

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

Государственное образовательное учреждение высшего профессионального образования

«ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Е.Ю. БУРКИН

**ЦИФРОВЫЕ СИСТЕМЫ УПРАВЛЕНИЯ
УСТРОЙСТВ СИЛОВОЙ ЭЛЕКТРОНИКИ**

ПРАКТИКУМ

Издательство ТПУ
Томск 2007

ББК 31.264.5.я73

УДК 621.314(075.8)

Б90

Буркин Е.Ю.

Б90 Лабораторный практикум по цифровым системам управления устройств силовой электроники: учебное пособие/ Е.Ю.Буркин. – Томск: Изд-во ТПУ, 2007. – 80 с.

Практикум включает семь работ, включающих следующие разделы: регуляторы постоянного напряжения; однофазные и трехфазные инверторы. Для каждой лабораторной работы приведены необходимые теоретические сведения, программа работы и контрольные вопросы. Лабораторный практикум предназначен для студентов, обучающихся по направлению 210100 (550770) «Электроника и микроэлектроника».

УДК 621.314(075.8)

ББК 31.264.5.я73

Рекомендовано к печати Редакционно-издательским советом
Томского политехнического университета

Рецензенты

Кандидат технических наук, зав. лаб. № 16 ГУП НИИАЭМ
при ТУСУРе
В.Ф. Кошевец

Кандидат физико-математических наук, доцент ТПУ
А.Н. Яковлев

© Томский политехнический университет, 2007
© Оформление. Издательство ТПУ, 2007

ВВЕДЕНИЕ

В современной электроэнергетике значительная часть первичной электроэнергии преобразуется с помощью силовых полупроводниковых преобразователей. Постоянное расширение областей применения устройств преобразовательной техники требует адекватного улучшения и оптимизации их параметров и характеристик. Немаловажным фактором в решении задач увеличения коэффициента полезного действия, уменьшения массогабаритных показателей, улучшения электромагнитной совместимости преобразователей играет тип их системы управления. В настоящее время развитие цифровой техники позволяет реализовать практически для всех типов преобразователей полностью цифровое управление. Предлагаемый лабораторный практикум посвящен изучению основных схем преобразователей электрической энергии: преобразователей постоянного напряжения и инверторов с цифровыми системами управления. Все лабораторные работы проводятся на универсальном стенде фронтальным методом.

ОПИСАНИЕ ЛАБОРАТОРНОГО СТЕНДА

Универсальный стенд с набором сменных модулей позволяет подробно исследовать процессы в базовых схемах преобразователей электроэнергии, изучаемых в курсе "Энергетическая электроника". Все работы выполняются фронтально на 8 стендах, содержащих наборные поля. Схемы собираются из соответствующих модулей на наборном поле, соединения выполняются с помощью комплекта проводников.

В качестве измерительных приборов используются универсальный осциллограф С1–83, позволяющий измерять и снимать диаграммы тока и напряжения, цифровой вольтметр В7–38, позволяющий измерять переменные и постоянные токи и напряжения, а также величину сопротивления.

Так как при сборке схем возможны ошибки, во избежание длительных коротких замыканий источников при их использовании, в цепь **ОБЯЗАТЕЛЬНО** должны быть включены модули с предохранителями. Кроме съемных элементов-модулей, в работах применяются габаритные элементы – дроссели и конденсаторы, размещенные в стенде стационарно. Их выводы с соответствующими обозначениями расположены в верхней части наборного поля, а сами они – за стеклом в глубине стенда. В лабораторных работах используется отладочный комплект eZdspF2812. eZdspF2812(F2808) это независимая отладочная плата, ко-

торая предназначена для оценки работы цифрового сигнального процессора (DSP) фирмы Texas Instruments TMS320F2812(F2808) и обладает следующими особенностями:

- DSP TMS320F2812(F2808) – 150(100)МГц, 18К слов RAM, 128К слов Flash;
- Внешнее ОЗУ SRAM – 64К слов;
- Тактовый генератор – 30МГц;
- 2 соединителя (АЦП и порты ввода\вывода);
- JTAG контроллер (IEEE 1149.1);
- Разъем для внешнего JTAG контроллера.

На рис.1 показана общая структурная схема платы eZdspF2812(F2808).

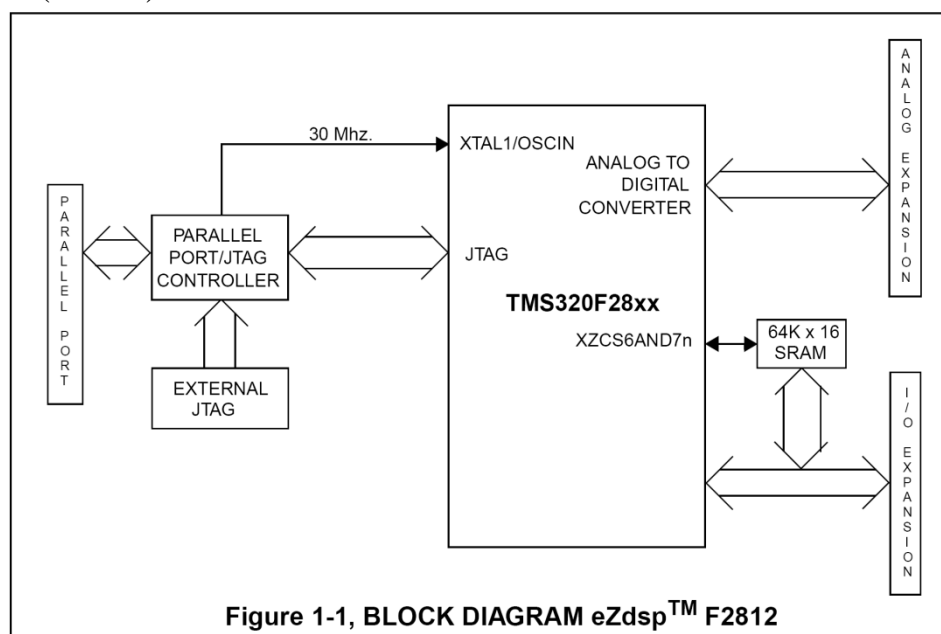


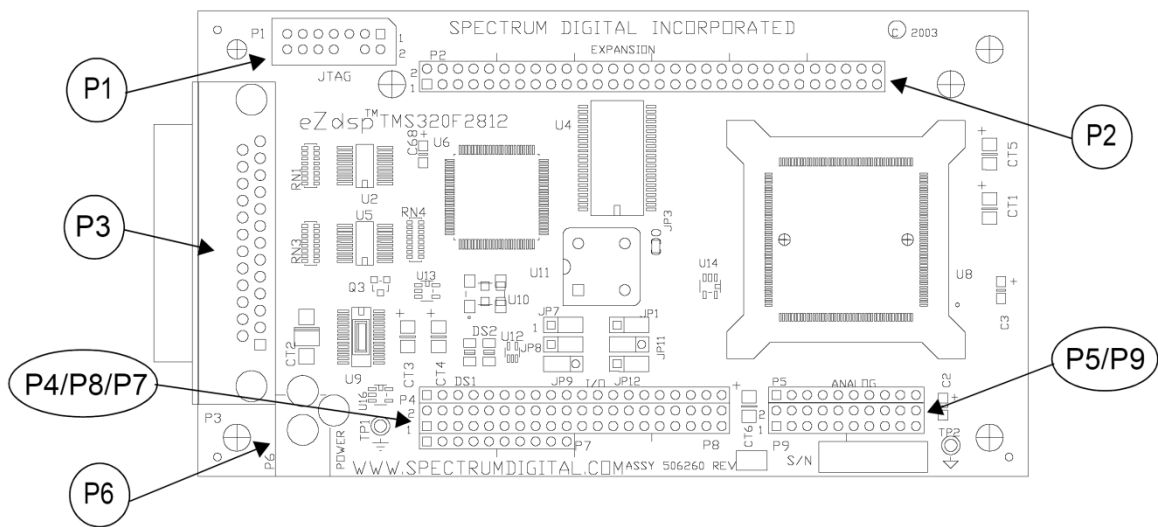
Рис.1

На плате присутствует ряд соединителей (разъемов) и перемычек (джамперов), которые настраивают работу платы. Все они обобщены в табл.1, а на рис.2 а, б показано их расположение.

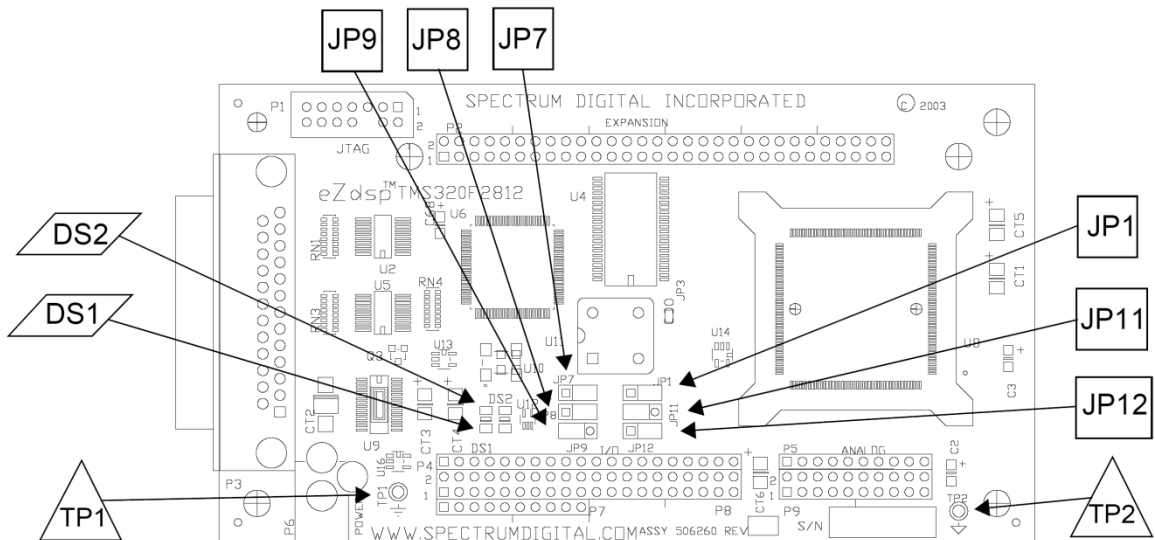
Таблица 1

Соединитель	Функция соединителя	Перемычка (заводское положение)	Функция перемычки
P1	JTAG интерфейс	JP1 (2-3)	ХМР/МСп 1-2 Микропроцессорный режим 2-3 Микрокомпьютерный режим

P2	Расширение памяти (интерфейс внеш.памяти)	JP7 (2-3)	Boot Mode 3 (SCITXDA)
P3	Параллельный порт/Интерфейс JTAG контроллера	JP8 (2-3)	Boot Mode 2 (MDXA)
P4/P8/P9	Интерфейс портов ввода-вывода	JP9 (1-2)	PLL disable (2-3)
P5/P9	Аналоговый интерфейс (входы АЦП)	JP11 (1-2)	Boot Mode 1 (SPISTEA)
P6	Соединитель питания	JP12 (2-3)	Boot Mode 0 (SPICLKA)
Перемычки должны быть обязательно в одном из двух положений (не должны отсутствовать)			



а



б

Рис.2 Схема расположения соединителей – а, и перемычек – б

Режим загрузки программы определяется переключками JP7, JP8, JP11, JP12 согласно табл.2

Таблица 2

JP7, BOOT3 SCITXDA	JP8, BOOT2 MDXA	JP11, BOOT1 SPISTEA	JP12, BOOT0 SPICLKA	Режим загрузки
1	x	x	x	FLASH
0	1	x	x	SPI
0	0	1	1	SCIA
0	0	1	0	H0 (def)
0	0	0	1	OTP
0	0	0	0	PARALLEL

На плате присутствуют два светодиода DS1 – +5V, и DS2 – XF бит (порт GPIOF14), а также две тестовые точки (аналоговая и цифровая земля TP1 и TP2 соответственно рис.2 б).

Функциональное назначение выводов основных разъемов показано на рис.3

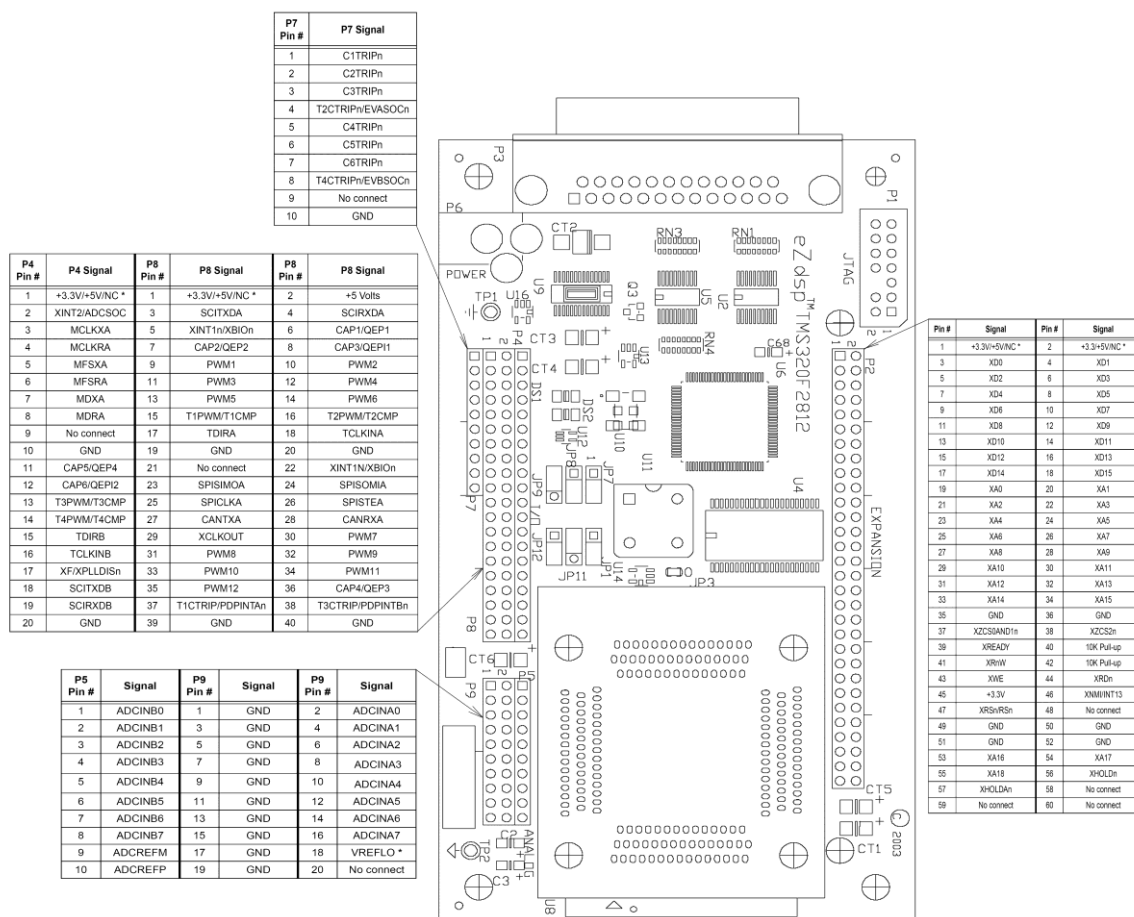


Рис.3. Схема расположения и функциональное назначение разъемов

С внешней схемой силовой части отладочный комплект соединяется через согласующую плату, которая защищает входы процессора, буферизует выходы и содержит в своем составе ряд дополнительных модулей. Принципиальная схема основных блоков и схема расположения элементов согласующей платы показаны на рис.4-6 соответственно.

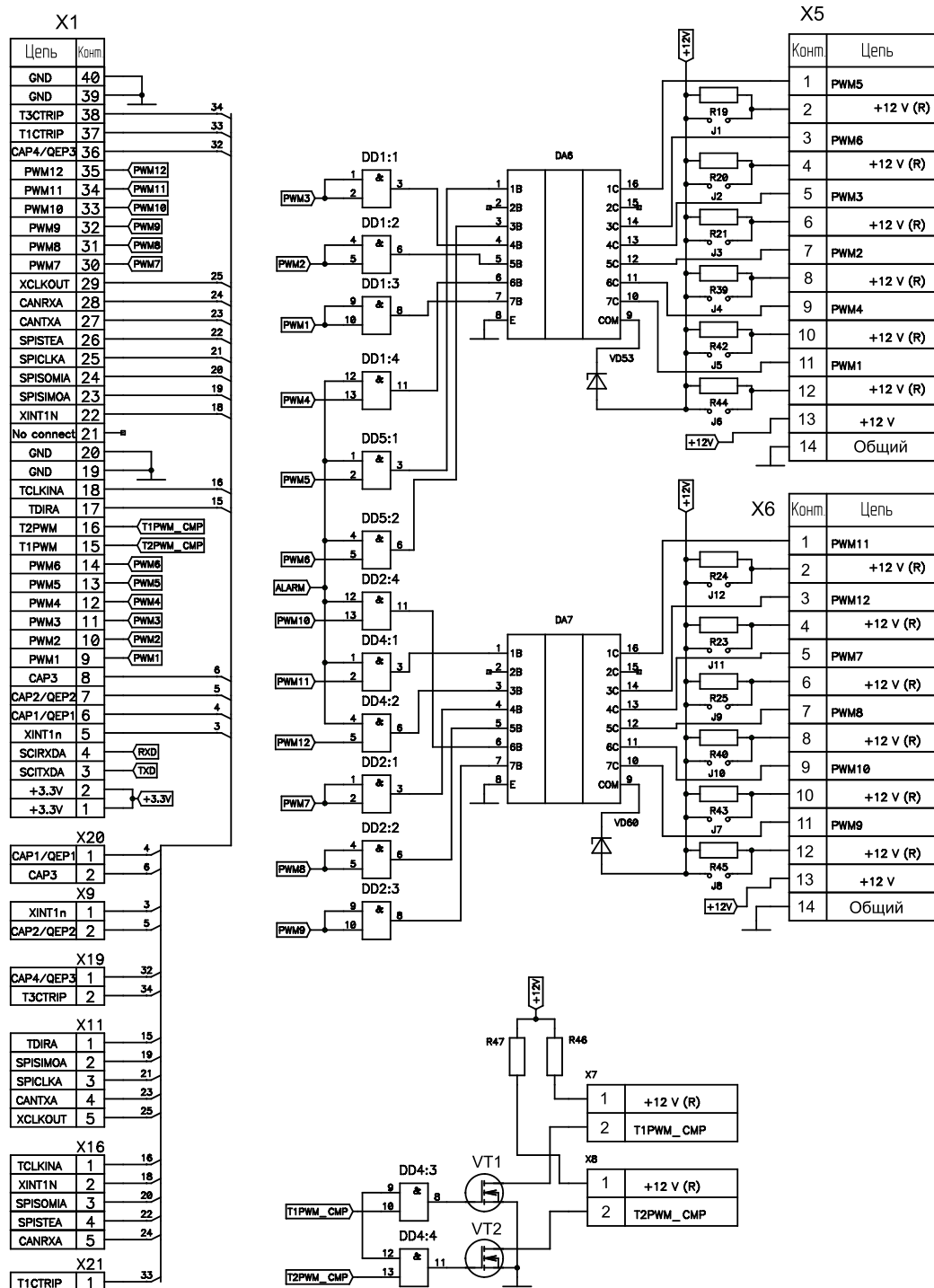


Рис.4 Принципиальная схема буферизования ШИМ-выходов

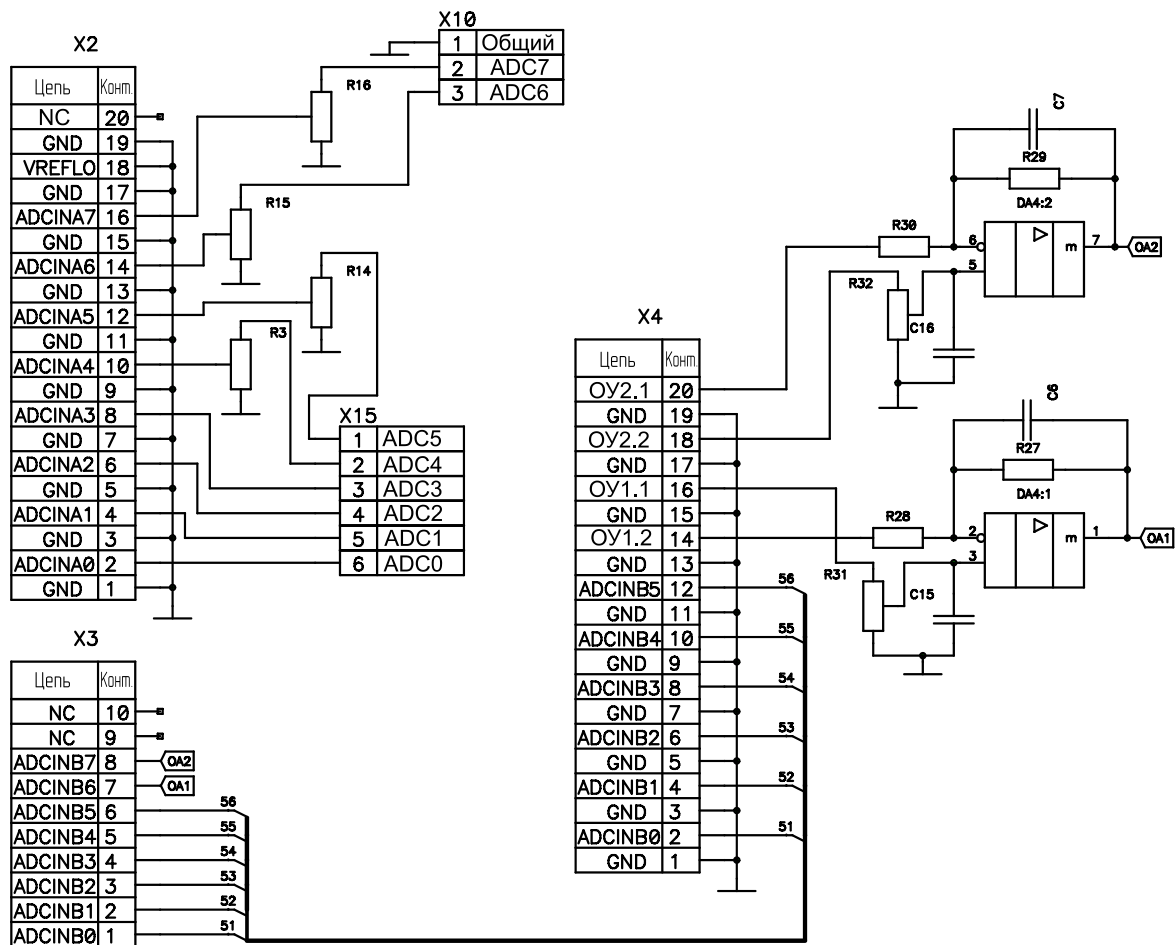


Рис.5 Схема подключения входов аналого-цифрового преобразователя

Согласующая плата подключается и работает с соответствующими выводами отладочного комплекта следующим образом. Разъемы платы X1, X2 и X3 подключаются к разъемам P8, P9 и P5 отладочного комплекта соответственно. При этом нумерация выводов у этих разъемов совпадает. Выводы ШИМ процессора буферизируются логическими элементами DD1, DD2, DD4, DD5, транзисторными сборками DA6, DA7 и транзисторами VT1, VT2, обеспечивая выход с открытым коллектором (стоком) и нагрузочной способностью до 100 мА. Сигнал разрешения (блокировки) ALARM в предлагаемом лабораторном практикуме не используется и всегда равен логической единице.

