

СОДЕРЖАНИЕ

Предисловие	11
Глава 1. Введение	13
1.1. Архитектура	13
1.2. Гибкая система тактирования	14
1.3. Внутрисхемная эмуляция	14
1.4. Адресное пространство	15
1.4.1. Флэш/ПЗУ	16
1.4.2. ОЗУ	16
1.4.3. Периферийные устройства	16
1.4.4. Регистры специальных функций (SFR)	16
1.4.5. Организация памяти	16
1.5. Расширенные возможности семейства MSP430x2xx	17
Глава 2. Сброс, прерывания и режимы работы	19
2.1. Сброс и инициализация системы	19
2.1.1. Сброс по снижению напряжения питания (BOR)	20
2.1.2. Состояние устройства после сброса	21
2.2. Прерывания	21
2.2.1. Немаскируемые прерывания (NMI)	22
2.2.2. Маскируемые прерывания	25
2.2.3. Обработка прерывания	25
2.2.4. Векторы прерываний	27
2.3. Режимы работы	29
2.3.1. Вход в режимы пониженного энергопотребления и выход из них	30
2.4. Принципы программирования устройств с низким энергопотреблением	31
2.5. Подключение неиспользуемых выводов	32
Глава 3. 16-битное RISC ЦПУ MSP430	33
3.1. Введение в ЦПУ	33
3.2. Регистры ЦПУ	35
3.2.1. Счётчик команд (PC)	35
3.2.2. Указатель стека (SP)	35

3.2.3. Регистр состояния (SR)	36
3.2.4. Регистры генератора констант CG1 и CG2	37
3.2.5. Регистры общего назначения R4...R15	38
3.3. Режимы адресации	39
3.3.1. Регистровый режим адресации	39
3.3.2. Индексный режим адресации	40
3.3.3. Относительный режим адресации	41
3.3.4. Абсолютный режим адресации	42
3.3.5. Косвенный регистровый режим адресации	43
3.3.6. Косвенный регистровый режим адресации с автоинкрементом	44
3.3.7. Непосредственный режим адресации	45
3.4. Система команд	45
3.4.1. Команды с двумя операндами (формат I)	46
3.4.2. Команды с одним операндом (формат II)	46
3.4.3. Команды перехода	48
3.4.4. Время выполнения и размер команд	90
3.4.5. Описание набора команд	92
Глава 4. 16-битное RISC ЦПУ MSP430X	94
4.1. Введение в ЦПУ	94
4.2. Прерывания	96
4.3. Регистры ЦПУ	96
4.3.1. Счётчик команд (PC)	96
4.3.2. Указатель стека (SP)	97
4.3.3. Регистр состояния (SR)	99
4.3.4. Регистры генератора констант CG1 и CG2	100
4.3.5. Регистры общего назначения R4...R15	101
4.4. Режимы адресации	103
4.4.1. Регистровый режим	104
4.4.2. Индексный режим адресации	105
4.4.3. Относительный режим адресации	110
4.4.4. Абсолютный режим адресации	114
4.4.5. Косвенный регистровый режим адресации	117
4.4.6. Косвенный регистровый режим адресации с автоинкрементом	118
4.4.7. Непосредственный режим адресации	119
4.5. Команды MSP430 и MSP430X	121
4.5.1. Команды MSP430	121
4.5.2. Команды MSP430X	126
4.6. Описание набора команд	137
4.6.1. Подробные описания расширенных команд	137
4.6.2. Команды MSP430	139
4.6.3. Расширенные команды	184
4.6.4. Адресные команды	220
Глава 5. Модуль синхронизации Basic Clock Module+	230
5.1. Введение	230

5.2. Функционирование модуля синхронизации	231
5.2.1. Возможности модуля синхронизации и приложения с низким энергопотреблением	231
5.2.2. Встроенный низкочастотный генератор со сверхнизким потреблением	233
5.2.3. Генератор LFXT1	233
5.2.4. Генератор XT2	234
5.2.5. Генератор с цифровым управлением (DCO)	235
5.2.6. Модулятор DCO	236
5.2.7. Отказоустойчивая работа модуля синхронизации	237
5.2.8. Синхронизация тактовых сигналов	239
5.3. Регистры модуля синхронизации	239
Глава 6. Контроллер DMA	244
6.1. Введение	244
6.2. Функционирование контроллера DMA	246
6.2.1. Режимы адресации контроллера DMA	246
6.2.2. Режимы пересылки контроллера DMA	247
6.2.3. Инициация передачи данных с использованием DMA	250
6.2.4. Прерывание DMA-пересылок	254
6.2.5. Приоритеты каналов DMA	254
6.2.6. Длительность DMA-пересылки	254
6.2.7. Функционирование DMA и прерывания	255
6.2.8. Прерывания контроллера DMA	255
6.2.9. Использование модуля USCI_B в режиме I2C с контроллером DMA	257
6.2.10. Использование модуля ADC12 с контроллером DMA	257
6.2.11. Использование модуля DAC12 с контроллером DMA	257
6.2.12. Запись в флэш-память с использованием контроллера DMA	258
6.3. Регистры контроллера DMA	258
Глава 7. Контроллер флэш-памяти	265
7.1. Введение	265
7.2. Сегментная организация флэш-памяти	265
7.2.1. Сегмент А	267
7.3. Функционирование флэш-памяти	267
7.3.1. Тактовый генератор контроллера флэш-памяти	268
7.3.2. Стирание флэш-памяти	269
7.3.3. Запись в флэш-память	271
7.3.4. Обращение к флэш-памяти во время записи или стирания	277
7.3.5. Останов циклов записи или стирания	278
7.3.6. Режим чтения при граничных условиях	278
7.3.7. Конфигурирование контроллера флэш-памяти и организация доступа к нему	279
7.3.8. Прерывания контроллера флэш-памяти	279
7.3.9. Программирование флэш-памяти	279
7.4. Регистры контроллера флэш-памяти	281

