

Переход с микроконтроллеров MSP430F1xx на MSP430F2xx

Алексей ПАНТЕЛЕЙЧУК
pantel@compel.ru

Сравнение микроконтроллеров семейства MSP430F1xx с MSP430F2xx

Микроконтроллеры MSP430F2xx построены на базе семейства MSP430F1xx и являются их улучшенной версией, они обеспечивают большую производительность, меньшее энергопотребление, дополнительную функциональность. Переход на новое семейство позволяет улучшить характеристики приложения и уменьшить его стоимость. В таблице 1 представлены отличительные характеристики семейств.

Несмотря на то, что многие представители нового семейства микроконтроллеров могут рассматриваться как совместимые по выводам прямые замены микроконтроллерам старого семейства, существует несколько моментов, на которые стоит обратить особое внимание. Зная эти моменты, разработчику будет легче перевести приложение на новый микроконтроллер, получив, таким образом, преимущества, обозначенные в таблице 1.

Переход с MSP430F1xx на MSP430F2xx: аппаратная часть

Совместимость по корпусу и выводам

Больше всего повезло разработчикам, использующим MSP430F16x и MSP430F13x/14x в 64-выводном корпусе LQFP, так как они могут напрямую заменить эти микроконтроллеры на MSP430F261x и MSP430F23x/24x соответственно. Как корпус, так и площадь, занимаемая на печатной плате (PCB foot-print), идентичны.

Все выводы MSP430F23x/24x могут быть использованы для тех же целей, что и выводы MSP430F13x/14x (выводы аналоговых и цифровых модулей, источников питания, JTAG-программатора), что позволяет осуществить переход на новое семейство без изменения печатной платы. Это утверждение справедливо для MSP430F16x и MSP430F261x с одним исключением. Если в приложении используется последовательный интерфейс USART микроконтроллера MSP430F16x в режиме I²C,

Цель настоящей статьи показать целесообразность перехода с микроконтроллеров MSP430F1xx на MSP430F2xx, а также облегчить сам переход, обозначив основные отличия двух семейств с точки зрения аппаратной и программной части.

прямую замену на MSP430F261x осуществить не получится, из-за несовместимости по выводам (табл. 2). Подробное описание выводов можно найти в техническом описании конкретного микроконтроллера.

Потребляемый ток

При переходе на MSP430F2xx разработчик получает выигрывает по энергопотреблению. Так, например, в режиме ожидания при синхронизации от 32-кГц осциллятора (LFXT1) потребляемый ток MSP430F261x составляет 1,1 мкА (3 В, 25 °С), в то время как для MSP430F16x это значение составляет от 2 до 2,6 мкА при выполнении тех же функций. Таким образом, перейдя на новый микроконтроллер, разработчик может в несколько раз увеличить жизнь батарейки (если в приложении такая используется).

Потребляемый ток в активном режиме старого и нового семейств при работе на одной и той же частоте, температуре и напряжении примерно одинаков. Но так как потребляе-

мый ток в активном режиме линейно зависит от частоты работы микроконтроллера, а максимальная тактовая частота MSP430F2xx увеличена в два раза, устройство из нового семейства может потреблять дополнительный ток.

Тактовая частота и напряжение питания

Максимальная частота работы MSP430 зависит от напряжения питания. Эта зависимость приводится в документации для каждого устройства. В общем, зависимости максимальной частоты от напряжения питания для MSP430F1xx и MSP430F2xx различаются, но MSP430F2xx всегда может работать на той же частоте, что и MSP430F1xx, при одном и том же напряжении питания. Однако, если разработчик, переходящий на MSP430F2xx, хочет увеличить тактовую частоту ядра, очень важно внимательно изучить зависимость частоты ядра микроконтроллера от напряжения питания

