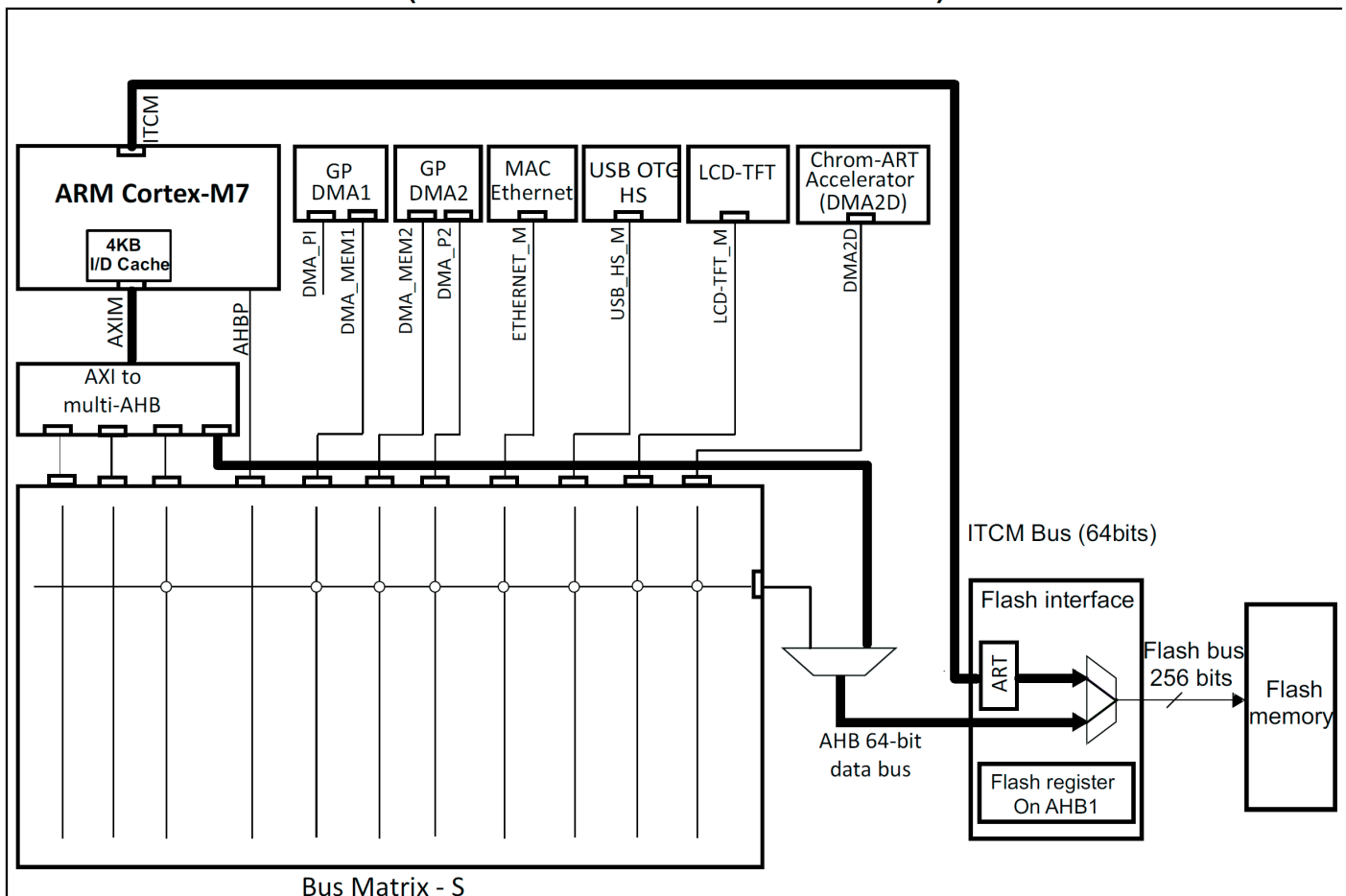
Организация основной памяти и информационных блоков показана в Табл. 3

3.3.2 Задержка доступа на чтение

Для правильного считывания данных из флэш-памяти количество состояний ожидания (LATENCY) должно быть правильно запрограммировано в регистре управления доступом к флэш-памяти (FLASH_ACR) в соответствии с частотой тактового сигнала процессора (HCLK) и напряжением питания устройства.

Соответствие между состояниями ожидания и тактовой частотой ЦП приведено в Таблице 4 и Таблице 5.

Figure 3. Flash memory interface connection inside system architecture (STM32F75xxx and STM32F74xxx)



Заметка:

- когда VOS [1: 0] = '0x01', максимальное значение fHCLK составляет 144 МГц.
 - когда VOS [1: 0] = '0x10', максимальное значение fHCLK составляет 168 МГц. Его можно расширить до 180 МГц, активировав режим over-drive.
 - когда VOS [1: 0] = '0x11', максимальное значение fHCLK составляет 180 МГц. Его можно расширить до 216 МГц, активировав режим овердрайва.
 - Режим Over-Drive недоступен, когда VDD находится в диапазоне от 1,8 до 2,1 В.
- Обратитесь к Разделу 4.1.4: Регулятор напряжения для получения подробной информации о том, как активировать режим повышенной скорости

После сброса тактовая частота процессора составляет 16 МГц, и в регистре FLASH_ACR настраивается 0 состояний ожидания (WS).

