

Современные 32-разрядные ARM-микроконтроллеры серии STM32: блок резервных данных ВКР

Олег Вальпа

Приведено описание блока резервных данных ВКР 32-разрядных ARM-микроконтроллеров серии STM32 от компании STMicroelectronics. Рассмотрены архитектура, состав и назначение регистров конфигурирования ВКР, а также примеры программ для работы с этим блоком.

ВВЕДЕНИЕ

Блок резервных данных ВКР 32-разрядных ARM-микроконтроллеров серии STM32 [1] представляет собой область данных, сохранение которых при пропадании основного питания микро-

контроллера обеспечивается за счет энергии резервной батареи. Аббревиатура ВКР является сокращением слова Backup, которое в переводе с английского означает «резервный». В англоязычных источниках аббревиатура ВКР применяется для обозначения слов Backup registers (резервные регистры), однако в отечественной технической литературе обычно используется перевод — резервные данные. Поэтому данный блок

Таблица 1. Формат регистров ВКР																																		
Сдвиг	Регистр	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	
0 × 00	Резерв	Резерв																																
0 × 04	BKP_DR1	Резерв															D[15:0]																	
	Исх.значение																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0 × 08	BKP_DR2	Резерв															D[15:0]																	
	Исх.значение																0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 0C	BKP_DR3	Резерв															D[15:0]																	
	Исх.значение																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 10	BKP_DR4	Резерв															D[15:0]																	
	Исх.значение																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 14	BKP_DR5	Резерв															DIV[15:0]																	
	Исх.значение																1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 18	BKP_DR6	Резерв															D[15:0]																	
	Исх.значение																0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 1C	BKP_DR7	Резерв															D[15:0]																	
	Исх.значение																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 20	BKP_DR8	Резерв															D[15:0]																	
	Исх.значение																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 24	BKP_DR9	Резерв															D[15:0]																	
	Исх.значение																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 28	BKP_DR10	Резерв															D[15:0]																	
	Исх.значение																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0 × 2C	BKP_RTCCR	Резерв																						ASOS	ASOE	CCO	CAL[6:0]							
	Исх.значение																							0	0	0	0	0	0	0	0	0	0	0
0 × 30	BKP_CR	Резерв																												TPAL	TPE			
	Исх.значение																													0	0			
0 × 34	BKP_CSR	Резерв																						TIF	TEF	Резерв								
	Исх.значение																							0	0					0	0	0		

назван как «блок резервных данных» или ВКР (англ.).

Регистры, расположенные в данной области, не сбрасываются ни при перезагрузке системы, ни при восстановлении подачи питания, ни при пробуждении устройства из режима ожидания Standby.

После сброса микроконтроллера доступ к регистрам области резервных данных заблокирован, и блок ВКР защищен от возможной случайной записи. Чтобы разрешить доступ по записи к регистрам этой области, необходимо выполнить определенную процедуру. Такой аппаратный способ блокировки обеспечивает защиту записанных в блоке ВКР данных.

СОСТАВ И ОСОБЕННОСТИ БЛОКА ВКР

В области резервных данных располагаются регистры, используемые для работы часов реального времени RTC, и 16-разрядные регистры резервных данных. В зависимости от модели микроконтроллера количество этих регистров может варьироваться от 10 до 42. Их удобно использовать, например, для хранения энергонезависимых данных, предназначенных для настройки системы. Фактически они являются аналогом памяти EEPROM с небольшим объемом. Конечно, в отличие от EEPROM энергетическая независимость данных в этих регистрах обеспечивается резервной

батареей, но зато количество циклов записи для них не ограничено.

Поскольку блок ВКР может питаться от резервной батареи, когда основное питание отсутствует, и процессор микроконтроллера не работает, сброс регистров, относящихся к этой области, отличается от сброса остальных регистров контроллера. Сбросить регистры этого блока можно программно, выполнив определенные процедуры, или аппаратно, подав сигнал на вывод TAMPER. Причем аппаратный сброс требуется предварительно программно разрешить.

РЕГИСТРЫ БЛОКА ВКР

Блок ВКР включает в свой состав следующие регистры:

- ВКР_DRx — группа информационных регистров;
- ВКР_RTCCR — регистр управления блоком RTC;
- ВКР_CR — регистр управления назначением многоцелевого вывода;
- ВКР_CSR — регистр управления прерыванием от многоцелевого вывода.

