

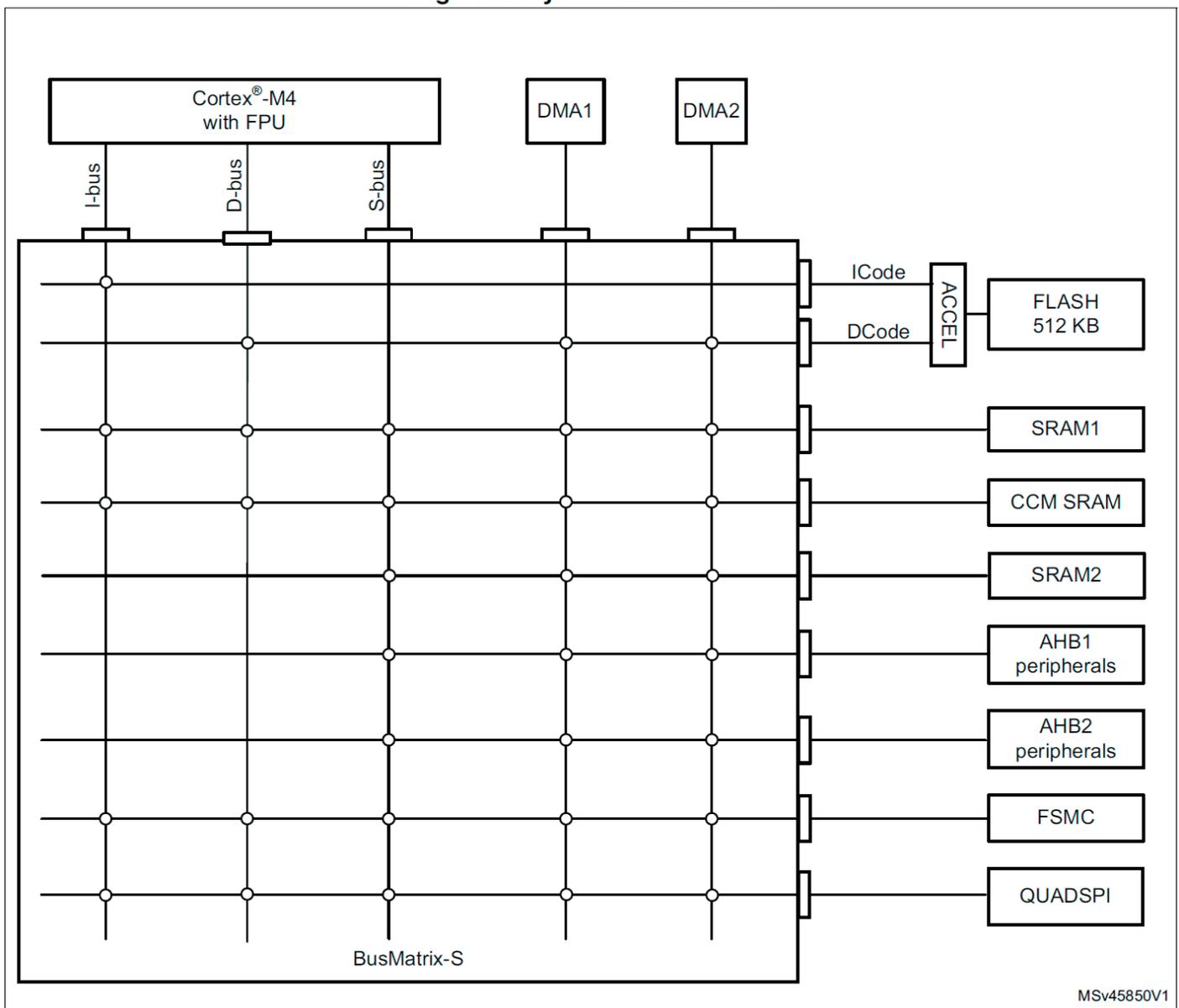
2 Обзор системы и памяти

2.1 Архитектура системы

Основная система состоит из 32-битной многослойной матрицы шины АНВ, которая соединяет между собой:

- До пяти мастеров:
 - Cortex®-M4 с ядром FPU I-bus
 - Cortex®-M4 с ядром FPU D-bus
 - Cortex®-M4 с ядром FPU S-bus
 - DMA1
 - DMA2
- До девяти ведомых устройств:
 - Внутренняя флэш-память на шине ICode
 - Внутренняя флэш-память на шине DCode
 - Внутренняя SRAM1
 - Внутренняя SRAM2
 - Внутренняя CCM SRAM
 - Периферийные устройства АНВ1, включая мосты АНВ-АРВ и периферийные устройства АРВ (подключенные к АРВ1 и АРВ2)

Figure 1. System architecture



- Периферийные устройства АHB2
- Гибкий контроллер статической памяти (FSMC)
- Интерфейс памяти QUAD SPI (QUADSPI)

Матрица шины обеспечивает доступ от ведущего к ведомому, обеспечивая одновременный доступ и эффективную работу, даже когда несколько высокоскоростных периферийных устройств работают одновременно. Эта архитектура показана на рисунке 1:

2.1.1 I-шина

Эта шина соединяет шину команд Cortex®-M4 с ядром FPU с BusMatrix.

Эта шина используется ядром для получения инструкций. Целью этой шины является память, содержащая код (внутреннюю флэш-память, внутреннюю SRAM или внешнюю память через FSMC или QUADSPI).

2.1.2 D-шина

Эта шина соединяет шину данных Cortex®-M4 с ядром FPU с BusMatrix. Эта шина используется ядром для буквальной загрузки и отладки. Целью этой шины является память, содержащая код (внутренняя флэш-память, внутренняя SRAM или внешняя память через FSMC или QUADSPI).

2.1.3 S-шина

Эта шина соединяет системную шину Cortex®-M4 с ядром FPU с BusMatrix. Эта шина используется ядром для доступа к данным, расположенным в периферийной области или области SRAM. Целями этой шины являются внутренняя SRAM, периферийные устройства АHB1, включая периферийные устройства APB1 и APB2, периферийные устройства АHB2 и внешняя память через QUADSPI или FSMC.

CCM SRAM также доступна на этой шине, чтобы обеспечить непрерывное сопоставление с SRAM1 и SRAM2.

2.1.4 DMA-шина

Эта шина соединяет главный интерфейс АHB DMA с BusMatrix. Целями этой шины являются SRAM1, SRAM2 и CCM SRAM, периферийные устройства АHB1, включая периферийные устройства APB1 и APB2, периферийные устройства АHB2 и внешние запоминающие устройства через QUADSPI или FSMC.

2.1.5 Матрица шины

BusMatrix управляет арбитражем доступа между мастерами. Арбитраж использует алгоритм Round Robin. BusMatrix состоит из пяти ведущих устройств (CPU АHB, системная шина, шина DCode, шина ICode, DMA1, DMA2,) и до девяти подчиненных устройств (FLASH, SRAM1, SRAM2, CCM SRAM, АHB1 (включая APB1 и APB2), АHB2, QUADSPI и FSMC).

Мосты АHB/APB

Два моста АHB/APB обеспечивают полное синхронное соединение между АHB и двумя шинами APB, позволяя гибко выбирать периферийную частоту.

Обратитесь к Разделу 2.2.2: Карта памяти и адреса регистров на странице 82 для отображения адресов периферийных устройств, подключенных к этому мосту.

После каждого сброса устройства все периферийные часы отключаются (кроме SRAM1/2 и интерфейса флэш-памяти). Перед использованием периферийного устройства вы должны включить его часы в регистрах RCC_AHBxENR и RCC_APBxENR.

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

2.2 Организация памяти

2.2.1 Введение

Память программ, память данных, регистры и порты ввода-вывода организованы в одном и том же линейном адресном пространстве размером 4 Гбайт.

Байты кодируются в памяти в формате Little Endian. Байт с наименьшим номером в слове считается младшим значащим байтом слова, а байт с наибольшим номером — старшим.

Адресуемое пространство памяти разделено на восемь основных блоков по 512 Мбайт каждый.

2.2.2 Карта памяти и граничные адреса регистров

Все области карты памяти, которые не выделены для встроенной памяти и периферийных устройств, считаются «зарезервированными». Подробное отображение доступных областей памяти и регистров см. в следующей таблице.

В таблице 3 приведены граничные адреса периферийных устройств, доступных в устройствах.

*1.4 Определение категории продукта

В таблице 1 представлен обзор плотности памяти по сравнению с линейкой продуктов.

Настоящее справочное руководство описывает расширенный набор функций для каждой категории продуктов.

См. Таблицу 2 для списка функций по категориям.

Table 1. STM32G4 Series memory density

Memory density	Category 2	Category 3	Category 4
128 Kbytes	STM32G431 STM32G441 (AES)	STM32G471 STM32G473 STM32G474	-
256 Kbytes	-	STM32G471 STM32G473 STM32G474	-
512 Kbytes	-	STM32G471 STM32G473 STM32G474 STM32G483 (AES) STM32G484 (AES)	STM32G491 STM32G4A1 (AES)

Figure 2. Memory map

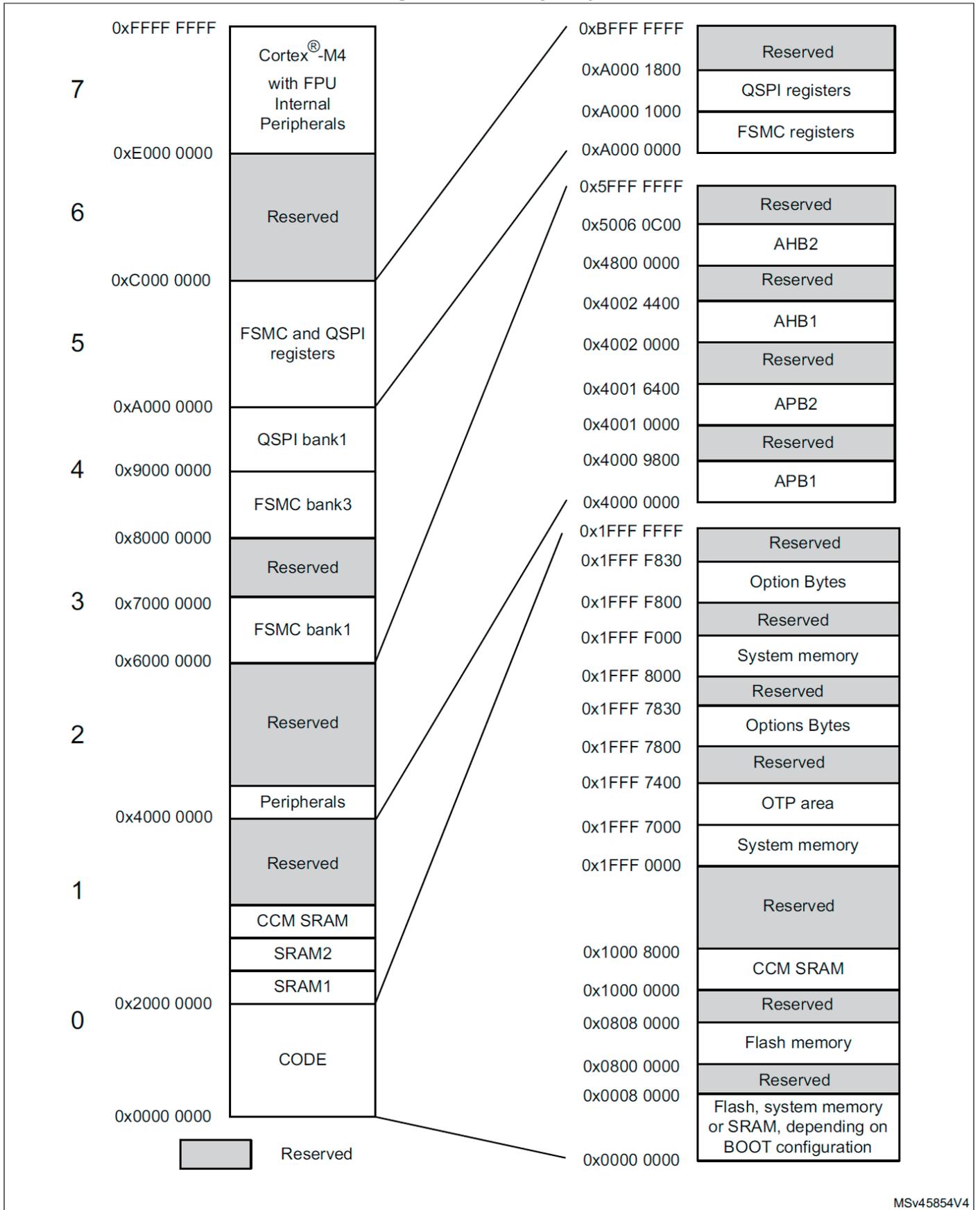


Таблица 3. Карта памяти серии STM32G4 и граничные адреса периферийных регистров ⁽¹⁾

Bus	Boundary address	Size (bytes)	Peripheral	Peripheral register map
-	0xA000 1400 - 0xAFFF FFFF	262 MB	Reserved	-
	0xA000 1000 - 0xA000 13FF	1 KB	QUADSPI control Registers	Section 20.5.14: QUADSPI register map
	0xA000 0400 - 0xA000 0FFF	3 KB	Reserved	-
	0xA000 0000 - 0xA000 03FF	1 KB	FSMC	Section 19.7.8: FMC register map
AHB2	0x5006 0C00 - 0x5FFF FFFF	256MB	Reserved	-
	0x5006 0800 - 0x5006 0BFF	1 KB	RNG	Section 26.7.4: RNG register map
	0x5006 0400 - 0x5006 07FF	1 KB	Reserved	-
	0x5006 0000 - 0x5006 03FF	1 KB	AES	Section 34.7.18: AES register map
	0x5000 1800 - 0x5005 FFFF	377 KB	Reserved	-
	0x5000 1400 - 0x5000 17FF	1 KB	DAC4	Section 22.7.24: DAC register map
	0x5000 1000 - 0x5000 13FF	1 KB	DAC3	Section 22.7.24: DAC register map
	0x5000 0C00 - 0x5000 0FFF	1 KB	DAC2	Section 22.7.24: DAC register map
	0x5000 0800 - 0x5000 0BFF	1 KB	DAC1	Section 22.7.24: DAC register map
	0x5000 0400 - 0x5000 07FF	1 KB	ADC3 - ADC4 - ADC5	Section 21.9: ADC register map
	0x5000 0000 - 0x5000 03FF	1 KB	ADC1 - ADC2	Section 21.9: ADC register map
	0x4800 1C00 - 0x4FFF FFFF	127 MB	Reserved	-
	0x4800 1800 - 0x4800 1BFF	1 KB	GPIOG	Section 9.4.12: GPIO register map
	0x4800 1400 - 0x4800 17FF	1 KB	GPIOF	Section 9.4.12: GPIO register map
	0x4800 1000 - 0x4800 13FF	1 KB	GPIOE	Section 9.4.12: GPIO register map
	0x4800 0C00 - 0x4800 0FFF	1 KB	GPIOD	Section 9.4.12: GPIO register map
	0x4800 0800 - 0x4800 0BFF	1 KB	GPIOC	Section 9.4.12: GPIO register map
	0x4800 0400 - 0x4800 07FF	1 KB	GPIOB	Section 9.4.12: GPIO register map
	0x4800 0000 - 0x4800 03FF	1 KB	GPIOA	Section 9.4.12: GPIO register map
	0x4002 3400 - 0x47FF FFFF	127 MB	Reserved	-
	0x4002 3000 - 0x4002 33FF	1 KB	CRC	Section 16.4.6: CRC register map
	0x4002 2400 - 0x4002 2FFF	3 KB	Reserved	-
	0x4002 2000 - 0x4002 23FF	1 KB	Flash interface	Section 5.7.14: FLASH register map
0x4002 1400 - 0x4002 1FFF	3 KB	FMAC	Section 18.4.9: FMAC register map	
0x4002 1000 - 0x4002 13FF	1 KB	RCC	Section 7.4.31: RCC register map	
0x4002 0C00 - 0x4002 0FFF	1 KB	CORDIC	Section 17.4.4: CORDIC register map	
0x4002 0800 - 0x4002 0BFF	1 KB	DMAMUX	Section 13.6.7: DMAMUX register map	
0x4002 0400 - 0x4002 07FF	1 KB	DMA 2	Section 12.6.7: DMA register map	
0x4002 0000 - 0x4002 03FF	1 KB	DMA 1	Section 12.6.7: DMA register map	
AHB1	0x4001 7800 - 0x4001 FFFF	2 KB	Reserved	-
	0x4001 6800 - 0x4001 77FF	3 KB	HRTIM	Section 27.5.82: HRTIM register map
	0x4001 5800 - 0x4001 67FF	4 KB	Reserved	-

APB2	0x4001 5400 - 0x4001 57FF	1 KB	SAI1	Section 40.6.19: SAI register map
	0x4001 5000 - 0x4001 53FF	1 KB	TIM20	Section 28.6.31: TIMx register map
	0x4001 4C00 - 0x4001 4FFF	1 KB	Reserved	-
	0x4001 4800 - 0x4001 4BFF	1 KB	TIM17	Section 30.8.22: TIM16/TIM17 register map
	0x4001 4400 - 0x4001 47FF	1 KB	TIM16	Section 30.8.22: TIM16/TIM17 register map
	0x4001 4000 - 0x4001 43FF	1 KB	TIM15	Section 30.7.23: TIM15 register map
	0x4001 3C00 - 0x4001 3FFF	1 KB	SPI4	Section 39.9.10: SPI/I2S register map
	0x4001 3800 - 0x4001 3BFF	1 KB	USART1	Section 37.8.15: USART register map
	0x4001 3400 - 0x4001 37FF	1 KB	TIM8	Section 28.6.31: TIMx register map
	0x4001 3000 - 0x4001 33FF	1 KB	SPI1	Section 39.9.10: SPI/I2S register map
	0x4001 2C00 - 0x4001 2FFF	1 KB	TIM1	Section 28.6.31: TIMx register map
	0x4001 0800 - 0x4001 2BFF	9 KB	Reserved	-
	0x4001 0400 - 0x4001 07FF	1 KB	EXTI	Section 15.5.13: EXTI register map
	0x4001 0300 - 0x4001 03FF	1 KB	OPAMP	Section 25.5.13: OPAMP register map
	0x4001 0200 - 0x4001 02FF		COMP	Section 24.6.2: COMP register map
	0x4001 0030 - 0x4001 01FF		VREFBUF	Section 23.4.3: VREFBUF register map
	0x4001 0000 - 0x4001 0029		SYSCFG	Section 10.2.11: SYSCFG register map
	0x4000 AF00 - 0x4000 FFFF	23 KB	Reserved	-
	0x4000 AC00 - 0x4000 AFFF	1 KB	FDCANs Message RAM	Section 44.4.38: FDCAN register map
	0x4000 A800 - 0x4000 ABFF	1 KB		
	0x4000 A400 - 0x4000 A7FF	1 KB		
	0x4000 A000 - 0x4000 A3FF	1 KB	UCPD1	Section 46.7.15: UCPD register map
	0x4000 8800 - 0x4000 9FFF	6 KB	Reserved	-
0x4000 8400 - 0x4000 87FF	1 KB	I2C4	Section 41.7.12: I2C register map	
0x4000 8000 - 0x4000 83FF	1 KB	LPUART1	Section 38.7.13: LPUART register map	
0x4000 7C00 - 0x4000 7FFF	1 KB	LPTIM1	Section 32.7.10: LPTIM register map	
0x4000 7800 - 0x4000 7BFF	1 KB	I2C3	Section 41.7.12: I2C register map	
0x4000 7400 - 0x4000 77FF	1 KB	Reserved	-	
APB1	0x4000 7000 - 0x4000 73FF	1 KB	PWR	Section 6.4.23: PWR register map and reset value table
	0x4000 6C00 - 0x4000 6FFF	1 KB	FDCAN3	Section 44.4.38: FDCAN register map
	0x4000 6800 - 0x4000 6BFF	1 KB	FDCAN2	Section 44.4.38: FDCAN register map
	0x4000 6400 - 0x4000 67FF	1 KB	FDCAN1	Section 44.4.38: FDCAN register map
	0x4000 6000 - 0x4000 63FF	1 KB	USB SRAM 1 Kbyte	-
	0x4000 5C00 - 0x4000 5FFF	1 KB	USB device FS	Section 45.6.3: USB register map
	0x4000 5800 - 0x4000 5BFF	1 KB	I2C2	Section 41.7.12: I2C register map
	0x4000 5400 - 0x4000 57FF	1 KB	I2C1	Section 41.7.12: I2C register map
	0x4000 5000 - 0x4000 53FF	1 KB	UART5	Section 37.8.15: USART register map
	0x4000 4C00 - 0x4000 4FFF	1 KB	UART4	Section 37.8.15: USART register map
	0x4000 4800 - 0x4000 4BFF	1 KB	USART3	Section 37.8.15: USART register map

	0x4000 4400 - 0x4000 47FF	1 KB	USART2	Section 37.8.15: USART register map
APB1 Cont.	0x4000 4000 - 0x4000 43FF	1 KB	Reserved	-
	0x4000 3C00 - 0x4000 3FFF	1 KB	SPI3/I2S3	Section 39.9.10: SPI/I2S register map
	0x4000 3800 - 0x4000 3BFF	1 KB	SPI2/I2S2	Section 39.9.10: SPI/I2S register map
	0x4000 3400 - 0x4000 37FF	1 KB	Reserved	-
	0x4000 3000 - 0x4000 33FF	1 KB	IWDG	Section 42.4.6: IWDG register map
	0x4000 2C00 - 0x4000 2FFF	1 KB	WWDG	Section 43.5.4: WWDG register map
	0x4000 2800 - 0x4000 2BFF	1 KB	RTC & BKP Registers	Section 35.6.21: RTC register map
	0x4000 2400 - 0x4000 27FF	1 KB	TAMP	Section 36.6.9: TAMP register map
	0x4000 2000 - 0x4000 23FF	1 KB	CRS	Section 8.7.5: CRS register map
	0x4000 1C00 - 0x4000 1FFF	1 KB	Reserved	-
	0x4000 1800 - 0x4000 1BFF	1 KB	Reserved	-
	0x4000 1400 - 0x4000 17FF	1 KB	TIM7	Section 29.5.31: TIMx register map Section 31.4.9: TIMx register map
	0x4000 1000 - 0x4000 13FF	1 KB	TIM6	Section 31.4.9: TIMx register map
	0x4000 0C00 - 0x4000 0FFF	1 KB	TIM5	Section 29.5.31: TIMx register map
	0x4000 0800 - 0x4000 0BFF	1 KB	TIM4	Section 29.5.31: TIMx register map
	0x4000 0400 - 0x4000 07FF	1 KB	TIM3	Section 29.5.31: TIMx register map
0x4000 0000 - 0x4000 03FF	1 KB	TIM2	Section 29.5.31: TIMx register map	

¹ См. Таблицу 1: Плотность памяти серии STM32G4, Таблицу 2: Специфические функции продукта и спецификации устройства для портов GPIO и периферийных устройств, доступных на вашем устройстве. область памяти, соответствующая недоступным портам GPIO или периферийным устройствам, зарезервирована (выделена серым цветом).