Глава 8. Цифровые порты ввода/вывода	285
8.1. Введение	285
8.2. Функционирование цифровых портов ввода/вывода	285
8.2.1. Регистр данных входа PxIN	285
8.2.2. Регистр данных выхода PxOUT	286
8.2.3. Регистр направления PxDIR	286
8.2.4. Регистр включения подтягивающих резисторов PxREN	286
8.2.5. Регистры выбора функции PxSEL и PxSEL2	286
8.2.6. Прерывания от портов P1 и P2	287
8.2.7. Конфигурация неиспользуемых выводов портов	288
8.3. Регистры цифровых портов ввода/вывода	288
Глава 9. Супервизор напряжения питания	290
9.1. Введение	290
9.2. Функционирование супервизора	290
9.2.1. Конфигурирование супервизора	290
9.2.2. Функционирование компаратора супервизора	291
9.2.3. Изменение битов VLDx	292
9.2.4. Рабочий диапазон супервизора	292
9.3. Регистры супервизора	293
Глава 10. Сторожевой таймер	295
10.1. Введение	295
10.2. Функционирование сторожевого таймера	296
10.2.1. Счётчик сторожевого таймера	297
10.2.2. Режим сторожевого таймера	297
10.2.3. Режим интервального таймера	297
10.2.4. Прерывания сторожевого таймера	298
10.2.5. Отказоустойчивое тактирование сторожевого таймера	298
10.2.6. Функционирование в режимах пониженного энергопотребления	299
10.2.7. Примеры кода	299
10.3. Регистры сторожевого таймера	299
Глава 11. Аппаратный умножитель	302
11.1. Введение	302
11.2. Функционирование аппаратного умножителя	302
11.2.1. Регистры операндов	303
11.2.2. Регистры результата	304
11.2.3. Примеры кода	305
11.2.4. Косвенная адресация RESLO	305
11.2.5. Использование прерываний	306
11.3. Регистры аппаратного умножителя	306
Глава 12. Таймер А	307
12.1. Введение	307
12.2. Функционирование Таймера А	307

12.2.1. 16-битный таймер/счётчик	308
12.2.2. Запуск таймера	309
12.2.3. Управление режимом работы таймера	309
12.2.4. Блоки захвата/сравнения	314
12.2.5. Модуль вывода	316
12.2.6. Прерывания Таймера А	319
12.3. Регистры Таймера А	321
Глава 13. Таймер В	326
13.1. Введение	326
13.1.1. Сходства и различия с Таймером А	326
13.2. Функционирование Таймера В	328
13.2.1. 16-битный таймер/счётчик	328
13.2.2. Запуск таймера	329
13.2.3. Управление режимом работы таймера	329
13.2.4. Блоки захвата/сравнения	334
13.2.5. Модуль вывода	337
13.2.6. Прерывания Таймера В	340
13.3. Регистры Таймера В	343
Глава 14. Универсальный последовательный интерфейс	348
14.1. Введение	348
14.2. Функционирование модуля USI	351
14.2.1. Инициализация модуля USI	351
14.2.2. Генерация тактового сигнала USI	351
14.2.3. Режим SPI	352
14.2.4. Режим I ² C	354
14.3. Регистры модуля USI	358
Глава 15. Универсальный последовательный коммуникационный интерфейс: режим UART	362
15.1. Введение	362
15.2. Введение в модуль USCI: режим UART	362
15.3. Функционирование модуля USCI: режим UART	363
15.3.1. Инициализация и сброс модуля USCI	363
15.3.2. Формат символа	365
15.3.3. Форматы асинхронного обмена	365
15.3.4. Автоматическое определение скорости передачи	368
15.3.5. Кодирование и декодирование сигналов IrDA	370
15.3.6. Автоматическое обнаружение ошибок	371
15.3.7. Разрешение приёма USCI	372
15.3.8. Разрешение передачи USCI	373
15.3.9. Контроллер скорости передачи UART	374
15.3.10. Установка скорости обмена	376
15.3.11. Синхронизация при передаче	377
15.3.12. Синхронизация при приёме	378