Сигналы, поступающие на АЦП, проходят, либо без изменения (только защитные диоды и резисторы) – ADCINA0-ADCINA3, ADCINB0-ADCINB6, либо через делитель образованный подстроечными резисторами R3, R14-R15 – ADCINA4-ADCINA7. Кроме того к двум входам АЦП – ADCINB6,7 подключены выходы вспомогательных уси-

лителей. Входы усилителей доступны на контактах разъема X4 согласующей платы, рис.5.

Оставшиеся сигналы разъема P8-X1 выведены через защитные диоды и резисторы на вспомогательные разъемы (X9, X11, X16, X19-21), рис.5.

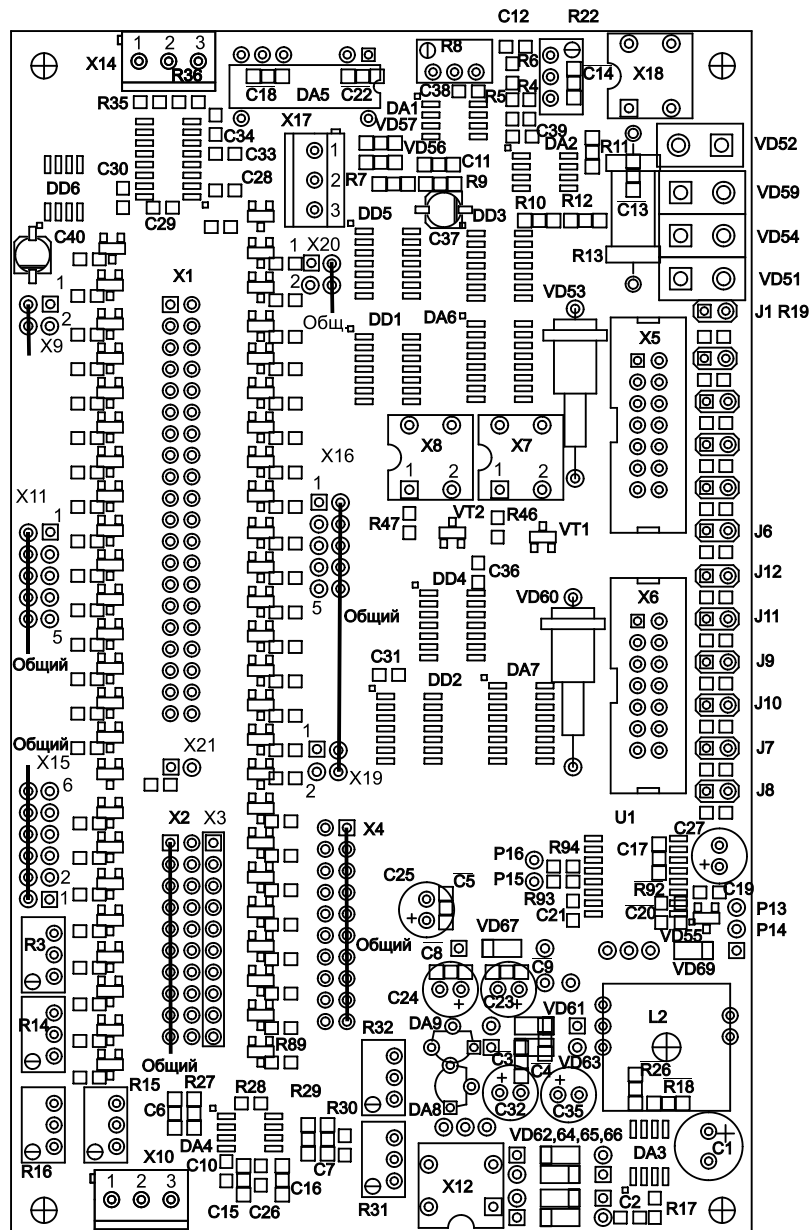


Рис.6. Схема расположения элементов согласующей платы

Кроме блоков, показанных на рис.4,5 согласующая плата содержит в своем составе вспомогательный источник питания, гальванически развязанный интерфейс RS232, изолирующий усилитель HCPL788J и два компаратора LM211, формирующих сигнал блокировки ALARM.

ЛАБОРАТОРНАЯ РАБОТА № 1

ИССЛЕДОВАНИЕ ПОНИЖАЮЩЕГО ПРЕОБРАЗОВАТЕЛЯ ПОСТОЯННОГО НАПРЯЖЕНИЯ С ОБРАТНОЙ СВЯЗЬЮ ПО ВЫХОДНОМУ НАПРЯЖЕНИЮ

ЦЕЛЬ РАБОТЫ

Целью настоящей лабораторной работы является практическое изучение основных статических и динамических характеристик понижающего преобразователя постоянного напряжения с обратной связью по выходному напряжению и цифровой системой управления.

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

Рассчитать регулятор напряжения по схеме рис.1.1 с выходной мощностью $P_H = 10$ (Вт), выходным напряжением 5 В, входным напряжением 10 В. По результатам расчета определить параметры малосигнальной модели, амплитудную и частотную характеристики для П и ПИ-регуляторов. В программе Simulink (MATLAB) реализовать малосигнальную и реальную модели преобразователя, при частоте работы ключевого элемента $f = 20$ кГц (частота квантования $f_s = 200$ кГц для цифрового регулятора). По моделям определить границу устойчивости П- и ПИ-регулятора и снять динамические характеристики при скачкообразном изменении нагрузки (уменьшение/увеличение на 50%) и входного напряжения (уменьшение/увеличение на 30%). Сравнить характеристики между собой и сделать вывод.

ИСПОЛЬЗУЕМЫЕ ЭЛЕМЕНТЫ

Модульные:

- Отладочный комплект А1 – ezDSP28xx.
- Модуль согласования А2.
- Модуль транзисторных ключей А3.
- Резисторы: $R_H = 36$ Ом – 2 шт.
- Конденсатор С1 – 100 мкФ 63 В
- Дроссель L1 – 360 мкГн.
- Шунт для измерения тока $R_{ш} = 0.1$ Ом.
- Набор соединительных проводов – 15 шт.

ПРОГРАММА РАБОТЫ

2.1. Собрать схему исследуемого регулятора согласно рис.1.1.

2.2. Написать программу осуществляющую стабилизацию выходного напряжения на уровне 5 В, при изменении входного напряжения источника питания в диапазоне 10 – 20 В. Последовательно реализовать П, ПИ-регуляторы.

2.3. Изменяя коэффициент пропорциональности для П-регулятора определить границу устойчивости и снять семейство зависимостей выходного напряжения в функции входного для различных значений коэффициента пропорциональности (4–5 зависимостей, 8–10 точек). Сравнить полученные экспериментально характеристики с теоретическими по малосигнальной модели преобразователя [1].

2.4. При выходном напряжении 5В, изменяя сопротивление нагрузки в диапазоне 5–72 Ом, снять семейство внешних характеристик для различных значений коэффициента заполнения (изменяя входное напряжение). Сравнить полученные экспериментально характеристики с теоретическими по малосигнальной модели преобразователя [1].

2.5. Реализовать ПИ-алгоритм. Варьируя постоянную интегрирования (фиксированный коэффициент пропорциональности, соответствующий устойчивой работе П-регулятора) определить границу устойчивости преобразователя. Сравнить с теоретическими значениями.

2.6. Для устойчивой работы ПИ-регулятора, изменяя сопротивление нагрузки, снять внешнюю характеристику преобразователя.

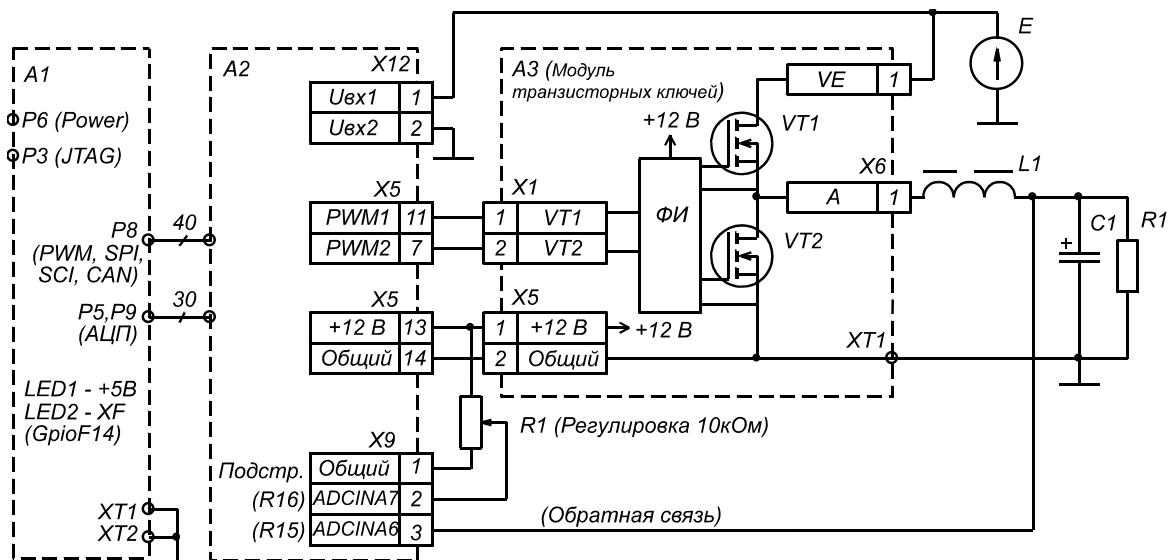


Рис.1.1

СОДЕРЖАНИЕ ОТЧЕТА

1. Название и программа работы.
2. Исследуемые схемы, диаграммы, таблицы.
3. Выводы по работе.
4. Выполненное предварительное задание.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните методику получения малосигнальных моделей импульсных преобразователей напряжения и тока. При каких условиях эта модель неудовлетворительно отражает энергетические процессы преобразователей?
2. Какие существуют методы определения устойчивости системы для линейных систем? Поясните на примере малосигнальной модели преобразователя понижающего типа их использование.
3. Получите модель понижающего преобразователя напряжения работающего в режиме прерывистого тока дросселя. Сравните частотные характеристики для режима непрерывного и прерывистого тока дросселя.
4. Поясните влияние активного сопротивления дросселя на энергетические процессы в схеме.
5. Поясните влияние активного последовательного сопротивления фильтрового конденсатора на энергетические процессы в схеме. Какие типы конденсаторов предпочтительно использовать в данном случае?
6. Методом усреднения покажите влияние времени переключения ключевого элемента на выражение для коэффициента передачи.
7. Расскажите об основных способах организации обратной связи. Запишите законы управления и нарисуйте структурную схему.
8. Приведите варианты схем понижающих преобразователей с гальванической развязкой выходного напряжения. Поясните принцип их действия и получите малосигнальные модели этих преобразователей.
9. Приведите вариант квазирезонансного преобразователя на основе схемы понижающего преобразователя постоянного напряжения. Поясните принцип его работы и приведите основные диаграммы и расчетные соотношения.
10. Поясните, какими способами в схемах ключевых преобразователей осуществляется защита ключевых элементов от перенапряжения и сквозного тока. Приведите схемы и диаграммы работы.

ЛАБОРАТОРНАЯ РАБОТА № 2

ИССЛЕДОВАНИЕ ПОНИЖАЮЩЕГО ПРЕОБРАЗОВАТЕЛЯ ПОСТОЯННОГО НАПРЯЖЕНИЯ С ОБРАТНОЙ СВЯЗЬЮ ПО ВЫХОДНОМУ НАПРЯЖЕНИЮ И ТОКУ ДРОССЕЛЯ

ЦЕЛЬ РАБОТЫ

Целью настоящей лабораторной работы является практическое изучение основных статических и динамических характеристик понижающего преобразователя постоянного напряжения с обратной связью по выходному напряжению и току дросселя с цифровой системой управления.

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

Рассчитать регулятор напряжения по схеме рис. 2.1 с выходной мощностью $P_H = 10$ (Вт), выходным напряжением 5 В, входным напряжением 10 В. По результатам расчета определить параметры малосигнальной модели, амплитудную и частотную характеристики для П и ПИ-регуляторов. В программе Simulink (MATLAB) реализовать малосигнальную и реальную модели преобразователя, при частоте работы ключевого элемента $f = 20$ кГц (частота квантования $f_s = 200$ кГц для цифрового регулятора). По моделям определить границу устойчивости П- и ПИ-регулятора и снять динамические характеристики при скачкообразном изменении нагрузки (уменьшение/увеличение на 50%) и входного напряжения (уменьшение/увеличение на 30%). Сравнить характеристики между собой и сделать вывод.

ИСПОЛЬЗУЕМЫЕ ЭЛЕМЕНТЫ

Модульные:

- Отладочный комплект А1 – ezDSP28xx.
- Модуль согласования А2.
- Модуль транзисторных ключей А3.
- Резисторы: $R_H = 36$ Ом – 2 шт.
- Конденсатор С1 – 100 мкФ 63 В
- Дроссель L1 – 360 мкГн.
- Шунт для измерения тока $R_{ш} = 0.1$ Ом.
- Набор соединительных проводов – 15 шт.

ПРОГРАММА РАБОТЫ

- 2.1. Собрать схему исследуемого регулятора согласно рис. 2.1.
- 2.2. Написать программу осуществляющую стабилизацию выходного напряжения на уровне 5 В, при изменении входного напряжения источника питания в диапазоне 10 – 20 В. Последовательно реализовать П, ПИ регуляторы.
- 2.3. Изменяя коэффициент пропорциональности (использовать различные комбинации коэффициентов по току и напряжению) для П-регулятора определить границу устойчивости и снять семейство зависимостей выходного напряжения в функции входного для различных значений коэффициента пропорциональности (4–5 зависимостей, 8–10 точек). Сравнить полученные экспериментально характеристики с теоретическими по малосигнальной модели преобразователя [1].
- 2.4. При выходном напряжении 5В, изменяя сопротивление нагрузки в диапазоне 5–72 Ом, снять семейство внешних характеристик для различных значений коэффициента заполнения (изменяя входное напряжение). Сравнить полученные экспериментально характеристики с теоретическими по малосигнальной модели преобразователя [1].
- 2.5. Реализовать ПИ-алгоритм. Для устойчивой работы ПИ-регулятора, изменяя сопротивление нагрузки, снять внешнюю характеристику преобразователя.

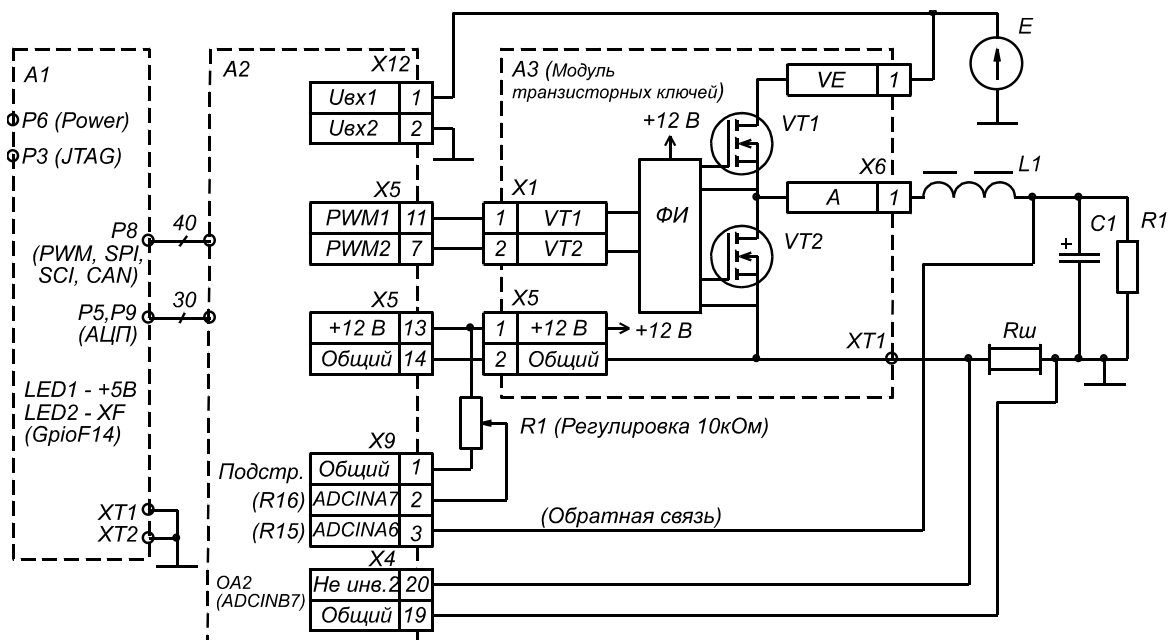


Рис.2.1

СОДЕРЖАНИЕ ОТЧЕТА

- 2.6. Название и программа работы.
- 2.7. Исследуемые схемы, диаграммы, таблицы.
- 2.8. Выводы по работе.
- 2.9. Выполненное предварительное задание.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для каких целей используют многоконтурные обратные связи в импульсных преобразователях электроэнергии? Приведите примеры структурных схем с несколькими контурами обратной связи и поясните принцип их действия.
2. Поясните методику получения малосигнальной модели преобразователя постоянного напряжения понижающего типа для режима непрерывного тока дросселя. Нарисуйте структурную схему преобразователя с учетом системы управления.
3. Приведите варианты схем и диаграммы работы нерассеивающих пассивных и активных снабберов преобразователей постоянного напряжения понижающего типа. Поясните принцип действия и методику расчета и выбора элементов этих схем.
4. Поясните влияние активного сопротивления дросселя на энергетические процессы в схеме.
5. Поясните влияние активного последовательного сопротивления фильтрового конденсатора на энергетические процессы в схеме. Какие типы конденсаторов предпочтительно использовать в данном случае?
6. Методом усреднения покажите влияние времени переключения ключевого элемента на выражение для коэффициента передачи.
7. Расскажите об основных способах организации обратной связи. Запишите законы управления и нарисуйте структурную схему.
8. Приведите варианты схем понижающих преобразователей с гальванической развязкой выходного напряжения. Поясните принцип их действия и получите малосигнальные модели этих преобразователей.
9. Приведите вариант квазирезонансного преобразователя на основе схемы понижающего преобразователя постоянного напряжения. Поясните принцип его работы и приведите основные диаграммы и расчетные соотношения.

ЛАБОРАТОРНАЯ РАБОТА № 3

ИССЛЕДОВАНИЕ ПОВЫШАЮЩЕГО ПРЕОБРАЗОВАТЕЛЯ ПОСТОЯННОГО НАПРЯЖЕНИЯ С ОБРАТНОЙ СВЯЗЬЮ ПО ВЫХОДНОМУ НАПРЯЖЕНИЮ

ЦЕЛЬ РАБОТЫ

Целью настоящей лабораторной работы является практическое изучение основных статических и динамических характеристик повышающего преобразователя постоянного напряжения с обратной связью по выходному напряжению и цифровой системой управления.

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

Рассчитать регулятор напряжения по схеме рис. 3.1 с выходной мощностью $P_H = 10$ (Вт), выходным напряжением 20 В, входным напряжением 10 В. По результатам расчета определить параметры малосигнальной модели, амплитудную и частотную характеристики для П и ПИ-регуляторов. В программе Simulink (MATLAB) реализовать малосигнальную и реальную модели преобразователя, при частоте работы ключевого элемента $f = 20$ кГц (частота квантования $f_s = 200$ кГц для цифрового регулятора). По моделям определить границу устойчивости П- и ПИ-регулятора и снять динамические характеристики при скачкообразном изменении нагрузки (уменьшение/увеличение на 50%) и входного напряжения (уменьшение/увеличение на 30%). Сравнить характеристики между собой и сделать вывод.

ИСПОЛЬЗУЕМЫЕ ЭЛЕМЕНТЫ

Модульные:

- Отладочный комплект А1 – ezDSP28xx.
- Модуль согласования А2.
- Модуль транзисторных ключей А3.
- Резисторы: $R_H = 36$ Ом – 2 шт.
- Конденсатор С1 – 100 мкФ 63 В
- Дроссель L1 – 360 мкГн.
- Шунт для измерения тока $R_{ш} = 0.1$ Ом.
- Набор соединительных проводов – 15 шт.

ПРОГРАММА РАБОТЫ

- 2.1. Собрать схему исследуемого регулятора согласно рис. 3.1.
- 2.2. Реализовав П- и ПИ-алгоритмы регулирования, изменяя сопротивление нагрузки, снять внешнюю характеристику преобразователя.
- 2.3. Изменяя коэффициент пропорциональности определить границу устойчивой работы, введя интегральную составляющую определить ее влияние на качество регулирования и устойчивость работы преобразователя по сравнению с чисто пропорциональным алгоритмом регулирования.