2.3 Битовая полоса

Карта памяти Cortex®-M4 с FPU включает две области битового диапазона. Эти области сопоставляют каждое слово в псевдонимной области памяти с битом в битовой области памяти. Запись слова в области псевдонима имеет тот же эффект, что и операция чтения-изменения-записи для целевого бита в области битового диапазона.

В устройствах серии STM32G4 и периферийные регистры, и SRAM отображаются в область битового диапазона, поэтому разрешены операции записи и чтения в одном битовом диапазоне. Операции доступны только для Cortex® -M4 с доступом к FPU, а не от других мастеров шины (например, DMA).

Формула отображения показывает, как сослаться на каждое слово в области псевдонима на соответствующий бит в области битового диапазона. Формула отображения:

$$\text{bit_word_addr} = \text{bit_band_base} + (\text{byte_offset} \times 32) + (\text{bit_number} \times 4)$$

где:

– bit_word_addr – это адрес слова в псевдонимной области памяти, который отображается на целевой бит

- `bit_band_base` – начальный адрес области псевдонима
- `byte_offset` – это номер байта в области битового диапазона, который содержит целевой бит
- `bit_number` – это битовая позиция (0-7) целевого бита

Пример

В следующем примере показано, как отобразить бит 2 байта, расположенного по адресу SRAM1.

0x20000300 в область псевдонима:

$0x22006008 = 0x22000000 + (0x300 * 32) + (2 * 4)$

Запись по адресу 0x22006008 имеет тот же эффект, что и операция чтения-изменения-записи для бита 2 байта по адресу SRAM1 0x20000300.

Чтение адреса 0x22006008 возвращает значение (0x01 или 0x00) бита 2 байта по адресу SRAM1 0x20000300 (0x01: бит установлен; 0x00: бит сброшен)

2.4 Встроенная статическая память

Устройства серии STM32G4 категории 3 имеют до 128 Кбайт SRAM:

- 80 Кбайт SRAM1 (отображается по адресу 0x2000 0000)
- 16 Кбайт SRAM2 (отображается по адресу 0x2001 4000)
- 32 Кбайт CCM SRAM (сопоставляется с адресом 0x1000 0000 и концом SRAM2)

Устройства серии STM32G4 категории 4 имеют до 112 Кбайт SRAM:

- 80 Кбайт SRAM1 (отображается по адресу 0x2000 0000)
- 16 Кбайт SRAM2 (отображается по адресу 0x2001 4000)
- 16 Кбайт CCM SRAM (сопоставляется с адресом 0x1000 0000 и концом SRAM2)

Устройства серии STM32G4 категории 2 имеют до 32 Кбайт SRAM:

- 16 Кбайт SRAM1 (отображается по адресу 0x2000 0000)
- 6 Кбайт SRAM2 (отображается по адресу 0x2000 4000)
- 10 Кбайт CCM SRAM (сопоставляется с адресом 0x1000 0000 и концом SRAM2)

К этим SRAM можно обращаться как к байтам, полусловам (16 бит) или целым словам (32 бита). Эти памяти могут быть адресованы на максимальной тактовой частоте системы без состояния ожидания и, таким образом, как ЦП, так и DMA.

ЦП может получить доступ к SRAM1 через системную шину или через шины ICode/DCode, когда выбрана загрузка из SRAM1 или когда выбрано физическое переназначение (Раздел 10.2.1: Регистр переназначения памяти SYSCFG (SYSCFG_MEMRMP) в контроллере SYSCFG). Чтобы получить максимальную производительность при выполнении SRAM1, следует выбрать физическое переназначение (выбор загрузки или программного обеспечения).

CCM SRAM отображается по адресу 0x1000 0000.

Выполнение может быть выполнено из CCM SRAM с максимальной производительностью без переназначения благодаря доступу через шину ICode.

CCM SRAM имеет псевдоним по адресу, следующему за концом SRAM2 (0x2000 5800 для устройств категории 2, 0x2001 8000 для устройств категории 3, 0x2001 8000 для устройств категории 4), предлагая непрерывное адресное пространство с SRAM1 и SRAM2.

2.4.1 Проверка четности

На устройствах категорий 3 и 4 проверка четности выполняется для первых 32 Кбайт SRAM1 и для всего CCM SRAM.

На устройствах категории 2 проверка четности реализована для всего SRAM1 и CCM SRAM.

Пользователь может включить проверку четности, используя бит опции SRAM_PE в байте опции пользователя (см. Раздел 3.4.1: Описание байтов опции).

Ширина шины данных составляет 36 бит, потому что 4 бита доступны для проверки на четность (1 бит на байт) для повышения надежности памяти, как того требуют, например, нормы класса B или SIL.

Биты четности вычисляются и сохраняются при записи в SRAM. Затем они автоматически проверяются при чтении. Если один бит терпит неудачу, генерируется NMI. Эта же ошибка также может быть связана с входом BRK_IN Break TIM1/TIM8/TIM15/TIM16/TIM17/TIM20 и с hrtim_sys_flt с управляющим битом SPL в Разделе 10.2.8: Регистр конфигурации SYSCFG 2 (SYSCFG_CFGR2). Флаг ошибки четности SRAM (SPF) доступен в Разделе 10.2.8: Регистр конфигурации 2 SYSCFG (SYSCFG_CFGR2).

***Примечание.** При включении проверки четности SRAM рекомендуется программно инициализировать всю память SRAM в начале кода, чтобы избежать ошибок четности при чтении неинициализированных ячеек.*

2.4.2 CCM SRAM Защита от записи

CCM SRAM может быть защищен от записи со степенью детализации страниц 1 Кбайт.

Table 4. CCM SRAM organization

Page number	Start address	End address
Page 0	0x1000 0000	0x1000 03FF
Page 1	0x1000 0400	0x1000 07FF
Page 2	0x1000 0800	0x1000 0BFF
Page 3	0x1000 0C00	0x1000 0FFF
Page 4	0x1000 1000	0x1000 13FF
Page 5	0x1000 1400	0x1000 17FF
Page 6	0x1000 1800	0x1000 1BFF
Page 7	0x1000 1C00	0x1000 1FFF
Page 8	0x1000 2000	0x1000 23FF
Page 9	0x1000 2400	0x1000 27FF
Page 10 ⁽¹⁾	0x1000 2800	0x1000 2BFF
Page 11 ⁽¹⁾	0x1000 2C00	0x1000 2FFF
Page 12 ⁽¹⁾	0x1000 3000	0x1000 33FF
Page 13 ⁽¹⁾	0x1000 3400	0x1000 37FF
Page 14 ⁽¹⁾	0x1000 3800	0x1000 3BFF

Page 15 ⁽¹⁾	0x1000 3C00	0x1000 3FFF
Page 16 ⁽²⁾	0x1000 4000	0x1000 43FF
Page 17 ⁽²⁾	0x1000 4400	0x1000 47FF
Page 18 ⁽²⁾	0x1000 4800	0x1000 4BFF
Page 19 ⁽²⁾	0x1000 4C00	0x1000 4FFF
Page 20 ⁽²⁾	0x1000 5000	0x1000 53FF
Page 21 ⁽²⁾	0x1000 5400	0x1000 57FF
Page 22 ⁽²⁾	0x1000 5800	0x1000 5BFF
Page 23 ⁽²⁾	0x1000 5C00	0x1000 5FFF
Page 24 ⁽²⁾	0x1000 6000	0x1000 63FF
Page 25 ⁽²⁾	0x1000 6400	0x1000 67FF
Page 26 ⁽²⁾	0x1000 6800	0x1000 6BFF
Page 27 ⁽²⁾	0x1000 6C00	0x1000 6FFF
Page 28 ⁽²⁾	0x1000 7000	0x1000 73FF
Page 29 ⁽²⁾	0x1000 7400	0x1000 77FF
Page 30 ⁽²⁾	0x1000 7800	0x1000 7BFF
Page 31 ⁽²⁾	0x1000 7C00	0x1000 7FFF

^{1.} Доступно только на устройствах категорий 3 и 4.

^{2.} Доступно только на устройствах категории 3.

Защиту от записи можно включить в Разделе 10.2.9: Регистр защиты от записи SYSCFG CCM SRAM (SYSCFG_SWPR) в блоке SYSCFG. Это регистр с механизмом однократной записи «1», что означает, что запись «1» в бит устанавливает защиту от записи для этой страницы SRAM, и ее можно удалить/очистить только путем перезагрузки системы.

2.4.3 Защита от чтения CCM SRAM

CCMSRAM защищен защитой от чтения (RDP). Дополнительные сведения см. в разделе 3.5.1: Защита от чтения (RDP).

2.4.4 Стирание CCM SRAM

CCMSRAM можно стереть с помощью сброса системы, используя бит опции CCMSRAM_RST в байте опции пользователя (см. Раздел 3.4.1: Описание байтов опции).

Стирание CCM SRAM также может быть запрошено программным обеспечением путем установки бита CCMSR в разделе 10.2.7: SYSCFG CCM SRAM, управляющий регистром и регистром состояния (SYSCFG_SCSR).

2.5 Обзор флэш-памяти

Флэш-память состоит из двух отдельных физических областей:

- Основной блок флэш-памяти. Он содержит прикладную программу и пользовательские данные, если это необходимо.
- Информационный блок. Он состоит из трех частей:
 - Байты опций для аппаратной защиты и пользовательской конфигурации защиты памяти.
 - Системная память, содержащая собственный код ST.
 - OTP (однократно программируемая) область

Интерфейс Flash реализует доступ к инструкциям и доступ к данным на основе протокола АНВ. Он также реализует логику, необходимую для выполнения операций с флэш-памятью (программирование/стирание), управляемых через регистры флэш-памяти. Дополнительные сведения см. в разделе 3: Встроенная флэш-память (FLASH) для устройств категории 3, в разделе 4: Встроенная флэш-память (FLASH) для устройств категории 4 и в разделе 5: Встроенная флэш-память (FLASH) для устройств категории 2.

2.6 Конфигурация загрузки

2.6.1 Конфигурация загрузки

Можно выбрать три разных режима загрузки через вывод BOOT0 или бит nBOOT0 в регистре FLASH_OPTR (если бит nSWBOOT0 очищен в регистре FLASH_OPTR) и бит nBOOT1 в регистре FLASH_OPTR, как показано в следующей таблице.

Table 5. Boot modes

BOOT_LOCK	nBOOT1 FLASH_OPTR[23]	nBOOT0 FLASH_OPTR[27]	BOOT0 pin PB8	nSWBOOT0 FLASH_OPTR[26]	Boot Memory Space Alias
1	X	X	X	X	Main Flash memory
0	X	X	0	1	Main Flash memory is selected as boot area
0	X	1	X	0	Main Flash memory is selected as boot area
0	0	X	1	1	Embedded SRAM1 is selected as boot area
0	0	0	X	0	Embedded SRAM1 is selected as boot area
0	1	X	1	1	System memory is selected as boot area
0	1	0	X	0	System memory is selected as boot area

Значения как на выводе BOOT0 (поступающие от вывода или бита опции), так и на бите nBOOT1 фиксируются на 4-м фронте внутреннего источника синхронизации запуска после сброса сброса. Пользователь может установить nBOOT1 и BOOT0 для выбора требуемого режима загрузки.

Вывод BOOT0 или бит пользовательской опции (в зависимости от значения бита nSWBOOT0 в регистре FLASH_OPTR) и бит nBOOT1 также пересчитываются при выходе из режима ожидания. Следовательно, они должны быть сохранены в тре-

буемой конфигурации режима загрузки в режиме ожидания. По истечении этой задержки запуска ЦП извлекает значение вершины стека по адресу 0x0000 0000, а затем начинает выполнение кода из загрузочной памяти по адресу 0x0000 0004.

В зависимости от выбранного режима загрузки основная флэш-память, системная память или SRAM1 доступны следующим образом:

- Загрузка из основной флэш-памяти: основная флэш-память имеет псевдоним в области загрузочной памяти (0x0000 0000), но по-прежнему доступна из исходной области памяти (0x0800 0000). Другими словами, доступ к содержимому флэш-памяти возможен, начиная с адреса 0x0000 0000 или 0x0800 0000.
- Загрузка из системной памяти: системная память имеет псевдоним в области загрузочной памяти (0x0000 0000), но по-прежнему доступна из исходной области памяти (0x1FFF 0000).
- Загрузка со встроенной SRAM1: SRAM1 имеет псевдоним в области загрузочной памяти (0x0000 0000), но по-прежнему доступен из исходной области памяти (0x2000 0000).

PB8/BOOT0 GPIO настроен в:

- Режим ввода во время фазы полного сброса, если опциональный бит nSWBOOT0 установлен в регистре FLASH_OPTR, а затем автоматически переключается в аналоговый режим после сброса (вывод BOOT0).
- Режим ввода от фазы сброса до завершения загрузки байта опции, если бит nSWBOOT0 очищен в регистре FLASH_OPTR (значение BOOT0 поступает из бита опции). Затем он автоматически переключается в аналоговый режим, даже если фаза сброса не завершена.

Примечание. Когда устройство загружается из SRAM, в коде инициализации приложения необходимо переместить таблицу векторов в SRAM, используя таблицу исключений NVIC и регистр смещения. При загрузке с основной флэш-памяти прикладное программное обеспечение может загружаться как из банка 1, так и из банка 2 (только для устройств категории 3). По умолчанию выбрана загрузка из банка 1. Чтобы выбрать загрузку из банка флэш-памяти 2, установите бит VFB2 в байтах опций пользователя. Когда этот бит установлен и контакты загрузки находятся в конфигурации загрузки из основной флэш-памяти, устройство загружается из системной памяти, и загрузчик переходит к выполнению пользовательского приложения, запрограммированного в банке флэш-памяти 2. Для получения дополнительной информации см. AN2606.

См. Таблица 13: Статус доступа в зависимости от уровня защиты и режимов выполнения для функции загрузчика для разных уровней RDP.

Принудительная загрузка с пользовательской флэш-памяти

Независимо от конфигурации загрузки можно принудительно загрузиться с уникальной точки входа в основную флэш-память. См. раздел Встроенная флэш-память (FLASH).

Физическое переназначение

Как только выбран режим загрузочных контактов, прикладное программное обеспечение может изменить память, доступную в области кода (таким образом, код может выполняться через шину ICode вместо системной шины). Эта модифи-

кация выполняется путем программирования раздела 10.2.1: регистр переназначения памяти SYSCFG (SYSCFG_MEMRMP) в контроллере SYSCFG.

Таким образом, следующая память может быть переназначена:

- Основная флэш-память
- Системная память
- Встроенная SRAM1
- Банк FSMC 1 (NOR/PSRAM 1 и 2)
- Память QUADSPI

Таблица 6. Отображение памяти в зависимости от режима загрузки/физического перераспределения ⁽¹⁾

Addresses	Boot/remap in main Flash memory	Boot/remap in embedded SRAM 1	Boot/remap in system memory	Remap in FSMC	Remap in QUADSPI
0x2000 0000 - 0x2002 3FFF	SRAM1	SRAM1	SRAM1	SRAM1	SRAM1
0x1FFF 7000 - 0x1FFF FFFF	System memory/OTP/Options bytes	System memory/OTP/Options bytes	System memory/OTP/Options bytes	System memory/OTP/Options bytes	System memory/OTP/Options bytes
0x1000 8000 - 0x1FFE FFFF	Reserved	Reserved	Reserved	Reserved	Reserved
0x1000 0000 - 0x1000 7FFF	CCM SRAM	CCM SRAM	CCM SRAM	CCM SRAM	CCM SRAM
0x0808 0000 - 0x0FFF FFFF	Reserved	Reserved	Reserved	Reserved	Reserved
0x0800 0000 - 0x0807 FFFF	Flash memory	Flash memory	Flash memory	Flash memory	Flash memory
0x0400 0000 - 0x07FF FFFF	Reserved	Reserved	Reserved	FSMC bank 1 NOR/PSRAM 2 (128 MB) Aliased	QUADSPI bank (128 MB) Aliased
0x0010 0000 - 0x03FF FFFF	Reserved	Reserved	Reserved	FSMC bank 1 NOR/PSRAM 1 (128 MB) Aliased	QUADSPI bank (128 MB) Aliased
0x0000 0000 - 0x0007 FFFF (2) (3)	Flash Aliased	SRAM1 Aliased	System memory (28 KB) Aliased	FSMC bank 1 NOR/PSRAM 1 (128 MB) Aliased	QUADSPI Aliased)

1. Зарезервированная область памяти выделена в таблице серым цветом.

2. Когда FSMC переназначается по адресу 0x0000 0000, могут быть переназначены только первые две области контроллера памяти банка 1 (банк 1 NOR/PSRAM 1 и NOR/PSRAM 2). Когда FSMC переназначается по адресу 0x0000 0000, переназначается только 128 МБ. В режиме переназначения ЦП может обращаться к внешней памяти через шину ICode вместо системной шины, что повышает производительность.

3. Даже если в области загрузочной памяти используется псевдоним, соответствующая память по-прежнему доступна в исходной области памяти.

Встроенный загрузчик

Встроенный загрузчик находится в системной памяти, запрограммирован ST при производстве. См. режим загрузки системной памяти микроконтроллера AN2606 STM32.

3 Встроенная флэш-память (FLASH) для устройств категории 3

3.1 Введение

Интерфейс флэш-памяти управляет доступом CPU АНВ ICode и DCode 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 с помощью системы предварительной выборки инструкций и строк кэша.