15.3.13. Типовые скорости обмена и величины ошибок	379
15.3.14. Использование модуля USCI в режиме UART совместно с режимами пониженного энергопотребления	382
15.3.15. Прерывания модуля USCI	382
15.4. Регистры модуля USCI: режим UART	383
Глава 16. Универсальный последовательный коммуникационный интерфейс: режим SPI	392
16.1. Введение	392
16.2. Введение в модуль USCI: режим SPI	392
16.3. Функционирование модуля USCI: режим SPI	394
16.3.1. Инициализация и сброс модуля USCI	394
16.3.2. Формат символа	395
16.3.3. Режим ведущего	395
16.3.4. Режим ведомого	396
16.3.5. Разрешение обмена по интерфейсу SPI	397
16.3.6. Управление тактовым сигналом	397
16.3.7. Использование режима SPI совместно с режимами пониженного энергопотребления	398
16.3.8. Прерывания в режиме SPI	399
16.4. Регистры модуля USCI: режим SPI	401
Глава 17. Универсальный последовательный коммуникационный интерфейс: режим I²C	408
17.1. Введение	408
17.2. Введение в модуль USCI: режим I ² C	408
17.3. Функционирование модуля USCI: режим I ² C	409
17.3.1. Инициализация и сброс модуля USCI	409
17.3.2. Передача данных по шине I ² C	411
17.3.3. Режимы адресации I ² C	412
17.3.4. Режимы работы модуля I ² C	413
17.3.5. Генерация и синхронизация тактового сигнала I ² C	421
17.3.6. Использование модуля USCI в режиме I ² C совместно с режимами пониженного энергопотребления	425
17.3.7. Прерывания в режиме I ² C	425
17.4. Регистры модуля USCI: режим I ² C	427
Глава 18. Модуль операционного усилителя ОА	435
18.1. Введение	435
18.2. Функционирование модуля ОА	435
18.2.1. Операционный усилитель	437
18.2.2. Входы модуля ОА	437
18.2.3. Выход модуля ОА и организация обратной связи	437
18.2.4. Конфигурация модуля ОА	438
18.3. Регистры модулей ОА	444

Глава 19. Модуль аналогового компаратора Comparator_A+	447
19.1. Введение	447
19.2. Функционирование модуля Comparator_A+	447
19.2.1. Компаратор	447
19.2.2. Входные аналоговые ключи	448
19.2.3. Ключ замыкания входов	449
19.2.4. Выходной фильтр	450
19.2.5. Генератор опорного напряжения	450
19.2.6. Компаратор и регистр отключения порта CAPD	451
19.2.7. Прерывания компаратора	451
19.2.8. Использование компаратора для измерения сопротивления	452
19.3. Регистры модуля Comparator_A+	454
Глава 20. Модуль 10-битного АЦП ADC10	456
20.1. Введение	456
20.2. Функционирование модуля ADC10	458
20.2.1. Ядро 10-битного АЦП	458
20.2.2. Входы модуля ADC10 и мультиплексор	459
20.2.3. Генератор опорного напряжения	459
20.2.4. Автоматическое отключение	460
20.2.5. Синхронизация выборки и преобразования	461
20.2.6. Режимы преобразования	462
20.2.7. Контроллер передачи данных модуля ADC10	467
20.2.8. Использование встроенного датчика температуры	472
20.2.9. Заземление и борьба с помехами при использовании модуля ADC10	473
20.2.10. Прерывания модуля ADC10	474
20.3. Регистры модуля ADC10	475
Глава 21. Модуль 12-битного АЦП ADC12	481
21.1. Введение	481
21.2. Функционирование модуля ADC12	482
21.2.1. Ядро 12-битного АЦП	482
21.2.2. Входы модуля ADC12 и мультиплексор	484
21.2.3. Генератор опорного напряжения	485
21.2.4. Синхронизация выборки и преобразования	485
21.2.5. Сохранение результатов преобразования	488
21.2.6. Режимы преобразования	488
21.2.7. Использование встроенного датчика температуры	493
21.2.8. Заземление и борьба с помехами при использовании модуля ADC12	494
21.2.9. Прерывания модуля ADC12	495
21.3. Регистры модуля ADC12	497
Глава 22. Структура TLV	504
22.1. Введение	504
22.2. Поддерживаемые теги	504
22.2.1. Структура TLV калибровочных значений DCO	506

22.2.2. Структура TLV калибровочных значений модуля ADC12	506
22.3. Проверка целостности содержимого сегмента А	509
22.4. Анализ содержимого сегмента А	510
Глава 23. Модуль 12-битного ЦАП DAC12	511
23.1. Введение	511
23.2. Функционирование модуля ADC12	513
23.2.1. Ядро 12-битного ЦАП	513
23.2.2. Опорное напряжение модуля DAC12	514
23.2.3. Обновление состояния выхода модуля ADC12	514
23.2.4. Формат содержимого DAC12_xDAT	514
23.2.5. Калибровка смещения выходного усилителя модуля DAC12	515
23.2.6. Группирование нескольких модулей DAC12	517
23.2.7. Прерывания модуля DAC12	518
23.3. Регистры модуля DAC12.....	518
Глава 24. Модуль 16-битного АЦП SD16_A	521
24.1. Введение	521
24.2. Функционирование модуля SD16_A	523
24.2.1. Ядро АЦП	523
24.2.2. Диапазон входного аналогового сигнала и усилитель с программируемым коэффициентом усиления (PGA)	523
24.2.3. Генератор опорного напряжения	523
24.2.4. Автоматическое отключение	524
24.2.5. Выбор входного канала	524
24.2.6. Параметры аналогового входа	525
24.2.7. Цифровой фильтр	526
24.2.8. Регистр данных SD16MEM0	530
24.2.9. Режимы преобразования	531
24.2.10. Использование встроенного датчика температуры	532
24.2.11. Обработка прерываний	533
24.3. Регистры модуля SD16_A.....	534
Глава 25. Встроенный модуль эмуляции ЕЕМ.	539
25.1. Введение	539
25.2. Функциональные узлы модуля ЕЕМ	541
25.2.1. Триггеры	541
25.2.2. Секвенсор триггеров	541
25.2.3. Внутренний буфер трассировки	542
25.2.4. Управление тактовыми сигналами	542
25.3. Конфигурации модуля ЕЕМ	542