Регистры ВКР_DRx используются для хранения произвольных данных, сохранение которых необходимо даже после

выключения питания контроллера. Они имеют разрядность 16 бит, а их количество в зависимости от модели контроллера может достигать 42. Таким образом, в этой области можно хранить до 84 байт данных.

Регистр ВКР_RTCCR служит для управления блоком RTC и также является энергонезависимым.

Последние два регистра позволяют настроить микроконтроллер для использования многоцелевого вывода PC13-TAMPER-RTC. Можно выполнить настройки таким образом, чтобы при появлении сигнала на этом выводе выполнялся сброс регистров области резервных данных. Также можно настроить прерывание от этого события.

Формат регистров блока ВКР с названиями входящих в них разрядов представлен в таблице 1.

Рассмотрим поочередно структуру и назначение этих регистров.

Структура регистров **ВКР_DRx** приведена в таблице 2.

Все разряды этих регистров являются информационными и служат для хранения произвольных данных. Они имеют доступ по записи и чтению.

Здесь и далее способ обращения к разрядам регистров имеет следующие условные обозначения:

Таблица 2. Назначение разрядов регистров ВКР_DRx

Разряд	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Обозначение	D[15...0]															
Обращение	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw



АВТОРИЗОВАННЫЙ ДИСТРИБ'ЮТОР



Авторизований дистриб'ютор STMicroelectronics, NXP та Vishay в Україні

м. Київ, вул. Бориспільська, 9Д
тел. +38 (044) 567-44-48, (067) 219-27-86

+38 (044) 566-79-03
info@mastek.com.ua
www.mastek.com.ua

- *rw* — допускается чтение и запись разряда;
- *r* — допускается только чтение разряда;
- *w* — допускается только запись разряда.

Регистр **BKP_RTCCR** служит для управления блоком RTC. Назначение всех разрядов этого регистра приведено в таблице 3.

Разряд ASOS определяет, какой из сигналов поступает на многоцелевой вывод PC13-TAMPER-RTC. Если этот разряд имеет состояние «0» — поступает сигнальный выход RTC, а если «1» — секундный выход RTC. Этот разряд может быть сброшен только при сбросе всего блока BKP.

Разряд ASOE разрешает работу выходного сигнала, определяемого разрядом ASOS. Этот разряд также может быть сброшен только при сбросе всего блока BKP.

Разряд CCO подключает сигнал калибровки к многоцелевому выводу PC13-TAMPER-RTC. Если этот разряд имеет состояние «0» — сигнал отключен, а если «1» — подключен сигнал калибровки, представляющий собой импульсы тактовой частоты генератора RTC, поделенные на 64. Этот разряд сбрасывается при отключении питания микроконтроллера.

Разряды CAL[6:0] задают величину калибровки часов реального времени RTC. Эта величина указывает количество тактовых импульсов для RTC, которые будут проигнорированы каждые 2^{20} часов, т.е. каждые 1 048 576 часов. Это позволяет выполнить калибровку RTC, замедляющую часы с шагом 1 000 000/ 2^{20} PPM, где PPM — это одна миллионная часть величины (от англ. *Parts Per Million* — частей на миллион). С помощью разрядов CAL часы RTC могут быть замедлены от 0 до 121 PPM.

Регистр **BKP_CR** служит для управления назначением многоцелевого вывода. Назначение разрядов этого регистра приведено в таблице 4.

Разряд TRAL определяет альтернативное назначение многоцелевого вывода PC13-TAMPER-RTC. Если этот разряд имеет состояние «0», то все информационные регистры BKP сбрасываются от высокого уровня сигнала на выводе PC13-TAMPER-RTC, а если состояние «1» — от низкого уровня сигнала на том же выводе. При этом должен быть предварительно установлен в единичное состояние разряд TPE.