3.2 Основные характеристики флэш-памяти

- До 512 Кбайт флэш-памяти с двухбанковой архитектурой, поддерживающей возможность чтения во время записи (RWW).
- Операции чтения флэш-памяти с двумя поддерживаемыми режимами ширины данных:
 - Режим одного банка DBANK=0: доступ для чтения 128 бит
 - Режим двойного банка DBANK=1: доступ для чтения 64 бит
- Стирание страницы, стирание банка и массовое стирание (оба банка)

Особенности интерфейса флэш-памяти:

- Операции чтения флэш-памяти
- Операции программирования/стирания флэш-памяти
- Защита от чтения активирована опцией (RDP)
- 4 области защиты от записи (2 на банк при DBANK=1 и 4 на полную память при DBANK=0)
 - 2 зоны защиты от чтения собственного кода (по 1 на банк при DBANK=1, 2 на всю память при DBANK=0)
 - 2 защищенные области памяти, определяемые параметром (по 1 на банк при DBANK = 1, 1 на всю память при DBANK = 0).
- Предварительная выборка на ICODE
- Кэш инструкций: 32 строки кэша по 4 x 64 или 2 x 128 бит в ICode (1 КБ ОЗУ)
- Кэш данных: 8 строк кэша по 4 x 64 бита или 2 x 128 на DCode (256 байт ОЗУ)
- Коррекция кода ошибки ECC: 8 бит на 64-битное двойное слово
 - DBANK=1: 8 + 64 = 72 бита, 2 бита обнаружения, 1 бит коррекции
 - DBANK=0: (8+64) + (8+64) = 144 бита, 2 бита обнаружения, 1 бит коррекции
- Опциональный загрузчик байтов
- Режим пониженного энергопотребления

3.3 Функциональное описание флэш-памяти

3.3.1 Организация флэш-памяти

Флэш-память имеет следующие основные характеристики:

- Емкость до 512 Кбайт в режиме одного банка (ширина чтения 128 бит) или в режиме двойного банка (ширина чтения 64 бита)

- Поддерживает режим двойной загрузки благодаря опциональному биту VFB2 (только в режиме двух банков)
 - Режим двойного банка, когда бит DBANK установлен:
 - 512 КБ организовано в 2 банка для основной памяти
 - Размер страницы 2 Кбайт
 - Чтение данных шириной 72 бита (64 бита плюс 8 бит ECC)
 - Банк и массовое стирание
 - Режим одного банка при сбросе DBANK:
 - 512 КБ организовано в один банк для основной памяти
 - Размер страницы 4 Кбайт
 - Чтение данных шириной 144 бита (128 бит плюс 2x8 бит ECC)
 - Массовое стирание

Флэш-память организована следующим образом:

- Блок основной памяти организован в зависимости от бита конфигурации двойного банка:
 - Когда включен двойной банк (установлен бит DBANK), флэш-память делится на 2 банка по 256 КБ, и каждый банк организован следующим образом:
 - Блок основной памяти, содержащий 128 страниц по 2 Кбайт.
 - Каждая страница состоит из 8 строк по 256 байт.
 - Когда двойной банк отключен (бит DBANK сброшен), блок основной памяти организован как один единственный банк размером 512 КБ следующим образом:
 - Блок основной памяти, содержащий 128 страниц по 4 Кбайт.
 - Каждая страница состоит из 8 строк по 512 байт.
 - Информационный блок, содержащий:
 - Системная память, из которой устройство загружается в режиме загрузки системной памяти. Область зарезервирована для использования STMicroelectronics и содержит загрузчик, который используется для перепрограммирования флэш-памяти через один из следующих интерфейсов: USART, SPI, I2C, FDCAN, USB. Он запрограммирован STMicroelectronics при изготовлении устройства и защищен от ложных операций записи/стирания. Для получения дополнительной информации см. AN2606, доступный на сайте www.st.com.
 - 1 Кбайт (128 двойных слов) OTP (однократно программируемых) байтов для пользовательских данных. Область OTP доступна только в Банке 1. Данные OTP не могут быть стерты и могут быть записаны только один раз. Если только один бит равен 0, все двойное слово больше не может быть записано, даже со значением 0x0000 0000 0000 0000.
 - Байты опций для пользовательской конфигурации.
- Организация памяти основана на основной области и информационном блоке, как показано в таблице 28.

3.3.2 Исправление кода ошибки (ECC)

Режим двойного банка (DBANK=1, разрядность данных 64 бита)

Данные во флэш-памяти представляют собой 72-битные слова: на каждое двойное слово добавляется 8 бит (64 бита). Механизм ECC поддерживает:

- Обнаружение и исправление одной ошибки

Table 7. Flash module - 512/256/128 KB dual bank organization (64 bits read width)

Flash area		Flash memory addresses	Size (bytes)	Name	
Main memory (512/256/128 KB)	Bank 1 ⁽¹⁾ (256/128/64 KB)	0x0800 0000 - 0x0800 07FF	2 K	Page 0	
		0x0800 0800 - 0x0800 0FFF	2 K	Page 1	
		0x0800 1000 - 0x0800 17FF	2 K	Page 2	
		0x0800 1800 - 0x0800 1FFF	2 K	Page 3	
		-	-	-	
		-	-	-	
			0x0803 F800 - 0x0803 FFFF	2 K	Page 127
	Bank 2 ⁽¹⁾ (256/128/64 KB)	0x0804 0000 - 0x0804 07FF	2 K	Page 0	
		0x0804 0800 - 0x0804 0FFF	2 K	Page 1	
		0x0804 1000 - 0x0804 17FF	2 K	Page 2	
		0x0804 1800 - 0x0804 1FFF	2 K	Page 3	
		-	-	-	
		-	-	-	
			0x0807 F800 - 0x0807 FFFF	2 K	Page 127
Information block	Bank 1	0x1FFF 0000 - 0x1FFF 6FFF	28 K	System memory	
	Bank 2	0x1FFF 8000 - 0x1FFF EFFF	28 K		
	Bank 1	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP area	
	Bank 1	0x1FFF 7800 - 0x1FFF 782F	48	Option bytes	
	Bank 2	0x1FFF F800 - 0x1FFF F82F	48		

1. Для устройств 256 КБ: со страницы 0 до страницы 63.
Для устройств 128 КБ: со страницы 0 на страницу 31

- Обнаружение двух ошибок

При обнаружении и исправлении одной ошибки в регистре Flash ECC (FLASH_ECCR) устанавливается флаг ECCS (коррекция ECC). Если установлен ECCSIE, генерируется прерывание.

При обнаружении двух ошибок в регистре FLASH_ECCR устанавливается флаг ECCD (обнаружение ECC). В этом случае создается NMI.

При обнаружении ошибки ECC адрес неисправного двойного слова и связанный с ним банк сохраняются в ADDR_ECC[20:0] и BK_ECC в регистре FLASH_ECCR.

ADDR_ECC[2:0] всегда очищаются.

Когда установлены ECCS или ECCD, ADDR_ECC и BK_ECC не обновляются, если возникает новая ошибка ECC. FLASH_ECCR обновляется только после сброса флагов ECC.

Table 8. Flash module - 512/256/128 KB single bank organization (128 bits read width)

Flash area		Flash memory addresses	Size (bytes)	Name
Main memory ⁽¹⁾ (512/256/128 KB)		0x0800 0000 - 0x0800 0FFF	4 K	Page 0
		0x0800 1000 - 0x0800 1FFF	4 K	Page 1
		0x0800 2000 - 0x0800 2FFF	4 K	Page 2
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	0x0807 F000 - 0x0807 FFFF	4 K
Information block	Bank 1	0x1FFF 0000 - 0x1FFF 6FFF	28 K	System memory
	Bank 2	0x1FFF 8000 - 0x1FFF EFFF	28 K	
	Bank 1	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP area
	Bank 1	0x1FFF 7800 - 0x1FFF 782F	48	Option bytes
	Bank 2	0x1FFF F800 - 0x1FFF F82F	48	

1. *Для устройств 256 КБ: со страницы 0 до страницы 63.
Для устройств 128 КБ: со страницы 0 на страницу 31*

Режим одного банка (DBANK=0, разрядность данных 128 бит)

Данные во флэш-памяти представляют собой 144-битные слова: на каждое двойное слово добавляется 8 бит. Механизм ECC поддерживает:

- Обнаружение и исправление одной ошибки
- Обнаружение двух ошибок на 64 двойных слова

Пользователь должен сначала проверить бит SYSF_ECC, и если он установлен, пользователь должен обратиться к модели программирования DBANK=1 (поскольку системная флэш-память всегда использует 2 банка). Если бит не установлен, пользователь должен обратиться к следующей модели программирования:

Каждое двойное слово (биты 63:0 и биты 127:64) имеет ECC.

Когда одна ошибка обнаружена в 64 младших битах (биты 63:0) и исправлена, в регистре FLASH_ECCR устанавливается флаг ECC (коррекция ECC).

Когда одна ошибка обнаружена в 64 битах MSB (биты 127:64) и исправлена, в регистре FLASH_ECCR устанавливается флаг ECC2 (коррекция ECC2).

Если ECCIE установлен, генерируется прерывание. Пользователь должен прочитать ECC и ECC2, чтобы увидеть, какая часть 128-битных данных была исправлена (63:0, 127:64 или оба).

При обнаружении двух ошибок в 64 битах LSB в регистре FLASH_ECCR устанавливается флаг ECCD (обнаружение ECC).

При обнаружении двух ошибок в 64 битах MSB (биты 127:64) в регистре FLASH_ECCSR устанавливается флаг ECCD2 (обнаружение ECC2).

В этом случае создается NMI. Пользователь должен прочитать ECCD и ECCD2, чтобы увидеть, какая часть 128-битных данных содержит обнаружение ошибок (63:0, 127:64 или оба).

При обнаружении ошибки ECC адрес ошибочного двойного слова сохраняется в ADDR_ECC[20:0] в FLASH_ECCSR. ADDR_ECC[20:0] содержит адрес 2-кратного двойного слова.

ADDR_ECC[3:0] всегда очищается. BK_ECC в этом режиме не используется.

Когда установлены ECCC/ECCC2 или ECCD/ECCD2, если возникает новая ошибка ECC, ADDR_ECC не обновляется. FLASH_ECCSR обновляется, только если флаги ECC (ECCC/ECCC2/ECCD/ECCD2) очищены.

Примечание. Для первичных данных: 0xFF FFFF FFFF FFFF FFFF обнаружена и исправлена одна ошибка, но обнаружение двух ошибок не поддерживается.

Когда сообщается об ошибке ECC, новое чтение по ошибочному адресу может не генерировать ошибку ECC, если данные все еще присутствуют в текущем буфере, даже если ECCC и ECCD очищены.

3.3.3 Задержка доступа для чтения

Для корректного чтения данных из флэш-памяти необходимо правильно запрограммировать количество состояний ожидания (LATENCY) в регистре управления доступом к флэш-памяти (FLASH_ACR) в соответствии с частотой тактовой частоты процессора (HCLK) и диапазоном внутреннего напряжения устройства V_{CORE}. См. Раздел 6.1.5: Управление динамическим масштабированием напряжения. В таблице 9 показано соответствие между состояниями ожидания и тактовой частотой процессора.

Table 9. Number of wait states according to CPU clock (HCLK) frequency

Wait states (WS) (LATENCY)	HCLK (MHz)		
	V _{CORE} Range 1 boost mode	V _{CORE} Range 1 normal mode	V _{CORE} Range 2
0 WS (1 CPU cycles)	≤ 34	≤ 30	≤ 12
1 WS (2 CPU cycles)	≤ 68	≤ 60	≤ 24
2 WS (3 CPU cycles)	≤ 102	≤ 90	≤ 26
3 WS (4 CPU cycles)	≤ 136	≤ 120	-
4 WS (5 CPU cycles)	≤ 170	≤ 150	-

После сброса тактовая частота ЦП составляет 16 МГц, а в регистре FLASH_ACR настроено 1 состояние ожидания (WS).

При изменении частоты ЦП должны применяться следующие программные последовательности, чтобы настроить количество состояний ожидания, необходимых для доступа к флэш-памяти:

Увеличение частоты процессора:

1. Запрограммируйте новое количество состояний ожидания в битах LATENCY в регистре управления доступом к флэш-памяти (FLASH_ACR).

2. Убедитесь, что новое количество состояний ожидания учитывается для доступа к флэш-памяти, читая регистр FLASH_ACR.

3. Проанализируйте изменение изменения частоты процессора, вызванное:

- изменение источника тактового сигнала, определяемого битами SW в регистре RCC_CFGR
- или с помощью предделителя тактовой частоты ЦП, определяемого битами HPRE в RCC_CFGR.

Если какой-то из двух вышеперечисленных шагов снижает частоту процессора, сначала выполните этот шаг, а затем остальные. В противном случае измените источник тактовой частоты ЦП, записав биты SW в регистр RCC_CFGR, а затем (при необходимости) измените предварительный делитель тактовой частоты ЦП, записав биты HPRE в регистр RCC_CFGR.

4. Убедитесь, что новый источник тактовой частоты ЦП и/или новое значение предварительного делителя тактовой частоты ЦП учитываются, прочитав состояние источника тактовой частоты (биты SWS) или/и значение предварительного делителя АНВ (биты HPRE), соответственно, в Регистр RCC_CFGR.

Уменьшение частоты процессора:

1. Измените источник тактового сигнала ЦП, записав биты SW в регистр RCC_CFGR.

2. При необходимости измените предварительный делитель тактовой частоты ЦП, записав биты HPRE в RCC_CFGR.

3. Проанализируйте изменение изменения частоты процессора, вызванное:

- изменение источника тактового сигнала, определяемого битами SW в регистре RCC_CFGR
- или с помощью предделителя тактовой частоты ЦП, определяемого битами HPRE в RCC_CFGR.

Если какой-либо из двух вышеперечисленных шагов увеличивает частоту процессора, сначала выполните другой шаг, а затем этот шаг. В противном случае измените источник тактовой частоты ЦП, записав биты SW в регистр RCC_CFGR, а затем (при необходимости) измените предварительный делитель тактовой частоты ЦП, записав биты HPRE в регистр RCC_CFGR.

4. Убедитесь, что новый источник тактовой частоты ЦП и/или новое значение предварительного делителя тактовой частоты ЦП учитываются, прочитав состояние источника тактовой частоты (биты SWS) или/и значение предварительного делителя АНВ (биты HPRE), соответственно, в Регистр RCC_CFGR.

5. Запрограммируйте новое количество состояний ожидания в битах LATENCY в регистре управления доступом к флэш-памяти (FLASH_ACR).

6. Убедитесь, что для доступа к флэш-памяти используется новое количество состояний ожидания, прочитав регистр FLASH_ACR.

3.3.4 Адаптивный ускоритель памяти в реальном времени (ART Accelerator)

Запатентованный ускоритель памяти Adaptive real-time (ART) оптимизирован для отраслевого стандарта STM32 Arm® Cortex®-M4 с процессорами FPU. Он уравнивает присущее Arm® Cortex®-M4 преимущество в производительности с FPU по сравнению с технологиями флэш-памяти, которые обычно требуют, чтобы процессор ожидал флэш-память на более высоких рабочих частотах.

Для высвобождения полной производительности процессора ускоритель реализует очередь предварительной выборки инструкций и кэш ветвлений, что увеличивает скорость выполнения программы из 64-битной флэш-памяти. Согласно бенчмарку CoreMark, производительность, достигнутая благодаря ускорителю ART, эквивалентна выполнению программы в состоянии ожидания 0 из флэш-памяти при частоте процессора до 170 МГц.

Предварительная выборка инструкций

Cortex®-M4 получает инструкции по шине ICode и литеральный пул (константа/данные) по шине DCode. Блок предварительной выборки направлен на повышение эффективности доступа к шине ICode.

В случае режима одного банка (бит опции DBANK сброшен) каждая операция чтения флэш-памяти предоставляет 128 бит либо из четырех инструкций по 32 бита, либо из восьми инструкций по 16 бит, в зависимости от запущенной программы. Эта 128-битная текущая строка инструкции сохраняется в текущем буфере, и в случае последовательного кода для выполнения предыдущей строки инструкции чтения требуется не менее четырех циклов ЦП.

В режиме двойного банка (установлен бит опции DBANK) каждая операция чтения флэш-памяти предоставляет 64 бита либо из двух инструкций по 32 бита, либо из четырех инструкций по 16 бит, в зависимости от запущенной программы. Эта 64-битная текущая строка инструкции сохраняется в текущем буфере, и в случае последовательного кода для выполнения предыдущей строки инструкции чтения требуется не менее двух циклов ЦП.

Предварительная выборка на шине ICode может использоваться для чтения следующей последовательной строки инструкций из флэш-памяти, в то время как текущая строка инструкций запрашивается ЦП.

Предварительная выборка включается установкой бита PRFTEN в регистре управления доступом к флэш-памяти (FLASH_ACR). Эта функция полезна, если для доступа к флэш-памяти требуется хотя бы одно состояние ожидания.

На рис. 9 показано выполнение последовательных 16-битных инструкций с предварительной выборкой и без нее, когда для доступа к флэш-памяти требуется 3 WS.

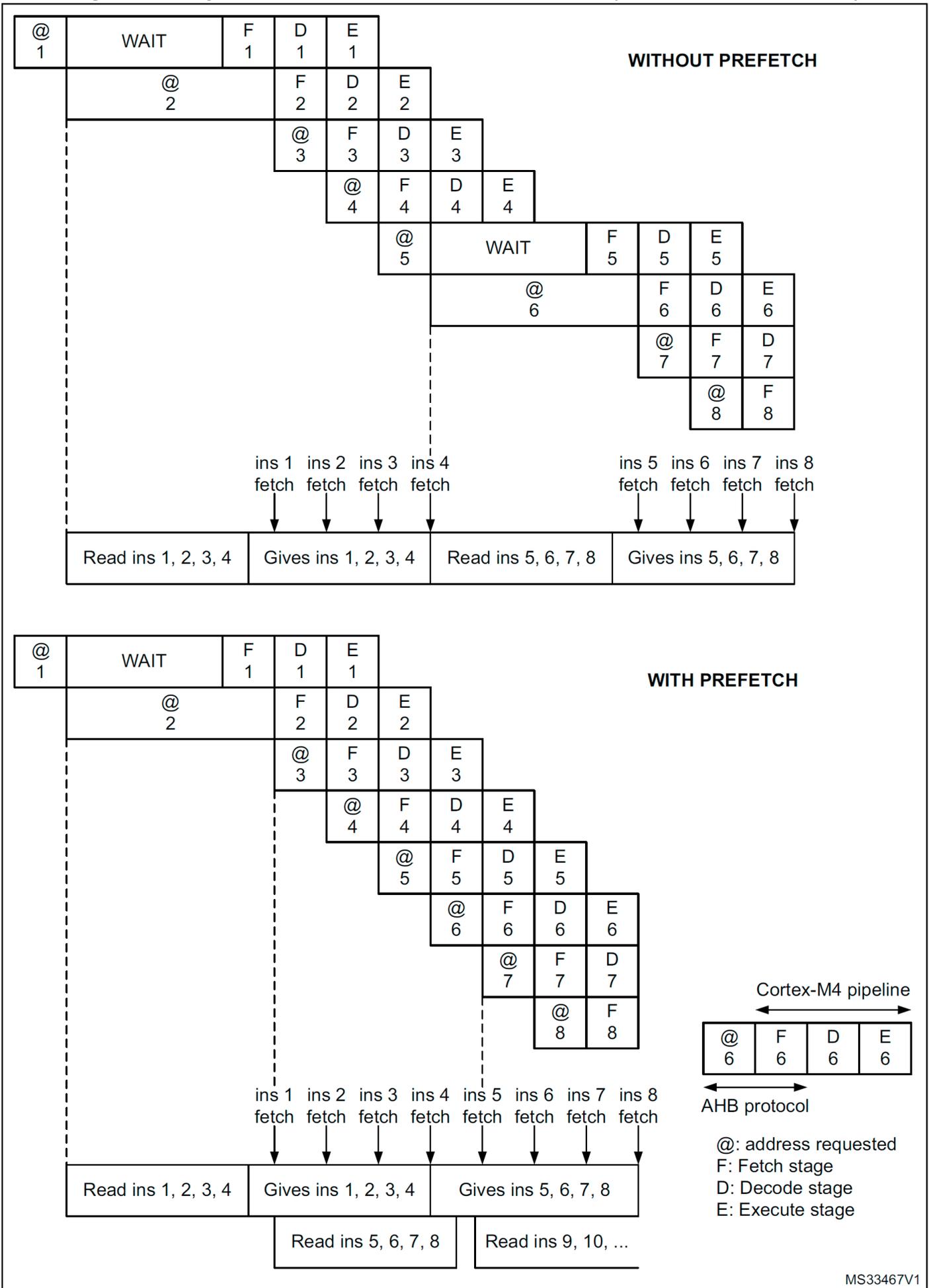
Когда код не последовательный (ветвь), инструкция может отсутствовать в текущей используемой строке инструкции или в строке инструкции с предварительной выборкой. В этом случае (промах) штраф с точки зрения количества циклов как минимум равен количеству состояний ожидания.

Если в текущем буфере присутствует цикл, новый доступ к флэш-памяти не выполняется.

Кэш-память инструкций (I-Cache)

Чтобы ограничить время, потерянное из-за переходов, можно сохранить 32 строки по 4 x 64 бита в режиме двойного банка или 32 строки по 2 x 128 бит в режиме одного банка в кэш-памяти инструкций. Эту функцию можно включить, установив бит разрешения кэша инструкций (ICEN) в регистре управления доступом к флэш-памяти (FLASH_ACR). Каждый раз, когда происходит промах (запрошенные данные отсутствуют в используемой в данный момент строке инструкции, в предварительно выбранной строке инструкции или в кэш-памяти инструкций), прочитанная строка копируется в кэш-память инструкций. Если некоторые данные, содержащиеся в кэш-памяти инструкций, запрашиваются ЦП, они предо-

Figure 3. Sequential 16-bit instructions execution (64-bit read data width)



ставляются без вставки какой-либо задержки. Как только все строки кэш-памяти инструкций заполнены, используется политика LRU (наименее недавно использовавшаяся) для определения строки, подлежащей замене в кэш-памяти инструкций. Эта функция особенно полезна в случае кода, содержащего циклы.

Кэш-память инструкций включается после сброса системы.

Кэш-память данных (D-Cache)

Пулы литералов извлекаются из флэш-памяти через шину DCode на этапе выполнения конвейера ЦП. Каждый доступ к чтению шины DCode извлекает 64 или 128 бит, которые сохраняются в текущем буфере. Следовательно, конвейер ЦП останавливается до тех пор, пока не будет предоставлен запрошенный литеральный пул. Чтобы ограничить потери времени из-за литеральных пулов, доступ через DCode шины данных АНВ имеет приоритет над доступом через ICode шины команд АНВ.