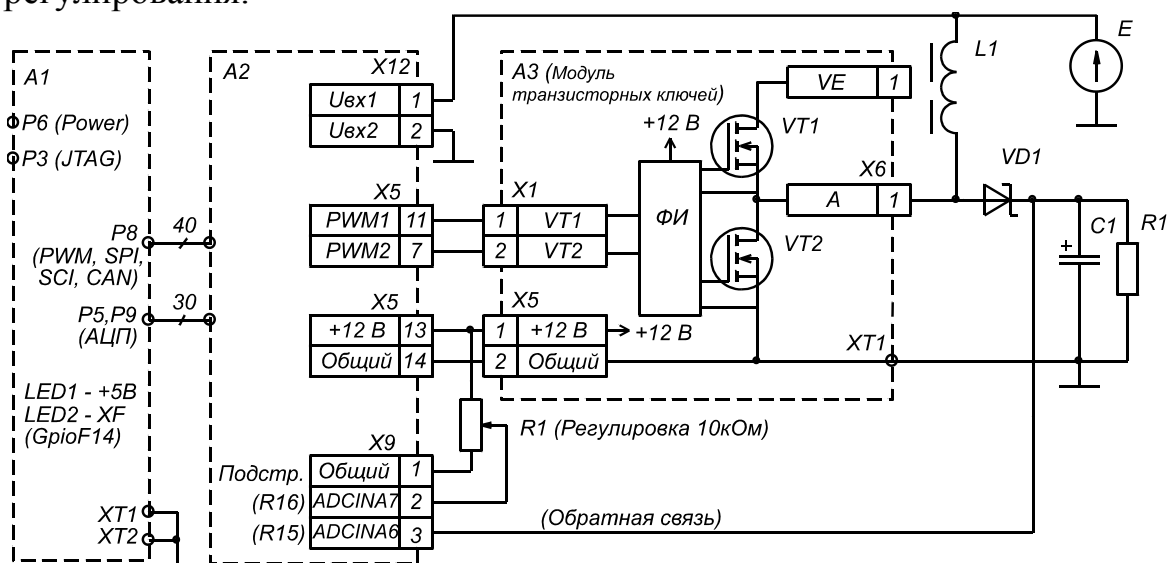


Рис.3.1

СОДЕРЖАНИЕ ОТЧЕТА

- 2.4. Название и программа работы.
- 2.5. Исследуемые схемы, диаграммы, таблицы.
- 2.6. Выводы по работе.
- 2.7. Выполненное предварительное задание.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните методику получения малосигнальной модели импульсного преобразователя напряжения повышающего типа. При каких условиях эта модель неудовлетворительно отражает энергетические процессы преобразователей?

2. Какие существуют методы определения устойчивости системы для линейных систем? Поясните на примере малосигнальной модели преобразователя повышающего типа их использование.
3. Получите модель повышающего преобразователя напряжения работающего в режиме прерывистого тока дросселя. Сравните частотные характеристики для режима непрерывного и прерывистого тока дросселя.
4. Поясните влияние активного сопротивления дросселя на энергетические процессы в схеме.
5. Поясните влияние активного последовательного сопротивления фильтрового конденсатора на энергетические процессы в схеме. Какие типы конденсаторов предпочтительно использовать в данном случае?
6. Что такое корректор коэффициента мощности? Приведите вариант однофазного корректора и поясните способы управления им.
7. Расскажите об основных способах организации обратной связи в импульсных преобразователях напряжения и тока. Запишите законы управления и нарисуйте структурные схемы.
8. Приведите варианты схем повышающих преобразователей с гальванической развязкой выходного напряжения. Поясните принцип их действия и получите малосигнальные модели этих преобразователей.
9. Приведите вариант квазирезонансного преобразователя на основе схемы повышающего преобразователя постоянного напряжения. Поясните принцип его работы, приведите основные диаграммы и расчетные соотношения.
10. Поясните, какими способами в схемах ключевых преобразователей осуществляется защита ключевых элементов от перенапряжения и сквозного тока. Приведите вариант схемы и диаграммы работы нерассеивающего снаббера для преобразователя повышающего типа.

ЛАБОРАТОРНАЯ РАБОТА № 4

ИССЛЕДОВАНИЕ ОБРАТНОХОДОВОГО (ИНВЕРТИРУЮЩЕГО) ПРЕОБРАЗОВАТЕЛЯ ПОСТОЯННОГО НАПРЯЖЕНИЯ С ОБРАТНОЙ СВЯЗЬЮ ПО ВЫХОДНОМУ НАПРЯЖЕНИЮ

ЦЕЛЬ РАБОТЫ

Целью настоящей лабораторной работы является практическое изучение основных статических и динамических характеристик обрат-

ноходового преобразователя постоянного напряжения с обратной связью по выходному напряжению и цифровой системой управления.

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

Рассчитать регулятор напряжения по схеме рис. 4.1 с выходной мощностью $P_H = 10$ (Вт), выходным напряжением 10 В, входным напряжением 10 В. По результатам расчета определить параметры малосигнальной модели, амплитудную и частотную характеристики для П и ПИ-регуляторов. В программе Simulink (MATLAB) реализовать малосигнальную и реальную модели преобразователя, при частоте работы ключевого элемента $f = 20$ кГц (частота квантования $f_s = 200$ кГц для цифрового регулятора). По моделям определить границу устойчивости П- и ПИ-регулятора и снять динамические характеристики при скачкообразном изменении нагрузки (уменьшение/увеличение на 50%) и входного напряжения (уменьшение/увеличение на 30%). Сравнить характеристики между собой и сделать вывод.

ИСПОЛЬЗУЕМЫЕ ЭЛЕМЕНТЫ

Модульные:

- Отладочный комплект А1 – ezDSP28xx.
- Модуль согласования А2.
- Модуль транзисторных ключей А3.
- Резисторы: $R_H = 36$ Ом – 2 шт.
- Конденсатор С1 – 100 мкФ 63 В
- Дроссель L1 – 360 мкГн.
- Шунт для измерения тока $R_{ш} = 0.1$ Ом.
- Набор соединительных проводов – 15 шт.

ПРОГРАММА РАБОТЫ

- 2.1. Собрать схему исследуемого регулятора согласно рис. 4.1.
- 2.2. Реализовав П- и ПИ-алгоритмы регулирования, изменяя сопротивление нагрузки, снять внешнюю характеристику преобразователя.
- 2.3. Изменяя коэффициент пропорциональности определить границу устойчивой работы, введя интегральную составляющую определить ее влияние на качество регулирования и устойчивость работы

преобразователя по сравнению с чисто пропорциональным алгоритмом регулирования.

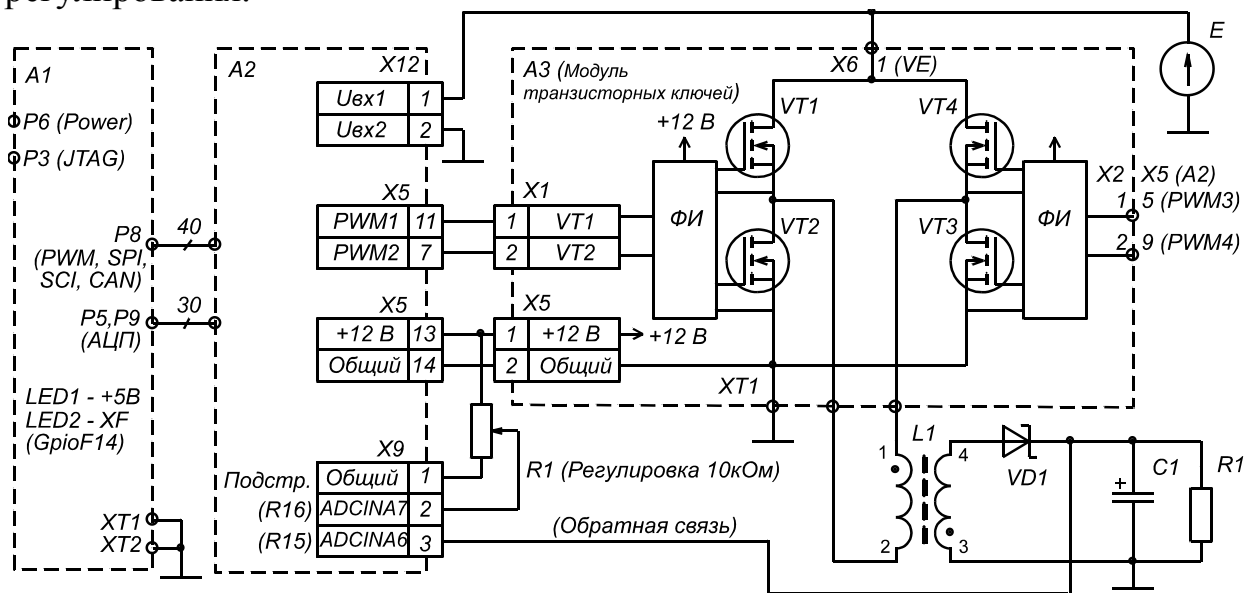


Рис.4.1

СОДЕРЖАНИЕ ОТЧЕТА

- 2.4. Название и программа работы.
- 2.5. Исследуемые схемы, диаграммы, таблицы.
- 2.6. Выводы по работе и ответ на контрольный вопрос.
- 2.7. Выполненное предварительное задание.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните методику получения малосигнальной модели преобразователя напряжения инвертирующего типа в режиме непрерывного тока дросселя. Сравните параметры модели с моделями преобразователей понижающего и повышающего типов.
2. Поясните статические и динамические характеристики импульсных преобразователей. Как структура и параметры закона управления влияют на эти характеристики?
3. Получите модель инвертирующего преобразователя напряжения работающего в режиме прерывистого тока дросселя. Сравните частотные характеристики для режима непрерывного и прерывистого тока дросселя.
4. Поясните влияние активного сопротивления дросселя на энергетические процессы в схеме. Выведите выражение для коэффициента

- передачи с учетом только этого сопротивления и определите максимально достижимое его значение.
5. Как нелинейность зависимости коэффициента передачи влияет на построение системы управления преобразователями постоянного напряжения повышающего и инвертирующего типов?
 6. Поясните структуру системы управления с цифровым регулятором. Расскажите о способах реализации цифрового регулятора на примере процессора TMS320F2812.
 7. Поясните, каким образом в цифровом регуляторе осуществляется интегрирование и дифференцирование сигналов. Приведите диаграммы и основные соотношения.
 8. Приведите варианты схем инвертирующих преобразователей с гальванической развязкой выходного напряжения. Поясните принцип их действия и получите малосигнальные модели этих преобразователей.
 9. Приведите вариант квазирезонансного преобразователя на основе схемы инвертирующего преобразователя постоянного напряжения. Поясните принцип его работы и приведите основные диаграммы и расчетные соотношения.
 10. Расскажите о z-преобразовании. Приведите основные определения и теоремы z-преобразования.

ЛАБОРАТОРНАЯ РАБОТА № 5

ИССЛЕДОВАНИЕ МОСТОВОГО ИНВЕРТОРА НАПРЯЖЕНИЯ С ВЫХОДНЫМ LC-ФИЛЬТРОМ И МНОГОКРАТНОЙ ШИРОТНО-ИМПУЛЬСНОЙ МОДУЛЯЦИЕЙ ВЫХОДНОГО НАПРЯЖЕНИЯ

ЦЕЛЬ РАБОТЫ

Целью настоящей лабораторной работы является практическое изучение основных статических характеристик мостового инвертора напряжения с многократной широтно-импульсной модуляцией выходного напряжения.

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

Рассчитать мостовой инвертор напряжения по схеме рис. 5.1 с выходной мощностью $P_H = 10$ (Вт), выходным напряжением 10 В. В программе Simulink (MATLAB) реализовать реальную модель преобразова-

теля, при частоте работы ключевого элемента $f = 20$ кГц (частота квантования $f_s = 200$ кГц для цифрового регулятора). В модели реализовать равномерную многократную ШИМ по синусоидальной функции построения, а также неравномерную многократную ШИМ с различным числом импульсных интервалов. Для различного числа импульсных интервалов рассчитать и промоделировать параметры выходного Г-образного и резонансного LC-фильтра. Подготовить алгоритмы и программы для реализации программы лабораторной работы с использованием отладочного комплекта ezDSP28xx.

ИСПОЛЬЗУЕМЫЕ ЭЛЕМЕНТЫ

Модульные:

- Отладочный комплект А1 – ezDSP28xx.
- Модуль согласования А2.
- Модуль транзисторных ключей А3.
- Трансформатор TV1.
- Резисторы: $R_H = 36$ Ом – 2 шт.
- Конденсатор С1 – 1 мкФ 63 В
- Дроссель L1 – 360 мкГн.
- Шунт для измерения тока $R_{ш} = 0.1$ Ом.
- Набор соединительных проводов – 15 шт.

ПРОГРАММА РАБОТЫ

2.1. Собрать схему исследуемого регулятора согласно рис. 5.1. Частоту работы инвертора задать равной 20 кГц, а коэффициент заполнения импульса изменять сопротивлением R1.

2.2. Реализовать режим однократной однополярной модуляции. Снять и сфазировать осциллограммы напряжения и тока элементов силовой части дискретно изменяя коэффициент заполнения. С помощью цифрового осциллографа определить гармонический состав напряжения до и после фильтра. Рассчитать по измеренным гармоникам коэффициент гармоник и сравнить его значение с теоретически рассчитанными значениями.

2.3. Реализовать режим многократной однополярной модуляции выходного напряжения по синусоидальной функции построения. Снять и сфазировать осциллограммы напряжения и тока элементов силовой части для различных значений коэффициента (8-10 точек) заполнения и различного числа импульсных интервалов (5, 10, 15). С помощью циф-

рового осциллографа определить гармонический состав напряжения до и после фильтра. Рассчитать по измеренным гармоникам коэффициент гармоник и сравнить его значение с теоретически рассчитанными значениями.

2.4. Реализовать режим многократной однополярной неравномерной модуляции выходного напряжения для различного числа импульсных интервалов. С помощью цифрового осциллографа определить гармонический состав напряжения до и после фильтра. Рассчитать по измеренным гармоникам коэффициент гармоник и сравнить его значение с теоретически рассчитанными значениями и со значениями, полученными и рассчитанными для равномерной модуляции.

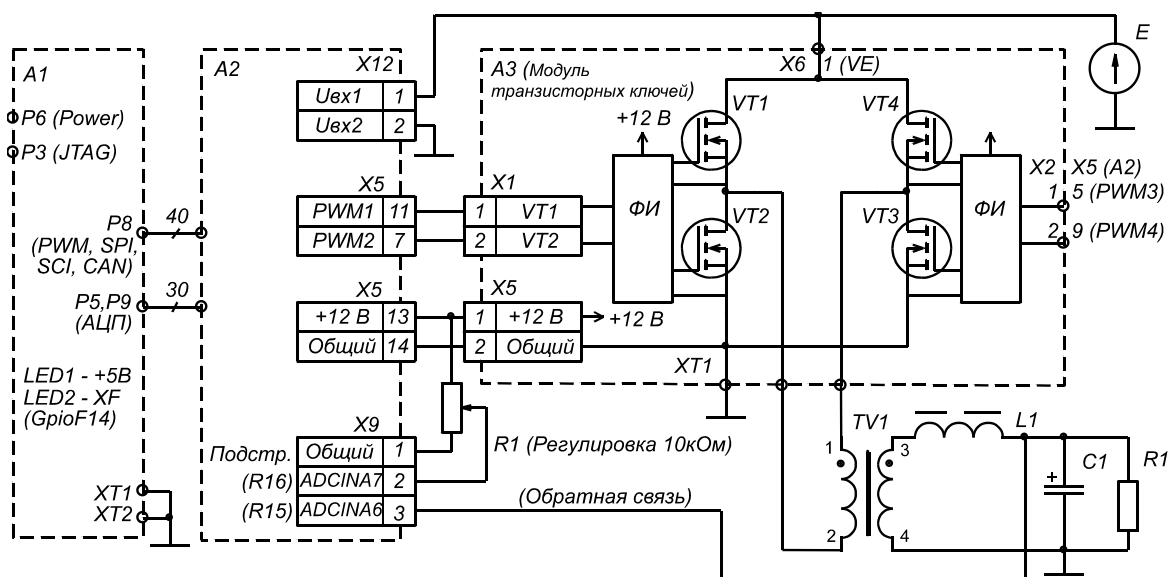


Рис.5.1

СОДЕРЖАНИЕ ОТЧЕТА

- 2.5. Название и программа работы.
- 2.6. Исследуемые схемы, диаграммы, таблицы.
- 2.7. Выводы по работе и ответ на контрольный вопрос.
- 2.8. Выполненное предварительное задание.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Расскажите о разложении периодической функции в ряд Фурье. Поясните понятие коэффициента гармоник. Для режима однократной ШИМ выведите выражение для коэффициента гармоник.
2. Для режим однократной ШИМ получите выражения для тока дросселя и выходного напряжения.

3. Расскажите о способах формирования выходного напряжения инверторов на основе АИМ.
4. Расскажите о способах формирования выходного напряжения инверторов на основе ШИМ.
5. Поясните влияние характера нагрузки на коэффициент гармоник выходного напряжения при наличии Г-образного LC-фильтра и однократной ШИМ.
6. Приведите варианты и анализ схем выходных фильтров инверторов напряжения с синусоидальным выходным напряжением.
7. Расскажите об основных способах организации обратной связи с использованием цифрового регулятора в схемах инверторов.
8. Приведите варианты схем инверторов тока. Поясните принцип их действия и область применения.
9. Приведите вариант квазирезонансного преобразователя на основе схемы мостового инвертора напряжения. Поясните принцип его работы и приведите основные диаграммы и расчетные соотношения.
10. Поясните, какими способами в схемах инверторов осуществляется защита ключевых элементов от перенапряжения и сквозного тока. Приведите схемы и диаграммы работы.

ЛАБОРАТОРНАЯ РАБОТА № 6

ИССЛЕДОВАНИЕ МОСТОВОГО РЕЗОНАНСНОГО ИНВЕРТОРА НАПЯЖЕНИЯ С ФАЗОВОЙ АВТОПОДСТРОЙКОЙ ЧАСТОТЫ

ЦЕЛЬ РАБОТЫ

Целью настоящей лабораторной работы является практическое изучение основных статических характеристик мостового резонансного инвертора напряжения с ФАПЧ. В программе Simulink (MATLAB) реализовать реальную модель преобразователя, при частоте работы инвертора $f = 10$ кГц (частота квантования $f_s = 200$ кГц для цифрового регулятора). Подготовить алгоритмы и программы для реализации программы лабораторной работы с использованием отладочного комплекта ezDSP28xx.

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

Рассчитать резонансный инвертор напряжения по схеме рис. 6.1 с выходной мощностью $P_H = 10$ (Вт), входным напряжением 15 В.

ИСПОЛЬЗУЕМЫЕ ЭЛЕМЕНТЫ

Модульные:

- Отладочный комплект А1 – ezDSP28xx.
- Модуль согласования А2.
- Модуль транзисторных ключей А3.
- Резисторы: $R_H = 36 \text{ Ом}$ – 2 шт.
- Конденсатор С1 – 100 мкФ 63 В
- Дроссель L1 – 360 мкГн.
- Шунт для измерения тока $R_{ш} = 0.1 \text{ Ом}$.
- Набор соединительных проводов – 15 шт.

ПРОГРАММА РАБОТЫ

2.1. Собрать схему исследуемого регулятора согласно рис. 6.1. Написать простую программу управления с формированием прямоугольного равномерного напряжения с изменяемой частотой в диапазоне 5-25кГц (регулировка осуществляется резистором R1)

2.2. Изменяя регулировочным резистором R1 частоту работы снять и сфазировать осциллограммы напряжения и тока на основных элементах схемы для различного режимов: $a - f > fp$; $f = fp$; $f < fp$. Здесь fp – резонансная частота колебательного контура.

2.3. Реализовать алгоритм фазовой автоподстройки частоты и проверить его работу изменяя сопротивление нагрузки. На одном графике зарисовать диаграмму контурного тока для различных значений добротности колебательного контура.

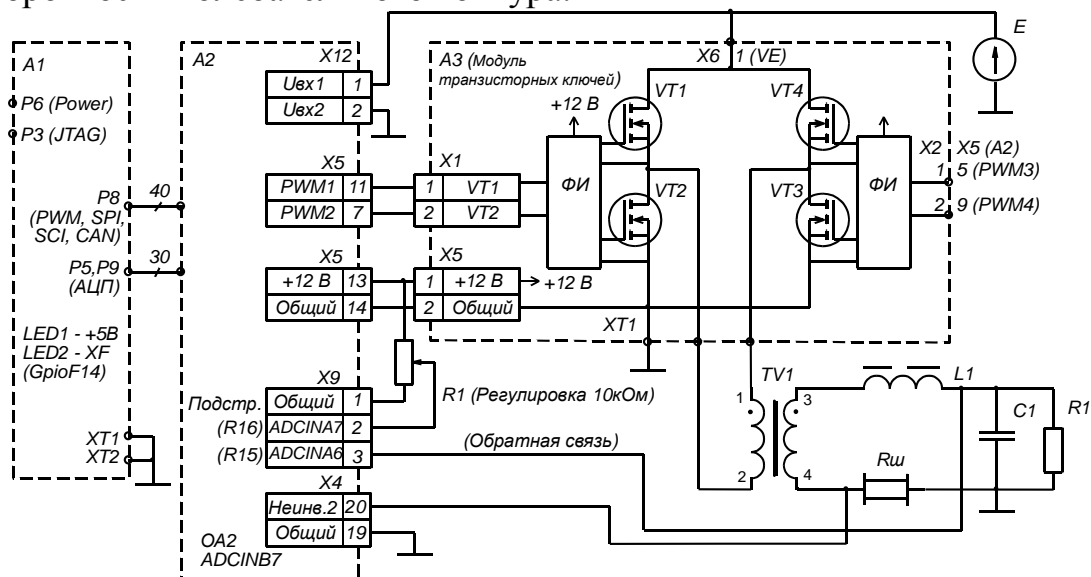


Рис. 6.1

СОДЕРЖАНИЕ ОТЧЕТА

- 2.4. Название и программа работы.
- 2.5. Исследуемые схемы, диаграммы, таблицы.
- 2.6. Выводы по работе.
- 2.7. Выполненное предварительное задание.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните особенности резонансных инверторов по отношению к другим типам инверторов. Приведите основные схемы резонансных инверторов напряжения и тока.
2. Получите выражения для тока дросселя и выходного напряжения.
3. Поясните основные способы регулирования выходного напряжения в резонансных инверторах. Приведите структурные схемы и поясните алгоритмы реализации с точки зрения цифрового регулятора.
4. Расскажите о фазовой автоподстройке частоты и способах ее организации с помощью аналоговых и цифровых регуляторов.
5. Расскажите о резонансных инверторах тока.