Разряд TPE определяет назначение многоцелевого вывода PC13-TAMPER-

Таблица 3. Назначение разрядов регистра BKP_RTCCR

Разряд	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Обозначение	Резерв						ASOS	ASOE	CCO	CAL[6...0]						
Обращение							rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Таблица 4. Назначение разрядов регистра BKP_CR

Разряд	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Обозначение	Резерв														TRAL	TRE
Обращение															rw	rw

Таблица 5. Назначение разрядов регистра BKP_CSR

Разряд	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Обозначение							TIF	TEF						TPIE	CTI	CTE
Обращение							r	r						rw	w	w

RTC. Если этот разряд имеет состояние «0», то вывод является свободно определяемым двунаправленным выводом, а если «1», то этот вывод выполняет альтернативную функцию.

Одновременная установка разрядов TRAL и TPE может привести к случайному назначению многоцелевого вывода. Поэтому рекомендуется изменять разряд TRAL только при условии, когда разряд TPE сброшен.

Регистр **BKP_CSR** служит для обслуживания прерывания от многоцелевого вывода. Назначение всех разрядов этого регистра приведено в таблице 5.

Разряд TIF представляет собой флаг прерывания сигнала TAMPER. Данный разряд устанавливается аппаратно, когда обнаружено событие сигнала TAMPER и разряд TPIE установлен. Сброс этого разряда производится записью 1 в разряд CTI, при этом сбрасывается и само прерывание. Состояния разряда означают: 0 — нет прерывания от сигнала TAMPER, 1 — произошло прерывание от сигнала TAMPER. Этот разряд сбрасывается только при системном сбросе и при пробуждении из спящего режима.

Разряд TEF представляет собой флаг события сигнала TAMPER и устанавливается аппаратно, когда обнаружено событие сигнала TAMPER. Сброс этого разряда производится записью 1 в разряд CTE. Состояния разряда означают: 0 — нет событий сигнала

TAMPER, 1 — произошло событие сигнала TAMPER.

Разряд TPIE разрешает прерывания от сигнала TAMPER. Сброс разряда в состояние «0» запрещает прерывания от сигнала TAMPER, а установка в состояние «1» — разрешает. В этом случае должен быть установлен разряд TPE регистра BKP_CR.

Разряд CTI очищает прерывание сигнала TAMPER, когда устанавливается в состояние «1». При этом он также сбрасывает флаг TIF.

Разряд CTE при установке его в состояние «1» очищает события сигнала.

Более подробное описание назначения регистров BKP можно найти в источнике [2].

ПРОГРАММИРОВАНИЕ

Для инициализации блока BKP и работы с ним необходимо выполнить следующие действия:

- разрешить тактирование и доступ к резервной области данных;
- разрешить доступ к области резервных данных для записи;
- произвести операции чтения или записи данных;
- запретить доступ к области резервных данных для записи.

Рассмотрим конкретный пример программы обращения к регистрам резервных данных BKP (см. листинг 1).

Листинг 1

```
uint16_t t; // Вспомогательная переменная t
// Разрешить тактирование управления питанием и управления резервной областью
RCC->APB1ENR |= RCC_APB1ENR_PWREN | RCC_APB1ENR_BKPEN;
t = BKP->DR4; // Читать регистр данных DR4
t++; // Увеличить прочтенное значение на 1
PWR->CR |= PWR_CR_DBP; // Разрешить запись в область BKP
BKP->DR4 = t; // Записать новое значение в регистр DR4
PWR->CR &= ~PWR_CR_DBP; // Запретить запись в область BKP
```