Если некоторые литеральные пулы используются часто, кэш-память данных можно включить, установив бит включения кэша данных (DCEN) в регистре управления доступом к флэш-памяти (FLASH_ACR). Эта функция работает так же, как и кэш-память инструкций, но размер сохраняемых данных ограничен 8 строками по 4*64 бита в режиме двойного банка и 8 строками по 2*128 бит в режиме одного банка.

Кэш-память данных включается после сброса системы.

***Примечание.** D-Cache активен только тогда, когда данные запрашиваются ЦП (а не DMA1 и DMA2). Данные в блоке байтов опций не кэшируются.*

3.3.5 Программирование флэш-памяти и операции стирания

Встроенная флэш-память серии STM32G4 может быть запрограммирована с помощью внутрисхемного программирования или программирования внутри приложения.

Метод внутрисхемного программирования (ICP) используется для обновления всего содержимого Flash-памяти с использованием протокола JTAG, SWD или загрузчика для загрузки пользовательского приложения в микроконтроллер. ICP предлагает быстрые и эффективные итерации проектирования и устраняет ненужную обработку пакетов или подключение устройств.

В отличие от метода ICP, ***внутриприкладное программирование (IAP)*** может использовать любой коммуникационный интерфейс, поддерживаемый микроконтроллером (входы-выходы, USB, CAN, UART, I²C, SPI и т. д.), для загрузки данных программирования в память. IAP позволяет пользователю перепрограммировать флэш-память во время работы приложения. Тем не менее, часть приложения должна быть предварительно запрограммирована во флэш-памяти с использованием ICP.

Содержимое флэш-памяти не гарантируется, если во время работы с флэш-памятью происходит сброс устройства.

Текущая операция с флэш-памятью не блокирует ЦП, пока ЦП не обращается к тому же банку флэш-памяти. Выборка кода или данных возможна в одном банке, в то время как операция записи/стирания выполняется в другом банке (см. Раздел 3.3.8: Чтение во время записи (RW), доступное только в режиме двойного банка (DBANK=1)).

Стирание и программирование флэш-памяти возможно только в диапазоне масштабирования напряжения 1. Биты VOS[1:0] в PWR_CR1 должны быть запрограммированы на 01b.

Наоборот, во время операции программирования/стирания флэш-памяти любая попытка чтения того же банка флэш-памяти останавливает шину. Операция чтения проходит правильно после завершения операции программирования/стирания.

Разблокировка флэш-памяти

После сброса запись в регистр управления флэш-памятью (FLASH_CR) запрещена для защиты флэш-памяти от возможных нежелательных операций, например, из-за электрических помех. Для разблокировки этого регистра используется следующая последовательность:

1. Записать KEY1 = 0x45670123 в регистр Flash key (FLASH_KEYR)
2. Запишите KEY2 = 0xCDEF89AB в регистр FLASH_KEYR.

Любая неправильная последовательность блокирует регистр FLASH_CR до следующего сброса системы. В случае неправильной последовательности клавиш обнаруживается ошибка шины и генерируется прерывание Hard Fault.

Регистр FLASH_CR может быть снова заблокирован программой путем установки бита LOCK в регистре FLASH_CR.

***Примечание.** Регистр FLASH_CR не может быть записан, если установлен бит BSY в регистре состояния флэш-памяти (FLASH_SR). Любая попытка записи в него с установленным битом BSY приводит к остановке шины AHB до тех пор, пока бит BSY не будет очищен.*

3.3.6 Последовательности стирания основной флэш-памяти

Операция стирания флэш-памяти может выполняться на уровне страницы, банка или всей флэш-памяти (массовое стирание). Массовое стирание не влияет на информационный блок (системная флэш-память, OTP и байты опций).

Стереть страницу

Чтобы стереть страницу, выполните следующую процедуру:

1. Убедитесь, что операции с флэш-памятью не выполняются, проверив бит BSY в регистре состояния флэш-памяти (FLASH_SR).
2. Проверьте и очистите все флаги программирования ошибок из-за предыдущего программирования. Если нет, устанавливается PGSERR.
3. В режиме двойного банка (бит опции DBANK установлен) установите бит PER и выберите страницу для стирания (PNB) с соответствующим банком (BKER) в регистре управления флэш-памятью (FLASH_CR). В режиме одного банка (бит опции DBANK сброшен) установите бит PER и выберите страницу для стирания (PNB). Бит BKER в регистре управления флэш-памятью (FLASH_CR) должен быть очищен.
4. Установите бит STRT в регистре FLASH_CR.
5. Дождитесь сброса бита BSY в регистре FLASH_SR.

***Примечание.** Внутренний генератор HSI16 (16 МГц) включается автоматически, когда бит STRT установлен, и отключается автоматически, когда бит STRT очищается, за исключением случаев, когда HSI16 ранее был включен с помощью HSION в регистре RCC_CR.*

Если стирание страницы является частью области, защищенной от записи (с помощью WRP или PCROP), устанавливается WRPERR и запрос на стирание страницы отменяется.

Банк 1, Банк 2 Массовое стирание (доступно только в режиме двойного банка, когда DBANK=1)

Чтобы выполнить массовое стирание банка, выполните следующую процедуру:

1. Убедитесь, что операции с флэш-памятью не выполняются, проверив бит BSY в регистре FLASH_SR.
2. Проверьте и очистите все флаги программирования ошибок из-за предыдущего программирования. Если нет, устанавливается PGSEERR.
3. Установите бит MER1 или MER2 (в зависимости от банка) в регистре управления флэш-памятью (FLASH_CR). Оба банка могут быть выбраны в одной и той же операции, в этом случае это соответствует массовому стиранию.
4. Установите бит STRT в регистре FLACH_CR.
5. Дождитесь сброса бита BSY в регистре состояния флэш-памяти (FLASH_SR).

Массовое стирание

Чтобы выполнить массовое стирание, выполните следующую процедуру:

1. Убедитесь, что операции с флэш-памятью не выполняются, проверив бит BSY в регистре FLASH_SR.
2. Проверьте и очистите все флаги программирования ошибок из-за предыдущего программирования. Если нет, устанавливается PGSEERR.
3. Установите бит MER1 и MER2 в регистре управления флэш-памятью (FLASH_CR).
4. Установите бит STRT в регистре FLACH_CR.
5. Дождитесь сброса бита BSY в регистре состояния флэш-памяти (FLASH_SR).

***Примечание.** Внутренний генератор HSI16 (16 МГц) включается автоматически, когда бит STRT установлен, и отключается автоматически, когда бит STRT очищается, за исключением случаев, когда HSI16 ранее был включен с помощью HSION в регистре RCC_CR.*

Когда DBANK=0, если установлен только бит MERA или MERB, устанавливается PGSEERR и операция стирания не выполняется.

Если стираемый банк или один из стираемых банков содержит область, защищенную от записи (с помощью WRP или PCROP), устанавливается WRPERR и запрос на массовое стирание отменяется (для обоих банков, если выбраны оба).

3.3.7 Последовательности программирования основной флэш-памяти

Флэш-память программируется по 72 бита за раз (64 бита + 8 бит ECC).

Программирование по ранее запрограммированному адресу не допускается, за исключением случаев, когда данные для записи полные нули, и любая попытка устанавливает флаг PROGERR в регистре состояния флэш-памяти (FLASH_SR).

Можно запрограммировать только двойное слово (2 x 32-битные данные).

- Любая попытка записи байта или полуслова устанавливает флаг SIZERR в регистр FLASH_SR.
- Любая попытка записи двойного слова, которое не соответствует адресу двойного слова, устанавливает флаг PGAERR в регистре FLASH_SR.

Стандартное программирование

Последовательность программирования Flash-памяти в стандартном режиме следующая:

1. Убедитесь, что никакие операции с основной флэш-памятью не выполняются, проверив бит BSY в регистре состояния флэш-памяти (FLASH_SR).
2. Проверьте и очистите все флаги программирования ошибок из-за предыдущего программирования. Если нет, устанавливается PGSERR.
3. Установите бит PG в регистре управления флэш-памятью (FLASH_CR).
4. Выполните операцию записи данных по нужному адресу памяти, внутри блока основной памяти или области OTP. Можно запрограммировать только двойное слово.
 - Напишите первое слово в адресе, выровненное по двойному слову
 - Напиши второе слово.
5. Дождитесь сброса бита BSY в регистре FLASH_SR.
6. Убедитесь, что в регистре FLASH_SR установлен флаг EOP (это означает, что операция программирования прошла успешно), и удалите его с помощью программного обеспечения.
7. Очистите бит PG в регистре FLASH_SR, если больше нет запросов на программирование.

Примечание. Когда интерфейс флэш-памяти получил правильную последовательность (двойное слово), автоматически запускается программирование и устанавливается бит BSY. Внутренний генератор HSI16 (16 МГц) включается автоматически при установке бита PG и автоматически отключается при сбросе бита PG, за исключением случаев, когда HSI16 ранее был включен с помощью HSION в регистре RCC_CR.

Если пользователю нужно запрограммировать только одно слово, двойное слово должно быть дополнено значением стирания 0xFFFF FFFF для автоматического запуска программирования.

ЕСС вычисляется из двойного слова для программы.

Быстрое программирование для строки (64 двойных слова, если DBANK=1) или для половины строки (64 двойных слова, если DBANK=0)

Этот режим позволяет запрограммировать строку (64 двойных слова, если DBANK=1) или половину строки (64 двойных слова, если DBANK=0), а также сократить время программирования страницы, устраняя необходимость проверки местоположения флэш-памяти перед их программированием и чтобы избежать времени нарастания и спада высокого напряжения для каждого двойного слова. При быстром программировании тактовая частота процессора (HCLK) должна быть не менее 8 МГц.

В режиме быстрого программирования можно запрограммировать только основную память.

Последовательность программирования оперативной памяти Flash в стандартном режиме следующая:

1. В режиме одного банка (DBANK=0) выполните массовое стирание. Если нет, устанавливается PGSERR. Быстрое программирование возможно только в том случае, если код выполняется из оперативной памяти или из загрузчика. В режиме двойного банка (DBANK=1) выполните массовое стирание банка для программирования. Если нет, устанавливается PGSERR.

2. Убедитесь, что никакие операции с основной флэш-памятью не выполняются, проверив бит *BSY* в регистре состояния флэш-памяти (*FLASH_SR*).
3. Проверьте и очистите все флаги программирования ошибок из-за предыдущего программирования.
4. Установите бит *FSTPG* в регистре управления флэш-памятью (*FLASH_CR*).
5. Напишите 64 двойных слова, чтобы запрограммировать строку или половину строки. Можно запрограммировать только двойные слова:
 - Запишите первое слово в адрес, выровненный с двойным словом
 - Запишите второе слово.
6. Дождитесь сброса бита *BSY* в регистре *FLASH_SR*.
7. Убедитесь, что в регистре *FLASH_SR* установлен флаг *EOP* (это означает, что операция программирования прошла успешно), и удалите его с помощью программного обеспечения.
8. Очистите бит *FSTPG* в регистре *FLASH_SR*, если больше нет запросов на программирование.

Примечание: Если во флэш-память делается попытка записи в режиме быстрого программирования во время выполнения операции чтения в том же банке, программирование прерывается без какого-либо уведомления системы (флажок ошибки не устанавливается).

Когда интерфейс *Flash* получает первое двойное слово, автоматически запускается программирование. Бит *BSY* устанавливается, когда на первое двойное слово подается высокое напряжение, и сбрасывается, когда запрограммировано последнее двойное слово или в случае ошибки. Внутренний генератор *HSI16* (16 МГц) включается автоматически, когда бит *FSTPG* установлен, и отключается автоматически, когда бит *FSTPG* очищается, за исключением случаев, когда *HSI16* ранее был включен с помощью *HSION* в регистре *RCC_CR*.

64 двойных слова должны быть записаны последовательно. Высокое напряжение сохраняется на вспышке на протяжении всего программирования. Максимальное время между двумя запросами на запись двойных слов определяется временем программирования (около 2 x 25 мкс). Если после этого времени программирования приходит второе двойное слово, быстрое программирование прерывается и устанавливается *MISSERR*.

Высокое напряжение не должно превышать 8 мс для полной строки между двумя стираниями. Это гарантируется последовательностью 64 двойных слов, записанных последовательно с тактовой частотой, большей или равной 8 МГц. Внутренний счетчик тайм-аута отсчитывает 7 мс, когда установлено быстрое программирование, и останавливает программирование по истечении тайм-аута. В этом случае устанавливается бит *FASTERR*.

Если происходит ошибка, высокое напряжение прекращается, и следующее двойное слово не запрограммировано. В любом случае, все предыдущие двойные слова были правильно запрограммированы.

Ошибки программирования

Обнаружено несколько видов ошибок. В случае ошибки операция флэш-памяти (программирование или стирание) прерывается.

- **PROGERR**: ошибка программирования

В стандартном программировании: **PROGERR** устанавливается, если записываемое слово не было ранее стерто (кроме случаев, когда значение для программирования равно полному нулю).

- **SIZERR**: Ошибка программирования размера

При стандартном программировании или быстром программировании: можно запрограммировать только двойное слово и записать только 32-битные данные. **SIZERR** устанавливается, если записывается байт или полуслово.

- **PGAERR**: Ошибка программирования выравнивания

PGAERR устанавливается, если происходит одно из следующих условий:

– При стандартном программировании: первое программируемое слово не соответствует адресу двойного слова или второе слово не принадлежит тому же адресу двойного слова.

– При быстром программировании: данные для программирования не принадлежат к той же строке, что и предыдущие запрограммированные двойные слова, или адрес для программирования не больше предыдущего.

- **PGSERR**: ошибка последовательности программирования

PGSERR устанавливается, если происходит одно из следующих условий:

– В стандартной последовательности программирования или быстрой последовательности программирования: данные записываются, когда **PG** и **FSTPG** очищаются.

– В стандартной последовательности программирования или быстрой последовательности программирования: **MER1**, **MER2** и **PER** не сбрасываются при установке **PG** или **FSTPG**.

– В последовательности быстрого программирования: массовое стирание не выполняется до установки бита **FSTPG**.

– В последовательности массового стирания: **PG**, **FSTPG** и **PER** не стираются, если установлены **MER1** или **MER2**.

– В последовательности стирания страниц: **PG**, **FSTPG**, **MER1** и **MER2** не сбрасываются при установке **PER**.

– **PGSERR** устанавливается также, если **PROGERR**, **SIZERR**, **PGAERR**, **WRPERR**, **MISSERR**, **FASTERR** или **PGSERR** установлены из-за предыдущей ошибки программирования.

– Когда **DBANK=0**, в случае установки только **MER1** или **MER2**, устанавливается **PGSERR** (массовое стирание банка не допускается).

- **WRPERR**: Ошибка защиты от записи

WRPERR устанавливается, если происходит одно из следующих условий:

– Попытаться запрограммировать или стереть в области, защищенной от записи (**WRP**), или в области **PCROP**, или в защищенной области памяти.

– Попытка выполнить очистку банка, когда одна или несколько страниц защищены **WRP** или **PCROP**.

– Подключены функции отладки или загрузка выполняется из **SRAM** или системной флэш-памяти, когда для защиты от чтения (**RDP**) задан уровень 1.

– Попытка изменить байты опции, когда защита от чтения (**RDP**) установлена на уровень 2.

- **MISSERR**: Ошибка пропуска данных быстрого программирования

В быстром программировании: все данные должны быть записаны последовательно. MISSERR устанавливается, если предыдущее программирование данных завершено, а следующие данные для программирования еще не записаны.

- FASTERR: Ошибка быстрого программирования

При быстром программировании: FASTERR устанавливается, если возникает одно из следующих условий:

- Когда бит FSTPG установлен более чем на 7 мс, генерируется обнаружение тайм-аута.
- Когда быстрое программирование было прервано MISSERR, PGAERR, WRPERR или SIZERR.

Если во время программы или операции стирания возникает ошибка, в регистре FLASH_SR устанавливается один из следующих флагов ошибки:

PROGERR, SIZERR, PGAERR, PGSEERR, MISSERR (флаги ошибок программы), WRPERR (флаг ошибки защиты)

В этом случае, если в регистре состояния флэш-памяти (FLASH_SR) установлен бит разрешения прерывания из-за ошибки ERRIE, генерируется прерывание, а в регистре FLASH_SR устанавливается флаг ошибки операции OPERR.

***Примечание.** При обнаружении нескольких последовательных ошибок (например, в случае передачи DMA во флэш-память) флаги ошибок не могут быть очищены до окончания последовательных запросов на запись.*

Программирование и кеширование

Если доступ для записи во флэш-память касается некоторых данных в кэше данных, доступ для записи во флэш-память изменяет данные во флэш-памяти и данные в кэше.

Если операция стирания во флэш-памяти также касается данных в кэше данных или инструкций, вы должны убедиться, что эти данные перезаписываются до того, как к ним будет осуществлен доступ во время выполнения кода. Если это невозможно сделать безопасно, рекомендуется очистить кэши, установив биты DCRST и ICRST в регистре управления доступом к флэш-памяти (FLASH_ACR).

***Примечание.** Кэш I/D следует очищать только тогда, когда он отключен (I/DCEN = 0).*

3.3.8 Чтение во время записи (RWW)

доступно только в режиме двойного банка (DBANK=1)

Режим двойного банка доступен только тогда, когда установлен бит опции DBANK, разрешающий операции чтения во время записи. Эта функция позволяет выполнять операцию чтения из одного банка, в то время как операция стирания или программирования выполняется для другого банка.

***Примечание.** Операции записи во время записи не допускаются. Например, невозможно выполнить операцию стирания в одном банке при программировании другого.*

Чтение из банка 1 при стирании страниц в банке 2 (или наоборот)

При выполнении программного кода из банка 1 можно выполнить операцию стирания страницы в банке 2 (и наоборот). Следуйте приведенной ниже процедуре:

1. Убедитесь, что операции с флэш-памятью не выполняются, проверив бит BSY в регистре состояния флэш-памяти (FLASH_SR) (BSY активен, когда выполняется операция стирания/программирования в банке 1 или банке 2).

2. Установите бит PER, PSB для выбора страницы и BKER для выбора банка в регистре управления флэш-памятью (FLASH_CR).

3. Установите бит STRT в регистре FLASH_CR.

4. Дождитесь сброса бита BSY (или используйте прерывание EOP).

Чтение из банка 1 при массовом стирании банка 2 (или наоборот)

При выполнении программного кода из банка 1 можно выполнить операцию массового стирания в банке 2 (и наоборот). Следуйте приведенной ниже процедуре:

1. Убедитесь, что операции с флэш-памятью не выполняются, проверив бит BSY в регистре состояния флэш-памяти (FLASH_SR) (BSY активен, когда выполняется операция стирания/программирования в банке 1 или банке 2).

2. Установите MER1 или MER2 в регистре управления флэш-памятью (FLASH_CR).

3. Установите бит STRT в регистре FLASH_CR.

4. Дождитесь сброса бита BSY (или используйте прерывание EOP).

Чтение из банка 1 при программировании банка 2 (или наоборот)

Выполняя программный код из банка 1, можно выполнять программную операцию в банке 2 (и наоборот). Следуйте приведенной ниже процедуре:

1. Убедитесь, что операции с флэш-памятью не выполняются, проверив бит BSY в регистре состояния флэш-памяти (FLASH_SR) (BSY активен, когда выполняется операция стирания/программирования в банке 1 или банке 2).

2. Установите бит PG в регистре управления флэш-памятью (FLASH_CR).

3. Выполните операции записи данных по нужному адресу памяти внутри основного блока памяти или области OTP.

4. Дождитесь сброса бита BSY (или используйте прерывание EOP).

3.4 Байты опции FLASH

3.4.1 Описание байтов опций

Байты опций настраиваются конечным пользователем в зависимости от требований приложения. В качестве примера конфигурации сторожевой таймер может быть выбран в аппаратном или программном режиме (см. Раздел 5.4.2: Программирование дополнительных байтов).

Двойное слово разбивается в байтах опций следующим образом:

Table 10. Option byte format

63-24	23-16	15 -8	7-0	31-24	23-16	15 -8	7-0
Complemented option byte 3	Complemented option byte 2	Complemented option byte 1	Complemented option byte 0	Option byte 3	Option byte 2	Option byte 1	Option byte 0

Организация этих байтов внутри информационного блока показана в таблице 31: Организация дополнительных байтов.

Байты опций можно считать из ячеек памяти, перечисленных в Таблица 31: Организация байтов опций, или из регистров байтов опций:

- Регистр опций флэш-памяти (FLASH_OPTR)
- Регистр начального адреса флэш-памяти PCROP1 (FLASH_PCROP1SR)
- Регистр конечного адреса флэш-памяти PCROP1 (FLASH_PCROP1ER)
- Адресный регистр области A WRP флэш-банка 1 (FLASH_WRP1AR)
- Адресный регистр области B WRP флэш-банка 1 (FLASH_WRP1BR)

- Регистр начального адреса флэш-памяти PCROP2 (FLASH_PCROP2SR)
- Регистр конечного адреса флэш-памяти PCROP2 (FLASH_PCROP2ER)
- Адресный регистр области A WRP флэш-банка 2 (FLASH_WRP2AR)
- Адресный регистр области B WRP флэш-банка 2 (FLASH_WRP2BR).