ЛАБОРАТОРНАЯ РАБОТА № 7

ИССЛЕДОВАНИЕ ТРЕХФАЗНОГО МОСТОВОГО ИНВЕРТОРА НАПРЯЖЕНИЯ

ЦЕЛЬ РАБОТЫ

Целью настоящей лабораторной работы является практическое изучение основных статических и динамических характеристик трехфазного мостового инвертора напряжения.

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

Смоделировать в программе Simulink (MATLAB) схему трехфазного инвертора, работающего на R - LR нагрузку для режимов: 180; 120; многократная ШИМ по синусоидальной функции построения. Смоделировать работу трехфазного инвертора на двигатель переменного тока с векторным регулированием. Подготовить алгоритмы и программы для реализации программы лабораторной работы с использованием отладочного комплекта ezDSP28xx.

ИСПОЛЬЗУЕМЫЕ ЭЛЕМЕНТЫ

Модульные:

- Отладочный комплект А1 – ezDSP28xx.
- Модуль согласования А2.
- Модуль транзисторных ключей А3.
- Резисторы: $R_1 \dots R_3 = 36 \text{ Ом}$ – 3 шт.
- Дроссель $L_1..L_3$ – 360 мкГн.
- Шунт для измерения тока $R_{ш} = 0.1 \text{ Ом}$.
- Набор соединительных проводов – 15 шт.

ПРОГРАММА РАБОТЫ

2.1. Собрать схему исследуемого регулятора согласно рис. 7.1. без дросселей (чисто активная нагрузка).

2.2. Снять и сфазировать фазные и линейные напряжения для трех режимов (180, 120, многократная ШИМ по синусоидальной функции построения).

2.3. Подключить дроссели $L_1 \dots L_3$ и повторить измерения пункта 2.2, добавив диаграммы напряжения на нагрузке.

2.4. Изменяя коэффициент заполнения импульса для многократной ШИМ (резистор R_1), снять цифровым вольтметром зависимость действующих значений фазного и линейного напряжений от коэффициента заполнения. Для 3-4 точек зарисовать диаграммы тока и напряжения.

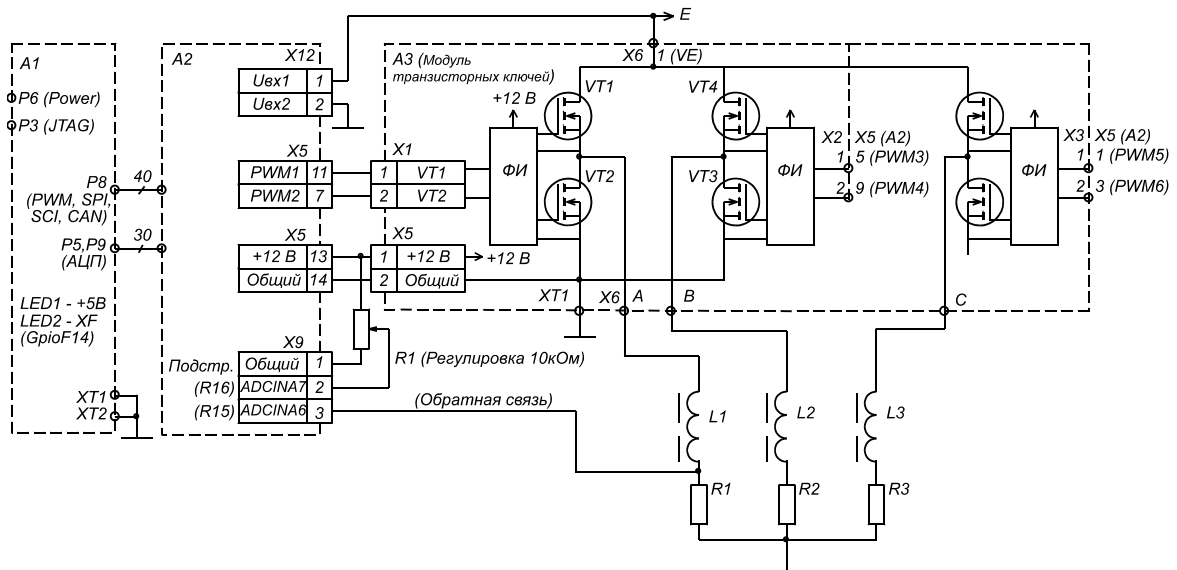


Рис. 7.1

СОДЕРЖАНИЕ ОТЧЕТА

- 2.5. Название и программа работы.
- 2.6. Исследуемые схемы, диаграммы, таблицы.
- 2.7. Выводы по работе.
- 2.8. Выполненное предварительное задание.

СОЗДАНИЕ ПРОЕКТА И ОПИСАНИЕ ОСНОВНЫХ ФУНКЦИОНАЛЬНЫХ БЛОКОВ ПРОЦЕССОРА TMS320F28XX

1. СОЗДАНИЕ ПРОЕКТА

Один из способов начать работу заключается в использовании файлов шаблонов, которые содержат описание структур для работы с периферийными устройствами, такими как порты ввода-вывода, менеджер событий, АЦП и пр. Для семейства 281x эти шаблоны и примеры проектов находятся в архиве sprc097.zip. После установки получаются четыре каталога содержащие документацию, примеры и файлы заголовков.

Для того чтобы создать новый проект нужно сделать следующее:

1. Запустить среду программирования Code Composer Studio (CCS) и создать новый проект (Project – New). Вписать имя проекта (автоматически создастся одноименная поддиректория), тип проекта (Executable) и тип прибора (TMS320C28xx).

2. Скопировать в созданную директорию каталоги \DSP281x_common и \DSP281x_headers, которые находятся в общем рабочем каталоге D:\DSP\.

3. Создать исходный файл проекта и подключить его (Project – Add files to project, или кликнуть правой клавишей мышки на заголовке проекта и в выпадающем меню выбрать тот же пункт – Add files to project).

* Обратите внимание на расширение файла, его нужно писать. Копирование и создание файла можно сделать Проводником, например. Другой способ создания файла – скопировать исходный файл из какого-нибудь примера, каталог \DSP281x_examples\xxx\, и отредактировать его.

В предлагаемом практикуме в качестве исходного файла используется файл шаблона template.c, который необходимо скопировать из общего каталога D:\DSP\ и переименовать его так как вам удобно.

4. Включить в проект файлы заголовков \DSP281x_headers\include\DSP281x_Device.h и \DSP281x_common\include\DSP281x_Examples.h. Для этого в исходном файле должно быть включение этих файлов, т.е. должно быть следующее

```
#include "DSP281x_Device.h"
```

```
#include "DSP281x_Examples.h"
```

5. Прописать пути поиска для этих файлов. (Project – Build Options – Compiler tab – Preprocessor (Category)) В поле Include search path (-i) вписать через точку с запятой полные пути (только каталоги) к этим файлам (можно, например, скопировать из Проводника). Правая клавиша на заголовке сработает и в этом случае.

6. Подключить к проекту следующие исходные файлы:
\\DSP281x_headers\\source\\DSP281x_GlobalVariableDefs.c

\\DSP281x_common\\source\\DSP281x_SysCtrl.c

\\DSP281x_common\\source\\DSP281x_DefaultIsr.c

\\DSP281x_common\\source\\DSP281x_PieCtrl.c

\\DSP281x_common\\source\\DSP281x_PieVect.c

\\DSP281x_common\\source\\DSP281x_SysCtrl.c

6. Подключить командный файл компилятора для заголовочных файлов и основного проекта. В нашем случае это – \\DSP281x_headers\\cmd\\DSP281x_Headers_nonBIOS.cmd (или DSP281x_Headers_BIOS.cmd если используется DSP\\BIOS) и F2812_EzDSP_RAM_lnk.cmd.

7. Подключить библиотеку:

X:\\CCStudio_v3.1\\C2000\\cgtools\\lib\\rts2800_ml.lib (путь может несколько отличаться, в зависимости от компьютера)

2. ТАКТИРОВАНИЕ ПРОЦЕССОРА

2.1. Блоки OSC и PLL. Процессоры семейства TMS320F28xx содержат в своем составе модуль ФАПЧ (фазовой автоподстройки частоты, PLL), который обеспечивает тактирование всех блоков процессора. Этот модуль может работать либо с кварцевым резонатором от собственного генератора (OSC), либо от внешнего генератора (подсоединяется к выводу XTAL1/CLKIN). Структурная схема показана на рис.1.1. С точки зрения работы модуля PLL выделяются два режима работы – включен PLL и выключен. Режим определяется уровнем на выводе XPLLDIS в момент сброса процессора (используется триггер, который сохраняет значение XPLLDIS) . Низкий уровень XPLLDIS выключает модуль PLL, а высокий, соответственно включает (если оставить вывод не подключенным, то из-за наличия подтягивающего резистора модуль PLL включится).

Во включенном состоянии входной сигнал умножается на значение регистра PLLCR (4 младших разряда – битовое поле DIV) и делится

на два (рис.7). Таким образом, тактовая частота процессора определяется следующим выражением

$$\text{CLKIN} = \text{OSCCLK} * \text{PLL} * \text{DIV} / 2.$$

В том случае, когда $\text{PLL} = 0$ умножения не происходит, но деление на 2 остается, $\text{CLKIN} = \text{OSCCLK} / 2$.

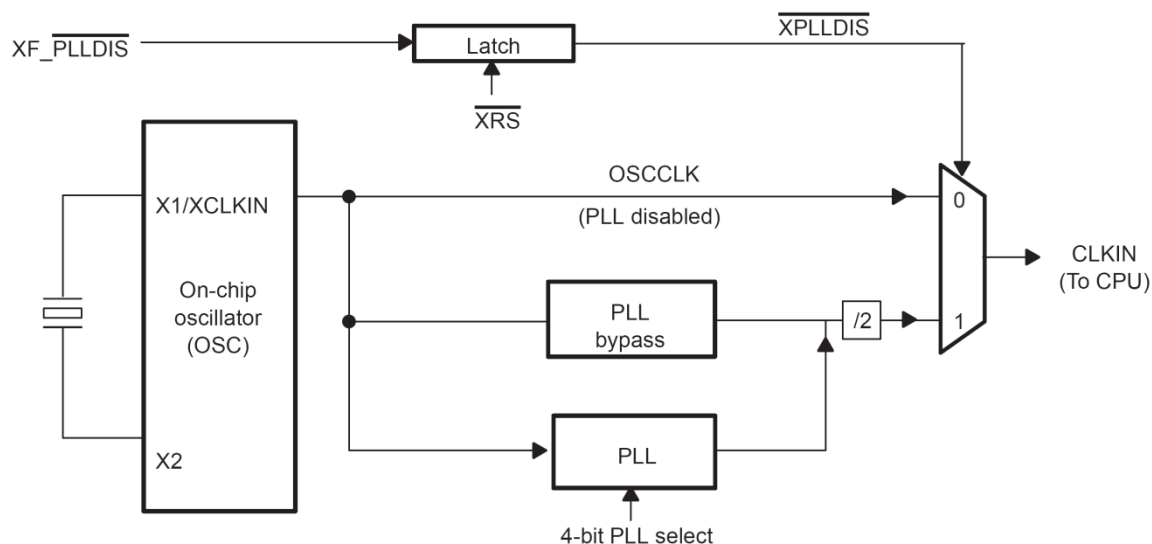


Рис.7

Периферийные устройства тактируются с помощью дополнительных делителей. Делитель, формирующий сигнал LSPCLK определяет работу SCI, SPI, McBSP, а делитель, формирующий сигнал HSPCLK тактирует EVA(B) и ADC. Регистры, конфигурирующие эти делители – HISPCP (битовое поле HSPCLK0:2) и LOSPCP (битовое поле LSPCLK0:2).

$$\text{LSPCLK} = \begin{cases} \frac{\text{SYSCLKOUT}}{2 * \text{LOSPCP}}, & \text{LOSPCP} \neq 0 \\ \text{SYSCLKOUT}, & \text{LOSPCP} = 0 \end{cases} \quad (\text{SYSCLKOUT}/4 - \text{по умолчанию})$$

$$\text{HSPCLK} = \begin{cases} \frac{\text{SYSCLKOUT}}{2 * \text{HISPCP}}, & \text{HISPCP} \neq 0 \\ \text{SYSCLKOUT}, & \text{HISPCP} = 0 \end{cases} \quad (\text{SYSCLKOUT}/2 - \text{по умолчанию})$$

Для уменьшения потребляемой мощности тактирование периферии может быть выключено путем сброса соответствующего бита в регистре PCLKCR.

*Все регистры сгруппированы в объединении SysCtrlRegs.xx и настраиваются в файле DSP281x_SysCtrl.c (Основная функция – InitSysCtrl()).

2.2. Тактирование ADC и EV. Как было замечено выше, тактирование (синхронизация) блоков аналого-цифрового преобразователя(ADC) и менеджера событий (EV) осуществляется сигналом HSPCLK. В дальнейшем этот сигнал проходит через делители частоты, коэффициенты деления которых, настраиваются в соответствующих периферийных регистрах. Если говорить о менеджере событий, то тактирующим сигналом для него могут выступать: внутренний сигнал процессора HSPCLK, внешний тактовый сигнал TCLKIN или сигнал от энкодера QEP, пропущенные через делитель. Настройка типа сигнала синхронизации и коэффициента деления осуществляется в регистре контроля TxCON, битовые поля TCLKS[1-0] и TPS[2-0] соответственно. Коэффициент деления определяется по следующему выражению $Tclk = x/2^{TPS}$.

Структурная схема тактирования блока ADC показана на рис.8. Сигнал HSPCLK проходит через 4-х битный делитель частоты ADCTRL3.ADCLKPS. Затем этот сигнал делится на два в случае, если бит ADCTRL1.CPS=1 и поступает в качестве тактового сигнала на АЦП – ADCCLK и на генератор ширины импульса захвата, который определяет длительность (число тактов – ADCTRL.ACQPS) заряда конденсатора устройства выборки и хранения (УВХ, S/H). Частота тактирования АЦП определяется по следующему выражению

$$ADCCLK = HSPCLK / 2 \times ADCLKPS[3 - 0] \times (1 + CPS)$$

за исключением $ADCLKPS[3-0]=0$. Для этого случая коэффициент деления равен 1, если $ADCTRL1.CPS=0$ и 1/2, если $ADCTRL1.CPS=1$.

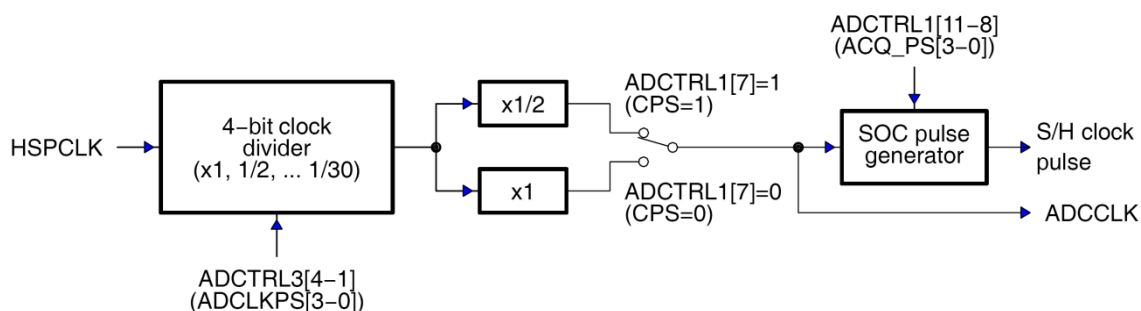


Рис.8

3. ПОРТЫ ВВОДА-ВЫВОДА [3]

Порты в семействах F281x и F280x несколько отличаются друг от друга. В семействе F281x функция порта переключается между обслуживанием стандартного ввода-вывода и обслуживанием выводов пери-

ферийных устройств, таких как интерфейсы SPI, CAN и пр., причем максимально к каждому выводу может быть подсоединено только одно периферийное устройство. Структурную схему можно увидеть на рис.9.

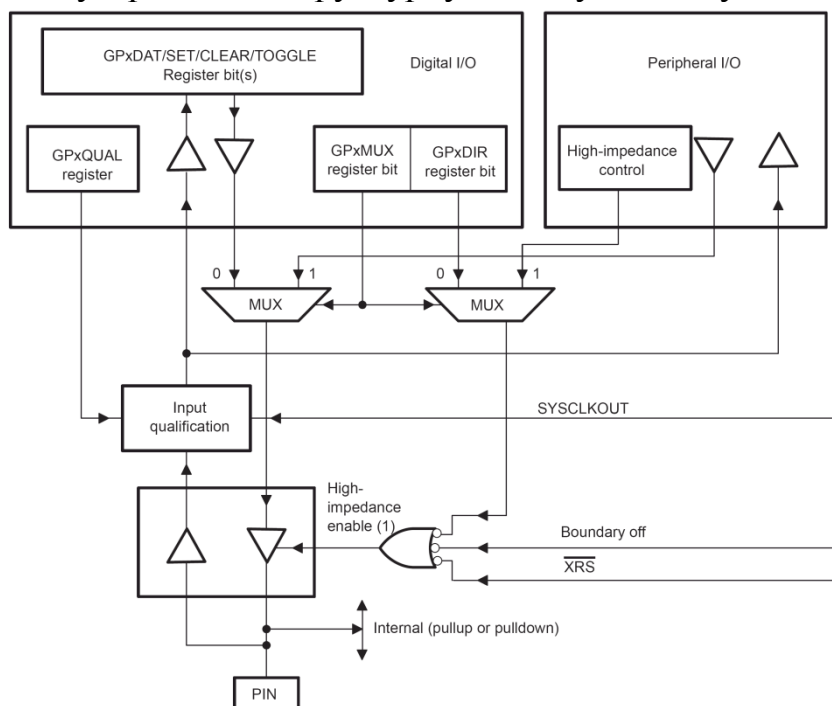


Рис.9

Как можно видеть из схемы, в независимости от того, что в данный момент подключено входной сигнал поступает на входные буферы порта ввода-вывода, и периферийного устройства. Поэтому, с одной стороны необходимо контролировать работу периферийного устройства, так чтобы при переключении вывода в режим порта ввода-вывода не произошло нежелательного прерывания. С другой стороны в любой момент состояние вывода может быть проконтролировано через регистр GPxDAT. Также, входной сигнал для ограничения влияния шума может быть подвержен определенного рода фильтрации путем многократного чтения (6 успешных чтений) с регулируемой частотой. Работа этого блока поясняется на рис.10.

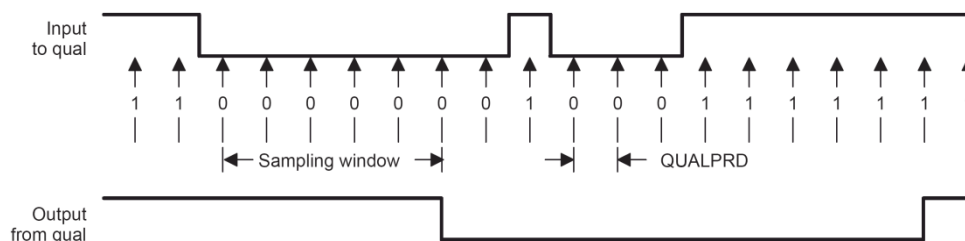


Рис.10

Видно, что сигнал проходит на вход буферов порта или периферии только после шести успешных чтений, период которых определяется для каждого из портов в соответствующем регистре – GPxQUAL.

Работа порта определяется набором следующих контролируемых регистров:

1) В TMS320F281x каждый из портов имеет один регистр мультиплексор (GPxMUX), который определяет, что будет работать – периферия или общий ввод-вывод. В начальный момент порты настроены на общий ввод-вывод (ввод).

GpioMuxRegs.GPxMUX.bit = 0; – I/O

GpioMuxRegs.GPxMUX.bit = 1; – периферия для этого вывода

*структуры похожие на GpioMuxRegs определены для всех периферийных устройств в соответствующих файлах заголовках и подключаются путем включения в проект родительского файла x:\work_directory\DSP281x_headers\DSP281x_Device.h.

2) Регистр направления, определяет, будет ли порт работать на ввод или на вывод.

GpioMuxRegs.GPxDIR.bit = 0; – input

GpioMuxRegs.GPxDIR.bit = 1; – output

3) Регистр данных. Отражает текущее состояние входного сигнала после входной обработки (многократное постоянное чтение, значение не меняется).

GpioDataRegs.GPxDAT.bit = 0; AND pin output, то выход становится равен 0

GpioDataRegs.GPxDAT.bit = 1; AND pin output, то выход становится равен 1

4) Регистр установки.

GpioDataRegs.GPxSET.bit = 0; игнорируется

GpioDataRegs.GPxSET.bit = 1; AND pin output, то выход становится равен 1

5) Регистр очистки.

GpioDataRegs.GPxCLEAR.bit = 0; игнорируется
 GpioDataRegs.GPxCLEAR.bit = 1; AND pin output, то выход становится равен 0

б) Регистр переключения.

GpioDataRegs.GPxTOGGLE.bit = 0; игнорируется
 GpioDataRegs.GPxTOGGLE.bit = 1; AND pin is output, то выход переключается в противоположное состояние

7) Регистр многократной выборки (6 выборок) GpioMuxRegs.GPxQUAL. Этот регистр определяет период выборки.
 15-8 – зарезервированы;
 7-0 – QUALPRD: 0x00 – SYSCLK; 0x01 – 2 SYSCLK; 0x02 – 4 SYSCLK; ... 0xFF – 510 SYSCLK.

Примечания.

а) x – номер (буква) порта: А, В, D, Е, F, G.