ПРЕДИСЛОВИЕ

Об этой книге

В настоящей книге рассматриваются модули и периферийные устройства микроконтроллеров семейства MSP430х2хх. В каждой главе приводится обобщённый обзор отдельного модуля или периферийного устройства. Не все модели семейства обладают полным набором функций и возможностей того или иного модуля или периферийного устройства. Кроме того, в различных семействах микроконтроллеров одни и те же модули и периферийные устройства могут быть реализованы по-разному. Некоторые модули в отдельных моделях или во всём семействе могут быть реализованы не в полном объёме.

Назначение выводов, подключение внутренних сигналов и рабочие параметры отличаются от устройства к устройству. Для получения более точной информации пользователю необходимо изучить справочную документацию на конкретный микроконтроллер.

Дополнительная документация

Дополнительную информацию по рассматриваемой теме можно найти на сайте <http://www.ti.com/msp430>.

Принятые обозначения

Примеры кода набраны особым шрифтом.

Глоссарий

ACLK (Auxiliary CLocK) — вспомогательный тактовый сигнал
ADC (Analog-to-Digital Converter) — аналого-цифровой преобразователь, АЦП
BOR (Brown-Out Reset) — сброс по снижению напряжения питания
BSL (BootStrap Loader) — начальный загрузчик
CPU (Central Processing Unit) — центральный процессор, ЦПУ
DAC (Digital-to-Analog Converter) — цифро-аналоговый преобразователь, ЦАП
DCO (Digitally Controlled Oscillator) — генератор с цифровым управлением
dst (Destination) — получатель
FLL (Frequency Locked Loop) — система автоматической подстройки частоты
GIE (General Interrupt Enable) — общее разрешение прерываний

INT($N/2$) (Integer portion of $N/2$) — целая часть от $N/2$
 I/O (Input/Output) — ввод/вывод
 ISR (Interrupt Service Routine) — процедура обработки прерывания
 LSB (Least-Significant Bit) — младший значащий бит
 LSD (Least-Significant Digit) — младший значащий разряд
 LPM (Low-Power Mode) — режим пониженного энергопотребления
 MAB (Memory Address Bus) — шина адреса
 MCLK (Master CLoCK) — основной тактовый сигнал
 MDB (Memory Data Bus) — шина данных
 MSB (Most-Significant Bit) — старший значащий бит
 MSD (Most-Significant Digit) — старший значащий разряд
 NMI (Non-Maskable Interrupt) — немаскируемое прерывание
 PC (Program Counter) — счётчик команд
 POR (Power-On Reset) — сброс по включению питания
 PUC (Power-Up Clear) — очистка по включению питания
 RAM (Random Access Memory) — оперативное запоминающее устройство, ОЗУ
 SCG (System Clock Generator) — генератор системного тактового сигнала
 SFR (Special Function Register) — регистр специальных функций
 SMCLK (Sub-system Master CLoCK) — дополнительный тактовый сигнал
 SP (Stack Pointer) — указатель стека
 SR (Status Register) — регистр состояния
 src (Source) — источник
 TOS (Top-of-Stack) — вершина стека
 WDT (WatchDog Timer) — сторожевой таймер

Соглашения по обозначению битов регистров

Формат всех регистров приводится с ключом, обозначающим доступность каждого отдельного бита и его исходное состояние.

Доступность бита регистра и его начальное состояние

Ключ	Доступность бита
rw	Чтение/запись
r	Только чтение
r0	Читается как 0
r1	Читается как 1
w	Только запись
w0	Записывается как 0
w1	Записывается как 1
(w)	Бит в регистре не реализован; запись 1 приводит к формированию импульса. Всегда читается как 0
h0	Сбрасывается аппаратно
h1	Устанавливается аппаратно
—0, —1	Состояние после сигнала PUC
—(0), —(1)	Состояние после сигнала POR

ГЛАВА 1

ВВЕДЕНИЕ

В этой главе описывается архитектура MSP430.

1.1. Архитектура

Микроконтроллеры семейства MSP430 имеют фон-неймановскую архитектуру (**Рис. 1.1**) и содержат 16-битное RISC ЦПУ, периферийные модули, а также гибкую систему тактирования, объединённые общими шинами адреса (MAB) и данных (MDB). Сочетание современного ЦПУ и отображаемых в памяти аналоговых и цифровых периферийных модулей делает семейство MSP430 пригодным для работы в приложениях, связанных с обработкой смешанных сигналов.