Настоятельно рекомендуется использовать следующие программные последовательности для настройки количества состояний ожидания для доступа к флэш-памяти с частотой процессора.

Увеличение частоты процессора

1. Запрограммируйте новое количество состояний ожидания в биты LATENCY в регистре FLASH_ACR.
2. Убедитесь, что новое число состояний ожидания учтено для доступа к флэш-памяти, прочитав регистр FLASH_ACR
3. Измените источник тактовой частоты процессора, записав биты SW в регистр RCC_CFGR.
4. При необходимости измените прескалер процессора, записав биты HPRE в RCC_CFGR.
5. Убедитесь, что новый источник тактовых импульсов ЦП и / и новое значение прескалерной частоты ЦП / учтены путем считывания состояния источника тактовой частоты (биты SWS) или / и значения прескалера АНВ (биты HPRE), соответственно, в RCC_CFGR зарегистрироваться.

Уменьшение частоты процессора

1. Модифицируйте источник тактовой частоты процессора, записав биты SW в регистр RCC_CFGR.
2. При необходимости измените прескалер процессора, записав биты HPRE в RCC_CFGR.
3. Убедитесь, что новый источник тактового сигнала CPU и / и новое значение прескалера тактового сигнала CPU / учтены путем считывания состояния источника тактового сигнала (биты SWS) или / и значения прескалера АНВ (биты HPRE), соответственно, в Регистр RCC_CFGR
4. Запрограммируйте новое количество состояний ожидания в биты LATENCY в FLASH_ACR.
5. Убедитесь, что новое число состояний ожидания используется для доступа к флэш-памяти, прочитав регистр FLASH_ACR

Заметка:

Изменение конфигурации тактовой частоты процессора или конфигурации состояния ожидания (WS) может быть неэффективным сразу. Чтобы убедиться, что текущая тактовая частота процессора является той, которую настроил пользователь, пользователь может проверить коэффициент предварительного масштаби-

Table 3. STM32F756xx and STM32F74xxx Flash memory organization

Block	Name	Bloc base address on AXIM interface	Block base address on ICTM interface	Sector size
Main memory block	Sector 0	0x0800 0000 - 0x0800 7FFF	0x0020 0000 - 0x0020 7FFF	32 Kbytes
	Sector 1	0x0800 8000 - 0x0800 FFFF	0x0020 8000 - 0x0020 FFFF	32 Kbytes
	Sector 2	0x0801 0000 - 0x0801 7FFF	0x0021 0000 - 0x0021 7FFF	32 Kbytes
	Sector 3	0x0801 8000 - 0x0801 FFFF	0x0021 8000 - 0x0021 FFFF	32 Kbytes
	Sector 4	0x0802 0000 - 0x0803 FFFF	0x0022 0000 - 0x0023 FFFF	128 Kbytes
	Sector 5	0x0804 0000 - 0x0807 FFFF	0x0024 0000 - 0x0027 FFFF	256 Kbytes
	Sector 6	0x0808 0000 - 0x080B FFFF	0x0028 0000 - 0x002B FFFF	256 Kbytes
	Sector 7	0x080C 0000 - 0x080F FFFF	0x002C 0000 - 0x02F FFFF	256 Kbytes
Information block	System memory	0x1FF0 0000 - 0x1FF0 EDBF	0x0010 0000 - 0x0010 EDBF	60 Kbytes
	OTP	0x1FF0 F000 - 0x1FF0 F41F	0x0010 F000 - 0x0010 F41F	1024 bytes
	Option bytes	0x1FFF 0000 - 0x1FFF 001F	-	32 bytes

Table 4. STM32F750xx Flash memory organization

Block	Name	Bloc base address on AXIM interface	Block base address on ICTM interface	Sector size
Main memory block	Sector 0	0x0800 0000 - 0x0800 7FFF	0x0020 0000 - 0x0020 7FFF	32 Kbytes
	Sector 1	0x0800 8000 - 0x0800 FFFF	0x0020 8000 - 0x0020 FFFF	32 Kbytes
Information block	System memory	0x1FF0 0000 - 0x1FF0 EDBF	0x0010 0000 - 0x0010 EDBF	60 Kbytes
	OTP	0x1FF0 F000 - 0x1FF0 F41F	0x0010 F000 - 0x0010 F41F	1024 bytes
	Option bytes	0x1FFF 0000 - 0x1FFF 001F	-	32 bytes