б) Допускается работать целиком с портом, например: GPxMUX.all = 0;

в) Регистры мультиплексоры и регистры многократной выборки находятся в защищенной области памяти. Для доступа необходима следующая конструкция:

EALLOW; //открываем доступ

...; //настраиваем порты

EDIS; //закрываем доступ

Соответствие между номером порта и функцией периферийного устройства для семейства F281x приведены в табл. 2.1.

Таблица 2.1

GPxMUX, GPxDIR бит	Периферия	Порт	Тип	Состояние после сброса
GPAMUX, GPADIR (EV-A Peripheral)				
0	PWM1 (O)	GPIOA0	R/W	0
1	PWM2 (O)	GPIOA1	R/W	0
2	PWM3 (O)	GPIOA2	R/W	0
3	PWM4 (O)	GPIOA3	R/W	0
4	PWM5 (O)	GPIOA4	R/W	0
5	PWM6 (O)	GPIOA5	R/W	0
6	T1PWM_T1CMP (O)	GPIOA6	R/W	0
7	T2PWM_T2CMP (O)	GPIOA7	R/W	0

8	CAP1_QEP1 (I)	GPIOA8	R/W	0
9	CAP2_QEP2 (I)	GPIOA9	R/W	0
10	CAP3_QEPI1 (I)	GPIOA10	R/W	0
11	TDIRA (I)	GPIOA11	R/W	0
12	TCLKINA (I)	GPIOA12	R/W	0
13	C1TRIP (I)	GPIOA13	R/W	0
14	C2TRIP (I)	GPIOA14	R/W	0
15	C3TRIP (I)	GPIOA15	R/W	0
GPBMUX, GPBDIR (EV-B Peripheral)				
0	PWM7 (O)	GPIOB0	R/W	0
1	PWM8 (O)	GPIOB1	R/W	0
2	PWM9 (O)	GPIOB2	R/W	0
3	PWM10 (O)	GPIOB3	R/W	0
4	PWM11 (O)	GPIOB4	R/W	0
5	PWM12 (O)	GPIOB5	R/W	0
6	T3PWM_T3CMP (O)	GPIOB6	R/W	0
7	T4PWM_T4CMP (O)	GPIOB7	R/W	0
8	CAP4_QEP3 (I)	GPIOB8	R/W	0
9	CAP5_QEP4 (I)	GPIOB9	R/W	0
10	CAP6_QEPI2 (I)	GPIOB10	R/W	0
11	TDIRB (I)	GPIOB11	R/W	0
12	TCLKINB (I)	GPIOB12	R/W	0
13	C4TRIP (I)	GPIOB13	R/W	0
14	C5TRIP (I)	GPIOB14	R/W	0
15	C6TRIP (I)	GPIOB15	R/W	0
GPDMUX, GPDDIR				
0	T1CTRIP_PDPINTA (I)	GPIOD0	R/W	0
1	T2CTRIP (I)	GPIOD1	R/W	0
2 – 4	Зарезерв.	GPIOD2-4	-	-
5	T3CTRIP_PDPINTB (I)	GPIOD5	R/W	-
6	T4CTRIP (I)	GPIOD6	R/W	-
7 – 15	Зарезерв.	GPIOD7-15	-	-
GPEMUX, GPEDIR				
0	XINT1_XBIO (I)	GPIOE0	R/W	0
1	XINT2_ADCSOC (I)	GPIOE1	R/W	0
2	XNMI_XINT13 (I)	GPIOE2	R/W	0
3 – 15	Зарезерв.	GPIOE3-15	3,4 R, остал.R=0	0
GPFMUX, GPFDIR				
<i>SPI Peripheral</i>				
0	SPISIMO (O)	GPIOF0	R/W	0
1	SPISOMI (I)	GPIOF1	R/W	0
2	SPICLK (I/O)	GPIOF2	R/W	0
3	SPISTE (I/O)	GPIOF3	R/W	0
<i>SCIA Peripheral</i>				
4	SCITXDA (O)	GPIOF4	R/W	0
5	SCIRXDA (I)	GPIOF5	R/W	0

<i>CAN Peripheral</i>				
6	CANTX (O)	GPIOF6	R/W	0
7	CANRX (I)	GPIOF7	R/W	0
<i>McBSP Peripheral</i>				
8	MCLKX (I/O)	GPIOF8	R/W	0
9	MCLKR (I/O)	GPIOF9	R/W	0
10	MFSX (I/O)	GPIOF10	R/W	0
11	MFSR (I/O)	GPIOF11	R/W	0
12	MDX (O)	GPIOF12	R/W	0
13	MDR (I)	GPIOF13	R/W	0
<i>XF CPU Output Signal</i>				
14	XF (O)	GPIOF14	R/W	0
GPGMUX, GPGDIR				
0-3	Зарезерв.	GPIOG0-3	R	0
<i>SCI-B Peripheral</i>				
4	SCITXDB (O)	GPIOG4	R/W	0
5	SCIRXDB (I)	GPIOG5	R/W	0
6-15	Зарезерв.	GPIOG6-15	R	0

Семейство процессоров F280x имеет отличную от семейства F281x структуру порта [4], которая показана на рис.11. В основном отличия заключаются в более глубоком мультиплексировании функций порта. В зависимости от значения в регистре мультиплексоре (GRAMUX1, GRAMUX2, GPBMUX1) вывод порта может подключаться к одному из трех периферийных устройств, помимо основной функции порта ввода-вывода. Соответствие между номером порта и функцией периферийного устройства для семейства F280x приведены в табл. 2.2.

Таблица 2.2

GRAMUX1				
Bits	Bits = 0,0 (Default)	Bits = 0,1	Bits = 1,0	Bits = 1,1
1:0	GPIO0	EPWM1A(O)	Reserved	
3:2	GPIO1	EPWM1B(O)	SPISIMOD(I/O)	Reserved
5:4	GPIO2	EPWM2A(O)	Reserved	
7:6	GPIO3	EPWM2B(O)	SPISOMID(I/O)	Reserved
9:8	GPIO4	EPWM3A(O)	Reserved	
11:10	GPIO5	EPWM3B(O)	SPICLKD(I/O)	ECAP1(I/O)
13:12	GPIO6	EPWM4A(O)	EPWMSYNCl(I)	EPWMSYNCO(O)
15:14	GPIO7	EPWM4B(O)	SPISTED(I/O)	ECAP2(I/O)
17:16	GPIO8	EPWM5A(O)	CANTXB(O)	ADCSOCAO(O)
19:18	GPIO9	EPWM5B(O)	SCITXB(O)	ECAP3(I/O)
21:20	GPIO10	EPWM6A(O)	CANRXB(I)	ADCSOCBO(O)
23:22	GPIO11	EPWM6B(O)	SCIRXB(I)	ECAP4(I/O)
25:24	GPIO12	TZ1(I)	CANTXB(O)	SPISIMOB(I/O)
27:26	GPIO13	TZ2(I)	CANRXB(I)	SPISOMIB(I/O)
29:28	GPIO14	TZ3(I)	SCITXB(O)	SPICLKB(I/O)

31:30	GPIO15	TZ4(I)	SCIRXB(I)	SPISTEB(I/O)
GPAMUX2				
Bits	Bits = 0,0 (Default)	Bits = 0,1	Bits = 1,0	Bits = 1,1
1:0	GPIO16	SPISIMOA(I/O)	CANTXB(O)	TZ5(I)
3:2	GPIO17	SPISOMIA(I/O)	CANRXB(I)	TZ6(I)
5:4	GPIO18	SPICLKA(I/O)	SCITXB(O)	Reserved
7:6	GPIO19	SPISTEA(I/O)	SCIRXB(I)	Reserved
9:8	GPIO20	EQEP1A(I)	SPISIMOC(I/O)	CANTXB(O)
11:10	GPIO21	EQEP1B(I)	SPISOMIC(I/O)	CANRXB(I)
13:12	GPIO22	EQEP1S(I/O)	SPICLKC(I/O)	SCITXB(O)
15:14	GPIO23	EQEP1I(I/O)	SPISTEC(I/O)	SCIRXB(I)
17:16	GPIO24	ECAP1(I/O)	EQEP2A(I)	SPISIMOB(I/O)
19:18	GPIO25	ECAP2(I/O)	EQEP2B(I)	SPISOMIB(I/O)
21:20	GPIO26	ECAP3(I/O)	EQEP2I(I/O)	SPICLKB(I/O)
23:22	GPIO27	ECAP4(I/O)	EQEP2S(I/O)	SPISTEB(I/O)
25:24	GPIO28	SCIRXDA(I)	Reserved	TZ5(I)
27:26	GPIO29	SCITXDA(O)	Reserved	TZ6(I)
29:28	GPIO30	CANRXA(I)	Reserved	Reserved
31:30	GPIO31	CANTXA(O)	Reserved	Reserved
GPBMUX1				
1:0	GPIO32	SDAA(I/OC)	EPWMSYNCl(I)	ADCSOCAO(O)
3:2	GPIO33	SCLA(I/OC)	EPWMSYNCO(O)	ADCSOCBO(O)
5:4	GPIO34	Reserved	Reserved	Reserved

Другие отличия заключаются в следующем.

1. Более детализирована входная фильтрация.
2. Входной «подтягивающий» резистор может быть включен или выключен путем установки соответствующих битов в регистре GPx-PUD.
3. Входами внешних прерываний могут быть назначены различные выводы.

Следующие регистры управляют работой портов.

1) GPARCTRL – квалификатор

Бит	Название	Описание
31-24	QUALPRD3	Для всех полей: 0x00 – QUALPRD=SYSCLK; 0x01 – QUALPRD=SYSCLK/2; 0x02 – QUALPRD=SYSCLK/4; ...; 0xFF – QUALPRD=SYSCLK/510
23-16	QUALPRD2	
15-8	QUALPRD1	
7-0	QUALPRD0	

* EALLOW (защищен)

Бит	Название	Описание
31-30	GPIO15	00 – синхронизация с SYSCLK
29-28	GPIO14	01 – 3 выборки
...	...	10 – 6 выборок
1-0	GPIO0	11 – асинхронная работа (для периферии)

* EALLOW (защищен)

4) GPBSEL1 (GPBSEL2 – ?)

Бит	Название	Описание
31-8	Зарезерв.	00 – синхронизация с SYSCLK
7-6	GPIO35	01 – 3 выборки
5-4	GPIO34	10 – 6 выборок
3-2	GPIO33	11 – асинхронная работа (для периферии)
1-0	GPIO32	

* EALLOW (защищен)

5) GPADIR (GPBDIR) – настройка вывода

0 – GPIO настраивается как вход

1 – GPIO настраивается как выход

* EALLOW (защищен)

6) GPAPUD (GPBPUD) – регистр подтягивающих резисторов (после перезагрузки не включены)

0 – включает подтягивающий регистр

1 – выключает его

* EALLOW (защищен)

GPADAT (GPBDAT), GPASET(GPBSET), GPACLEAR (GPBCLEAR), GPATOGGLE (GPBTOGGLE) – то же что и в предыдущем процессоре.

7) GPIOXINT1SEL (16p) – поле GPIOSEL (4-0) определяет, какой порт А будет использован в качестве источника внешнего прерывания XINT1: 00000 – GPIO0 (по умолчанию); 00001 – GPIO1; и т.д.

8) GPIOXINT2SEL – аналогично для XINT2

9) GPIOXNMI – аналогично для XNMI (немаскируемое прерывание)

10) GPIOLPMSEL – регистр определяет сможет ли сигнал поданный на этот вывод (порт) «разбудить» процессор, который находится в режиме HALT и STANDBY: 1 – сможет.

* EALLOW (защищен)

3. МЕНЕДЖЕР СОБЫТИЙ (EVENT MANAGER)

В TMS320F2812 сигнальном процессоре таких менеджеров два, соответственно EVA и EVB. Функции, выполняемые этими менеджерами одинаковы. Модули, из которых состоят блоки менеджеров событий и соответствующие им основные сигналы отражены в табл.3.

Таблица 3

	EVA		EVB	
	Модуль	Сигнал	Модуль	Сигнал
Таймеры (16 разр.) (GP Timers)	GP Timer 1 GP Timer 2	T1PWM/T1CMP T2PWM/T2CMP	GP Timer 3 GP Timer 4	T3PWM/T3CMP T4PWM/T4CMP
Блоки сравнения (Compare units)	Compare 1 Compare 2 Compare 3	PWM1/2 PWM3/4 PWM5/6	Compare 4 Compare 5 Compare 6	PWM7/8 PWM9/10 PWM11/12
Блоки захвата (Capture units)	Capture 1 Capture 2 Capture 3	CAP1 CAP2 CAP3	Capture 4 Capture 5 Capture 6	CAP4 CAP5 CAP6
(Квадратурный энкодер) QEP channels	QEP	QEP1 QEP2 QEP11	QEP	QEP3 QEP4 QEP12
Внешние входы таймеров (External timer inputs)	Timer-direction external clock	TDIRA TCLKINA	Timer-direction external clock	TDIRB TCLKINB
Входы блокировки модуля сравнения	Compare	C1TRIP C2TRIP C3TRIP	Compare	C4TRIP C5TRIP C6TRIP
Входы блокировки модуля сравнения таймер		T1CTRIP/ T2CTRIP		T3CTRIP/ T4CTRIP
Вход общей блокировки		PDPINTA†		PDPINTB
Выходы запуска АЦП		EVASOC		EVBSOC

На рис.12 показана блок схема менеджера событий.

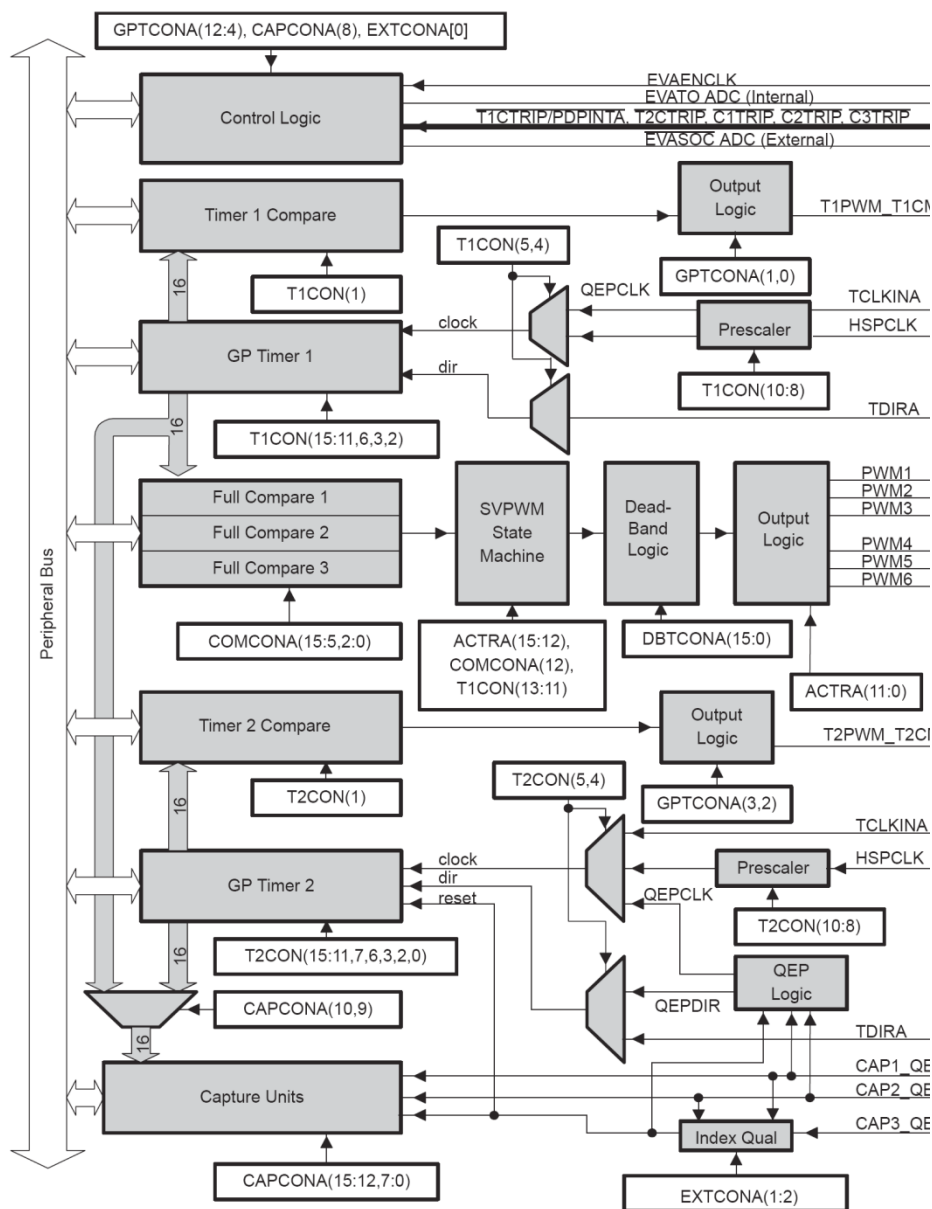


Рис.12

Как видно из табл.3 и рис.12 в блок менеджера событий входят следующие компоненты.

- 3.1. Два 16-ти разрядных таймера – GP Timer 1,2 со своими блоками (регистрами) сравнения Timer 1,2 Compare (выход T1PWM, T2PWM).
- 3.2. Три дополнительных модуля сравнения Full Compare1,2,3. Эти модули работают с таймером GP Timer 1 и формируют шесть дополнительных широтно-модулированных сигналов – PWM1-6. При формировании этих сигналов помимо выходной логики используется модуль формирования «мертвого» времени (Dead-Band Logic)

и модуль формирования последовательности сигналов для реализации режима векторного управления (SVPWM State Machine).

3.3. Блоки захвата – Capture Units, и связанный с ними модуль работы с квадратурным энкодером – QEP Logic.

Таймеры

В каждом менеджере есть два таймера общего назначения (16 разрядов). Работа таймеров и блоков сравнения поясняется на рис.13 и рис.14.

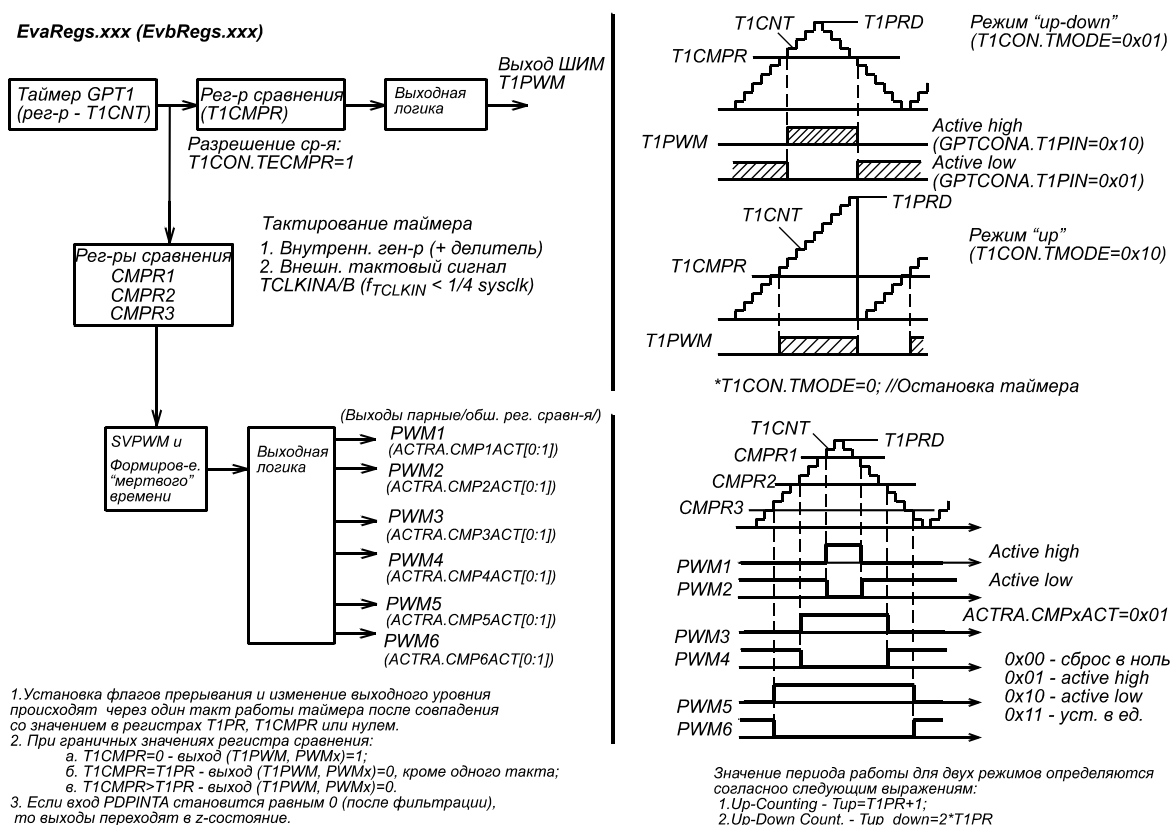


Рис.13

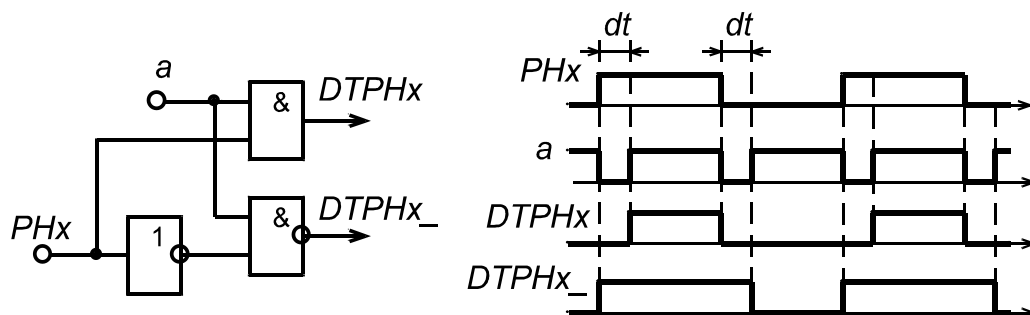


Рис.14. Схема и диаграмма формирования «мертвого» времени

В состав каждого таймера входят:

а) регистр-счетчик – **TCNTx** (текущее значение таймера x);

б) регистр сравнения – **TxCMP** (буферированный), а также только для таймеров 1 и 3 три дополнительных регистра сравнения – **CPMR1,2,3 (CMPR4,5,6)**;

в) регистр периода – **TxPR** (определяет максимальное значение и как следствие, частоту работы таймера);

г) регистр управления – **TxCON**;

Бит	Название	Описание	
15	Free	Определяют действие при запросе эмулятора. (Например точка остано- ва) 00 – останавливается сразу; 01 – останавливается по достижению периода; 10,11 – игнорируется запрос.	
14	Soft		
13	(Зарезерв.)		
12	TMODE1	Режим счета. 00 – Стоп (содержимое не меняется); 01 – вверх-вниз; 10 – вверх; 11 – вверх или вниз в зависимости от бита или вывода направ- ления (TDIRx).	
11	TMODE0		
10	TPS2	Делитель частоты входно- го сигнала. x=HSPCLK	
9	TPS1		000 – x/1 011 – x/8 110 – x/64
8	TPS0		001 – x/2 100 – x/16 111 – x/128
7	T2SWT1 (T4SWT3)	010 – x/4 101 – x/32	
7	T2SWT1 (T4SWT3)	Этот бит активен только для таймеров 2 и 4. Запускает T2 (T4) в тот момент, когда включается (разрешается работа) T1 (T3).	
6	TENABLE	1 – включение (разрешение) T1.	
5	TCLKS1	Источник синхронизации (генератор тактового сигнала). 00 – внутрен- ний (HSPCLK); 01 – внешний (TCLKINx); 10 – резерв; 11 – QEP.	
4	TCLKS0		
3	TCLD1	Условие при котором происходит загрузка регистра сравнения таймера (перегрузка из буферного, теневого регистра в основной). 00 – TxCNT=0; 01 – TxCNT=0 или TxCNT=TxPR; 10 – сразу; 11 – резерв.	
2	TCLD0		
1	TECMPR	1 – разрешает работу в режиме сравнения.	
0	SELT1PR (SELT3PR)	Этот бит активен только для таймеров 2 и 4. В случае, когда он равен 1, собственный регистр периода игнорируется, а в качестве одного исполь- зуется регистр периода T1 (T3).	