Table 11. Option byte organization

BANK	Address	[63:56]	[55:48]	[47:40]	[39:32]	[31:24]	[23:16]	[15:8]	[7:0]	
Bank 1	1FFF7800	USER OPT			RDP		USER OPT			RDP
	1FFF7808	Unused		Unused and PCROP1_STRT[14:0]		Unused		Unused and PCROP1_STRT[14:0]		
	1FFF7810	PCROP_RDP and Unused		Unused and PCROP1_END[14:0]		PCROP_RDP and Unused		Unused and PCROP1_END[14:0]		
	1FFF7818	Unused	WRP1A_END [6:0]	Unused	WRP1A_STRT [6:0]	Unused	WRP1A_END [6:0]	Unused	WRP1A_STRT [6:0]	
	1FFF7820	Unused	WRP2A_END [6:0]	Unused	WRP2A_STRT [6:0]	Unused	WRP2A_END [6:0]	Unused	WRP2A_STRT [6:0]	
	1FFF7828	Unused	BOOT_LOCK	Unused	SEC_SIZE1	Unused	BOOT_LOCK	Unused	SEC_SIZE1	
Bank 2	1FFFF800	Unused								
	1FFFF808	Unused		Unused and PCROP2_STRT[14:0]		Unused		Unused and PCROP2_STRT[14:0]		
	1FFFF810	Unused		Unused and PCROP2_END[14:0]		Unused		Unused and PCROP2_END[14:0]		
	1FFFF818	Unused	WRP2B_END [6:0]	Unused	WRP2B_STRT [6:0]	Unused	WRP2B_END [6:0]	Unused	WRP2B_STRT [6:0]	
	1FFFF820	Unused	WRP2B_END [6:0]	Unused	WRP2B_STRT [6:0]	Unused	WRP2B_END [6:0]	Unused	WRP2B_STRT [6:0]	
	1FFFF828	Unused			SEC_SIZE2		Unused			SEC_SIZE2

Пользователь и байты опций защиты от чтения

Адрес флэш-памяти: 0x1FFF 7800

Производственное значение ST: 0xFFEF F8AA

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRH_EN	NRST_MODE[1:0]	n BOOT0	nSW BOOT0	CCMSRAM_RST	SRAM_PE	n BOOT1	DBANK	Res.	BFB2	WWDG_SW	IWGD_STDBY	IWDG_STOP	IWDG_SW	
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	r	r	r		r	r	r	r	r	r	r	r	r	r	r

Бит 31 Зарезервировано, должно быть сохранено значение сброса.

Бит 30 **IRH_IN**: внутренняя рукоятка сброса для PG10

0: IRH отключен

1: IRH включен

- Биты 29:28 **NRST_MODE[1:0]**: режим пина PG10
- 00: Сброс ввода/вывода
 - 01: Сброс только входа
 - 10: GPIO
 - 11: Сброс ввода/вывода
- Бит 27 **nBOOT0**: дополнительный бит nBOOT0
- 0: nBOOT0 = 0
 - 1: nBOOT0 = 1
- Бит 26 **nSWBOOT0**: программный BOOT0
- 0: BOOT0 берется из бита опции nBOOT0
 - 1: BOOT0 взят с контакта PB8/BOOT0
- Бит 25 **CCMSRAM_RST**: стирание CCM SRAM при сбросе системы
- 0: CCM SRAM стирается при сбросе системы
 - 1: CCM SRAM не стирается при сбросе системы
- Бит 24 **SRAM_PE**: проверка четности SRAM1 и CCM SRAM включена
- 0: Проверка четности SRAM1 и CCM SRAM включена
 - 1: Проверка четности SRAM1 и CCM SRAM отключена
- Бит 23 **nBOOT1**: конфигурация загрузки
- Вместе с выводом BOOT0 этот бит выбирает режим загрузки из основной флэш-памяти, SRAM1 или системной памяти. См. Раздел 2.6: Конфигурация загрузки.
- Бит 22 **DBANK**:
- 0: Режим одного банка с шириной чтения данных 128 бит
 - 1: Режим двойного банка с 64-битными данными
- Этот бит может быть записан только тогда, когда PCROP1/2 отключен.
- Бит 21 Зарезервировано, должно быть сохранено значение сброса.
- Бит 20 **VFB2**: загрузка с двумя банками
- 0: Загрузка с двумя банками отключена
 - 1: Загрузка с двумя банками включена
- Бит 19 **WWDG_SW**: выбор сторожевого таймера окна
- 0: Аппаратный сторожевой таймер окна
 - 1: Программный сторожевой таймер окна
- Бит 18 **IWDG_STDBY**: зависание независимого сторожевого счетчика в режиме ожидания
- 0: Независимый сторожевой счетчик заморожен в режиме ожидания.
 - 1: Независимый сторожевой счетчик работает в режиме ожидания.
- Бит 17 **IWDG_STOP**: зависание независимого сторожевого счетчика в режиме остановки
- 0: Независимый сторожевой счетчик заморожен в режиме остановки.
 - 1: Независимый сторожевой счетчик работает в режиме остановки.
- Бит 16 **IWDG_SW**: выбор независимого сторожевого таймера
- 0: Аппаратный сторожевой таймер
 - 1: Программный сторожевой таймер
- Бит 15 Зарезервировано, должно быть сохранено значение сброса.
- Бит 14 **nRST_SHDW**:
- 0: Сброс генерируется при входе в режим отключения.
 - 1: Сброс не производится при входе в режим отключения.

Бит 31 **PCROP_RDP**: область PCROP сохраняется при снижении уровня RDP
Только этот бит установлен. Сбрасывается после полного массового стирания из-за смены RDP с уровня 1 на уровень 0.

0: область PCROP не стирается при снижении уровня RDP с уровня 1 до уровня 0.

1: область PCROP стирается, когда уровень RDP снижается с уровня 1 до уровня 0 (полное массовое стирание).

Биты 30:15 Зарезервировано, должно быть сохранено значение сброса.

Биты 14:0 **PCROP1_END[14:0]**: смещение конца области PCROP банка 1

DBANK=1

PCROP1_END содержит последнее двойное слово области PCROP банка 1.

DBANK=0

PCROP1_END содержит последние 2 двойных слова области PCROP для всей памяти.

WRP1 Байты опции адреса области A

Адрес флэш-памяти: 0x1FFF 7818

Значение сброса: 0xFF00 FFFF (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP1A_END[6:0]														
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP1A_STRT[6:0]														
									r	r	r	r	r	r	r

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP1A_END[6:0]**: смещение конца первой области WRP «A»

DBANK=1

WRP1A_END содержит последнюю страницу первой области WRP в банке1.

DBANK=0

WRP1A_END содержит последнюю страницу первой области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP1A_STRT[6:0]**: начальное смещение WRP первой области «A»

DBANK=1

WRP1A_STRT содержит первую страницу первой области WRP для банка1.

DBANK=0

WRP1A_STRT содержит первую страницу первой области WRP для всей памяти.

WRP2 Байты опции адреса области A

Адрес флэш-памяти: 0x1FFF 7820

Значение сброса: 0xFF00 FFFF (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP2A_END[6:0]														
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP2A_STRT[6:0]														
									r	r	r	r	r	r	r

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP2A_END[6:0]**: смещение конца второй области WRP «B»

DBANK=1

WRP2A_END содержит последнюю страницу второй области WRP для банка1.

DBANK=0

WRP2A_END содержит последнюю страницу второй области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP2A_STRT[6:0]**: смещение начала второй области WRP

DBANK=1

WRP2A_STRT содержит последнюю страницу второй области WRP для банка1.

DBANK=0

WRP2A_STRT содержит последнюю страницу второй области WRP для всей памяти.

Защищаемая область памяти, банк 1, байты опций

Адрес флэш-памяти: 0x1FFF7828

Значение сброса: 0xFF00 FF00 (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK								
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SEC_SIZE1[7:0]														
								r	r	r	r	r	r	r	r

Биты 31:17 Зарезервировано, должно быть сохранено значение сброса.

Бит 16 **BOOT_LOCK**: используется для принудительной загрузки из пользовательской области.

0: загрузка на основе конфигурации пэда/опционального бита

1: принудительная загрузка из основной флэш-памяти.

Биты 15:8 Зарезервированы, должны быть сохранены в значении сброса.

Биты 7:0 **SEC_SIZE1[7:0]**: размер защищаемой области памяти.

Содержит количество страниц защищенной флэш-памяти.

PCROP2 Байты опции начального адреса

Адрес флэш-памяти: 0x1FFFF808

Значение сброса: 0xFFFF FFFF (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCROP2_STRT[14:0]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Биты 31:15 Зарезервировано, должно быть сохранено значение сброса.

Биты 14:0 **PCROP2_STRT[14:0]**: смещение начала области PCROP

DBANK=1

PCROP2_STRT содержит первое двойное слово области PCROP для банка 2.

DBANK=0

PCROP2_STRT содержит первую область PCROP из двойного слова для всей памяти.

PCROP2 Байты опции конечного адреса

Адрес флэш-памяти: 0x1FFF F810

Значение сброса: 0x00FF 0000 (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCROP2_END[14:0]														
	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Биты 31:15 Зарезервировано, должно быть сохранено значение сброса.

Биты 14:0 **PCROP2_END[14:0]**: смещение конца области PCROP

DBANK=1

PCROP2_END содержит последнее двойное слово области PCROP для банка 2.

DBANK=0

PCROP2_END содержит последние 2 двойных слова области PCROP для всей памяти.

WRP1 Байты опции адреса области В

Адрес флэш-памяти: 0x1FFF F818

Значение сброса: 0xFF00 FFFF (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP1B_END[6:0]														
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP1B_STRT[6:0]														
									r	r	r	r	r	r	r

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP1B_END[6:0]**: смещение конца первой области WRP «В»

DBANK=1

WRP1B_END содержит последнюю страницу первой области WRP для банка 2.

DBANK=0

WRP1B_END содержит последнюю страницу третьей области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP1B_STRT[6:0]**: начальное смещение WRP первой области «B»

DBANK=1

WRP1B_STRT содержит первую страницу первой области WRP для банка 2.

DBANK=0

WRP1B_STRT содержит первую страницу третьей области WRP для всей памяти.

WRP2 Байты опции адреса области B

Адрес флэш-памяти: 0x1FFF F820

Значение сброса: 0xFF00 FFFF (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP2B_END[6:0]														
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP2B_STRT[6:0]														
									r	r	r	r	r	r	r

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP2B_END[6:0]**: смещение конца второй области WRP «B»

DBANK=1

WRP2B_END содержит последнюю страницу второй области WRP для банка 2.

DBANK=0

WRP2B_END содержит последнюю страницу четвертой области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP2B_STRT[6:0]**: смещение начала второй области WRP «B»

DBANK=1

WRP2B_STRT содержит первую страницу второй области WRP для банка 2.

DBANK=0

WRP2B_STRT содержит первую страницу второй области WRP для всей памяти.

Защищаемая область памяти, банк 2, байты опций

Адрес флэш-памяти: 0x1FFF F828

Значение сброса: 0xFF00 FF00 (производственное значение ST)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SEC_SIZE2[7:0]														
								r	r	r	r	r	r	r	r

Биты 31:8 Зарезервированы, должны быть сохранены в значении сброса.
Биты 7:0 **SEC_SIZE2[7:0]**: Размер защищаемой области памяти содержит количество страниц защищаемой флэш-памяти.

3.4.2 Программирование байтов опций

После сброса соответствующие биты в регистре управления флэш-памятью (FLASH_CR) защищены от записи. Для выполнения любой операции на странице байтов опций необходимо сбросить бит блокировки опций OPTLOCK в регистре управления флэш-памятью (FLASH_CR). Для разблокировки этого регистра используется следующая последовательность:

1. Разблокируйте FLASH_CR с помощью последовательности сброса LOCK (см. Разблокировка флэш-памяти).
2. Запишите OPTKEY1 = 0x08192A3B в регистр дополнительных ключей флэш-памяти (FLASH_OPTKEYR).
3. Запишите OPTKEY2 = 0x4C5D6E7F в регистр FLASH_OPTKEYR.

Пользовательские опции могут быть защищены от нежелательных операций стирания/программирования путем установки бита OPTLOCK программно.

Примечание. Если LOCK устанавливается программно, OPTLOCK также устанавливается автоматически.

Изменение опций пользователя

Байты опции запрограммированы иначе, чем пользовательский адрес основной памяти. Невозможно независимо изменить параметры пользователя банка 1 или банка 2. Сначала изменяются параметры пользователя банка 1.

Чтобы изменить значение параметров пользователя, выполните следующую процедуру:

1. Убедитесь, что не выполняются операции с флэш-памятью, проверив бит BSY в регистре состояния флэш-памяти (FLASH_SR).
2. Сбросьте бит блокировки опции OPTLOCK с помощью последовательности очистки, описанной выше.
3. Запишите требуемое значение опций в регистры опций: регистр опций флэш-памяти (FLASH_OPTR), регистр начального адреса флэш-памяти PCROP1 (FLASH_PCROP1SR), регистр конечного адреса флэш-памяти PCROP1 (FLASH_PCROP1ER), регистр адресов области A флэш-памяти WRP (FLASH_WRP1AR), область флэш-записи WRP. Регистр адреса B (FLASH_WRP1BR), Регистр начального адреса Flash PCROP2 (FLASH_PCROP2SR), Регистр конечного адреса Flash PCROP2 (FLASH_PCROP2ER), Регистр адреса области A WRP Flash Bank 2 (FLASH_WRP2AR), Регистр адреса области B WRP Flash Bank 2 (FLASH_WRP2BR).
4. Установите стартовый бит опций OPTSRT в регистре управления флэш-памятью (FLASH_CR).
5. Дождитесь сброса бита BSY.

Примечание. Любое изменение значения одной опции выполняется автоматически путем стирания сначала страниц байтов пользовательских опций (банк 1 и банк 2), а затем программирования всех байтов опций со значениями, содержащимися в регистрах опций флэш-памяти.

Загрузка опционального байта

После сброса бита BSY все новые параметры обновляются во флэш-памяти, но не применяются к системе. Они влияют на систему при загрузке. Загрузка опциональных байтов (OBL) выполняется в двух случаях:

- когда установлен бит OBL_LAUNCH в регистре управления флэш-памятью (FLASH_CR).

- после сброса питания (сброс BOR или выход из режимов Standby/Shutdown).

Загрузчик байтов опций выполняет чтение блока опций и сохраняет данные во внутренних регистрах опций. Эти внутренние регистры настраивают систему и не могут быть прочитаны программным обеспечением. Параметр OBL_LAUNCH генерирует сброс, поэтому загрузка байта опции выполняется при сбросе системы.

Каждый опциональный бит также имеет свое дополнение в том же двойном слове. Во время загрузки опции проверка бита опции и его дополнения позволяет проверить правильность загрузки.

Во время загрузки байта опций опции считываются двойным словом с ECC. Если слово и его дополнение совпадают, опциональное слово/байт копируется в регистр опций.

Если сравнение между словом и его дополнением не удастся, устанавливается бит состояния OPTVERR. Значения несоответствия принудительно заносятся в регистры опций:

- Для опции USR OPT значение несоответствия равно «1» для всех опций, кроме BOR_LEV, которое равно «000» (самый низкий порог).

- Для опции WRP значением несоответствия является значение по умолчанию «Нет защиты».

- Для параметра RDP значением несоответствия является значение по умолчанию «Уровень 1».

- Для PCROP значением несоответствия является «защита всей памяти». При сбросе системы внутренние регистры опций копируются в регистры опций, которые могут быть прочитаны и записаны программным обеспечением (FLASH_OTPR, FLASH_PCROP1/2SR, FLASH_PCROP1/2ER, FLASH_WRP1/2AR, FLASH_WRP1/2BR). Эти регистры также используются для изменения опций. Если эти регистры не изменены пользователем, они отражают состояния опций системы. См. Раздел : Изменение параметров пользователя для получения более подробной информации.

Активация режима двойного банка (переключение с DBANK=0 на DBANK=1)

При переключении с одного режима Flash на другой (например, с одинарного на двойной банк) рекомендуется выполнять код из SRAM или использовать загрузчик. Чтобы избежать чтения поврежденных данных из флэш-памяти при изменении организации памяти, перед перепрограммированием следует избегать любого доступа (будь то ЦП или DMA) к флэш-памяти.

- Отключите кэширование инструкций/данных и/или предварительную выборку, если они разрешены (сбросьте биты PRFTEN и ICEN/DCEN в регистре FLASH_ACR).

- Очистить кэш инструкций и данных, установив биты DCRST/ICRST в регистре FLASH_ACR.

- Установите бит опции DBANK и снимите всю защиту от записи WRP (следуйте процедуре модификации пользовательских опций и загрузчика байтов опций).
 - После завершения OBL с DBANK=1 выполните массовое стирание.
 - Начать новое программирование кода в 64-битном режиме с отображением памяти DBANK=1.
 - При необходимости установите новую схему WRP/PCROP с DBANK=1.
 - При необходимости установите PRFTEN и ICEN/DCEN.
- Новое программное обеспечение готово к запуску с использованием конфигурации банка.

Отключение режима двойного банка (переключение с DBANK=1 на DBANK=0)

При переключении с одного режима Flash на другой (например, с одинарного на двойной банк) рекомендуется выполнять код из SRAM или использовать загрузчик. Чтобы избежать чтения поврежденных данных из флэш-памяти при изменении организации памяти, перед перепрограммированием следует избегать любого доступа (будь то ЦП или DMA) к флэш-памяти.

- Отключите кэширование инструкций/данных и/или предварительную выборку, если они разрешены (сбросьте биты PRFTEN и ICEN/DCEN в регистре FLASH_ACR).

- Очистить кэш инструкций и данных, установив биты DCRST/ICRST в регистре FLASH_ACR.

- Очистите бит опции DBANK и всю защиту от записи WRP (следуйте процедуре модификации пользовательских опций и загрузчика байтов опций).
 - После завершения OBL с DBANK=0 выполните массовое стирание.
 - Начать новое программирование кода в 128-битном режиме с отображением памяти DBANK=0.
 - При необходимости установите новую схему WRP/PCROP с DBANK=0. При необходимости установите PRFTEN и ICEN/DCEN.

Новое программное обеспечение готово к запуску с использованием конфигурации банка.

3.5 Защита флэш-памяти

Основная флэш-память может быть защищена от внешнего доступа с помощью защиты от чтения (RDP). Страницы флэш-памяти также могут быть защищены от нежелательной записи из-за потери контекста счетчика программ. Детализация защиты от записи (WRP) составляет одну страницу (2 КБ). Кроме того, флэш-память также может быть защищена от чтения и записи третьими лицами (PCROP). Степень детализации PCROP — двойное слово (64 бита).

3.5.1 Защита от чтения (RDP)

Защита от чтения активируется установкой байта опции RDP, а затем применением сброса системы для перезагрузки нового байта опции RDP. Защита от чтения защищает основную флэш-память, дополнительные байты, резервные регистры (RTC_VKPxR в RTC) и CCM SRAM.

***Примечание.** Если защита от чтения установлена, когда отладчик все еще подключен через JTAG/SWD, примените POR (сброс при включении) вместо сброса системы.*

Существует три уровня защиты от чтения: от отсутствия защиты (уровень 0) до максимальной защиты или отсутствия отладки (уровень 2).

Флэш-память защищена, если байт опции RDP и его дополнение содержат пару значений, показанных в таблице 32.

Табл. 12. Состояние защиты от чтения флэш-памяти

Значение байта RDP	Значение дополнения RDP	Уровень защиты чтения
0xAA	0x55	Level 0 (значение производителя)
Любое значение, кроме 0xAA или 0xCC	Любое значение (не обязательно дополнительное), кроме 0x55 и 0x33	Level 1
0xCC	0x33	Level 2

Область системной памяти доступна для чтения независимо от уровня защиты. Он никогда не доступен для операций программирования/стирания.

Уровень 0: нет защиты

Возможны операции чтения, программирования и стирания в основной области флэш-памяти. Байты опций, CCM SRAM и резервные регистры также доступны для всех операций.

Уровень 1: Защита от чтения

Это уровень защиты по умолчанию, когда байт опции RDP стирается. Он также определяется, когда значение RDP отличается от 0xAA и 0xCC, или даже если дополнение неверно.

- Пользовательский режим: код, выполняемый в пользовательском режиме (загрузочная флэш-память), может получить доступ к основной флэш-памяти, байтам параметров, CCM SRAM и резервным регистрам со всеми операциями.

- Режимы отладки, загрузочного ОЗУ и загрузчика: в режиме отладки или когда код запускается из загрузочного ОЗУ или загрузчика, основная флэш-память, резервные регистры (RTC_VKPxR в RTC) и CCM SRAM полностью недоступны. В этих режимах при доступе к флэш-памяти для чтения или записи возникает ошибка шины и прерывание Hard Fault.