Листинг 2

```
//-----
// Функция инициализации блока ВКР
//-----
void Init_BKP(void)
{
    // Разрешить тактирование управления питанием и управления резервной
    // областью
    RCC->APB1ENR |= RCC_APB1ENR_PWREN | RCC_APB1ENR_BKPER;
}
//-----
// Функция табличного получения физического адреса
// данных блока ВКР по адресу смещения 1...42
//-----
uint32_t BKP_GetAdr(uint8_t adr)
{
    static uint32_t t[] =
    {BKP_BASE+0x00, BKP_BASE+0x04, BKP_BASE+0x08, BKP_BASE+0x0c,
    BKP_BASE+0x10, BKP_BASE+0x14, BKP_BASE+0x18, BKP_BASE+0x1c,
    BKP_BASE+0x20, BKP_BASE+0x24, BKP_BASE+0x28, BKP_BASE+0x40,
    BKP_BASE+0x44, BKP_BASE+0x48, BKP_BASE+0x4c, BKP_BASE+0x50,
    BKP_BASE+0x54, BKP_BASE+0x58, BKP_BASE+0x5c, BKP_BASE+0x60,
    BKP_BASE+0x64, BKP_BASE+0x68, BKP_BASE+0x6c, BKP_BASE+0x70,
    BKP_BASE+0x74, BKP_BASE+0x78, BKP_BASE+0x7c, BKP_BASE+0x80,
    BKP_BASE+0x84, BKP_BASE+0x88, BKP_BASE+0x8c, BKP_BASE+0x90,
    BKP_BASE+0x92, BKP_BASE+0x94, BKP_BASE+0x98, BKP_BASE+0x9c,
    BKP_BASE+0xa0, BKP_BASE+0xa4, BKP_BASE+0xa8, BKP_BASE+0xac,
    BKP_BASE+0xb0, BKP_BASE+0xb4, BKP_BASE+0xb8, BKP_BASE+0xbc};
    if(adr<=42) return t[adr]; // Вернуть данные
    else return 0;
}
//-----
// Функция чтения данных из блока ВКР по адресу 1...42
//-----
uint16_t BKP_Rd(uint8_t adr)
{
    return (*(__IO uint32_t*) BKP_GetAdr(adr));
}
//-----
// Функция записи данных в блок ВКР по адресу
//-----
void BKP_Wr(uint8_t adr, uint16_t data)
{
    PWR->CR |= PWR_CR_DBP; // Разрешить запись в область ВКР
    *(__IO uint16_t*)BKP_GetAdr(adr) = (uint16_t)data;
    PWR->CR &= ~PWR_CR_DBP; // Запретить запись в область ВКР
}
```

Первая команда данной программы обеспечивает подачу тактовых импульсов для ВКР. Без этого область ВКР будет недоступна. Перед записью данных в область ВКР сначала разрешается доступ к области резервных данных для записи, а после записи ВКР вновь блокируется, что позволяет защитить эти данные.

Для удобства разработки программ в листинге 2 приведен набор функций, позволяющих выполнить все необходимые операции для работы с ВКР.

Представленные готовые функции позволяют избавиться от необходимости

сти запоминания названия специальных регистров и упрощают работу с блоком ВКР.

Литература:

1. <https://www.st.com>.
2. www.st.com/web/en/resource/technical/document/reference_manual/CD00246267.pdf.

CNY

* Статья перепечатана из журнала «Современная электроника», № 3, 2014 г., с разрешения редакции, тел. +7 (495) 232-00-87, www.soel.ru

STM32CUBEMX — НОВЫЙ ГРАФИЧЕСКИЙ ГЕНЕРАТОР КОДА ДЛЯ STM32

Компания **STMicroelectronics** представила новое программное обеспечение для работы с микроконтроллерами семейства STM32. Семейство микроконтроллеров STM32 на данный момент включает в свой состав все варианты ядер: Cortex-M0, Cortex-M0+, Cortex-M3, Cortex-M4. При таком разнообразии микроконтроллеров встает вопрос об общих библиотеках, инструментарии для настройки частот тактирования, настройки периферии и генерации кода.

Основная идея программного обеспечения (ПО) STM32CubeMX как раз и заключается в общем инструменте для настройки и создания кода инициализации для микроконтроллеров STM32:

- назначение выводов с автоматическим разрешением конфликтов;
- построение дерева тактирования с динамической проверкой конфигурации;
- инициализация периферии с проверкой параметров на валидность;
- инициализация питания с оценкой результирующего потребления.

STM32CubeMX на данный момент поддерживает две серии микроконтроллеров — STM32F4 и STM32F2. До конца года компания STMicroelectronics планирует добавить поддержку всего семейства STM32. Обновление можно скачать с официального сайта STMicroelectronics или просто обновить программу.

STM32CubeMX является частью программной платформы STM32Cube™, которая разработана для облегчения труда разработчика и сокращения времени написания программ для микроконтроллеров семейства STM32. Платформа STM32Cube™ включает в себя десятки примеров базовых программ, поддерживает библиотеки USB, Ethernet, FreeRTOS, FatFS, функцию просчета потребления микроконтроллеров с батарейным питанием — STM32L0 и STM32L1. Пользователь может генерировать проекты для сред IAR Embedded Workbench, Keil uVision и Atollic TrueStudio. В будущем планируется поддержка GCC компилятора.

www.st.com