д) регистр контроля – **GPTCONA** (Для EVB {T3/T4} – **GPTCONB**)

Бит	Название	Описание
15	(Зарезерв.)	
14	T2STAT	Состояние таймера T2. 1 – счет вверх (увеличение); 0 – счет вниз (умень- шение).
13	T1STAT	Состояние таймера T1. 1 – счет вверх (увеличение); 0 – счет вниз (умень- шение)
12	T2CTRIPE	Разрешение блокировки выхода сравнения T2. 1 – включено, в этом слу- чае, когда вывод T2CTRIPE = 0, выход T2PWM\T2CMP блокируется (z- состояние), бит GPTCON(5) сбрасывается в 0, и устанавливается флаг PDPINT [EVIFRA(0)].

		<i>Этот бит активен при условии, что $EXTCON(0) = 1$.</i>		
11	T1CTRIPE	То же, что и для бита [12], только для T1.		
10	T2TOADC	Запуск АЦП по событию T2. 00 – никакое событие не запускает АЦП; 01 – установка флага прерывания по равенству нулю; 10 – установка флага прерывания по периоду; 11 – установка флага прерывания по сравнению.		
9	T2TOADC			
8	T1TOADC	То же, что и для битов [10-9], только для T1.		
7	T1TOADC			
6	T2CMPOE	Разрешение работы вывода по сравнению обоих таймеров. 0 – выходы TxPWM/TxCMP находятся в z-состоянии; 1 – эти выходы управляются блоком сравнения. <i>Этот бит активен если $EXTCON(0) = 0$. Этот бит, когда активен сбрасывается в 0 (срабатывает блокировка) если оба вывода $PDPINT=0$ и $T1CTRIPE=0$, а $EVIMRA(0)=1$.</i>		
5	T1CMPOE	Разрешение работы вывода по сравнению T2. 0 – вывод T2PWM/T2CMP находится в z-состоянии; 1 – этот вывод управляется блоком сравнения. <i>Этот бит активен если $EXTCON(0) = 1$. Наоборот по сравнению с T2CMPOE.</i>		
4	T1CMPOE	То же, что и для бита [5], только для T1.		
3	T2PIN	Полярность сигнала на выводе сравнения T2 (T2PWM/T2CMP). 00 – форсированный 0 (сразу); 01 – активный 0; 10 – активный 1; 11 – форсированный 1.		
2	T2PIN			
1	T1PIN	То же, что и для битов [3-2], только для T1.		
0	T1PIN	Активным считается состояние, когда $TxCNT > TxCMP$.		

В качестве тактового сигнала, помимо внутреннего, может выступать сигнал с внешнего вывода TCLKINA, TCLKINB. Также в определенных режимах по сигналу с внешнего вывода TDIRA,B можно управлять характером изменения содержимого таймера (увеличение или уменьшение). Работа таймера может вызывать четыре типа прерываний:

- а) переполнения ($TCNT_x = 0xFFFF$);
- б) равенство нулю ($TCNT_x = 0x0000$);
- в) сравнения ($TCNT_x = TxCMP$);
- г) периода ($TCNT_x = TxPR$).

Таймеры могут работать либо независимо, либо синхронизировано друг с другом.

Регистр сравнения для каждого таймера может использоваться для генерации ШИМ (выход TxPWM).

Блоки сравнения

В каждом менеджере помимо регистра сравнения TxCMP со своим выходом TxPWM находятся три дополнительных блока сравнения. Синхронизация этих блоков осуществляется первым таймером менеджера (GP Timer1/3). Каждый блок имеет два выхода ШИМ со схемой

формирования мертвого времени рис.14 и схемой формирования векторов управления трехфазным инвертором (т.е. в сумме 6 выходов).

Работой этих блоков сравнения управляют регистры – COMCONA(B), ACTRA(B) и DBTCONA(B).

а) Регистр **COMCONA (COMCONB** – для менеджера EVB)

Бит	Название	Описание
15	CENABLE	0 – запрещает работу блока сравнения (регистры CMPRx, ACTRA становятся прозрачными); 1 – разрешает работу блока сравнения.
14	CLD1	Определяют момент перезагрузки регистров сравнения новым значением. 00 – T1CNT=0 (underflow), 01 – T1CNT=0 или T1CNT=T1PR, 10 – немедленно, 11 – зарезервировано (результат не предсказуем)
13	CLD0	
12	SVENABLE	1 – разрешает векторный режим ШИМ.
11	ACTRLD1	Определяют момент перезагрузки регистра управления (Action control reg) новым значением. 00 – T1CNT=0 (underflow), 01 – T1CNT=0 или T1CNT=T1PR, 10 – немедленно, 11 – зарезервировано (результат не предсказуем)
10	ACTRLD0	
9	FCMPOE	Этот бит, когда активен, разрешает или запрещает все выходы сравнения. 0 – все выходы PWM1/2/3/4/5/6 находятся в Z состоянии, 1 – эти выходы управляются соответствующей логикой. Этот бит активен, если бит EXTCONA(0) = 0. Если бит активен, то он сбрасывается в ноль когда PDPINTA/T1CTRIIP=0 и EVAIFRA(0)=1.
8	PDPINTA Status	Этот бит отражает текущее состояние вывода PDPINTA.
7	FCMP3OE	Этот бит, когда активен, разрешает или запрещает выходы сравнения 3. 0 – выходы PWM5/6 находятся в Z состоянии, 1 – эти выходы управляются соответствующей логикой. Этот бит активен, если бит EXTCONA(0) = 1. Если бит активен, то он сбрасывается в ноль когда C3TRIP=0 и также разрешен.
6	FCMP2OE	Аналогично предыдущему, только для блока сравнения 2 (PWM3/4 и C2TRIP)
5	FCMP1OE	Аналогично предыдущему, только для блока сравнения 1 (PWM1/2 и C1TRIP)
4	(Зарезерв.)	
3	(Зарезерв.)	
2	C3TRIP	Разрешение работы вывода блокировки C3TRIP. Активен, если EXTCONA(0) = 1. 1 – разрешает (C3TRIP = 0 – выходы PWM5/6 в Z состоянии), 0 – запрещает.
1	C2TRIP	Разрешение работы вывода блокировки C2TRIP. Активен, если EXTCONA(0) = 1. 1 – разрешает (C2TRIP = 0 – выходы PWM3/4 в Z состоянии), 0 – запрещает.
0	C1TRIP	Разрешение работы вывода блокировки C1TRIP. Активен, если EXTCONA(0) = 1. 1 – разрешает (C1TRIP = 0 – выходы PWM1/2 в Z состоянии), 0 – запрещает.

б) Регистр **ACTRA (ACTRB** – для менеджера EVB)

Бит	Название	Описание
-----	----------	----------

15	SVRDIR	Направление вращения векторов ШИМ. 0 – положительное (CCW), 1 – отрицательное (CW).
14	D2	Базовые вектора.
13	D1	
12	D0	
11	CMP6ACT1	Действие на сравнение выхода 6, CMP6. 00 – сбрасывается в 0; 01 – активный низкий уровень; 10 – активный высокий уровень; 11 – устанавливается в 1.
10	CMP6ACT0	
9	CMP5ACT1	То же для CMP5.
8	CMP5ACT0	
7	CMP4ACT1	То же для CMP4.
6	CMP4ACT0	
5	CMP3ACT1	То же для CMP3.
4	CMP3ACT0	
3	CMP2ACT1	То же для CMP2.
2	CMP2ACT0	
1	CMP1ACT1	То же для CMP1.
0	CMP1ACT0	

в) Регистр **DBTCONA (DBTCONB – для менеджера EVB)**

Бит	Название	Описание
15-12	(Зарезерв.)	
11-8	DBT3:0	Период 4х-битного таймера-счетчика, формирующего «мертвое» время.
7	EDBT3	1 – разрешает формирование «мертвого» времени для выводов PWM5 и PWM6.
6	EDBT2	1 – разрешает формирование «мертвого» времени для выводов PWM3 и PWM4.
5	EDBT1	1 – разрешает формирование «мертвого» времени для выводов PWM1 и PWM2.
4-2	DBTPS2:0	Делитель частоты (HISPCLK) для таймера «мертвого» времени: 000 – x/1; 001 – x/2; 010 – x/4; 011 – x/8; 100 – x/16; 101-111 – x/32
1-0	(Зарезерв.)	

Блоки захвата

В каждом менеджере событий находятся по три блока захвата с соответствующими выводами – CAPx (x = 1, 2, или 3 для EVA; и x = 4, 5, или 6 для EVB).

По заданному событию на этих выводах (таких событий три – фронт, срез или импульс) содержимое таймера TCNTx сохраняется (захватывается) в соответствующем двухуровневом стеке (FIFO). В качестве базового таймера можно задать любой таймер GPT1 или GPT2, для

EVA и GPT3,4 для EVB. Каждый из блоков захвата вырабатывает собственное прерывание.

Особенности блоков захвата

1. Один 16-разрядный регистр управления – CAPCONA (CAPCONB);
2. Один 16-разрядный статусный регистр FIFO – CAPFIFOA (CAPFIFOB);
3. Выбор таймера в качестве базового – GPT1,2 (GPT3,4);
4. Три 16-разрядных 2-уровневых FIFO стека, один для каждого блока;
5. 6 входов захвата-сравнения с триггером Шмитта – CAP1-6. Входы CAP1,2 (CAP4,5) имеют дополнительную альтернативную функцию – могут использоваться в качестве входов квадратурного энкодера (QEP).
6. Определяемые пользователем события, приводящие к захвату данных – фронт, срез или оба и фронт и срез.
7. Шесть маскируемых флагов прерывания.

а) Регистр управления – CAPCONA (CAPCONB для EVB, отличие – вместо блоков 1,2 и 3, блоки 4,5 и 6 с соответствующими индексами битов)

Бит	Название	Описание
15	CAPRES	Сброс блока захвате, обнуляет все регистры блока. Читается нулем.
14	CAP12EN	Включение блоков захвата 1 и 2. 00 – блоки захвата выключены, FIFO хранит значения, записанные до момента выключения; 01 – блоки захвата включены; 10 и 11 – зарезервированы.
13		
12	CAP3EN	Включение блока захвата 3. 0 – блок выключен; 1 – блок включен.
11	(Зарезерв.)	
10	CAP3TSEL	Выбор таймера для блока захвата 3. 0 – GPT2; 1 – GPT1.
9	CAP12TSEL	Выбор таймера для блоков захвата 1 и 2. 0 – GPT2; 1 – GPT1.
8	CAP3TOADC	Запуск АЦП по событию в блоке захвата 3. 0 – нет действия; 1 – запускает АЦП, если устанавливается флаг CAP3INT.
7	CAP1EDGE	Определяет фронт, по которому будет производиться захват значения таймера. Для блока захвата 1. 00 – нет действия; 01 – нарастающий фронт; 10 – срез импульса; оба фронта.
6		
5	CAP2DGE	Определяет фронт, по которому будет производиться захват значения таймера. Для блока захвата 2. 00 – нет действия; 01 – нарастающий фронт; 10 – срез импульса; оба фронта.
4		
3	CAP3DGE	Определяет фронт, по которому будет производиться захват значения таймера. Для блока захвата 3. 00 – нет действия; 01 – нарастающий фронт; 10 – срез импульса; оба фронта.
2		
1	(Зарезерв.)	
0	(Зарезерв.)	

б) Статусный регистр – CAPFIFOA (CAPFIFOB)

Бит	Название	Описание
15-14	(Зарезерв.)	
13-12	CAP3FIFO	Состояние (статус) стека захваченных значений FIFO блока захвата 3. 00 – пустой; 01 – содержит одно значение; 10 – содержит два значения; 11 – содержал два значения, но было принято еще одно, при этом первое значение «вытолкнулось» из стека и было потеряно.
11-10	CAP2FIFO	Состояние (статус) стека захваченных значений FIFO блока захвата 2. 00 – пустой; 01 – содержит одно значение; 10 – содержит два значения; 11 – содержал два значения, но было принято еще одно, при этом первое значение «вытолкнулось» из стека и было потеряно.
9-8	CAP1FIFO	Состояние (статус) стека захваченных значений FIFO блока захвата 1. 00 – пустой; 01 – содержит одно значение; 10 – содержит два значения; 11 – содержал два значения, но было принято еще одно, при этом первое значение «вытолкнулось» из стека и было потеряно.
7-0	(Зарезерв.)	

Прерывания менеджера событий (EV)

Прерывания менеджера событий организованы в три группы А, В и С. С каждой группой связан соответствующий регистр флагов – EVAIFR_x (EVBIFR_x, где x=A, B, C) и регистр масок (разрешений) – EVAIMR_x (EVBIMR_x, где x=A, B, C). Табл. 4 показывает все прерывания менеджера событий, их приоритет внутри группы и, собственно, группировку. Флаг в EVAIFR_x маскируется (не формируется запрос на прерывание) если соответствующий бит в регистре EVAIMR_x (EVBIMR_x) равен нулю. Когда какое-нибудь событие, формирующее прерывание происходит в менеджере событий, соответствующий флаг прерывания устанавливается в одном из флаговых регистров EVAIFR_x (EVBIFR_x). И в случае если соответствующий бит в регистре масок EVAIMR_x (EVBIMR_x) установлен в единицу формируется запрос контроллеру прерываний (PIE). Вектор прерывания соответствующий флагу прерывания с наивысшим приоритетом обрабатывается контроллером прерываний. **(Внимание! Флаг прерывания должен очищаться (путем записи единицы) программно в обработчике прерываний)**

Таблица 4

Группа	Прерывание	Приоритет внутри группы	Описание	Вход прерывания (PIE) INT
A	PDPINTA	1 (highest)	Power Drive Protection Interrupt A (блокировка выходов ШИМ)	1
A	CMP1INT	2	Compare Unit 1 compare interrupt (сравнение модуля 1, CMPR1)	2 (4 – EVB)
	CMP2INT	3	Compare Unit 2 compare interrupt	
	CMP3INT	4	Compare Unit 3 compare interrupt	
	T1PINT	5	GP timer 1 period interrupt (рав-во периоду, T1CNT=T1PR)	
	T1CINT	6	GP timer 1 compare interrupt (сравнение, T1CNT=T1CMP)	
	T1UFINT	7	GP timer 1 underflow interrupt (рав-во нулю, T1CNT=0)	
	T1OFINT	8	GP timer 1 overflow interrupt (переполнение, T1CNT=0xFFFF)	
B	T2PINT	1	GP timer 2 period interrupt	3 (5 – EVB)
	T2CINT	2	GP timer 2 compare interrupt	
	T2UFINT	3	GP timer 2 underflow interrupt	
	T2OFINT	4	GP timer 2 overflow interrupt	
C	CAP1INT	1	Capture Unit 1 interrupt (прерывание модуля захвата 1)	3 (5 – EVB)
	CAP2INT	2	Capture Unit 2 interrupt	
	CAP3INT	3 (lowest)	Capture Unit 3 interrupt	
Для менеджера событий EVB те же прерывания, но для таймеров 3 и 4 вместо таймеров 1 и 2, а также для модулей сравнения и захвата 4, 5 и 6 вместо модулей 1, 2 и 3.				

Регистры масок прерываний EVAIMRA, B, C имеют те же битовые поля, что и флаговые регистры EVAIFRA, B, C.

Регистры флагов прерываний менеджера событий – **EVAIFRA** (**EVVIFRA** , вместо таймера T1 таймер T3, и CMP4, 5, 6, соответственно)

Бит	Название	Описание
15-11	(Зарезерв.)	
10	T1OFINT	Флаг переполнения таймера 1, overflow
9	T1UFINT	Флаг достижения таймером 1 нулевого значения, underflow
8	T1CINT	Флаг сравнения таймера 1 (T1CNT=T1CMP)
7	T1PINT	Флаг достижения таймером значения регистра периода (T1CNT=T1PRD)
6-4	(Зарезерв.)	
3	CMP3INT	Флаг сравнения таймера 1 (T1CNT=CMP3)

2	CMP2INT	Флаг сравнения таймера 1 (T1CNT=CMP2)
1	CMP1INT	Флаг сравнения таймера 1 (T1CNT=CMP1)
0	PDPINTA	Флаг блокировки, устанавливается, если вход блокировки модуля сравнения равен нулю и эта блокировка разрешена (EXTCONA(0)=1). Если EXTCONA(0)=0, то определение этого флага становится таким же как и для семейства 240х, для совместимости кода.

EVAIFRB (EVBIFRB, вместо таймера T2 таймер T4)

Бит	Название	Описание
15-4	(Зарезерв.)	
3	T2OFINT	Флаг переполнения таймера 2, overflow
2	T2UFINT	Флаг достижения таймером 2 нулевого значения, underflow
1	T2CINT	Флаг сравнения таймера 2 (T1CNT=T2CMP)
0	T2PINT	Флаг достижения таймером значения регистра периода (T2CNT=T2PRD)

EVAIFRC (EVBIFRC – CAP4, 5, 6)

Бит	Название	Описание
15-3	(Зарезерв.)	
2	CAP3FINT	Флаг прерывания модуля захвата 3
1	CAP2FINT	Флаг прерывания модуля захвата 2
0	CAP1FINT	Флаг прерывания модуля захвата 1

Регистр функционального расширения EXTCONA (EXTCONB)

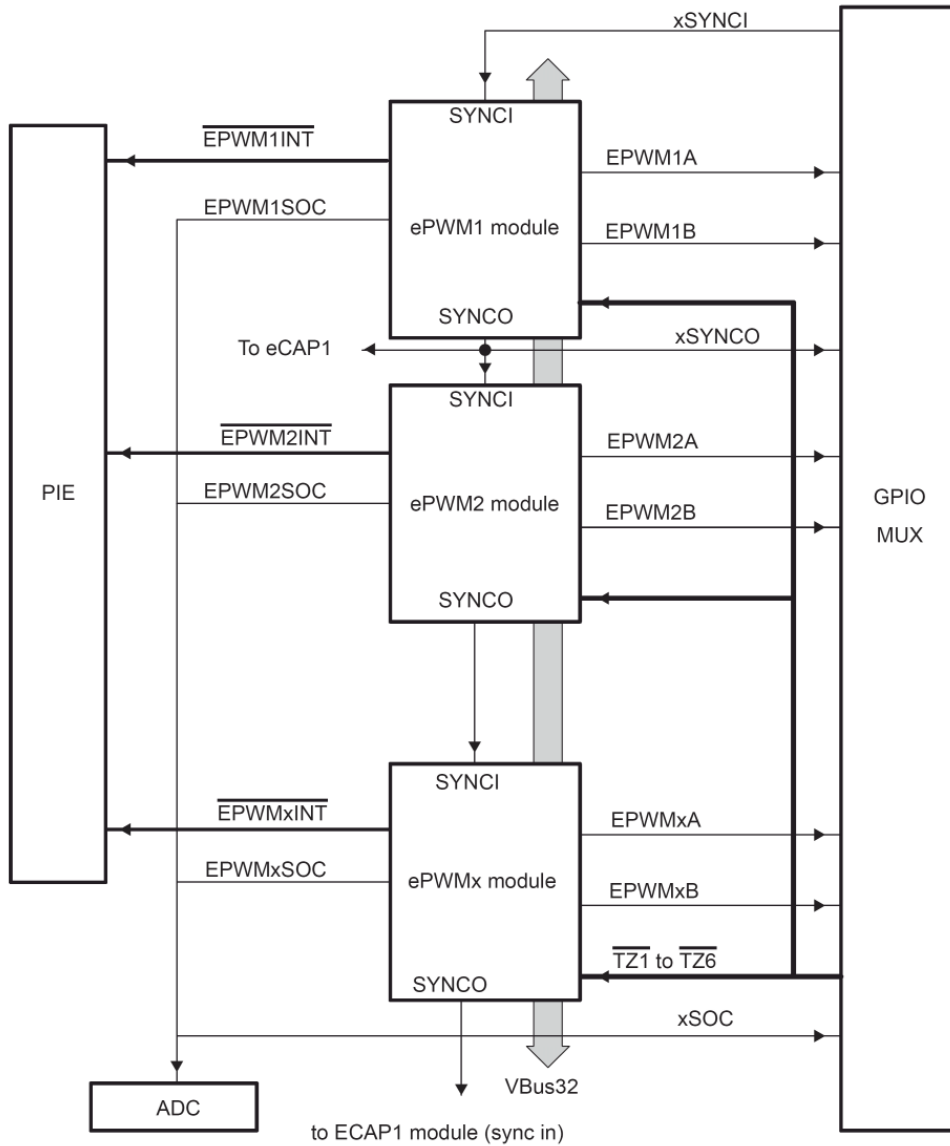
Этот регистр подключают дополнительные (по сравнению с семейством 240х) функциональные особенности. По умолчанию, для совместимости, эти особенности выключены.