Предостережение: Если настроен уровень 1 и область PCROP не определена, обязательно установить бит PCROP_RDP в 1 (полное массовое стирание, когда уровень RDP снижается с уровня 1 до уровня 0). Если сконфигурирован Уровень 1 и определена область PCROP, если код пользователя должен быть защищен RDP, а не PCROP, его нельзя помещать на страницу, содержащую область PCROP.

Уровень 2: без отладки

На этом уровне гарантируется уровень защиты 1. Кроме того, порт отладки Cortex®-M4, загрузка из ОЗУ (режим загрузочного ОЗУ) и загрузка из системной памяти (режим загрузчика) больше не доступны. В пользовательском режиме выполнения (режим загрузки FLASH) разрешены все операции с основной флэш-памятью. Наоборот, с байтами опций можно выполнять только операции чтения.

Байты опций нельзя ни запрограммировать, ни стереть. Таким образом, уровень 2 вообще нельзя удалить: это необратимая операция. При попытке изменить байты опций в регистре `Flash_SR` устанавливается флаг ошибки защиты `WRPERR` и может генерироваться прерывание.

***Примечание.** Функция отладки также отключается при сбросе.*

STMicroelectronics не может проводить анализ дефектных деталей, на которых установлен уровень защиты 2.

Изменение уровня защиты от чтения

Легко перейти с уровня 0 на уровень 1, изменив значение байта `RDP` на любое значение (кроме `0xCC`). Запрограммировав значение `0xCC` в байте `RDP`, можно перейти на уровень 2 либо напрямую с уровня 0, либо с уровня 1. **Оказавшись на уровне 2, изменить уровень защиты от чтения больше невозможно.**

Когда `RDP` перепрограммируется на значение `0xAA` для перехода с уровня 1 на уровень 0, выполняется массовое стирание основной флэш-памяти, если в регистре конечного адреса флэш-памяти `PCROP1 (FLASH_PCROP1ER)` установлено значение `PCROP_RDP`. Резервные регистры (`RTC_BKPxR` в `RTC`) и `CCM SRAM` также стираются. Для параметров пользователя, кроме защиты `PCROP`, устанавливаются их предыдущие значения, скопированные из `FLASH_OPTR`, `FLASH_WRPxR` ($x=1, 2$ и $y=A$ или B). `PCROP` отключен. Область одноразового пароля не подвергается массовому стиранию и остается неизменной.

Если бит `PCROP_RDP` сброшен в `FLASH_PCROP1ER`, полное массовое стирание заменяется частичным массовым стиранием, то есть последовательными стираниями страниц в банке, где активен `PCROP`, за исключением страниц, защищенных `PCROP`. Это делается для того, чтобы сохранить код `PCROP`. Если `PCROP` активен для обоих банков, оба банка стираются путем стирания страниц.

Только когда оба банка стираются, параметры перепрограммируются с их предыдущими значениями. Это также верно для регистров `FLASH_PCROPxSR` и `FLASH_PCROPxER` ($x=1,2$).

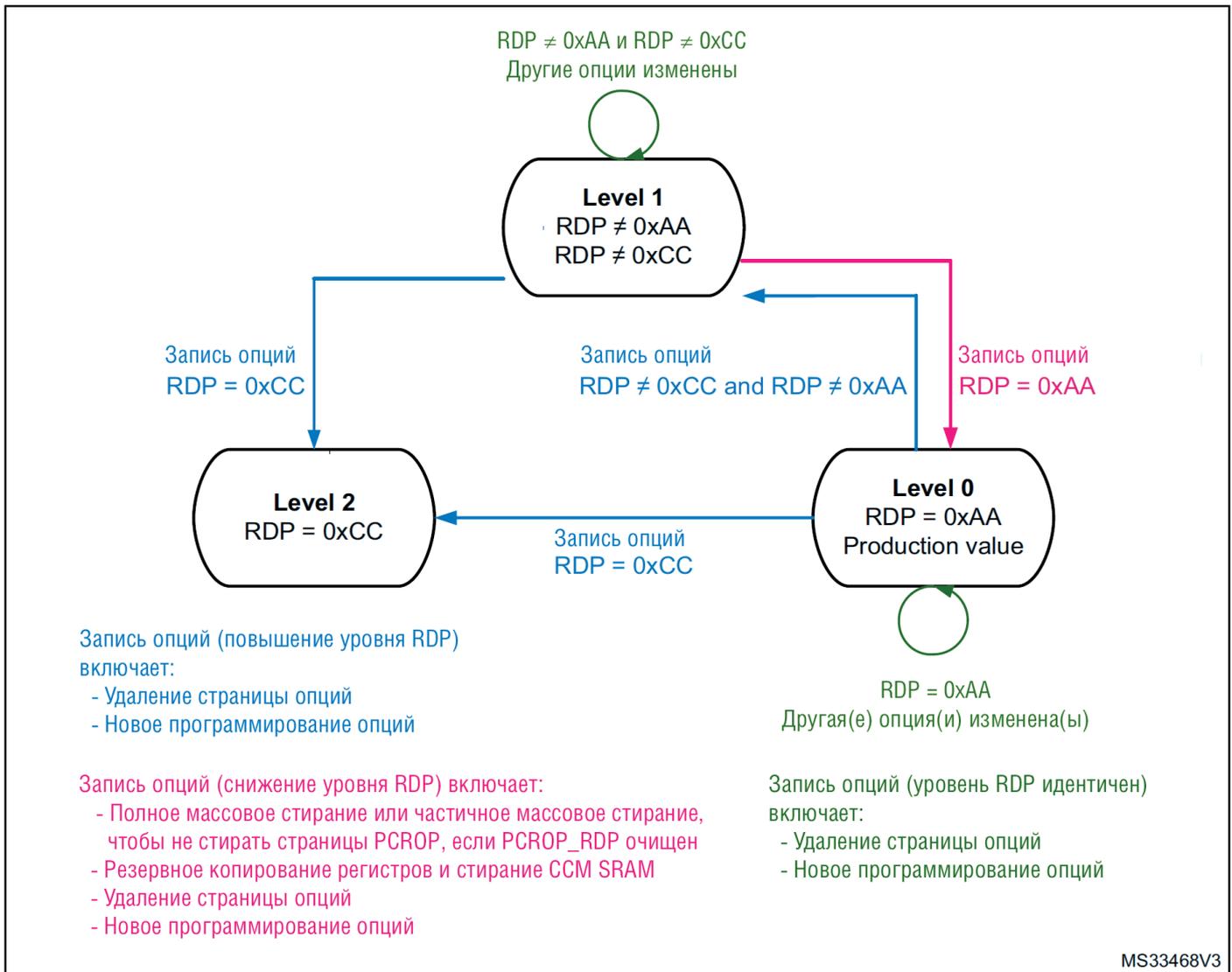
***Примечание.** Полное массовое стирание или частичное массовое стирание выполняется только тогда, когда активен уровень 1 и запрошен уровень 0. При повышении уровня защиты (0->1, 1->2, 0->2) массового стирания нет.*

Чтобы подтвердить изменение уровня защиты, байты опций должны быть перезагружены через бит `OBL_LAUNCH` в регистре управления флэш-памятью.

3.5.2 Защита от считывания собственного кода (PCROP)

Кроме того, флэш-память может быть защищена от чтения и записи третьими лицами. Защищенная область предназначена только для выполнения: она может быть доступна только `CPU STM32` в виде кода инструкции, в то время как все другие виды доступа (`DMA`, отладка и чтение, запись и стирание данных `CPU`) строго запрещены. В зависимости от режима `DBANK` он позволяет либо указать одну зону `PCROP` на банк в режиме двойного банка, либо указать две зоны `PCROP` для всей памяти. Дополнительный опциональный бит (`PCROP_RDP`) позволяет выбрать, будет ли стираться область `PCROP` при изменении защиты `RDP` с уровня 1 на уровень 0 (см. Изменение уровня защиты от чтения).

Рисунок 4. Изменение уровня защиты от чтения (RDP)



Каждая область PCROP определяется смещением начальной страницы и смещением конечной страницы, относящимся к базовому адресу физического банка флэш-памяти. Эти смещения определены в адресных регистрах PCROP Flash PCROP1 Регистр начального адреса (FLASH_PCROP1SR), Flash PCROP1 Регистр конечного адреса (FLASH_PCROP1ER), Flash PCROP2 Регистр начального адреса (FLASH_PCROP2SR), Flash PCROP2 Регистр конечного адреса (FLASH_PCROP2ER).

В режиме одного банка (DBANK=0):

- Область PCROP_x (x = 1,2) определяется от адреса: базовый адрес + [PCROP_x_STRT x 16] (включено) до адреса: базовый адрес + [(PCROP_x_END+1) x 16] (исключено). Минимальный размер области PCROP составляет два двойных слова (256 бит).

В режиме двойного банка (DBANK=1)

- Область PCROP_x (x = 1,2) определяется от адреса: базовый адрес банка «x» + [PCROP_x_STRT x 0x8] (включительно) до адреса: базовый адрес банка «x» + [(PCROP_x_END+1) x 0x8] (исключено). Минимальный размер области PCROP составляет два двойных слова (128 бит).

Например, для защиты PCROP с адреса 0x0806 2F80 (включительно) по адресу 0x0807 0004 (включено):

Табл. 13. Статус доступа в зависимости от уровня защиты и режимов выполнения

Area	Protection level	Пользовательское выполнение (загрузка с флэш-памяти)			Отладка / загрузка из оперативной памяти/ Загрузка из загрузчика ⁽¹⁾		
		Read	Write	Erase	Read	Write	Erase
Flash main memory	1	Yes	Yes	Yes	No	No	No ⁽³⁾
	2	Yes	Yes	Yes	N/A	N/A	N/A
System memory ⁽²⁾	1	Yes	No	No	Yes	No	No
	2	Yes	No	No	N/A	N/A	N/A
Option bytes	1	Yes	Yes ⁽³⁾	Yes	Yes	Yes ⁽³⁾	Yes
	2	Yes	No	No	N/A	N/A	N/A
OTP	1	Yes	Yes ⁽⁴⁾	N/A	No	No	N/A
	2	Yes	Yes ⁽⁴⁾	N/A	N/A	N/A	N/A
Backup registers	1	Yes	Yes	N/A	No	No	No ⁽⁵⁾
	2	Yes	Yes	N/A	N/A	N/A	N/A
CCM SRAM	1	Yes	Yes	N/A	No	No	No ⁽⁶⁾
	2	Yes	Yes	N/A	N/A	N/A	N/A

1. Когда уровень защиты 2 активен, порт отладки, загрузка из ОЗУ и загрузка из системной памяти отключены.

2. Системная память доступна только для чтения, независимо от уровня защиты (0, 1 или 2) и режима выполнения.

3. Основная флэш-память стирается, когда байт опции RDP запрограммирован с отключенными защитами всех уровней (0xAA).

4. OTP можно написать только один раз.

5. Резервные регистры стираются при изменении RDP с уровня 1 на уровень 0.

6. CCM SRAM стирается при изменении RDP с уровня 1 на уровень 0.

• если загрузка во флэш-памяти выполняется в банке 1, регистры FLASH_PCROP1SR и FLASH_PCROP1ER должны быть запрограммированы с помощью:

– PCROP1_STRT = 0xC5F0.

– PCROP1_END = 0xE000.

• Если два банка переставлены местами, защита должна применяться к банку 2, а регистры

FLASH_PCROP2SR и FLASH_PCROP2ER должны быть запрограммированы с помощью:

– PCROP2_STRT = 0xC5F0.

– PCROP2_END = 0xE000.

Любой доступ для чтения, выполненный через D-bus к защищенной области PCROP, вызывает ошибку флага RDERR.

Любой адрес, защищенный PCROP, также защищен от записи, и любой доступ для записи к одному из этих адресов запускает WRPERR.

Любая область PCROP также защищена от стирания. Следовательно, любое стирание страницы в этой зоне невозможно (включая страницу, содержащую начальный адрес и конечный адрес этой зоны). Более того, массовое стирание программного обеспечения не может быть выполнено, если одна зона защищена PCROP.

В предыдущем примере из-за постраничного стирания все страницы со страницы 0x62 до 0x70 защищены на случай стирания страницы. (Все адреса от 0x0806 2000 до 0x080 70FFF не могут быть стерты).

Деактивация PCROP может происходить только при изменении RDP с уровня 1 на уровень 0. Если модификация пользовательских опций пытается очистить PCROP или уменьшить область PCROP, запускается программирование опций, но область PCROP остается неизменной. Наоборот, можно увеличить площадь PCROP.

Когда опциональный бит PCROP_RDP сброшен, когда RDP изменяется с уровня 1 на уровень 0, полное массовое стирание заменяется частичным массовым стиранием, чтобы сохранить область PCROP (см. Изменение уровня защиты от чтения). В этом случае PCROP1/2_STRT и PCROP1/2_END также не стираются.

Примечание. При использовании PCROP_RDP рекомендуется выровнять область PCROP с детализацией страницы или оставить свободной остальную часть страницы, где начинается или заканчивается зона PCROP.

Таблица 14. Защита PCROP⁽¹⁾

PCROP _x регистрирует значения (x = 1,2)	Зона защиты PCROP
PCROP _x _offset_strt > PCROP _x _offset_end	Нет области PCROP.
PCROP _x _offset_strt < PCROP _x _offset_end	Область между PCROP _x _offset_strt и PCROP _x _offset_end защищена. можно писать: – PCROP _x _offset_strt с меньшим значением – PCROP _x _offset_end с более высоким значением

1. Когда DBANK=1, минимальный размер области PCROP составляет 2 двойных слова: PCROP_x_offset_strt и PCROP_x_offset_end.

Когда DBANK=0, минимальный размер области PCROP равен 2x(2хдвойных слов): PCROP_x_offset_strt и PCROP_x_offset_end.

Когда DBANK=1, ответственность за отсутствие перекрытия зон PCROP лежит на пользователе.

3.5.3 Защита от записи (WRP)

Пользовательскую область во флэш-памяти можно защитить от нежелательных операций записи. В зависимости от битовой конфигурации опции DBANK, она позволяет указать:

- В режиме одного банка (DBANK=0): в каждом банке можно определить четыре защищенные от записи (WRP) области с разбивкой по размеру страницы (4 КБ).
- В режиме двойного банка (DBANK=1): в каждом банке можно определить две защищенные от записи (WRP) области со степенью детализации страниц (2 КБ).

Каждая область определяется смещением начальной страницы и смещением конечной страницы, относящимся к базовому адресу физического банка

флэш-памяти. Эти смещения определены в адресных регистрах WRP: регистр адреса области A Flash WRP (FLASH_WRP1AR), регистр адреса области B Flash WRP (FLASH_WRP1BR), регистр адреса области A WRP Flash Bank 2 (FLASH_WRP2AR), регистр адреса области B WRP Flash Bank 2 (FLASH_WRP2BR).

Режим двойного банка (DBANK=1)

Область банка «x» WRP «y» (x=1,2 и y=A,B) определяется от адреса: Bank «x» Base address + [WRPxy_STRT x 0x800] (включительно) до адреса: Bank «x» Базовый адрес + [(WRPxy_END+1) x 0x800] (исключено).

Режим одного банка (DBANK=0)

Область «y» WRPx (x=1,2 и y=A,B) определяется от адреса: Базовый адрес + [WRPy_STRT x 0x1000] (включительно) до адреса: Базовый адрес + [(WRPy_END+1) x 0x1000] (исключено).

Например, для защиты по WRP с адреса 0x0806 2800 (включительно) по адресу 0x0807 07FF (включено):

- если загрузка во флэш-памяти выполняется в банке 1, регистр FLASH_WRP1AR должен быть запрограммирован следующим образом:

- WRP1A_STRT = 0x62.

- WRP1A_END = 0x70.

Вместо этого можно использовать WRP1B_STRT и WRP1B_END в FLASH_WRP1BR (область «B» в банке 1).

- Если два банка переставлены местами, защита должна применяться к банку 2, а регистр FLASH_WRP2AR должен быть запрограммирован следующим образом:

- WRP2A_STRT = 0x62.

- WRP2A_END = 0x70.

- WRP2A_STRT = 0xC5.

- WRP2A_END = 0xE0.

Вместо этого можно использовать WRP2B_STRT и WRP2B_END в FLASH_WRP2BR (область «B в банке 2»).

Когда WRP активен, его нельзя стереть или запрограммировать. Следовательно, массовое стирание программного обеспечения не может быть выполнено, если одна область защищена от записи.

При попытке стирания/программирования части флэш-памяти, защищенной от записи, в регистре FLASH_SR устанавливается флаг ошибки защиты от записи (WRPERR). Этот флаг также устанавливается для любого доступа на запись к:

- зона OTP

- часть флэш-памяти, которая никогда не может быть записана как область ICP – PCROP.

Примечание. Если выбран уровень защиты от чтения памяти (уровень RDP = 1), невозможно запрограммировать или стереть флэш-память, если подключены функции отладки ЦП (JTAG или одиночный провод) или загрузочный код выполняется из ОЗУ или системы. flash, даже если WRP не активирован.

Примечание. Чтобы проверить параметры WRP, байты параметров должны быть повторно загружены через бит OBL_LAUNCH в регистре управления флэш-памятью.

Примечание. Когда DBANK=0, ответственность за отсутствие перекрытий в зоне WRP лежит на пользователе.

Table 15. WRP protection

WRP registers values (x=1/2 y= A/B)	WRP protection area
WRP _{xy} _STRT = WRP _{xy} _END	Page WRP _{xy} is protected.
WRP _{xy} _STRT > WRP _{xy} _END	No WRP area.
WRP _{xy} _STRT < WRP _{xy} _END	The pages from WRP _{xy} _STRT to WRP _{xy} _END are protected.

3.5.4 Защищаемая область памяти

Область защищенной памяти определяет область кода, которая может быть выполнена только один раз при загрузке и никогда больше, если не произойдет новый сброс.

Основная цель области защищенной памяти — защитить определенную часть флэш-памяти от нежелательного доступа. Это способ изолировать загрузку микропрограммы первого этапа вместе с ее конфиденциальными активами (ключами) от остального кода приложения. Это позволяет реализовать решение корня доверия, такое как безопасная загрузка. Защищаемая область памяти находится в основной флэш-памяти. Он предназначен для выполнения надежного кода. В незащищенном состоянии защищаемая память ведет себя как остальная часть основной флэш-памяти. При защите (установлен бит SEC_PROT1 (или SEC_PROT2) в регистре FLASH_CR) любая попытка программирования или стирания в защищенной области памяти вызывает ошибку защиты от записи (установлен флаг WRPERR), а любая попытка чтения из нее генерирует ошибку чтения (установлен флаг RDERR).

Размер защищенной области памяти определяется битовым полем SEC_SIZE1[7:0] (или SEC_SIZE2[7:0]) регистра FLASH_SEC1R (или FLASH_SEC2R). Его можно изменить только на уровне RDP 0. Его содержимое стирается при переходе с уровня RDP 1 на уровень 0, даже если оно перекрывается со страницами PCROP.

Защищаемая область памяти определяется:

В случае конфигурации с двумя банками (DBANK=1):

с адреса: Базовый адрес Bank1 (включено) на адрес: Базовый адрес Bank1 + (0x800 * SEC_SIZE1) (исключено) и с адреса: Базовый адрес Bank2 (включено) на адрес: Базовый адрес Bank2 + (0x800 * SEC_SIZE2) (Исключенный) .

В случае конфигурации с одним банком DBANK=0:

с адреса: Базовый адрес банка (включено) на адрес: Базовый адрес банка + (0x1000 * SEC_SIZE1) (исключено).

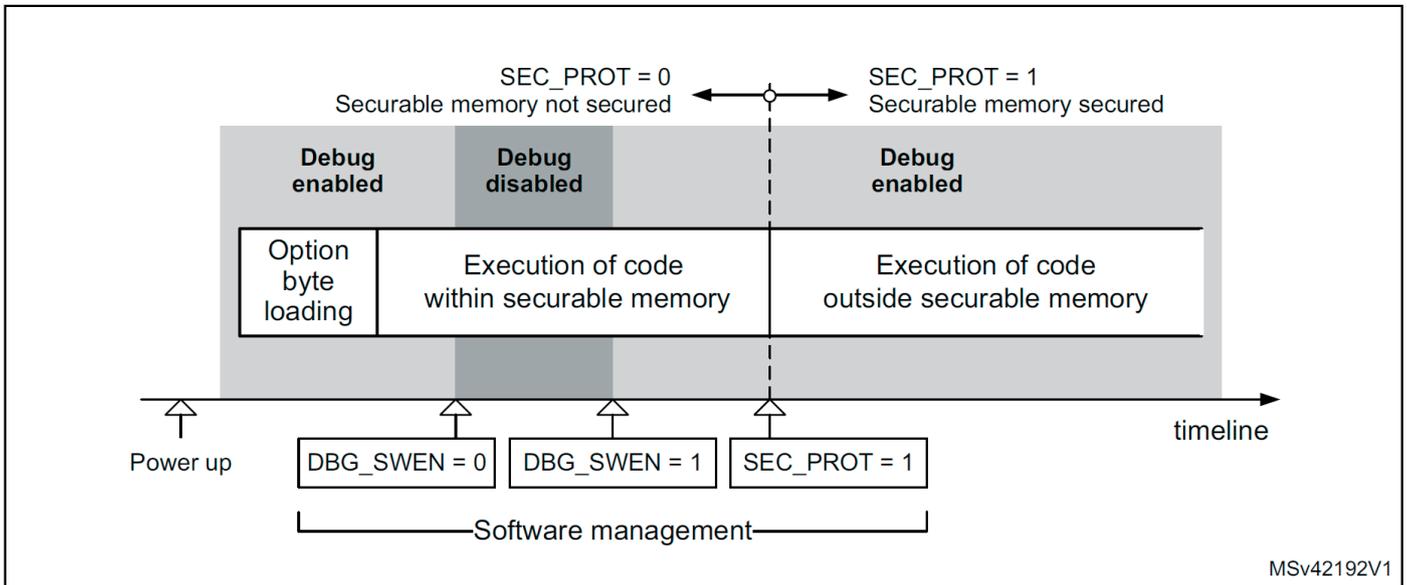
3.5.5 Отключение доступа к отладке ядра

Для выполнения конфиденциального кода или манипулирования конфиденциальными данными в защищенной области памяти отладочный доступ к ядру может быть временно отключен.