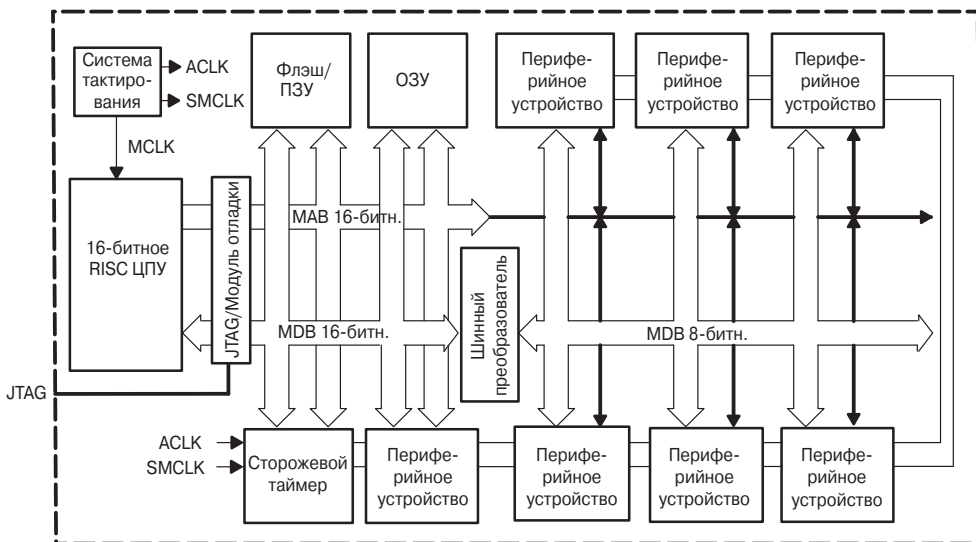


Рис. 1.1. Архитектура MSP430.

- Отличительные характеристики микроконтроллеров семейства MSP430x2xx:
- Архитектура со сверхнизким потреблением, позволяющая увеличить время работы при питании от батарей:
 - ток сохранения содержимого ОЗУ — не более 0.1 мкА;
 - ток потребления в режиме часов реального времени — не более 0.8 мкА;
 - ток потребления в активном режиме — 250 мкА/MIPS.
 - Высокоэффективная аналоговая подсистема, позволяющая выполнять точные измерения:
 - таймеры, управляемые компаратором, для измерения сопротивления резистивных элементов.
 - 16-битное RISC ЦПУ:
 - большой регистровый файл устраняет ограничения рабочего регистра;
 - произведённое по меньшему техпроцессу ядро позволяет снизить потребление и уменьшает стоимость кристалла;
 - оптимизировано для современных языков программирования высокого уровня;
 - набор команд состоит всего из 27 инструкций; поддерживается 7 режимов адресации;
 - векторная система прерываний с расширенными возможностями.
 - Флэш-память с возможностью внутрисхемного программирования позволяет гибко изменять программный код (в том числе, во время эксплуатации), а также производить сохранение данных.

1.2. Гибкая система тактирования

Система тактирования разработана специально для применения в устройствах с батарейным питанием. Низкочастотный вспомогательный тактовый сигнал ACLK формируется обычным «часовым» кварцем частотой 32 кГц. Сигнал ACLK может использоваться для периодического «пробуждения» часов реального времени, работающих в фоновом режиме. Встроенный высокочастотный генератор с цифровым управлением (DCO) может формировать основной тактовый сигнал (MCLK), используемый ЦПУ и быстродействующими периферийными модулями. Время выхода на режим этого генератора составляет менее 2 мкс при частоте 1 МГц. Решения на базе микроконтроллеров MSP430 эффективно используют высокопроизводительное 16-битное RISC ЦПУ в течение очень коротких интервалов времени:

- низкочастотный вспомогательный тактовый сигнал используется для реализации режима ожидания со сверхнизким потреблением;
- высокочастотный основной тактовый сигнал используется для эффективной обработки сигналов.

1.3. Внутрисхемная эмуляция

В составе микроконтроллеров имеется специальный модуль внутрисхемной эмуляции, доступ к которому осуществляется по интерфейсу JTAG без использования дополнительных системных ресурсов.

Преимущества внутрисхемной эмуляции:

- создание и отладка кода программы с возможностью его выполнения в реальном времени;
- поддержка точек останова и выполнения программы в пошаговом режиме;
- отлаживаемый объект имеет те же характеристики, что и конечное устройство;
- сохраняется целостность смешанных сигналов благодаря отсутствию взаимного влияния проводов.

1.4. Адресное пространство

Семейство MSP430 имеет фон-неймановскую архитектуру с единым адресным пространством, которое разделено между регистрами специальных функций (SFR), периферийными устройствами, ОЗУ и флэш-памятью в соответствии с **Рис. 1.2**. Подробное распределение памяти для конкретной модели микроконтроллера можно узнать из соответствующей документации. Обращение к исполняемому коду всегда выполняется по чётным адресам. Доступ к данным может осуществляться как побайтно, так и пословно. В настоящее время общий объём адресуемой памяти составляет 128 КБ.