Таблица 1. MSP430F1xx в сравнении с MSP430F2xx

	MSP430F1xx	MSP430F2xx
Максимальная частота ядра, МГц	8	16
Время выхода из режимов низкого энергопотребления, мкс	<6	<1
Энергопотребление в режиме LPM3, мкА	<2	<1
Поддержка функции сброса микроконтроллера при снижении напряжения питания ниже допустимого уровня	Некоторые	Все
Минимальное напряжение питания микроконтроллера при программировании Flash	2,7 В	2,2 В
Встроенные подтягивающие резисторы на портах ввода/вывода	Нет	Все порты
Неравномерность частоты высокочастотного осциллятора при дрейфе температуры и напряжения, %	±20	±2
Обнаружение неисправностей осциллятора	Высокочастотный	Высокочастотный и низкочастотный
Дополнительный встроенный низкочастотный осциллятор с низким энергопотреблением	Нет	12 кГц
Фильтр пульсаций сигнала синхронизации	Нет	Есть
Конфигурируемые встроенные нагрузочные конденсаторы кварцевого резонатора	Нет	Есть
Детектирование обращения к несуществующему адресу	Нет	Есть
Сторожевой таймер, устойчивый к выходу из строя осциллятора	Нет	Есть
Защищенность загрузчика от взлома	256-битный пароль	256-битный пароль, функция стирания
Flash-память, кбайт	до 60	до 120
ОЗУ, кбайт	до 10	до 8
Температурный диапазон, °С	-40...+85	-40...+105

Таблица 2. Изменения в выводах интерфейса I²C при переходе

Сигналы I ² C	Вывод/сигнал F16x	Вывод/сигнал F261x	Комментарии
SDA	вывод 29 (P3.1/SIM00/SDA)	вывод 29 (P3.1/UCB0SIM0/UCB0SDA)	Нет конфликта
SCL	вывод 31 (P3.3/UCLK0/SCL)	вывод 30 (P3.2/UCB0SOM1/UCB0SCL)	Меняется вывод

Особенно рекомендуется рассмотреть эти зависимости, если необходимо получить линейное изменение мощности. Напомним, что MSP430F1xx и MSP430F2xx содержат встроенный супервизор напряжения питания (SVS), который может помочь в решении этой задачи.

Документ об ошибках (Device Errata)

При переходе с MSP430F1xx на MSP430F2xx очень важно взглянуть в документ Errata Sheet, в котором собраны найденные ошибки микроконтроллера и предложены методы их обхода, так как этим новые семейства тоже могут отличаться друг от друга.

Переход с MSP430F1xx на MSP430F2xx: программная часть

В этой части статьи мы обсудим важные моменты относительно программного обеспечения при переходе с MSP430F1xx на MSP430F2xx. Несмотря на то, что оба семейства программно совместимы, в большинстве случаев при переходе на новый микроконтроллер нельзя просто загрузить бинарный код MSP430F1xx в MSP430F2xx.

В общем, потребуется перекомпоновать код приложения на уровне исходных кодов с использованием необходимых заголовочных файлов, библиотек и соответствующего командного файла компоновщика. Командные файлы компоновщика находятся в папке установки среды разработки и имеют расширение CMD (для Code Composer Essential) и XCL (для IAR Embedded Workbench).

Рассмотрим эти аспекты более подробно.

ЦПУ и система памяти

Расширенная архитектура памяти

Семейство MSP430F2xx построено на основе архитектуры MSP430X с улучшенным ЦПУ и системой памяти. Основное улучшение заключается в расширении разрядности адресной шины до 20 бит, что позволяет получить 1 Мбайт адресного пространства без использования страниц. Архитектура ЦПУ MSP430X обратно совместима с ЦПУ MSP430 микроконтроллеров MSP430F1xx.

Для использования адресного пространства более 64 кбайт нужно использовать команды CALLA и RETA (вместо CALL и RET), для того, чтобы адрес возврата из подпрограмм правильно сохранялся и извлекался как 20-битное число. В случае перекомпилирования программного кода на языке высокого уровня, такого как C, компилятор делает это автоматически. При переводе ассемблерного кода на новую архитектуру это нужно будет сделать вручную.

Существуют команды, изменяющие значение PC в соответствии с используемым режимом адресации. Например, команда MOV.W # value. PC сотрет старшие четыре

бита счетчика, так как это команда работает с 16-битными данными. Поэтому следует также пересмотреть команды, обращающиеся к PC напрямую. Опять же, при компиляции кода на языке высокого уровня это произойдет автоматически.

У большинства микроконтроллеров нового семейства увеличен объем ОЗУ. Чтобы воспользоваться этим преимуществом, нужно будет переписать программный код при переходе со старого семейства.