На уровне RDP 2 отладчик отключен аппаратно, но на других уровнях RDP отладчик может быть отключен программно с помощью бита `DBG_SWEN` в регистре `FLASH_ACR`.

На рис. 11 приведен пример управления битами `DBG_SWEN` и `SEC_PROT`.

Рисунок 5. Пример отключения доступа к отладке ядра



MSv42192V1

3.5.6 Принудительная загрузка с флэш-памяти

Чтобы повысить безопасность и установить цепочку доверия, бит опции `BOOT_LOCK` регистра `FLASH_SEC1R/FLASH_SEC2R` позволяет принудительно загрузить систему из основной флэш-памяти независимо от других параметров загрузки. Всегда можно установить бит `BOOT_LOCK`. Однако сбросить его можно только в том случае, если:

- RDP установлен на уровень 0, или
- RDP установлен на уровень 1, в то время как запрашивается уровень 0 и выполняется полное массовое стирание.

3.6 FLASH-прерывания

Таблица 16. Запрос прерывания флэш-памяти

Interrupt event	Event flag	Event flag/interrupt clearing method	Interrupt enable control bit
End of operation	EOP ⁽¹⁾	Write EOP=1	EOPIE
Operation error	OPERR ⁽²⁾	Write OPERR=1	ERRIE
Read error	RDERR	Write RDERR=1	RDERRIE
ECC correction	ECCC	Write ECCC=1	ECCCIE

1. EOP устанавливается, только если установлен EOPIE.

2. OPERR устанавливается, только если установлено ERRIE.

3.7 FLASH-регистры

3.7.1 Регистр управления доступом к флэш-памяти (FLASH_ACR)

Смещение адреса: 0x00

Значение сброса: 0x0004 0601

Доступ: без состояния ожидания, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_SWEN	Res.	Res.
													rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	LATENCY[3:0]			
	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw

Биты 31:19 Зарезервировано, должно быть сохранено значение сброса.

Бит 18 **DBG_SWEN**: Программное обеспечение включения отладки ПО может использовать этот бит для включения/отключения отладчика.

0: отладчик отключен

1: отладчик включен

Биты 17:15 Зарезервировано, должно быть сохранено значение сброса.

Бит 14 **SLEEP_PD**: Flash Режим пониженного энергопотребления в спящем режиме или в спящем режиме с низким энергопотреблением.

Этот бит определяет, находится ли флэш-память в режиме пониженного энергопотребления или в режиме ожидания, когда устройство находится в спящем режиме или в спящем режиме с низким энергопотреблением.

0: Flash в режиме ожидания в спящем режиме и в режимах ожидания с низким энергопотреблением.

1: Flash в режиме пониженного энергопотребления в спящем режиме и в режимах сна с низким энергопотреблением.

Предостережение. Нельзя отключать flash во время выполнения программы или операции стирания.

Бит 13 **RUN_PD**: flash в режиме пониженного энергопотребления во время работы или в режиме работы с низким энергопотреблением.

Этот бит защищен от записи с помощью FLASH_PDKEYR.

Этот бит определяет, находится ли флэш-память в режиме пониженного энергопотребления или в режиме ожидания, когда устройство находится в режиме работы или в режиме работы с низким энергопотреблением. Флэш-память можно перевести в режим пониженного энергопотребления только тогда, когда код выполняется из ОЗУ. Доступ к флэш-памяти невозможен, если установлено значение RUN_PD.

0: Flash в режиме ожидания

1: Flash в режиме пониженного энергопотребления

Предостережение. Нельзя отключать flash во время выполнения программы или операции стирания.

Бит 12 **DCRST**: сброс кэша данных

0: кэш данных не сбрасывается

1: кэш данных сбрасывается

Этот бит может быть записан только тогда, когда кэш данных отключен.

Бит 11 **ICRST**: сброс кэша инструкций

0: кэш инструкций не сбрасывается

1: кэш инструкций сбрасывается

Этот бит может быть записан только тогда, когда кэш инструкций отключен.

Бит 10 **DCEN**: включение кэша данных

0: кэш данных отключен

1: кэш данных включен

Бит 9 **ICEN**: включение кэша инструкций

0: кэш инструкций отключен

1: кэш инструкций включен

Бит 8 **PRFTEN**: разрешение предварительной выборки

0: предварительная выборка отключена

1: предварительная выборка включена

Биты 7:4 Зарезервированы, должны быть сохранены в значении сброса.

Биты 3:0 **LATENCY[3:0]**: задержка

Эти биты представляют собой отношение периода SYSCLK (системных часов) ко времени доступа к флэш-памяти.

0000: нулевое состояние ожидания

0001: одно состояние ожидания

0010: Два состояния ожидания

0011: Три состояния ожидания

0100: Четыре состояния ожидания

...1111: Пятнадцать состояний ожидания

3.7.2 Регистр ключа отключения питания флэш-памяти (FLASH_PDKEYR)

Смещение адреса: 0x04

Значение сброса: 0x0000 0000

Доступ: нет состояния ожидания, доступ по словам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PDKEYR[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PDKEYR[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Биты 31:0 **PDKEYR[31:0]**: Отключение питания в режиме работы Флэш-ключ

Следующие значения должны быть записаны последовательно, чтобы разблокировать бит RUN_PD в FLASH_ACR:

PDKEY1: 0x0415 2637

PDKEY2: 0xFAFB FCFD

3.7.3 Регистр флэш-ключа (FLASH_KEYR)

Смещение адреса: 0x08

Значение сброса: 0x0000 0000

Доступ: нет состояния ожидания, доступ по словам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Биты 31:0 **KEYR[31:0]**: флэш-ключ

Следующие значения должны быть записаны последовательно, чтобы разблокировать регистр FLASH_CR, позволяющий выполнять операции программирования/стирания флэш-памяти:

КЛЮЧ1: 0x4567 0123

КЛЮЧ2: 0xCDEF 89AB

3.7.4 Регистр дополнительных ключей флэш-памяти (FLASH_OPTKEYR)

Смещение адреса: 0x0C

Значение сброса: 0x0000 0000

Доступ: нет состояния ожидания, доступ по словам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OPTKEYR[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTKEYR[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Биты 31:0 **OPTKEYR[31:0]**: дополнительный байтовый ключ

Следующие значения должны быть записаны последовательно, чтобы разблокировать регистр FLASH_OTPR, позволяющий выполнять операции программирования/стирания байтов опций:

КЛЮЧ1: 0x0819 2A3B

КЛЮЧ2: 0x4C5D 6E7F

3.7.5 Регистр состояния флэш-памяти (FLASH_SR)

Смещение адреса: 0x10

Значение сброса: 0x0000 0000

Доступ: без состояния ожидания, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSY
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTV ERR	RD ERR	Res.	Res.	Res.	Res.	FAST ERR	MISS ERR	PGS ERR	SIZ ERR	PGA ERR	WRP ERR	PROG ERR	Res.	OP ERR	EOP
rc_w1	rc_w1					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

Биты 31:17 Зарезервировано, должно быть сохранено значение сброса.

Бит 16 **BSY**: занят

Это указывает на то, что выполняется операция флэш-памяти. Устанавливается в начале операции флэш-памяти и сбрасывается по завершении операции или при возникновении ошибки.

Бит 15 **OPTVERR**: ошибка достоверности опции

Устанавливается аппаратно, когда прочитанные параметры могут не совпадать с настроенными пользователем. Если опция не была загружена должным образом, OPTVERR устанавливается снова после каждого сброса системы.

Очищается записью 1.

Бит 14 **RDERR:** ошибка чтения PCROP

Устанавливается аппаратно, когда адрес для чтения через D-bus принадлежит защищенной от чтения области флэш-памяти (защита PCROP). Прерывание генерируется, если RDERRIE установлен в FLASH_CR.

Очищается записью 1.

Биты 13:10 Зарезервировано, должно быть сохранено значение сброса.

Бит 9 **FASTERR:** Ошибка быстрого программирования

Устанавливается аппаратно, когда последовательность быстрого программирования (активируемая FSTPG) прерывается из-за ошибки (выравнивание, размер, защита от записи или потеря данных). Одновременно устанавливается соответствующий бит состояния (PGAERR, SIZERR, WRPERR или MISSERR).

Очищается записью 1.

Бит 8 **MISSERR:** Ошибка отсутствия данных быстрого программирования

В режиме быстрого программирования 32 двойных слова должны быть отправлены во флэш-память последовательно, и новые данные должны быть отправлены в логическое управление флэш-памятью, прежде чем текущие данные будут полностью запрограммированы. MISSERR устанавливается аппаратно, когда новые данные отсутствуют вовремя.

Очищается записью 1.

Бит 7 **PGSERR:** ошибка последовательности программирования

Устанавливается аппаратно, когда доступ для записи во флэш-память выполняется кодом, а PG или FSTPG ранее не были установлены. Также устанавливается аппаратно, если PROGERR, SIZERR, PGAERR, WRPERR, MISSERR или FASTERR установлены из-за предыдущей ошибки программирования.

Установите также при попытке выполнить очистку банка, когда DBANK=0.

Очищается записью 1.

Бит 6 **SIZERR:** ошибка размера

Устанавливается аппаратно, когда размер доступа составляет байт или полу-слово во время программы или быстрой последовательности программы. Разрешено только программирование двойным словом (следовательно: доступ к слову).

Очищается записью 1.

Бит 5 **PGAERR:** ошибка выравнивания программирования

Устанавливается аппаратно, когда данные для программирования не могут содержаться в одной и той же 64-битной строке флэш-памяти в случае стандартного программирования, или если происходит смена страницы во время быстрого программирования.

Очищается записью 1.

Бит 4 **WRPERR:** ошибка защиты от записи

Устанавливается аппаратно, когда адрес, который нужно стереть/запрограммировать, принадлежит защищенной от записи части (WRP, PCROP или RDP level 1) флэш-памяти.

Очищается записью 1.

Бит 3 PROGERR: ошибка программирования

Устанавливается аппаратно, когда программируемый адрес из двойного слова содержит значение, отличное от «0xFFFF FFFF FFFF FFFF» перед программированием, за исключением случаев, когда данные для записи равны «0x0000 0000 0000 0000».

Очищается записью 1.

Бит 2 Зарезервирован, должен быть установлен на значение сброса.

Бит 1 OPERR: ошибка операции

Устанавливается аппаратно, когда операция с флэш-памятью (программирование/стирание) завершается неудачно.

Этот бит устанавливается, только если разрешены прерывания по ошибке (ERRIE = 1).

Очищается записью «1».

Бит 0 EOP: Конец операции

Устанавливается аппаратно после успешного завершения одной или нескольких операций с флэш-памятью (программирование/стирание).

Этот бит устанавливается, только если разрешены прерывания по окончании операции (EOPIE = 1).

Очищается записью 1.

3.7.6 Регистр управления флэш-памятью (FLASH_CR)

Смещение адреса: 0x14

Значение сброса: 0xC000 0000

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	OPT LOCK	SEC PROT2	SEC PROT1	OBL LAUNCH	RD ERRIE	ERR IE	EOP IE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPT STRT	START
rs	rs	rs	rs	rc_w1	rw	rw	rw						rw	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MER2	Res.	Res.	Res.	BKER	Res.	PNB[6:0]						MER1	PER	PG	
rw				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Бит 31 LOCK: Блокировка FLASH_CR

Только этот бит установлен. Когда установлено, регистр FLASH_CR заблокирован. Он очищается аппаратно после обнаружения последовательности разблокировки.

В случае неудачной операции разблокировки этот бит остается установленным до следующего сброса системы.

Бит 30 OPTLOCK: Блокировка опций

Только этот бит установлен. Когда установлено, все биты, касающиеся опций пользователя в регистре FLASH_CR и странице опций, заблокированы. Этот бит сбрасывается аппаратно после обнаружения последовательности разблокировки. Бит LOCK должен быть очищен перед выполнением последовательности разблокировки для бита OPTLOCK.

В случае неудачной операции разблокировки этот бит остается установленным до следующего сброса.

- Бит 29 **SEC_PROT2**: Бит защиты защищаемой области памяти для банка 2.
Этот бит установлен для блокировки доступа к защищаемой области в банке 2. Он устанавливается программно при выходе из защищаемой области и может быть записан только один раз. В случае DBANK=0 этот бит бесполезен.
- Бит 28 **SEC_PROT1**: Бит защиты защищаемой области памяти для банка 1.
Этот бит установлен для блокировки доступа к защищенной области памяти для банка 1 (или когда DBANK=0). Он устанавливается программно при выходе из Защищаемой области и может быть записан только один раз.
- Бит 27 **OBL_LAUNCH**: Принудительная загрузка байта опции
Если установлено значение 1, этот бит принудительно перезагружает байт опции. Этот бит сбрасывается только после завершения загрузки байта опции. Запись невозможна, если установлен OPTLOCK.
- 0: Загрузка байта опций завершена
1: Запрошена загрузка опционального байта
- Бит 26 **RDERRIE**: разрешение прерывания по ошибке чтения PCROP
Этот бит разрешает генерацию прерывания, когда бит RDERR в FLASH_SR установлен в 1.
- 0: Прерывание ошибки чтения PCROP отключено
1: Прерывание ошибки чтения PCROP разрешено
- Бит 25 **ERRIE**: разрешение прерывания по ошибке
Этот бит разрешает генерацию прерывания, когда бит OPERR в FLASH_SR установлен в 1.
- 0: Прерывание по ошибке OPERR отключено
1: Прерывание по ошибке OPERR разрешено
- Бит 24 **EOPIE**: разрешение прерывания в конце операции
Этот бит разрешает генерацию прерывания, когда бит EOP в FLASH_SR установлен в 1.
- 0: Прерывание EOP отключено
1: Прерывание EOP включено
- Биты 23:19 Зарезервировано, должно быть сохранено значение сброса.
- Бит 18 **FSTPG**: Быстрое программирование
0: Быстрое программирование отключено
1: Быстрое программирование включено
- Бит 17 **OPTSTRT**: запуск модификации опций
Этот бит запускает операцию опций, если он установлен.
Этот бит устанавливается только программно и сбрасывается, когда сбрасывается бит BSY в FLASH_SR.
- Бит 16 **START**: Start
Этот бит запускает операцию стирания, если он установлен. Если биты MER1, MER2 и PER сброшены, а бит STRT установлен, может произойти непредсказуемое поведение без создания флага ошибки. Это состояние должно быть запрещено.
Этот бит устанавливается только программно и сбрасывается, когда сбрасывается бит BSY в FLASH_SR.
- Бит 15 **MER2**: массовое стирание банка 2
Этот бит запускает массовое стирание банка 2 (всех пользовательских страниц банка 2), если он установлен.

Биты 14:12 Зарезервировано, должно быть сохранено значение сброса.

Бит 11 **BKER**: стирание банка

DBANK=1

0: Банк 1 выбран для стирания страницы

1: Банк 2 выбран для стирания страницы

DBANK=0

Зарезервировано, должно быть установлено значение сброса.

Бит 10 Зарезервировано, должно быть сохранено значение сброса.

Биты 9:3 **PNB[6:0]**: Выбор номера страницы

Эти биты выбирают страницу для стирания:

00000000: страница 0

00000001: страница 1

...

11111111: стр. 255

Бит 2 **MER1**: массовое стирание банка 1

Этот бит запускает массовое стирание банка 1 (всех пользовательских страниц банка 1), если он установлен.

Бит 1 **PER**: стирание страницы

0: Стирание страницы отключено

1: Стирание страницы включено

Бит 0 **PG**: Программирование

0: Программирование флэш-памяти отключено

1: Программирование флэш-памяти включено

3.7.7 Флэш-регистр ECC (FLASH_ECCR)

Смещение адреса: 0x18

Значение сброса: 0x0000 0000

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ECCD	ECCC	ECCD2	ECCC2	Res.	Res.	Res.	ECCC IE	Res.	SYSF_ECC	BK_ECC	Res.	Res.	ADDR_ECC[18:16]		
rc_w1	rc_w1	rc_w1	rc_w1				rw		r	r			r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR_ECC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Бит 31 **ECCD**: обнаружение ECC

DBANK=1

Устанавливается аппаратно при обнаружении двух ошибок ECC (только если ECCC/ECCC2/ECCD/ECCD2 предварительно очищены). Когда этот бит установлен, генерируется NMI.

Очищается записью 1.

DBANK=0

Устанавливается аппаратно при обнаружении двух ошибок ECC в 64-битном LSB (биты 63:0) (только если ECCC/ECCC2/ECCD/ECCD2 предварительно очищены). Когда этот бит установлен, генерируется NMI.

Очищается записью 1.

Бит 30 **ЕССС**: коррекция ECC

DBANK=1

Устанавливается аппаратно при обнаружении и исправлении одной ошибки ECC (только если ЕССС/ЕССС2/ЕССД/ЕССД2 предварительно очищены). Прерывание генерируется, если установлен ЕСССІЕ.

Очищается записью 1.

DBANK=0

Устанавливается аппаратно при обнаружении и исправлении одной ошибки ECC в 64-битном LSB (биты 63:0) (только если ЕССС/ЕССС2/ЕССД/ЕССД2 предварительно очищены).

Очищается записью 1.

Бит 29 **ЕССД2**: обнаружение ЕСС2

DBANK=0

Устанавливается аппаратно при обнаружении двух ошибок ECC в 64-битном старшем разряде (биты 127:64). Этот бит установлен (только если ЕССС/ЕССС2/ЕССД/ЕССД2 предварительно очищены). Когда этот бит установлен, генерируется NMI.

Очищается записью 1.

DBANK=1

Зарезервировано, должно быть сохранено значение сброса.

Бит 28 **ЕССС2**: коррекция ECC

DBANK=0

Устанавливается аппаратно при обнаружении и исправлении одной ошибки ECC в 64-битном старшем разряде (биты 127:64). Этот бит установлен (только если ЕССС/ЕССС2/ЕССД/ЕССД2 предварительно очищены). Прерывание генерируется, если установлен ЕСССІЕ.

Очищается записью 1.

DBANK=1

Зарезервировано, должно быть сохранено значение сброса.

Биты 27:25 Зарезервировано, должно быть сохранено значение сброса.

Бит 24 **ЕСССІЕ**: разрешение прерывания коррекции ECC

0: прерывание ЕССС отключено

1: прерывание ЕССС разрешено.

DBANK=0

Этот бит разрешает генерацию прерывания, когда установлены биты ЕССС или ЕССС2 в регистре FLASH_ECCR.

DBANK=1

Этот бит разрешает генерацию прерывания, когда установлен бит ЕССС в регистре FLASH_ECCR.

Бит 23 Зарезервировано, должно быть сохранено значение сброса.

Бит 22 **SYSF_ECC**: сбой системной флэш-памяти ECC

Этот бит указывает, что исправление ошибок ECC или обнаружение двойных ошибок ECC находится в системной флэш-памяти.

Бит 21 **ВК_ECC**: отказ банка ECC

DBANK=1

Этот бит указывает, к какому банку относится коррекция ошибок ECC или обнаружение двойных ошибок ECC.

0: банк 1

1: банк 2

DBANK=0

Если SYSF_ECC равен 1, это указывает, какой банк связан с ошибкой ECC.

Если SYSF_ECC равен 0, зарезервировано, его нужно оставить очищенным.

Биты 20:19 Зарезервировано, должно быть сохранено значение сброса.

Биты 18:0 **ADDR_ECC[18:0]:** адрес ошибки ECC

DBANK=0

Этот бит указывает, к какому адресу во флэш-памяти относится коррекция ошибок ECC или обнаружение двойных ошибок ECC.

DBANK=1

Этот бит указывает, к какому адресу в банке относится коррекция ошибок ECC или обнаружение двойной ошибки ECC.

3.7.8 Регистр опций флэш-памяти (FLASH_OTPR)

Смещение адреса: 0x20

Значение сброса: 0xFXXX XXXX. Биты регистра загружаются значениями из флэш-памяти в OBL.

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRHEN	NRST_MODE [1:0]		n BOOT0	nSW BOOT0	CCM SRAM_RST	SRAM _PE	nBOOT1	DBANK	Res.	BFB2	WWDG _SW	IWGD_STDBY	IWDG_STOP	IWDG _SW
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Бит 31 Зарезервировано, должно быть сохранено значение сброса.