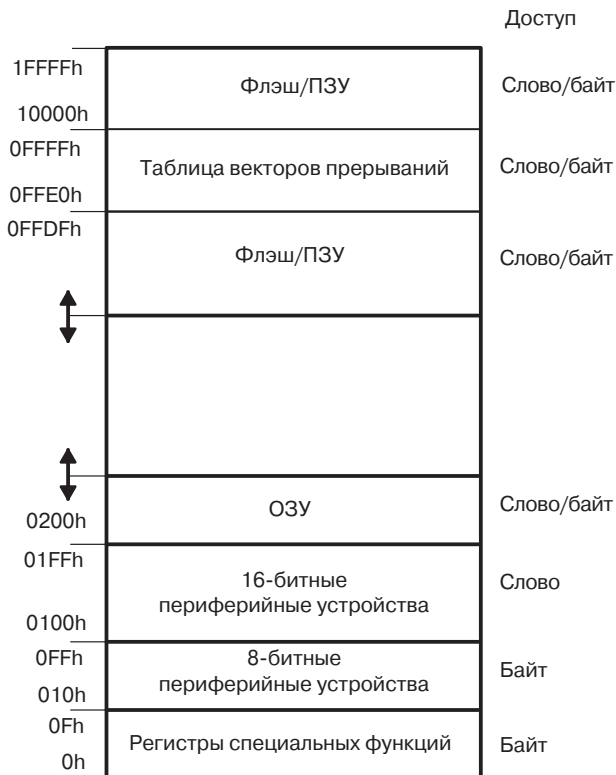


Рис. 1.2. Карта памяти.

1.4.1. Флэш/ПЗУ

Начальный адрес области флэш/ПЗУ зависит от объёма этой памяти и отличается для разных устройств. Конечный адрес области флэш/ПЗУ всегда равен 0x1FFFF. Флэш-память может использоваться как для хранения кода, так и для хранения данных. Двухбайтные и однобайтные данные (или таблицы данных) могут располагаться во флэш-памяти и использоваться непосредственно оттуда, без предварительного копирования в ОЗУ.

Таблица векторов прерываний занимает верхние 16 слов нижней области памяти размером 64 КБ. При этом вектор прерывания с наивысшим приоритетом располагается в последнем слове области по адресу 0x1FFFF.

1.4.2. ОЗУ

Область ОЗУ начинается с адреса 0200h. Конечный адрес области зависит от объёма ОЗУ и меняется от модели к модели. ОЗУ может использоваться как для хранения данных, так и для хранения программного кода.

1.4.3. Периферийные устройства

Регистры периферийных модулей (устройств) располагаются в общем адресном пространстве. Область адресов от 0100h до 01FFh зарезервирована для 16-битных периферийных модулей. Для обращения к таким устройствам необходимо использовать команды, оперирующие двухбайтными значениями. При использовании команд, работающих с однобайтными значениями, допускаются обращения только к чётным адресам памяти, при этом старший байт результата всегда будет равен нулю.

Область адресов от 010h до 0FFh зарезервирована для 8-битных периферийных модулей. Для обращения к этим устройствам необходимо использовать команды, оперирующие байтами. Если для чтения из такого модуля использовать команду, оперирующую словами, то содержимое старшего байта результата будет неопределённым. При записи в 8-битный модуль двухбайтного значения, в регистр устройства будет записан только младший байт.

1.4.4. Регистры специальных функций (SFR)

Некоторые функции периферийных устройств конфигурируются посредством регистров специальных функций. Эти 8-битные регистры располагаются в младших 16 байт адресного пространства. Для обращения к указанным регистрам можно использовать только команды, оперирующие байтами. Назначение отдельных битов регистров специальных функций описано в документации на конкретные модели.

1.4.5. Организация памяти

Однобайтные значения располагаются по чётным или нечётным адресам. Двухбайтные значения располагаются только по чётным адресам, как показано

на **Рис. 1.3**. При использовании команд, оперирующих словами, обращаться можно только к чётным адресам памяти. Младший байт двухбайтного значения всегда располагается по чётному адресу, а старший байт — по следующему нечётному адресу. Например, если слово данных расположено по адресу xxx4h, то младший байт значения находится по адресу xxx4h, а старший байт значения — по адресу xxx5h.



Рис. 1.3. Биты, байты и слова в памяти с побайтовой организацией.

1.5. Расширенные возможности семейства MSP430x2xx

В **Табл. 1.1** перечислены различные усовершенствования, появившиеся в микроконтроллерах семейства MSP430x2xx. Эти усовершенствования описываются в последующих главах книги или же, если улучшение касается параметров устройства, приводятся в документации на конкретные модели.

Таблица 1.1. Усовершенствования семейства MSP430x2xx

Подсистема	Усовершенствование
Сброс	Сброс по снижению напряжения питания реализован во всех моделях MSP430x2xx. В регистр IFG1 добавлены флаги PORIFG и RETIFG, показывающие причину сброса. Выборка команды из диапазона адресов 0x0000...0x01FF вызывает сброс устройства
Сторожевой таймер	Во всех моделях MSP430x2xx реализован модуль усовершенствованного сторожевого таймера (WDT+). Гарантируется бесперебойная генерация тактового сигнала для этого таймера

Таблица 1.1. Усовершенствования семейства MSP430x2xx (продолжение)