Бит	Название	Описание
15-4	(Зарезерв.)	
3	EVSOC	1 – разрешает формировать импульс запуска внешнего АЦП (т.е. EVASOCn и EVBSOCn). Активный уровень – низкий, формируется на выбранное событие генерирующее запуск АЦП.
2	QEPIE	1 – разрешает работу CAP3_QEP1 в качестве индексного входа. Это значит, что либо только фронт на входе CAP3_QEP1 (EXTCONA[1]=0), либо фронт плюс оба входа CAP1_QEP1 и CAP2_QEP2 равны 1 (EXTCONA[1]=0) приводят к сбросу в ноль таймера сконфигурированного в качестве таймера QEP.
1	QEPIQUAL	Режим квалификации (фильтрации) индексного входа. 1 – требуется не только фронт на входе CAP3_QEP1, но и высокий уровень на входах CAP1 и CAP2, чтобы произошел сброс таймера QEP.
0	INDCOE	1 – разрешает режим независимого разрешения (блокировки) выходов блоков сравнения.

4. МОДУЛЬ ШИМ ДЛЯ СЕМЕЙСТВА F280X(EPWM)

Блок таймеров общего назначения и модулей захвата\сравнения в семействе процессоров F280x имеет существенные отличия от аналогичного блока (менеджера событий) семейства F281x. Блок таймеров состоит из ряда идентичных модулей широтно-импульсных модуляторов (ePWMx), соединенных последовательно относительно сигнала синхронизации. Структурная схема взаимного соединения этих модулей показана на рис.15. Каждый модуль, в свою очередь, состоит из семи субмодулей, структура и взаимное соединение которых показаны на рис.16. Основные функции, выполняемые субмодулями состоят в следующем:

1. Модуль ТВ (Time Base) определяет временные характеристики работы таймера. Установка периода (16 разрядов) и типа генерации, характер загрузки регистра-периода а также синхронизация находятся в этом модуле.
2. Модуль СС (Counter Compare) вырабатывает сигналы сравнения между значением таймера-счетчика и одного из двух регистров сравнения (CMPA и CMPB). Эти сигналы используются для формирования широтно-модулированного выходного сигнала на выходах EPWMxA и EPWMxB.
3. Модуль АQ (Action qualifier) определяет действие (установка или сброс) по сигналам сравнения (модуль СС) и сигналам вырабатываемым модулем ТВ в моменты, когда значение таймера равно нулю (CTR=ZERO) или периоду (CTR=PRD).
4. Модуль DB (Dead Band) формирует временную паузу, «мертвое время» между импульсами.
5. Модуль РС позволяет генерировать пачку импульсов.
6. Модуль TZ управляет блокировкой выходных сигналов.
7. Модуль ET регулирует выработку сигналов прерывания для контроллера прерывания (PIE).



Puc.15

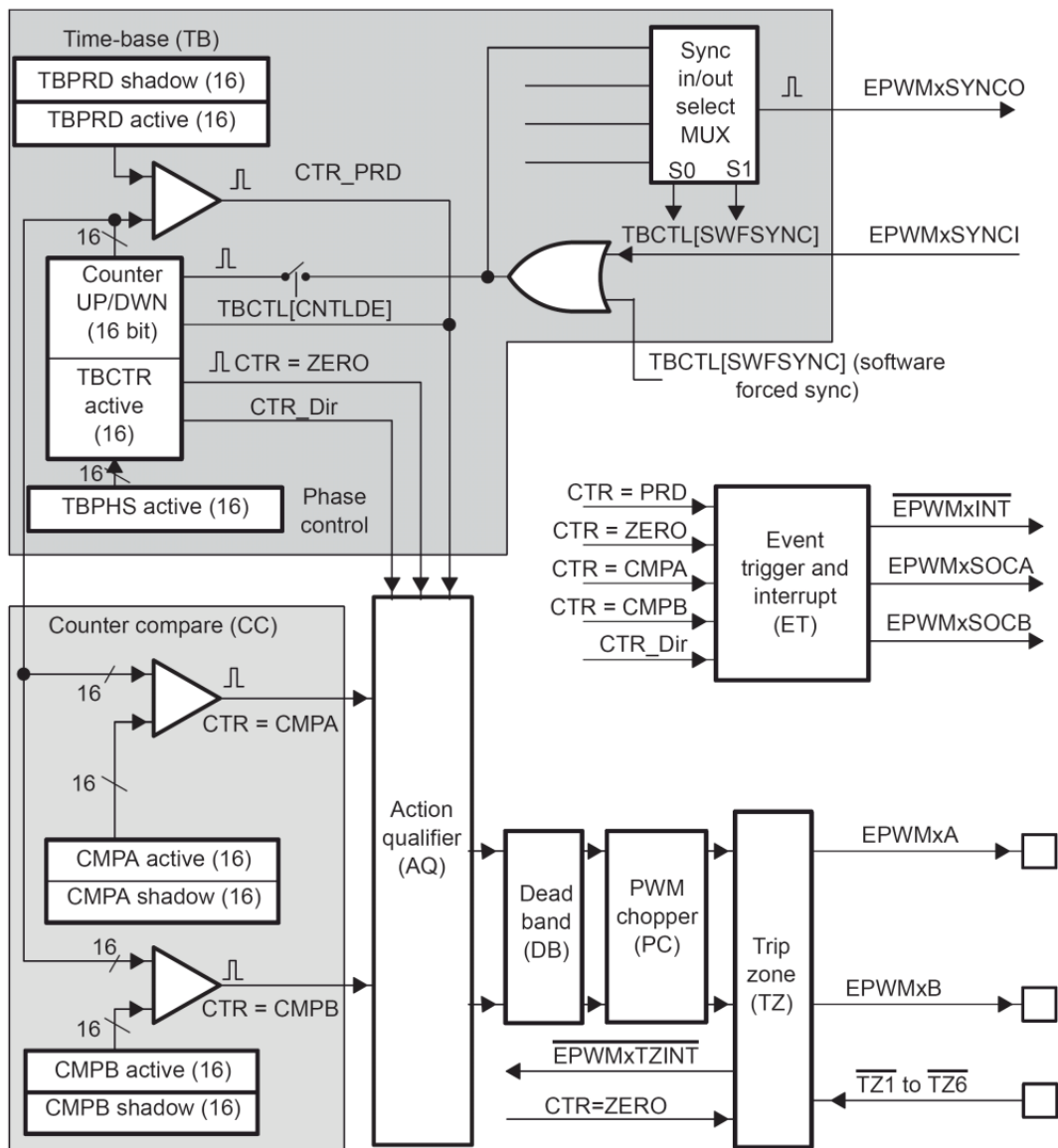


Рис.16

1.ТВ (Time Base) модуль

а) TBPRD – регистр периода, определяет частоту работы модуля (имеет теневой регистр по тому же адресу)

б) TBPHS – регистр фазы (используется для синхронизации, по синхроимпульсу содержимое этого регистра передвигается в регистр-счетчик)

в) TBCNT – регистр-счетчик.

г) TBCTL – регистр управления

Бит	Название	Описание
15	Free	Определяют действие при запросе эмулятора. (Например, точка остановки) 00 – останавливается сразу после следующего инкремента или декремента счетчика; 01 – останавливается по достижению периода в
14	Soft	

		режиме (up) и по достижению нуля в режиме (down и up-down); 10,11 – игнорируется запрос.			
13	PHSDIR	Бит направления счета после прихода импульса синхр. и переноса значения из TBPHS в TBCNT. 0 – счет «вниз» после синхр.; 1 – счет «вверх» после синхр. В режимах up и down этот бит игнорируется.			
12-10	CLKDIV	Делитель TBCLK=SYSCLKOUT/(HSPCLKDIV*CLKDIV)	000 – x/1 (def)	011 – x/8	110 – x/64
			001 – x/2	100 – x/16	111 – x/128
			010 – x/4	101 – x/32	
9-7	HSPCLKDIV	Скоростной делитель	000 – x/1	011 – x/6	110 – x/12
			001 – x/2 (def)	100 – x/8	111 – x/14
			010 – x/4	101 – x/10	
6	SWFSYNC	Программный синхроимпульс. 1 – вырабатывает одиночный синхроимпульс. Работает только, если SYNCOSSEL=00.			
5-4	SYNCOSSEL	Выбор выхода синхронизации. Эти биты определяют источник, который генерирует выходной синхросигнал. 00 – EPWMxSYNCl (и программный); 01 – CTR=ZERO; 10 – CTR=CMPB; 11 – выключен выход.			
3	PRDL	Условие при котором происходит загрузка регистра периода таймера (перегрузка из буферного, теневого регистра в основной). 0 – CNT=ZERO; 1 – сразу загружается активный регистр, минуя теневого.			
2	PHSEN	Бит разрешения загрузки регистра-счетчика из фазового регистра. 0 – НЕ позволяет загрузку, 1 – позволяет загрузку по сигналу синхронизации.			
1-0	CTRMODE	Режим работы (счета) таймера: 00 – up; 01 – down; 10 – up-down; 11 – стоп (заморозка).			

д) TBSTS – статусный регистр

Бит	Название	Описание
15-3	Зарезерв.	
2	CTRMАХ	Бит индицирует достижение счетчиком максимального значения 0xFFFF. 0 – показывает, что таймер никогда не достигал максимума, 1 – показывает, что таймер достигал максимума, запись единицы очищает этот бит.
1	SYNCl	Бит индикации синхроимпульса. 0 – не было синхроимпульса, 1 – был синхроимпульс. Запись единицы очищает этот бит.
0	CTRDlR	Бит направления счета (не режима). 0 – down (в настоящий момент), 1 – up.

2. СС (Модуль сравнения)

а) СМРА – регистр сравнения А

б) СМРВ – регистр сравнения В

в) СМРСТЛ – регистр управления

Бит	Название	Описание
15-10	Зарезерв.	
9	SHDWBFULL	0 – теновой FIFO еще не заполнился, 1 – показывает, что FIFO полный. Этот бит очищается при перезагрузки теневого регистра в активный.
8	SHDWAFULL	0 – теновой FIFO еще не заполнился, 1 – показывает, что FIFO полный. Этот бит очищается при перезагрузки теневого регистра в активный. Этот флаг устанавливается когда 32-бита записываются в

		СМРА:СМРАНР регистр или 16-битов записываются в СМРА регистр. 16-битная запись в регистр СМРАНР не устанавливает этот бит.
7	Зарезерв.	
6	SHDWBMODE	Режим загрузки регистра сравнения В. 0 – теневой режим, работает как двойной буфер; 1 – режим непосредственной записи сразу в активный регистр.
5	Зарезерв.	
4	SHDWA MODE	Режим загрузки регистра сравнения А. 0 – теневой режим, работает как двойной буфер; 1 – режим непосредственной записи сразу в активный регистр.
3-2	LOADBMODE	Режим загрузки активного регистра из теневого (В): 00 – CTR=ZERO; 01 – CTR=PRD; 10 – CTR=ZERO or PRD; 11 – нет загрузки.
1-0	LOADA MODE	Режим загрузки активного регистра из теневого (А): 00 – CTR=ZERO; 01 – CTR=PRD; 10 – CTR=ZERO or PRD; 11 – нет загрузки.

3. AQ

а) AQCTLA (AQCTLB) – регистры управления А и В

Бит	Название	Описание
15-12	Зарезерв.	
11-10	CBD	Действие при условии, что CTR=СМРВ при счете вниз (down): 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).
9-8	СВU	Действие при условии, что CTR=СМРВ при счете вверх (up): 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).
7-6	CAD	Действие при условии, что CTR=СМРА при счете вниз (down): 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).
5-4	CAU	Действие при условии, что CTR=СМРА при счете вверх (up): 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).
3-2	PRD	Действие при условии, что CTR=PRD при счете вверх (up): 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).
1-0	ZRO	Действие при условии, что CTR=Zero при счете вверх (up): 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).

б) AQSFRС – регистр программного форсирования выходов

Бит	Название	Описание
15-8	Зарезерв.	
7-6	RLDCSF	Загрузка активного регистра AQCSF из теневого: 00 – CTR=Zero; 01 – CTR=PRD; 10 – CTR=Zero or PRD; 11 – непосредственная загрузка.
5	OTSFB	Однократное программное событие для выхода В. 1 – инициирует это событие.
4-3	ACTSFB	Действие на OTSFB: 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).
2	OTSFA	Однократное программное событие для выхода А. 1 – инициирует это событие.

1-0	ACTSFA	Действие на OTSFA: 00 – нет действия; 01 – очистка (low); 10 – установка (high); 11 – смена значения (toggle).
-----	--------	--

в) AQCSFRC – регистр продолжительного программного формирования выходов

Бит	Название	Описание
15-4	Зарезерв.	
3-2	CSFB	01 – форсирует продолжительный низкий уровень на выходе В, 10 – высокий уровень. (00 и 11 не имеют эффекта)
1-0	CSFA	01 – форсирует продолжительный низкий уровень на выходе А, 10 – высокий уровень. (00 и 11 не имеют эффекта)

4. DB модуль (модуль «мертвого» времени)

а) DBCTL – регистр управления генератора

Бит	Название	Описание
15-4	Зарезерв.	
3-2	POLSEL	Управление полярностью (Обычно с MODE=11): 00 – активный высокий (AH); 01 – активный низкий комплементарный (ALC); 10 – активный высокий комплементарный; 11 – активный низкий (AL) режимы.
1-0	MODE	Управление режимом «мертвого» времени: 00 – нет DBM; 01 – DBA=AQA (нет задержки для А), DBB=FED(задержка по спадающему фронту для В); 10 – DBA=RED (задержка по нарастающему фронту для А), DBB=AQB (нет задержки для В); 11 – DBM полностью разрешена (т.е. активны оба режима: RED и FED).

б) DBRED, DBFED – регистры счетчики задержки по нарастающему и спадающему фронтам соответственно.

№Бит	Название	Описание
15-10	Зарезерв.	
9-0	DEL	10-разрядный счетчик

5. PC модуль

а) PCCTL – регистр управления

Бит	Название	Описание
15-11	Зарезерв.	
10-8	CHPDUTY	Коэффициент заполнения (chopping) 000-1/8, 001-2/8 ... 110-7/8, 111-резерв.
7-5	CHPFREQ	Делитель частоты (chopping) 000-1 (т.е. нет деления = 12,5 МГц при 100 МГц системной частоты), 001-1/2 (6.25МГц), 010-1/3,... 111-1/8 (1.65 МГц)
4-1	OSHTWTH	Ширина одного импульса $(xxxx+1)*sysclk/8$ (0000 – 80ns @ 100MHz)
0	CHPEN	Разрешение PWM-chopping (1 – разрешение)

6. TZ модуль

а) TZSEL – регистр выбора

Бит	Название	Описание
15-14	Зарезерв.	
13	OSHT6	1 – разрешает TZx выступать в качестве источника однократного импульса блокировки, 0 – запрещает.
12	OSHT5	
11	OSHT4	
10	OSHT3	
9	OSHT2	
8	OSHT1	
7-6	Зарезерв.	
5	CBC6	1 – разрешает TZx вступать в качестве блокировки CBC
4	CBC5	
3	CBC4	
2	CBC3	
1	CBC2	
0	CBC1	

б) TZCTL – регистр управления модулем TZ

Бит	Название	Описание
15-4	Зарезерв.	
3-2	TZB	Действие блокировки TZx на выход EPWMxB: 00 – z; 01 – Hi; 10 – Lo; 11 – нет действия (блокировка выкл.)
1-0	TZA	Действие блокировки TZx на выход EPWMxA: 00 – z; 01 – Hi; 10 – Lo; 11 – нет действия (блокировка выкл.)

5. МОДУЛЬ ЗАХВАТА СЕМЕЙСТВА F280X (ЕСАР)

Каждый канал модуля захвата обладает одним входным выводом ЕСАРx по сигналу, с которого содержимое 32х-разрядного счетчика TSCTRx запоминается в одном из 4-х регистров CAP1-CAP4 (рис.17). Номер регистра, в который происходит запоминание, определяется 2х-битным счетчиком, содержимое которого изменяется циклически по мере прихода входного импульса.

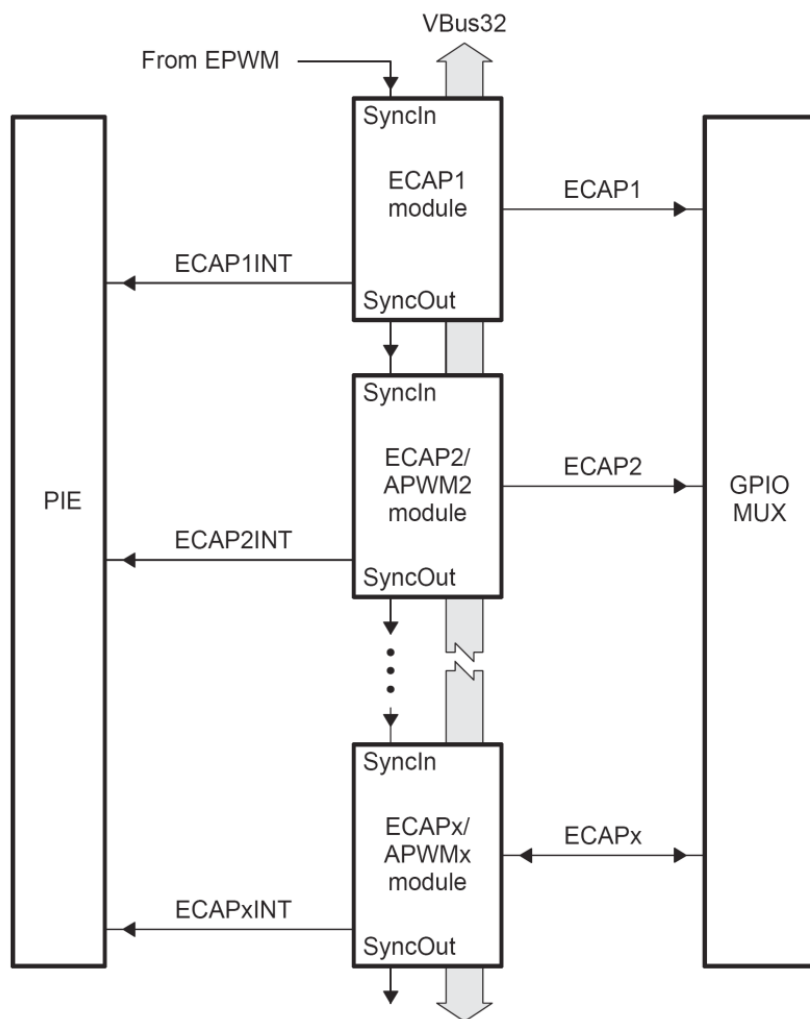


Рис.17

- а) TSCTR – счетчик (32-бита, временная основа для работы модуля захвата)
- б) CTRPHS – фазовый регистр. Значение этого регистра может вводиться в основной счетчик, либо по внешнему сигналу SYNCI, либо программно
- в) CAP1, 2, 3, 4 – регистры захвата. (программный доступ)
- г) ECCTL1 – регистр управления

Бит	Название	Описание
15-14	FREE/SOFT	Определяют действие при запросе эмулятора. (Например, точка останова) 00 – TSCTR останавливается; 01 – останавливается по достижению нуля; 10,11 – игнорируется запрос.
13-9	PRECALE	Входной делитель. $1/(N=xx*2)$, при $xx=0$ $N=1$ (нет деления), макс. $N=62$ (11111)
8	CAPLDEN	1 – разрешает загружать регистры захвата по событию; 0 – запрещает.
7	CTRRST4	1 – сброс счетчика по событию.
6	CAP4POL	0 – активным событием является фронт импульса; 1 – спрез.
5	CTRRST3	1 – сброс счетчика по событию.

4	CAP3POL	0 – активным событием является фронт импульса; 1 – срез.
3	CTRRST2	1 – сброс счетчика по событию.
2	CAP2POL	0 – активным событием является фронт импульса; 1 – срез.
1	CTRRST1	1 – сброс счетчика по событию.
0	CAP1POL	0 – активным событием является фронт импульса; 1 – срез.

д) ECCTL2 – регистр управления

Бит	Название	Описание
15-11	Зарезерв.	
10	APWMPOL	Полярность выходного сигнала в режиме ШИМ. 0 – активный высокий уровень (т.е. значение рег.сравн. определяет высокий уровень); 1 – активный низкий уровень.
9	CAP/APWM	Выбор режима. 0 – режим захвата; 1 – ШИМ режим.
8	SWSYNC	Программный синхросигнал. Запись 1 форсирует теньевую загрузку TSCTR текущего модуля, а также модулей следующих далее по цепочке (с большим номером) при условии, что биты SYNCO_SEL=00 (сквозная передача синхросигнала). После записи 1 этот бит автоматически обнуляется.
7-6	SYNCO_SEL	Выбор выходного синхросигнала. 00 – входной синхросигнал является выходным; 01 – CTR=PRD (актуально в ШИМ режиме); 1x – нет вых. синхро.
5	SYNCI_EN	0 – выключает опцию входной синхронизации; 1 – разрешает загружать счетчик из (фазового?) регистра либо по входному сигналу SYNCI, либо программно.
4	TSCTRSTOP	Остановка таймера. 0 – стоп (сохраняется достигнутое значение; 1 – свободный счет.
3	REARM	Управление однократным режимом захвата (перезапуск). 1 – перезапускает однократную последовательность захвата, т.е. сбрасывает счетчик Mod4 в ноль, «размораживает» его, разрешает загружать регистры захвата CAPx.
2-1	STOPVALUE	Значение остановки (количество захватов) в режиме однократного захвата. 00 – стоп после первого захвата (CAP1), ..., 11 – стоп после четвертого захвата (CAP4). После остановки прерывания блокируются.
0	CONT/ ONESHT	Режим работы. 0 – непрерывный захват; 1 – однократный захват.