Передача параметров подпрограмм и стековый фрейм

Для обмена данными с функциями компилятор обычно использует регистры ЦПУ. В случае, когда все данные не умещаются в регистры, они размещаются в стеке перед тем, как вызывается подпрограмма. В микроконтроллерах с большим адресным пространством MSP430F2xx вызов подпрограмм осуществляется с помощью команды CALLA, которая использует дополнительное 16-битное слово в стеке для размещения значения 20-битного счетчика команд (рис. 1).

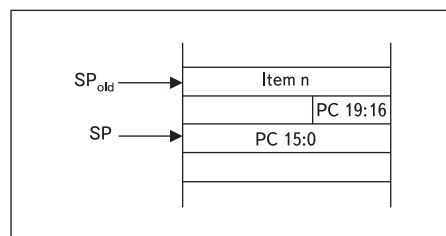


Рис. 1. Сохранение адреса возврата в стеке командой CALLA

Если подпрограмме нужно получить доступ к данным в стеке, необходимо учесть, что счетчик команд занимает дополнительное слово. При переносе приложения с MSP430F1xx на MSP430F2xx следует пересмотреть фрагменты кода, где происходит обращение к стеку.

Если осуществляется переход с MSP430F1xx на MSP430F2xx параллельно с обновлением версии компилятора, нужно иметь в виду, что в новой версии может измениться способ вызова подпрограмм, как, например, при переходе с IAR Embedded Workbench V2.xx/V3.xx на V4.xx.

Оптимизация кода по скорости выполнения

При переносе кода, который содержал оптимизированные вручную критичные к скорости выполнения участки, следует учитывать такой фактор: время выполнения некоторых команд ЦПУ сократилось на один или два цикла.

Другим улучшением архитектуры MSP430X является сокращение времени реакции на прерывание (пять циклов MCLK вместо шести) и времени возврата из прерывания (три цикла вместо пяти).

Объем ОЗУ

Микроконтроллер MSP430F1611 содержит 10 кбайт ОЗУ, в то время как максимальный объем ОЗУ MSP430F261x составляет 8 кбайт. Таким образом, если в приложении задействованы все 10 кбайт ОЗУ MSP430F1611, для того, чтобы перейти на MSP430F261x, потребуется переработка программного кода.

Таблица векторов прерывания MSP430F261x занимает 32 слова памяти, в то время как в MSP430F16x — только 16. Область памяти MSP430F261x с адресом 0xFFBE зарезервирована для специальных задач загрузчика.

Информационная Flash-память

Микроконтроллеры MSP430F1xx и MSP430F2xx содержат 256 байт информационной Flash-памяти, расположенной по адресам от 0x1000 до 0x10FF. Хотя объем этого сектора обоих семейств одинаков, его организация различна. Информационная память MSP430F1xx состоит из двух сегментов (INFOA и INFOB) по 128 байт каждый, в то время как MSP430F2xx содержит четыре сегмента (INFOA, INFOB, INFOC и INFOD) по 64 байта каждый.

Каждый сегмент информационной Flash-памяти должен стираться отдельно, следовательно, для MSP430F2xx потребуется четыре обращения вместо двух, как для MSP430F1xx. Сегмент INFOA микроконтроллеров MSP430F2xx поддерживает функцию защиты от стирания, что требует дополнительной обработки при обращении к этой области памяти. Однако не рекомендуется стирать сектор INFOA, а также записывать в него данные, так как там хранятся калибровочные коэффициенты высокочастотного осциллятора DCO. Для более подробной информации о системе памяти и обращении к сегменту INFOA смотрите [3, 4].

Последовательные интерфейсы: USCI вместо USART

Одно из самых значительных отличий MSP430F2xx от MSP430F1xx заключается в последовательном коммуникационном интерфейсе. Микроконтроллеры MSP430F2xx содержат USCI-модуль, а MSP430F1xx — USART, которые программно не совместимы, то есть при переходе на новый микроконтроллер требуется адаптация кода. С помощью USCI можно реализовать два SPI-канала или сочетание из I²C и UART. Стоит заметить, что MSP430F1xx не поддерживают интерфейс I²C.

Мы подчеркнем только основные отличия последовательных коммуникационных модулей старого и нового семейств. Для подробного ознакомления рекомендуется обратиться к документам [3, 4], а также рассмотреть программные примеры использования USCI-модуля, доступные на сайте производителя.

Режим UART

Функционирование USCI-модуля MSP430F2xx аналогично USART-модулю MSP430F1xx в режиме UART.