Table 5. Number of wait states according to CPU clock (HCLK) frequency

Wait states (WS) (LATENCY)	HCLK (MHz)			
	Voltage range 2.7 V - 3.6 V	Voltage range 2.4 V - 2.7 V	Voltage range 2.1 V - 2.4 V	Voltage range 1.8 V - 2.1 V
0 WS (1 CPU cycle)	0 < HCLK ≤ 30	0 < HCLK ≤ 24	0 < HCLK ≤ 22	0 < HCLK ≤ 20
1 WS (2 CPU cycles)	30 < HCLK ≤ 60	24 < HCLK ≤ 48	22 < HCLK ≤ 44	20 < HCLK ≤ 40
2 WS (3 CPU cycles)	60 < HCLK ≤ 90	48 < HCLK ≤ 72	44 < HCLK ≤ 66	40 < HCLK ≤ 60
3 WS (4 CPU cycles)	90 < HCLK ≤ 120	72 < HCLK ≤ 96	66 < HCLK ≤ 88	60 < HCLK ≤ 80
4 WS (5 CPU cycles)	120 < HCLK ≤ 150	96 < HCLK ≤ 120	88 < HCLK ≤ 110	80 < HCLK ≤ 100
5 WS (6 CPU cycles)	150 < HCLK ≤ 180	120 < HCLK ≤ 144	110 < HCLK ≤ 132	100 < HCLK ≤ 120
6 WS (7 CPU cycles)	180 < HCLK ≤ 210	144 < HCLK ≤ 168	132 < HCLK ≤ 154	120 < HCLK ≤ 140
7 WS (8 CPU cycles)	210 < HCLK ≤ 216	168 < HCLK ≤ 192	154 < HCLK ≤ 176	140 < HCLK ≤ 160
8 WS (9 CPU cycles)	-	192 < HCLK ≤ 216	176 < HCLK ≤ 198	160 < HCLK ≤ 180
9 WS (10 CPU cycles)	-	-	198 < HCLK ≤ 216	-

рования АНВ и значения состояния источника синхронизации. Чтобы убедиться, что число запрограммированных WS является эффективным, пользователь может прочитать регистр FLASH_ACR.

Instruction prefetch (предварительная загрузка инструкций)

Каждая операция чтения с флэш-памяти обеспечивает 256 битов, представляющих 8 инструкций по 32 бита по 16 инструкций по 16 бит в соответствии с запущенной программой. Таким образом, в случае последовательного кода, по крайней мере, 8 циклов ЦП необходимы для выполнения предыдущего чтения строки инструкции. Предварительная выборка на шине ITCM позволяет читать последовательную следующую строку инструкций во флэш-памяти, пока текущая строка инструкций запрашивается ЦПУ. Предварительную выборку можно включить, установив бит PRFTEN в регистре FLASH_ACR. Эта функция полезна, если для доступа к флэш-памяти требуется хотя бы одно состояние ожидания. Когда код не является последовательным (ветвящимся), команда может отсутствовать ни в текущей используемой строке команд, ни в строке предварительно выбранных команд. В этом случае (мисс) штраф в терминах количества циклов, по крайней мере, равен числу состояний ожидания.

Adaptive real-time memory accelerator (ART Accelerator™) (адаптивный ускоритель памяти в реальном времени (ART Accelerator™))

Фирменный ускоритель памяти Adaptive Real-Time (ART) оптимизирован для промышленного стандарта STM32 Arm® Cortex®-M7 с процессорами FPU. Он уравнивает преимущество в производительности Arm® Cortex®-M7 с FPU по сравнению с технологиями флэш-памяти, что обычно требует от процессора ожидания флэш-памяти на более высоких рабочих частотах.

Чтобы обеспечить полную производительность процессора, ускоритель реализует унифицированный кэш команд и кэш веток, что увеличивает скорость выполнения программы из 256-битной флэш-памяти. Исходя из теста CoreMark, производительность, достигнутая благодаря ускорителю ART, эквивалентна нулю выполнения программы состояния ожидания из флэш-памяти на частоте процессора до 216 МГц.

Ускоритель ART доступен только для флэш-доступа через интерфейс ITCM.

Чтобы ограничить время, потерянное из-за скачков, в ускорителе ART можно сохранить 64 строки по 256 бит. Эту функцию можно включить, установив бит ARTEN в регистре FLASH_CR. ART Accelerator унифицирован, содержит инструкции, а также пулы литералов данных. Каждый раз, когда происходит пропадание (запрошенные данные отсутствуют в текущей используемой строке данных или в кэш-памяти команд), строка чтения копируется в кэш-память команд ART. Если ЦПУ запрашивает данные, содержащиеся в кэш-памяти команд, данные предоставляются без вставки задержки. Как только все строки кэш-памяти заполнены, политика LRU (наименьшее недавно использованное) используется для определения строки, подлежащей замене в кэш-памяти. Эта функция особенно полезна в случае кода, содержащего циклы.

Заметка:

Данные в секторе конфигурации пользователя не кэшируются.