Подсистема	Усовершенствование
Базовая система синхронизации	<p>Генератор LFXT1 в режиме LF имеет встроенные нагрузочные конденсаторы с программируемой ёмкостью.</p> <p>Генератор LFXT1 в режиме HF позволяет использовать кварцевые резонаторы частотой до 16 МГц.</p> <p>Генератор LFXT1 в режиме LF имеет схему обнаружения сбоев.</p> <p>Выходы XIN и XOUT в 20- и 28-выводных моделях имеют дополнительные функции.</p> <p>Некоторые модели не поддерживают использование внешнего R_{OSC} с DCO. В этом случае младший бит регистра DCSCCTL2 всегда должен быть сброшен. За подробной информацией обращайтесь к документации на конкретный микроконтроллер.</p> <p>Значительно увеличена рабочая частота DCO.</p> <p>Значительно улучшена температурная стабильность DCO</p>
Флэш-память	<p>Информационная секция памяти содержит 4 сегмента по 64 байт.</p> <p>Сегмент А индивидуально блокируется битом LOCKA.</p> <p>Вся информация может быть защищена от группового стирания битом LOCKA.</p> <p>Стирание сегмента может приостанавливаться на время прерывания.</p> <p>Обновление флэш-памяти может быть отменено прерыванием.</p> <p>Напряжение программирования флэш-памяти снижено до 2.2 В.</p> <p>Уменьшена длительность операций записи/стирания.</p> <p>Сбой в работе тактового генератора отменяет обновление флэш-памяти</p>
Цифровые порты ввода/вывода	<p>Все порты имеют встроенные подтягивающие (pullup/pulldown) резисторы.</p> <p>В 20- и 28-выводных моделях добавлены линии ввода/вывода P2.6 и P2.7 в качестве альтернативных функций выводов XIN и XOUT. Если используется кварцевый резонатор, биты P2SELx для этих линий порта всегда должны быть установлены</p>
Компаратор А	<p>Количество входов, подключаемых к компаратору, увеличено за счет использования нового входного мультиплексора</p>
Энергопотребление	<p>Типовой потребляемый ток в режиме LPM3 снижен почти на 50% при напряжении питания 3 В.</p> <p>Значительно уменьшено время выхода DCO на режим</p>
Рабочая частота	<p>Максимальная рабочая частота увеличена до 16 МГц при напряжении питания 3.3 В</p>
Начальный загрузчик	<p>Некорректный пароль приводит к стиранию всей флэш-памяти.</p> <p>Усложнена процедура входа в загрузчик для предотвращения случайного стирания флэш-памяти</p>

СБРОС, ПРЕРЫВАНИЯ И РЕЖИМЫ РАБОТЫ

В этой главе описываются подсистема сброса, прерывания и режимы работы MSP430x2xx.

2.1. Сброс и инициализация системы

Схема сброса, показанная на **Рис. 2.1**, формирует сигналы сброса (POR) и очистки (PUC) по включению питания. Генерация данных сигналов вызывается различными событиями, при этом исходное состояние устройства зависит от того, какой из сигналов был сгенерирован.

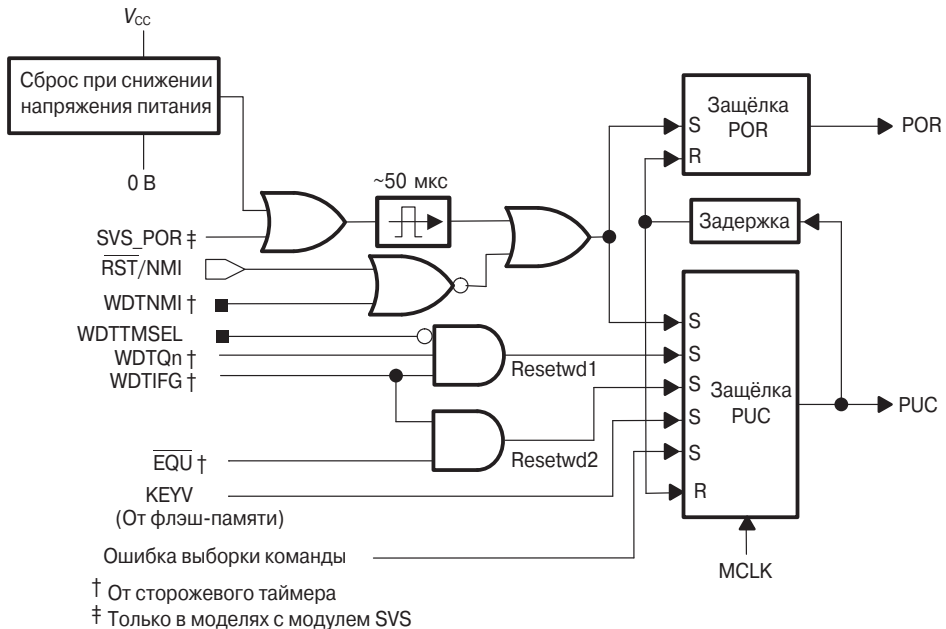


Рис. 2.1. Подсистема сброса.

Сигнал POR является сигналом сброса устройства. Этот сигнал генерируется при наступлении любого из следующих событий:

- включение устройства;
- сигнал НИЗКОГО уровня на выводе $\overline{\text{RST}}/\text{NMI}$, если последний сконфигурирован как вход сброса;
- низкий уровень напряжения питания при $\text{PORON} = 1$.