Основные отличия:

- Модуль USCI использует отдельный генератор скорости передачи. При этом используется новая схема модуляции, с двухуровневым модулятором с возможностью передискретизации. И при переходе на новый микроконтроллер потребуется пересчитать коэффициенты для настройки генератора скорости передачи.
- Детектирование фронта первого импульса посылки и схема активации системы синхронизации нового и старого семейств отличаются. В USCI для этого применяется более простая схема, в которой модуль автоматически активирует систему синхронизации при обнаружении фронта первого импульса и генерирует прерывание ЦПУ после приема целого символа. В UART-модуле MSP430F1xx прерывание генерируется сразу после обнаружения фронта первого импульса посылки, и приложение должно само активировать систему синхронизации после принятия первого символа.
- Флаги прерываний модуля USCI не сбрасываются автоматически при входе в функцию обработки прерывания.

Режим SPI

Функционирование USCI-модуля MSP430F2xx и USART-модуля MSP430F1xx в режиме SPI также аналогично.

Основные отличия:

- USART поддерживает один канал SPI, в то время как USCI поддерживает два канала, работающих одновременно (USCI_A0, USCI_B0).
- Каждому из SPI-каналов микроконтроллера MSP430F1xx соответствует отдельный вектор прерываний. Каждый USCI-модуль микроконтроллера MSP430F2xx содержит два совмещенных вектора прерываний, для приема и передачи.
- USCI в режиме SPI по умолчанию работает в режиме «самый младший бит — первый» (LSB-first). Порядок может меняться с помощью бита UCMSB в регистрах управления UCAxCTL0/UCBxCTL0. Модуль USART работает только в режиме MSB-first («самый старший бит — первый»), и этот режим изменить нельзя.
- Максимальная частота USCI при передаче битов в режиме SPI соответствует частоте BRCLK, в то время как для USART-модуля эта величина составляет BRCLK/2.

Режим I²C

Модуль USCI не поддерживает некоторых функций, доступных в модуле USART:

- Автоматический подсчет байтов (регистр I2CNDAT).
- 16-битный доступ к регистрам I²C (I2CDRW-регистр).
- Декодирование источника прерывания I²C (I2CIV-регистр).

Система синхронизации

Осцилляторы LFXT1 и XT2

Микроконтроллеры MSP430F2xx могут работать от тех же низко- и высокочастотных осцилляторов, что и MSP430F1xx, потребляя при этом меньше электроэнергии. Кроме того, в низкочастотном режиме (LF) доступны конфигурируемые нагрузочные конденсаторы кварцевого резонатора. По умолчанию значение емкости этих конденсаторов в LF-режиме составляет 6 пФ.

Если в приложении, в котором использовались внешние нагрузочные конденсаторы, производится переход на микроконтроллер MSP430F2xx, то нужно иметь в виду:

- Способность низкочастотного осциллятора микроконтроллера MSP430F2xx обнаруживать неисправности и сигнализировать о них с помощью установки флага LFXT1OF.
- Если в приложении на базе MSP430F1xx использовался внешний 32-кГц кварц для работы в режиме низкого энергопотребления (LPM3) и точность кристалла не так важна, то вместо него при переходе на MSP430F2xx можно использовать встроенный осциллятор VLO. В результате на один компонент приложения станет меньше и сократится энергопотребление микроконтроллера в режиме LPM3. Частота VLO составляет 12 кГц. Для более подробной информации обратитесь к [6].
- Если в приложении использовался внешний цифровой осциллятор, то при переходе на MSP430F2xx рекомендуется использовать новый режим ввода цифрового тактового сигнала (с помощью установки битов управления LFXT1S1 и LFXT1S0).

Высокочастотный осциллятор DCO

В микроконтроллерах MSP430F2xx DCO-осциллятор имеет более высокую точность, что позволяет использовать его в широком частотном диапазоне. Изначально во Flash-памяти хранятся калибровочные коэффициенты для упрощения настройки DCO.

При переходе с MSP430F1xx на MSP430F2xx следует обратить внимание на следующие аспекты:

- Частота DCO MSP430F1xx по умолчанию находится в диапазоне 800 кГц, а в MSP430F2xx — в диапазоне 1,2 МГц.
- Использование калибровочных коэффициентов для DCO MSP430F2xx позволяет обойтись без использования FLL-алгоритмов совместно с внешним кварцем для настройки DCO.
- MSP430F1xx содержит три бита для управления частотой DCO (RSELx в регистре

BCSCTL1), в то время как MSP430F2xx — четыре.