Бит 30 **IRHEN:** внутренний бит разрешения держателя сброса

0: Внутренние сбросы распространяются как простой импульс на выводе NRST.

1: Внутренние сбросы переводят вывод NRST в низкий уровень до тех пор, пока он не будет восприниматься как низкий уровень.

Биты 29:28 **NRST_MODE[1:0]:**

00: Зарезервировано

01: Только ввод сброса: низкий уровень на выводе NRST генерирует системный сброс, внутренний СБРОС не распространяется на вывод NSRT.

10: GPIO: стандартная функциональность площадки GPIO, возможен только внутренний СБРОС

11: Двухнаправленный сброс: вывод NRST настроен на режим ввода/вывода сброса (устаревший режим)

Бит 27 **nBOOT0:** дополнительный бит nBOOT0

0: nBOOT0 = 0

1: nBOOT0 = 1

Бит 26 **nSWBOOT0:** ПО BOOT0

0: BOOT0 берется из бита опции nBOOT0

1: BOOT0 взят с контакта PB8/BOOT0

- Бит 25 **CCMSRAM_RST**: стирание CCM SRAM при сбросе системы
0: CCM SRAM стирается при сбросе системы
1: CCM SRAM не стирается при сбросе системы
- Бит 24 **RAM_PE**: проверка четности SRAM1 и CCM SRAM включена
0: Проверка четности SRAM1 и CCM SRAM включена
1: Проверка четности SRAM1 и CCM SRAM отключена
- Бит 23 **nBOOT1**: конфигурация загрузки
Вместе с выводом BOOT0 этот бит выбирает режим загрузки из основной флэш-памяти, SRAM1 или системной памяти. См. Раздел 2.6: Конфигурация загрузки.
- Бит 22 **DBANK**:
0: Режим одного банка с шириной чтения данных 128 бит
1: Режим двойного банка с 64-битными данными
Этот бит может быть записан только тогда, когда PCROPA/V отключен.
- Бит 21 Зарезервировано, должно быть сохранено значение сброса.
- Бит 20 **BFB2**: загрузка с двумя банками
0: Загрузка с двумя банками отключена
1: Включение загрузки с двумя банками
- Бит 19 **WWDG_SW**: выбор сторожевого таймера окна
0: Аппаратный сторожевой таймер окна
1: Программный сторожевой таймер окна
- Бит 18 **WDG_STDBY**: зависание независимого сторожевого счетчика в режиме ожидания
0: независимый сторожевой счетчик заморожен в режиме ожидания.
1: Независимый сторожевой счетчик работает в режиме ожидания.
- Бит 17 **IWDG_STOP**: зависание независимого сторожевого счетчика в режиме остановки
0: Независимый сторожевой счетчик заморожен в режиме остановки.
1: Независимый сторожевой счетчик работает в режиме остановки.
- Бит 16 **IWDG_SW**: выбор независимого сторожевого таймера
0: Аппаратно-независимый сторожевой таймер
1: Независимый от программного обеспечения сторожевой таймер
- Бит 15 Зарезервирован, должен иметь значение сброса.
- Бит 14 **nRST_SHDW**
0: Сброс генерируется при входе в режим отключения.
1: Сброс не производится при входе в режим выключения.
- Бит 13 **nRST_STDBY**
0: Сброс генерируется при входе в режим ожидания.
1: Сброс не происходит при входе в режим ожидания.
- Бит 12 **nRST_STOP**
0: Сброс генерируется при входе в режим остановки
1: Сброс при входе в режим остановки не производится.
- Бит 11 Зарезервирован, должен иметь значение сброса.
- Биты 10:8 **BOR_LEV**: Уровень сброса BOR
Эти биты содержат пороговое значение уровня питания VDD, которое активирует/отпускает сброс.
000: Уровень BOR 0. Пороговое значение уровня сброса составляет около 1,7 В.

001: BOR Level 1. Пороговое значение уровня сброса составляет около 2,0 В.

010: BOR Level 2. Пороговое значение уровня сброса около 2,2 В.

011: BOR Level 3. Пороговое значение уровня сброса около 2,5 В.

100: Уровень BOR 4. Пороговое значение уровня сброса составляет около 2,8 В.

Биты 7:0 **RDP**: уровень защиты от чтения

0xAA: уровень 0, защита от чтения не активна

0xCC: уровень 2, активна защита от чтения чипа

Прочее: Уровень 1, активна защита от чтения памяти

Примечание. Позаботьтесь о настройке `PCROP_RDP` на уровне 1. Дополнительные сведения см. в разделе: *Level 1: Защита от чтения.*

3.7.9 Flash PCROP1 Регистр начального адреса (FLASH_PCROP1SR)

Смещение адреса: 0x24

Значение сброса: 0xFFFF XXXX

Биты регистра загружаются значениями из флэш-памяти в OBL.

Доступ: нет состояния ожидания, когда не выполняется никаких операций с флэш-памятью, доступ по словам.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCROP1_STRT[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:15 Зарезервировано, должно быть сохранено значение сброса.

Биты 14:0 **PCROP1_STRT[14:0]**: смещение начала области PCROP

DBANK=1

PCROP1_STRT содержит первое двойное слово области PCROP для банка1.

DBANK=0

PCROP1_STRT содержит первые 2 двойных слова области PCROP для всей памяти.

3.7.10 Flash PCROP1 Регистр конечного адреса (FLASH_PCROP1ER)

Смещение адреса: 0x28

Значение сброса: 0xX000 XXXX

Биты регистра загружаются значениями из флэш-памяти в OBL.

Доступ: нет состояния ожидания, когда не выполняется никаких операций с флэш-памятью, доступ по словам, полусловам.

Доступ к биту `PCROP_RDP` можно получить с помощью байтового доступа.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PCROP_RDP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rs															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCROP1_END[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Бит 31 **PCROP_RDP**: область PCROP сохраняется при снижении уровня RDP

Только этот бит установлен. Сбрасывается после полного массового стирания из-за смены RDP с уровня 1 на уровень 0.

0: область PCROP не стирается при снижении уровня RDP с уровня 1 до уровня 0.

1: область PCROP стирается, когда уровень RDP снижается с уровня 1 до уровня 0 (полное массовое стирание).

Биты 30:15 Зарезервировано, должно быть сохранено значение сброса.

Биты 14:0 **PCROP1_END[14:0]**: смещение конца области PCROP банка 1 DBANK=1

PCROP1_END содержит последнее двойное слово области PCROP банка 1. DBANK=0

PCROP1_END содержит последние два двойных слова первой области PCROP во всей памяти.

3.7.11 Флэш-банк 1 Адресный регистр зоны WRP A (FLASH_WRP1AR)

Смещение адреса: 0x2C

Значение сброса: 0x00XX 00XX

Биты регистра загружаются значениями из флэш-памяти в OBL.

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP1A_END[6:0]														
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP1A_STRT[6:0]														
									rw	rw	rw	rw	rw	rw	rw

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP1A_END[6:0]**: смещение конца первой области WRP «А» DBANK=1

WRP1A_END содержит последнюю страницу первой области WRP в банке1. DBANK=0

WRP1A_END содержит последнюю страницу первой области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP1A_STRT[6:0]**: начальное смещение WRP первой области «А» DBANK=1

WRP1A_STRT содержит первую страницу первой области WRP для банка1. DBANK=0

WRP1A_STRT содержит первую страницу первой области WRP для всей памяти.

3.7.12 Флэш-банк 1 Адресный регистр WRP-области B (FLASH_WRP1BR)

Смещение адреса: 0x30

Значение сброса: 0x00XX 00XX

Биты регистра загружаются значениями из флэш-памяти в OBL.

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP1B_END[6:0]														
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP1B_STRT[6:0]														
									rw	rw	rw	rw	rw	rw	rw

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP1B_END[6:0]**: смещение конца второй области WRP «B»

DBANK=1

WRP1B_END содержит последнюю страницу второй области WRP для банка 1.

DBANK=0

WRP1B_END содержит последнюю страницу второй области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP1B_STRT[6:0]**: смещение начала второй области WRP «B»

DBANK=1

WRP1B_STRT содержит последнюю страницу второй области WRP для банка 1.

DBANK=0

WRP1B_STRT содержит последнюю страницу второй области WRP для всей памяти.

3.7.13 Регистр начального адреса флэш-памяти

PCROP2 (FLASH_PCROP2SR)

Смещение адреса: 0x44

Значение сброса: 0xFFFF XXXX

Доступ: нет состояния ожидания, когда не выполняется никаких операций с флэш-памятью, доступ по словам.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCROP2_STRT[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:15 Зарезервировано, должно быть сохранено значение сброса.

Биты 14:0 **PCROP2_STRT[14:0]**: смещение начала области PCROP

DBANK=1

PCROP2_STRT содержит первое двойное слово области PCROP для банка 2.

DBANK=0

PCROP2_STRT содержит первую область PCROP из двойного слова для всей памяти.

3.7.14 Флэш-регистр регистра конечного адреса

PCROP2 (FLASH_PCROP2ER)

Смещение адреса: 0x48

Значение сброса: 0x0000 XXXX

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCROP2_END[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:15 Зарезервировано, должно быть сохранено значение сброса.

Биты 14:0 **PCROP2_END[14:0]**: смещение конца области PCROP

DBANK=1

PCROP2_END содержит последнее двойное слово области PCROP для банка2.

DBANK=0

PCROP2_END содержит последние 2 двойных слова области PCROP для всей памяти.

3.7.15 Флэш-банк 2 WRP-область адресного регистра A (FLASH_WRP2AR)

Смещение адреса: 0x4C

Значение сброса: 0x00XX 00XX

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP2A_END[6:0]														
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP2A_STRT[6:0]														
									rw	rw	rw	rw	rw	rw	rw

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP2A_END[6:0]**: смещение конца первой области WRP «А»

DBANK=1

WRP2A_END содержит последнюю страницу первой области WRP для банка 2.

DBANK=0

WRP2A_END содержит последнюю страницу третьей области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP2A_STRT[6:0]**: смещение начала первой области WRP «А»

DBANK=1

WRP2A_STRT содержит первую страницу первой области WRP для банка2.

DBANK=0

WRP2A_STRT содержит первую страницу третьей области WRP для всей памяти.

3.7.16 Флэш-банк 2 Адресный регистр WRP-области B (FLASH_WRP2BR)

Смещение адреса: 0x50

Значение сброса: 0x00XX 00XX

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	WRP2B_END[6:0]														
									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WRP2B_STRT[6:0]														
									rw	rw	rw	rw	rw	rw	rw

Биты 31:23 Зарезервировано, должно быть сохранено значение сброса.

Биты 22:16 **WRP2B_END[6:0]**: смещение конца второй области WRP «B»
DBANK=1

WRP2B_END содержит последнюю страницу второй области WRP для банка2.
DBANK=0

WRP2B_END содержит последнюю страницу четвертой области WRP для всей памяти.

Биты 15:7 Зарезервировано, должно быть сохранено значение сброса.

Биты 6:0 **WRP2B_STRT[6:0]**: смещение начала второй области WRP «B»
DBANK=1

WRP2B_STRT содержит первую страницу второй области WRP для банка2.
DBANK=0

WRP2B_STRT содержит первую страницу второй области WRP для всей памяти.

3.7.17 Флэш-регистр защищенной области банка 1 (FLASH_SEC1R)

Смещение адреса: 0x70

Значение сброса: 0xFFFF FFXX

Доступ: нет состояния ожидания, когда не выполняются операции с флэш-памятью, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK								
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SEC_SIZE1[7:0]														
								rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:17 Зарезервировано, должно быть сохранено значение сброса.

Бит 16 **BOOT_LOCK**: используется для принудительной загрузки из пользовательской области флэш-памяти.

0: загрузка на основе конфигурации pad/option бита

1: принудительная загрузка из основной флэш-памяти.

Биты 15:8 Зарезервированы, должны быть сохранены в значении сброса.

Биты 7:0 **SEC_SIZE1[7:0]**: устанавливает количество страниц, используемых в защищенной области памяти банка 1.

Защищаемая область начинается с адреса @ 0x0800 0000 и имеет размер SEC_SIZE1 * размер страницы.

Это поле можно изменить только на уровне 0.

Любая попытка модифицировать на уровне 1 автоматически терпит неудачу и не изменяет значение регистра.

3.7.18 Флэш-регистр защищенной области банка 2 (FLASH_SEC2R)

Смещение адреса: 0x74

Значение сброса: 0xFFFF FFXX

Доступ: нет состояния ожидания, когда не выполняется никаких операций с флэш-памятью, доступ по словам, полусловам и байтам.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SEC_SIZE2[7:0]														
								rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:8 Зарезервированы, должны быть сохранены в значении сброса.

Биты 7:0 **SEC_SIZE2[7:0]**: устанавливает количество страниц, используемых в защищенной области памяти банка 2.

Защищаемая область начинается с адреса @ 0x0804 0000 и имеет размер SEC_SIZE2 * размер страницы.

Когда DBANK=0, это поле бесполезно.

Это поле можно изменить только на уровне 0.

Любая попытка модифицировать на уровне 1 молча терпит неудачу и изменяет значение регистра.

Table 17. Flash interface - register map and reset values (continued)

Offset	Register	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	FLASH_WRP1BR	Res.	WRP1B_END[6:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT[6:0]													
	Reset value										X	X	X	X	X	X	X											X	X	X	X	X	X
0x44	FLASH_PCROP2SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP2_STRT[14:0]																							
	Reset value																		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x48	FLASH_PCROP2ER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP2_END[14:0]																							
	Reset value																		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x4C	FLASH_WRP2AR	Res.	WRP2A_END[6:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_STRT[6:0]													
	Reset value										X	X	X	X	X	X	X											X	X	X	X	X	X
0x50	FLASH_WRP2BR	Res.	WRP2B_END[7:0]						Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_STRT[7:0]													
	Reset value										X	X	X	X	X	X	X	X									X	X	X	X	X	X	X
0x70	FLASH_SEC1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1[7:0]															
	Reset value																	X								X	X	X	X	X	X	X	X
0x74	FLASH_SEC2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE2[7:0]															
	Reset value																									X	X	X	X	X	X	X	X

Обратитесь к разделу 2.2 на стр. 81 для адресов границ регистров.

Embedded Flash memory (FLASH) for category 2 devices

5.7.5 Регистр состояния флэш-памяти (FLASH_SR)

Смещение адреса: 0x10

Значение сброса: 0x0000 0000

Доступ: без состояния ожидания, доступ по словам, полусловам и байтам

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSY
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTV ERR	RD ERR	Res.	Res.	Res.	Res.	FAST ERR	MISS ERR	PGS ERR	SIZ ERR	PGA ERR	WRP ERR	PROG ERR	Res.	OP ERR	EOP
rc_w1	rc_w1					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

Биты 31:17 **Зарезервировано**, должно быть сохранено значение сброса.

Бит 16 **BSY**: занят

Это указывает на то, что выполняется операция флэш-памяти. Устанавливается в начале операции флэш-памяти и сбрасывается по завершении операции или при возникновении ошибки.

Бит 15 **OPTVERR**: ошибка достоверности опции

Устанавливается аппаратно, когда прочитанные параметры могут не совпадать с настроенными пользователем. Если опция не была загружена должным образом, OPTVERR устанавливается снова после каждого сброса системы.

Очищается записью 1.

Бит 14 **RDERR**: ошибка чтения PCROP

Устанавливается аппаратно, когда адрес для чтения через D-bus принадлежит защищенной от чтения области флэш-памяти (защита PCROP). Прерывание генерируется, если RDERRIE установлен в FLASH_CR.

Очищается записью 1.

Биты 13:10 **Зарезервировано**, должно быть сохранено значение сброса.

Бит 9 **FASTERR**: Ошибка быстрого программирования

Устанавливается аппаратно, когда последовательность быстрого программирования (активируемая FSTPG) прерывается из-за ошибки (выравнивание, размер, защита от записи или потеря данных). Одновременно устанавливается соответствующий бит состояния (PGAERR, SIZERR, WRPERR или MISSERR).

Очищается записью 1.

Бит 8 **MISSERR**: Ошибка отсутствия данных быстрого программирования

В режиме быстрого программирования 32 двойных слова должны быть отправлены во флэш-память последовательно, и новые данные должны быть отправлены в логическое управление флэш-памяти до того, как текущие данные будут полностью запрограммированы. MISSERR устанавливается аппаратно, когда новые данные отсутствуют вовремя.

Очищается записью 1.

Бит 7 **PGSERR**: ошибка последовательности программирования

Устанавливается аппаратно, когда доступ для записи во флэш-память выполняется кодом, а PG или FSTPG ранее не были установлены. Также устанавливается аппаратно, если PROGERR, SIZERR, PGAERR, WRPERR, MISSERR или FASTERR установлены из-за предыдущей ошибки программирования.

Очищается записью 1.

Бит 6 SIZERR: ошибка размера

Устанавливается аппаратно, когда размер доступа составляет байт или полу-слово во время программы или быстрой последовательности программы. Разрешено только программирование двойным словом (следовательно: доступ к слову).

Очищается записью 1.

Бит 5 PGAERR: ошибка выравнивания программирования

Устанавливается аппаратно, когда данные для программирования не могут содержаться в одной и той же строке 64-битной флэш-памяти в случае стандартного программирования, или если происходит смена страницы во время быстрого программирования.

Очищается записью 1.

Бит 4 WRPERR: ошибка защиты от записи

Устанавливается аппаратно, когда адрес, который нужно стереть/запрограммировать, относится к защищенной от записи части (WRP, PCROP или RDP level 1) флэш-памяти.

Очищается записью 1.

Бит 3 PROGERR: ошибка программирования

Устанавливается аппаратно, когда программируемый адрес из двойного слова содержит значение, отличное от «0xFFFF FFFF FFFF FFFF» перед программированием, за исключением случаев, когда данные для записи равны «0x0000 0000 0000 0000».

Очищается записью 1.

Бит 2 Зарезервирован, должен быть установлен на значение сброса.

Бит 1 OPERR: ошибка операции

Устанавливается аппаратно, когда операция с флэш-памятью (программирование/стирание) завершается неудачно.

Этот бит устанавливается, только если разрешены прерывания по ошибке (ERRIE = 1).

Очищается записью «1».

Бит 0 EOP: Конец операции

Устанавливается аппаратно, когда одна или несколько операций с флэш-памятью (программирование / стирание) были успешно завершены.

Этот бит устанавливается только в том случае, если разрешены прерывания по окончании операции (EOPIE = 1).

Очищается записью 1.

stm32g4xx_hal_flash.c

Периферийные функции FLASH

Интерфейс флэш-памяти управляет доступом I-кода и D-кода CPU АНВ 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 с помощью системы предварительной выборки инструкций и строк кэша.

Основные возможности FLASH:

- (+) Операции чтения флэш-памяти
- (+) Операции программирования/стирания флэш-памяти
- (+) Защита от чтения/записи
- (+) Программирование дополнительных байтов
- (+) Предварительная выборка в I-Code
- (+) 32 строки кэша 4*64 или 2*128 бит на I-Code
- (+) 8 строк кэша 4*64 или 2*128 бит на D-Code
- (+) Коррекция кода ошибки (ECC): данные во флэш-памяти представляют собой 72-битное слово (8 бит добавляются на двойное слово)

Как использовать этот драйвер

Этот драйвер предоставляет функции и макросы для настройки и программирования FLASH-памяти всех устройств STM32G4xx.

(#) Функции программирования ввода-вывода флэш-памяти:

- (++) Блокировка и разблокировка интерфейса FLASH с помощью функций HAL_FLASH_Unlock() и HAL_FLASH_Lock()
- (++) Программные функции: двойное слово и быстрая программа (программирование полного ряда)
- (++) Существует два режима программирования:
 - (+++) Режим опроса с использованием функции HAL_FLASH_Program()
 - (+++) Режим прерывания с использованием функции HAL_FLASH_Program_IT()

(#) Функции управления прерываниями и флагами:

- (++) Обработка прерываний FLASH вызовом HAL_FLASH_IRQHandler()
- (++) Функции обратного вызова вызываются после завершения операций флэш-памяти: HAL_FLASH_EndOfOperationCallback(), когда все в порядке, иначе HAL_FLASH_OperationErrorCallback()
- (++) Получить статус флага ошибки, вызвав HAL_GetError()

(#) Опциональные байты функций управления:

(++) Блокировка и разблокировка байтов опций с помощью функций HAL_FLASH_OB_Unlock() и HAL_FLASH_OB_Lock()

(++) Запустить перезагрузку байтов опций с помощью функции HAL_FLASH_Launch().

В этом случае генерируется сброс

В дополнение к этим функциям этот драйвер включает в себя набор макросов, позволяющих выполнять следующие операции:

(+) Установить задержку

(+) Включить/отключить буфер предварительной выборки

(+) Включить/отключить кеш инструкций и кеш данных

(+) Сброс кеша инструкций и кеша данных

(+) Включить/отключить отключение флэш-памяти в режимах работы с низким энергопотреблением и в спящем режиме.

(+) Включить/выключить прерывания Flash

(+) Мониторинг состояния флагов Flash