Сигнал PUC всегда генерируется при появлении сигнала POR, однако генерация последнего никак не связана с сигналом PUC. Сигнал PUC генерируется при наступлении любого из следующих событий:

- генерация сигнала POR;
- тайм-аут сторожевого таймера (только в соответствующем режиме);
- обращение к сторожевому таймеру с неверным ключом;
- обращение к контроллеру флэш-памяти с неверным ключом;
- попытка выборки команды из памяти в диапазоне адресов $0h...01Fh$.

2.1.1. Сброс по снижению напряжения питания (BOR)

Модуль BOR отслеживает изменения напряжения питания на выводе V_{CC} . Этот модуль сбрасывает устройство, вызывая генерацию сигнала POR при подаче и снятии напряжения питания. Диаграммы работы модуля BOR показаны на Рис. 2.2.

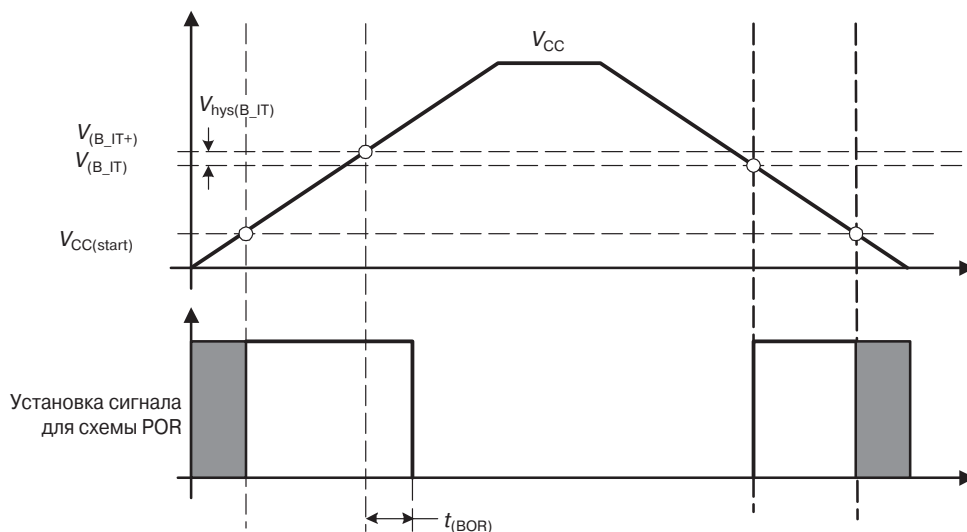


Рис. 2.2. Временные диаграммы работы модуля BOR.

Сигнал POR становится активным при переходе напряжением V_{CC} порогового значения $V_{CC(start)}$. Сигнал сброса удерживается в активном состоянии до тех пор, пока напряжение V_{CC} не превысит значения $V_{(B_IT+)}$ и не истечёт время $t_{(BOR)}$. Из-за наличия гистерезиса $V_{hys(B_IT-)}$ повторная активация сигнала сброса модулем BOR происходит при снижении напряжения питания ниже уровня $V_{(B_IT-)}$.

Поскольку величина $V_{(B_IT-)}$ намного больше уровня срабатывания схемы POR (V_{min}), модуль BOR обеспечивает сброс устройства при таких неисправностях источника питания, при которых его напряжение не опускается ниже уровня V_{min} . Точные значения параметров модуля BOR приводятся в документации на конкретные модели.

2.1.2. Состояние устройства после сброса

После снятия сигнала POR микроконтроллер переходит в следующее состояние:

- вывод \overline{RST}/NMI конфигурируется как вход сброса;
- все линии портов ввода/вывода конфигурируются как входы в соответствии с описанием, приведенным в главе 8 «Цифровые порты ввода/вывода»;
- прочие периферийные модули и регистры инициализируются так, как описано в соответствующих главах книги;
- регистр состояния (SR) сбрасывается;
- сторожевой таймер включается в сторожевом режиме;
- в счётчик команд (PC) загружается значение, находящееся по адресу вектора сброса (0FFFEh). Если по этому адресу записано 0FFFFh, то устройство отключается для минимизации потребления.

Инициализация программы

После сброса устройства пользовательская программа должна инициализировать его в соответствии с требованиями конкретного приложения. Процесс инициализации должен включать следующие этапы:

- инициализация указателя стека SP (как правило, указатель устанавливается на вершину ОЗУ);
- инициализация сторожевого таймера в соответствии с требованиями приложения;
- конфигурирование периферийных модулей в соответствии с требованиями приложения.

Дополнительно можно проверить состояние флагов сторожевого таймера, сбоя тактового генератора и флэш-памяти для определения причины сброса.

2.2. Прерывания

Приоритеты прерываний фиксированы и зависят от местонахождения конкретного модуля в цепочке, как показано на **Рис. 2.3**. Чем ближе расположен модуль к ЦПУ, тем выше приоритет его прерывания. Приоритеты определяют порядок обработки прерываний при одновременной генерации нескольких запросов.

В MSP430 имеется три типа прерываний:

- сброс системы;
- немаскируемые (NMI);
- маскируемые.

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

e-Univers.ru