- Если значения DCOx, MODx и RSELx в регистрах управления DCO микроконтроллера MSP430F1xx жестко запрограммированы, это приведет к другой частоте работы DCO-осциллятора MSP430F2xx.
- Если в MSP430F1xx использовался внешний резистор смещения (с помощью установки DCOR в регистре BCSCTL2), в MSP430F2xx при запуске такой программы DCO начнет работать точно так же, как и в MSP430F1xx, на той же самой частоте (при идентичной установке битов управления).

Загрузчик

Микроконтроллеры MSP430F2xx содержат новый загрузчик (BSL) с улучшенными защитными функциями. Память обоих семейств MSP430F1xx и MSP430F2xx защищена 256-битным паролем. Однако в MSP430F2xx при обращении к памяти с неправильным паролем стирается все содержимое Flash-памяти (включая калибровочные константы, хранящиеся в сегменте INFOA).

Векторы прерываний

Перекомпилирование кода MSP430F1xx с использованием библиотек для MSP430F2xx автоматически решает задачи, связанные с изменением расположения векторов в таблице прерываний. Однако в некоторых случаях при процедуре обработки прерывания нужно сделать это вручную.

Также изменилось расположение таблицы векторов прерываний. Для MSP430F1xx это была область адресов с 0xFFE0 до 0xFFFF (16 слов), а для MSP430F2xx — с 0xFFC0 до 0xFFFF (32 слова). Кроме этого, область памяти с адресом 0xFFBE зарезервирована в MSP430F2xx и используется для секретного ключа загрузчика BSL.

Зарезервированные участки памяти

Микроконтроллеры MSP430F2xx содержат обновленные версии базовой системы синхронизации (BCS+) и компаратора (Comparator_A+) периферийные устройства, такие как BCS+, Comparator+. Эти дополнительные устройства используют часть тех участков памяти, которые раньше имели статус «зарезервировано». Теперь при обращении к этим областям памяти нужно соблюдать осторожность, так как их изменение может привести к нежелательным последствиям в работе приложения. В качестве примера рассмотрим регистр управления компараторами Comparator_A и Comparator_A+.

7	6	5	4	3	2	1	0
Не используется				P2CA1	P2CA0	CAF	CAOUT
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	r-(0)

Рис. 2. Биты регистра CACTL2 микроконтроллеров MSP430F1xxx

7	6	5	4	3	2	1	0
CASHORT	P2CA4	P2CA3	P2CA2	P2CA1	P2CA0	CAF	CAOUT
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	r-(0)

Рис. 3. Биты регистра CACTL2 микроконтроллеров MSP430F2xxx

Допустим, что в ходе программы устанавливается бит 7 (рис. 2), и она работает правильно на MSP430F1xx. Если запустить эту же программу на MSP430F2xx (рис. 3), получится, что входы компаратора будут внутренне закорочены.

Заключение

Переход с MSP430F1xx на MSP430F2xx позволяет повысить производительность и функциональность приложения, снизить энергопотребление и конечную цену устройства. Как правило, из-за аппаратной совместимости многих микроконтроллеров старого и нового семейств такой переход требует только желания разработчика улучшить устройство. Но иногда для того, чтобы сделать устройство более конкурентоспособным, необходимо приложить кое-какие усилия. Надеем-

ся, что эта статья послужит не только руководством, но и стимулом к переходу на новое семейство. ■

Литература

1. Migrating From MSP430F16x to MSP430F261x Application Report (SLAA380).
2. Migrating From MSP430F13x/14x to MSP430F23x/24x Application Report (SLAA381A).
3. MSP430x1xx Family User's Guide (SLAU049).
4. MSP430x2xx Family User's Guide (SLAU144).
5. MSP430x15x, MSP430x16x, MSP430x161x data sheet (SLAS368).
6. MSP430x13x, MSP430x14x, MSP430x14x1 data sheet (SLAS272).
7. MSP430x241x, MSP430x261x data sheet (SLAS541).
8. MSP430x23x, MSP430x24x(1) data sheet (SLAS547).
9. Using the VLO Library (SLAA340).