е) ECEINT – регистр масок (разрешений) прерываний

ж) ECFLG – регистр флагов прерываний

з) ECCLR – регистр очистки прерываний

и) ECFRC – регистр установки (форсирования) прерываний

(таблица для регистров ECEINT, ECFLG, ECCLR, ECFRC)

Бит	Название	Описание
15-8	Зарезерв.	
7	CTR=CMP	Прерывание по сравнению. (1 – разрешение, флаг или очистка/установка флагов)
6	CTR=PRD	Прерывание по достижению периода.
5	CTROVF	Прерывание по переполнению счетчика.

4	CEVT4	Захват 4
3	CEVT3	Захват 3
2	CEVT2	Захват 2
1	CEVT1	Захват 1
0	Зарезерв. (INT рег. флагов и очистка)	В регистре флагов этот бит индицирует (1), что какое-то прерывание было сгенерировано.

- Если прерывание по захвату разрешено (регистр ECEINT) и какой-нибудь из флагов устанавливается в 1 и при этом флаг INT=0, то генерируется импульс прерывания и INT флаг устанавливается в 1. Никакие другие импульсы прерывания не генерируются до тех пор, пока флаг INT не будет очищен (с помощью записи 1 в регистр ECCLR) и не будет установлен один или более флагов прерывания.
- Запись логической единицы в регистр ECFRC приводит к тому, что устанавливается соответствующий флаг прерывания и генерируется импульс прерывания контроллеру прерываний.

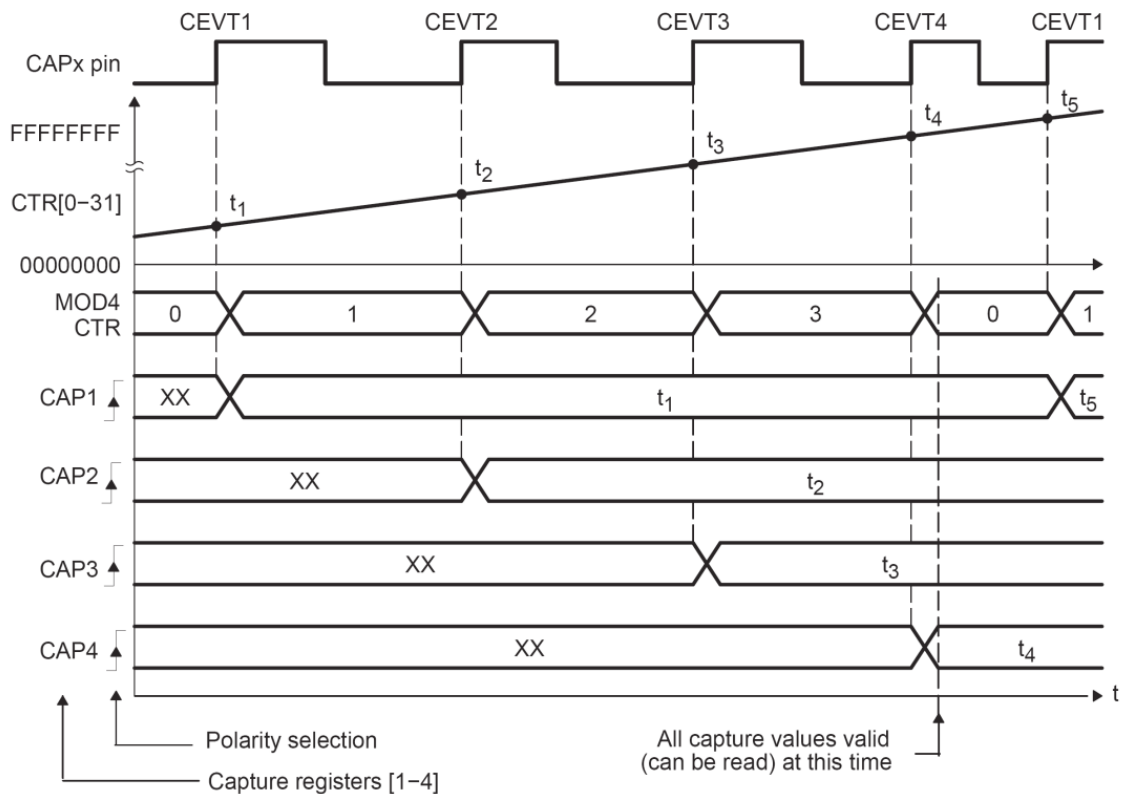
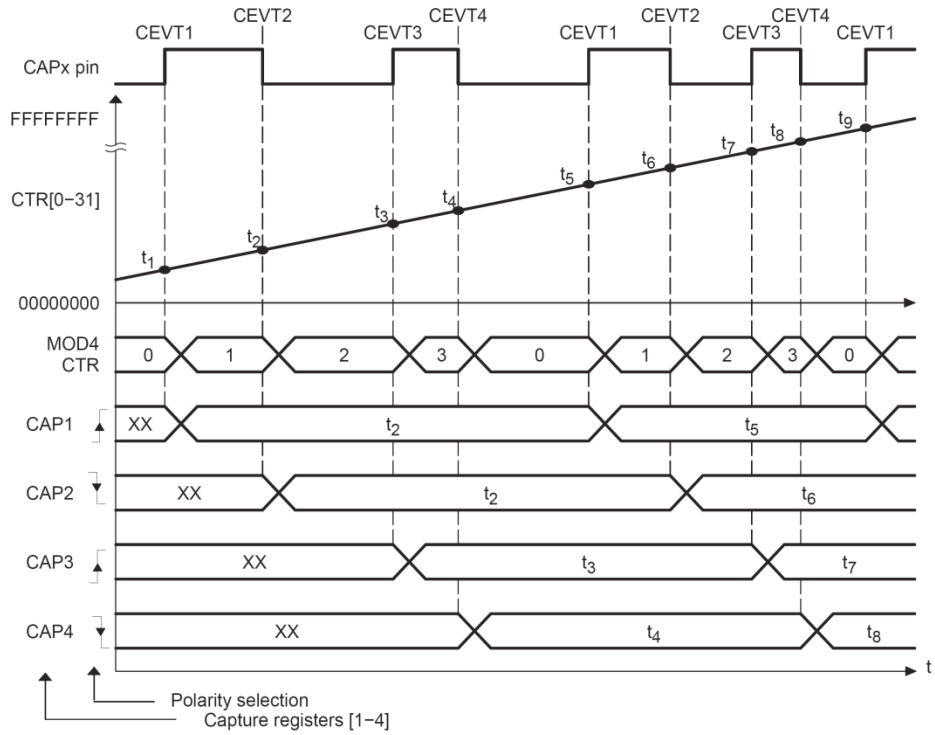
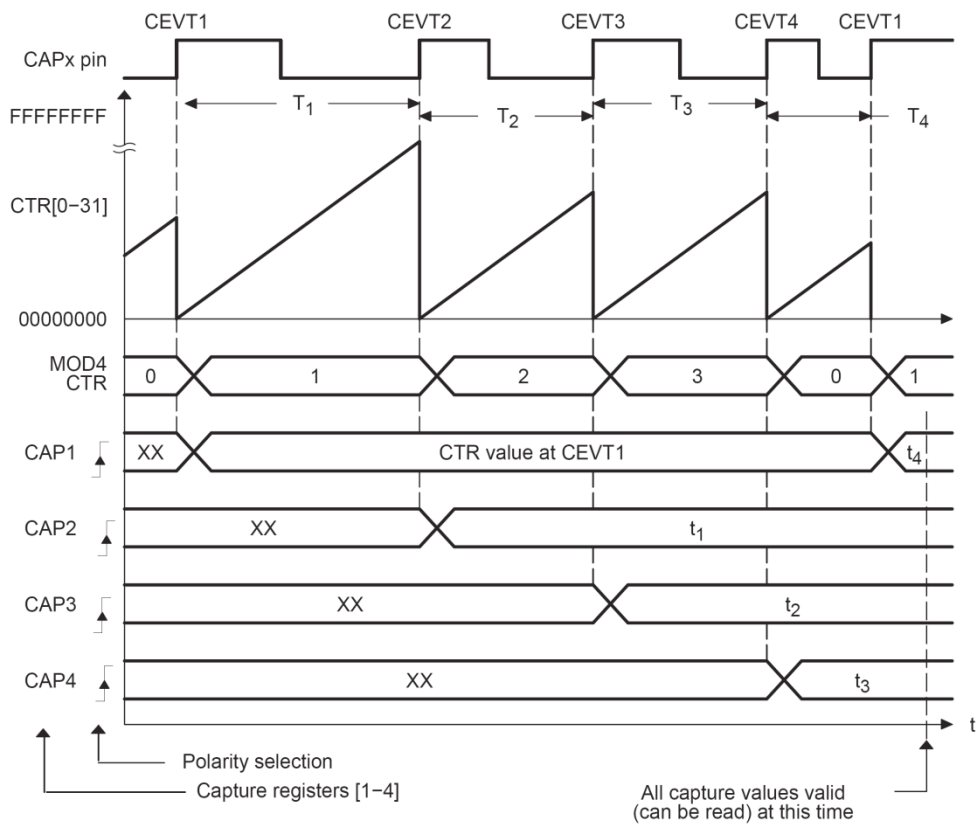


Рис.18



Puc.19



Puc.20

6. АНАЛОГО-ЦИФРОВОЙ ПРЕОБРАЗОВАТЕЛЬ (АЦП)

Блок АЦП состоит из двух 8-канальных модулей – SEQ1 и SEQ2, которые могут быть соединены последовательно, чтобы сформировать один 16-канальный модуль – SEQ. В блоке АЦП находятся два устройства выборки и хранения (УВХ) SHA и SHB, по одному на каждый 8-канальный модуль и один на все каналы и УВХ АЦП. Выделяют одиночный (16 каналов – SEQ) или двойной (2 по 8 каналов – SEQ1 и SEQ2) режимы преобразования. В обоих случаях АЦП способно выполнять автопреобразование, т.е. каждый раз как АЦП получает запрос на начало (старт) преобразования (SOC), оно может выполнить ряд преобразований автоматически, без вмешательства процессора или программы пользователя. Для каждого преобразования любой из 16-ти каналов (входов) может быть выбран через мультиплексор. После каждого преобразования результат сохраняется в соответствующем регистре результатов:

- 1-й результат – ADCRESULT0,
- 2-й результат – ADCRESULT1,
- ...
- 16-й результат – ADCRESULT15.

Также существует возможность делать выборку с одного и того же канала несколько раз (при этом порядок сохранения результата остается прежним), позволяя выполнить избыточную выборку (oversampling).

АЦП может работать в режиме одновременной выборки (POB) или последовательной выборки (РПВ). В режиме одновременной выборки происходит захват двух значений напряжения с помощью УВХ с последующим их преобразованием и сохранением (последовательным в последовательных регистрах результата). Причем пара входов образуется из групп А и В с одинаковым смещением (например: ADCINA0 & ADCINB0; ADCINA1 & ADCINB1). В режиме последовательной выборки захватывается и преобразуется одно значение напряжения. Для каждого преобразования (или пары преобразований в POB) текущее поле битов CONVxx (4 разряда) определяет вход (или пару входов) напряжение на которых будет выбрано и преобразовано. В режиме последовательной выборки все четыре бита CONVxx определяют номер входа. Старший бит (MSB) определяет с каким УВХ ассоциирован вход (SHA или SHB), а три младших разряда определяют номер входа (сдвигку).

Например: CONVxx = 0101b, значит выбран вход ADCINA5,
CONVxx = 1011b, значит выбран вход ADCINB3

В режиме одновременной выборки старший бит поля CONVxx игнорируется, а три младших бита определяют номер входа для обоих УВХ. Напряжение из устройства SHA преобразуется первым, а SHB вторым. Результат из SHA помещается в текущий регистр результата (ADCRESULTx), а результат из SHB помещается в следующий регистр. Указатель на регистр результата затем увеличивается на 2.

Как уже было сказано, аналоговый вход, напряжение с которого будет преобразовываться, выбирается с помощью поля CONVxx. 16 полей CONVxx распределяются по четырем регистрам ADCCHSELSEQn (n=1, 2, 3, 4). Эти поля могут иметь любое значение в диапазоне от 0 до 15. Входы могут быть выбраны в любом желаемом порядке, и один и тот же канал может быть выбран несколько раз.

Например:

1. Режим одновременной выборки (два блока SEQ1 и SEQ2)
(Simultaneous Sampling Dual Sequencer Mode)

```
AdcRegs.ADCCTRL3.bit.SMODE_SEL = 1;  
    // Устанавливаем режим одновременной выборки  
AdcRegs.ADCMAXCONV.all = 0x0003;  
    // 4 двойных преобразования (всего 8)  
AdcRegs.ADCCHSELSEQ1.bit.CONV00 = 0x0; // ADCINA0 & ADCINB0  
AdcRegs.ADCCHSELSEQ1.bit.CONV01 = 0x1; //ADCINA1 & ADCINB1  
AdcRegs.ADCCHSELSEQ1.bit.CONV02 = 0x2; //ADCINA2 & ADCINB2  
AdcRegs.ADCCHSELSEQ1.bit.CONV03 = 0x3; //ADCINA3 & ADCINB3
```

Если все преобразования были выполнены, то результат располагается в следующих регистрах:

```
ADCINA0 -> ADCRESULT0  
ADCINB0 -> ADCRESULT1  
ADCINA1 -> ADCRESULT2  
ADCINB1 -> ADCRESULT3  
ADCINA2 -> ADCRESULT4  
ADCINB2 -> ADCRESULT5  
ADCINA3 -> ADCRESULT6  
ADCINB3 -> ADCRESULT7
```

2. Режим последовательной выборки (каскадная работа, один SEQ)

(Simultaneous Sampling Cascaded Sequencer Mode)

```
AdcRegs.ADCCTRL3.bit.SMODE_SEL = 0;
```

```

// Устанавливаем режим последовательной выборки
AdcRegs.ADCCTRL1.bit.SEQ_CASC = 1;
// Устанавливаем каскадный режим выборки
AdcRegs.ADCMAXCONV.all = 0x0007;
// 8 преобразований
AdcRegs.ADCCHSELSEQ1.bit.CONV00 = 0x0; // ADCINA0
AdcRegs.ADCCHSELSEQ1.bit.CONV01 = 0x1; //ADCINA1
AdcRegs.ADCCHSELSEQ1.bit.CONV02 = 0x2; //ADCINA2
AdcRegs.ADCCHSELSEQ1.bit.CONV03 = 0x3; //ADCINA3
AdcRegs.ADCCHSELSEQ2.bit.CONV04 = 0x8; //ADCINB0
AdcRegs.ADCCHSELSEQ2.bit.CONV05 = 0x9; //ADCINB1
AdcRegs.ADCCHSELSEQ2.bit.CONV06 = 0xA; //ADCINB2
AdcRegs.ADCCHSELSEQ2.bit.CONV07 = 0xB; //ADCINB3

```

Если все преобразования были выполнены, то результат располагается в следующих регистрах:

```

ADCINA0 -> ADCRESULT0
ADCINA0 -> ADCRESULT1
ADCINA2 -> ADCRESULT2
ADCINA3 -> ADCRESULT3
ADCINB0 -> ADCRESULT4
ADCINB1 -> ADCRESULT5
ADCINB2 -> ADCRESULT6
ADCINB3 -> ADCRESULT7

```

В режиме автоматической выборки количество преобразований определяется с помощью регистра MAXCONV_n (3-х или 4-х битное поле) которое автоматически вводится в статусные биты SEQCNTR3-0 в регистре ADCASEQSR в начале каждой группы преобразований. Эта величина может иметь значения – 0-7 (или 0-15 при каскадировании блоков преобразования). Значение SEQCNTR3-0 уменьшается по мере выполнения преобразований из общей последовательности CONV00 – 01,02,..., до момента, когда не станет равным нулю. Количество преобразований определяется следующим образом:

$$N = \text{MAXCONV}_{n+1}.$$

Например: Допустим, мы хотим получить **семь(7)** преобразований автоматически от блока SEQ1, причем входы ADCINA2 и 3 должны выбираться дважды, а затем по разу А6,А7 и В4. Следовательно, величина MAXCONV_n должна быть равной **шести(6)** и регистры ADCCHSELSEQ_n должны быть установлены следующим образом:

Биты 15-12	Биты 11-8	Биты 7-4	Биты 3-0	
(CONV03) 3	(CONV02) 2	(CONV01) 3	(CONV00) 2	ADCCCHSELSEQ1
(CONV07) x	(CONV06) 12	(CONV05) 7	(CONV04) 6	ADCCCHSELSEQ2
(CONV11) x	(CONV10) x	(CONV09) x	(CONV08) x	ADCCCHSELSEQ3
(CONV15) x	(CONV14) x	(CONV13) x	(CONV12) x	ADCCCHSELSEQ4

Как только счетчик преобразований SEQCNTR3-0 достигнет нуля, может быть два развития событий:

а) Если COUNTRUN бит в ADCCTRL1 равен 1, то последовательность выборки начнется снова автоматически, SEQCNTR3-0 загрузится вновь из MAXCONVn, а указатель блока SEQ1 установится на первое преобразование (CONV00). В этом случае, чтобы избежать перезаписи необходимо гарантировать прочтение регистров результата.

б) Если COUNTRUN бит в ADCCTRL1 равен 0, то после выполнения первой последовательности преобразований блок остановится в текущем положении, т.е. SEQCNTR3-0 будет содержать 0, а указатель блока будет показывать на CONV06. Чтобы повторить последовательность выборки при следующем запросе на преобразование (SOC) АЦП должен быть сброшен путем записи 1 в бит RSTSEQn.

*Если установлен бит SEQOVRD, то автоматического сброса указателя блока выборки как в режиме при CONTRUN=1 не произойдет. По сигналу запуска преобразования очередная последовательность выборок будет начинаться с текущего положения указателя с соответствующим сохранением результата. Сброс указателя произойдет при достижении им максимального значения (ADCRESULT15).

Флаг прерывания устанавливается всякий раз, когда счетчик преобразований SEQCNTR3-0 становится равным нулю.

Регистры АЦП

а) Регистр управления 1 – ADCTRL1

Бит	Название	Описание
15	(Зарезерв.)	Чтение возвращает 0. Запись не имеет эффекта.
14	RESET	Программный сброс модуля АЦП. (1)
13	SUSMOD1	Определяют действие при запросе эмулятора. (Например точка останова) 00 – игнорируется запрос; 01 – останавливается по завершению всей последовательности преобразований; 10 – останавливается по завершению текущего преобразования (CONVn); 11 – останавливается сразу.
12	SUSMOD0	
11	ACQPS3	Определяют размер окна выборки. Контролируют ширину импульса SOC (старт преобразования). $SOC = (ADCCTRL1[11:8]+1) * \text{Период_ADCLK}$.
10	ACQPS2	

9	ACQPS1	
8	ACQPS0	
7	CPS	Делитель входной частоты: 0 – Fclk=CLK/1 (CLK – HSPCLK); 1 – Fclk=CLK/2
6	CONTRUN	Непрерывная работа. «0» – Режим «Старт-стоп». Преобразователь останавливается после достижения EOS (end of sequence – конец последовательности преобразований). При поступлении следующего стартового импульса SOC преобразование начинается снова с того места, где он остановился, при условии что не было сброса. «1» – Режим непрерывного преобразования. После достижения EOS, последовательность преобразований автоматически начинается снова с состояния CONV00 (для SEQ1 и каскада SEQ) и CONV08 (для SEQ2). Этот бит опрашивается только в момент EOS. Поэтому может быть изменен в любой момент времени до конца последовательности преобразований.
5	SEQOVRD	Режим перезаписи.
4	SEQCASC	«0» – два блока преобразования (SEQ1 и SEQ2) на 8 состояний «1» – один общий блок SEQ на 16 состояний.
3-0	(Зарезерв.)	

б) Регистр управления 2 – ADCTRL2

Бит	Название	Описание
15	EVBSOCSEQ	1 – разрешает запуск SOC от EVB в <i>каскадном</i> режиме
14	RSTSEQ1	1 – сброс SEQ1 сразу в CONV00
13	SOCSEQ1	Запуск АЦП (импульс SOC) для SEQ1. Этот бит может быть установлен следующими запускающими устройствами: программно (запись 1); EVA, EVB; внешний импульс (ADCSOC вывод). Замечание. RSTSEQ1 и SOCSEQ1 не должны быть установлены одновременно. Это сбросит SEQ1 но не начнет преобразование.
12	(Зарезерв.)	
11	INTENASEQ1	Разрешает (1) или запрещает (0) прерывание от SEQ1
10	INTMODSEQ1	Режим прерывания SEQ1. 0 – прерывание на каждый EOS, 1 – прерывание через раз.
9	(Зарезерв.)	
8	EVASOCSEQ1	0 – SEQ1 не может быть запущен от EVA, 1 – SEQ1/SEQ может быть запущен по сигналу от менеджера событий А (EVA), который может быть настроен на разные события.
7	EXTSOCSEQ1	0 – нет действия, 1 – разрешает запуск АЦП (SEQ1) от внешнего сигнала.
6	RSTSEQ2	1 – сброс SEQ2 сразу в CONV08. Игнорируется в каскадном режиме.
5	SOCSEQ2	Запуск АЦП (импульс SOC) для SEQ2. Этот бит может быть установлен следующими запускающими устройствами: программно (запись 1), EVB.
4	(Зарезерв.)	
3	INTENASEQ2	Разрешает (1) или запрещает (0) прерывание от SEQ2
2	INTMODSEQ2	Режим прерывания SEQ2. 0 – прерывание на каждый EOS, 1 – прерывание через раз.
1	(Зарезерв.)	
0	EVBSOCSEQ2	0 – SEQ2 не может быть запущен от EVB, 1 – SEQ2 может быть запущен по сигналу от менеджера событий В (EVB), который может быть

		настроен на разные события.
--	--	-----------------------------

в) Регистр управления 3 – ADCTRL3

Бит	Название	Описание
15-8	(Зарезерв.)	
7	ADCBGRFDN1	Включение (1) и выключение (0) опорного напряжения
6	ADCBGRFDN0	
5	ADCPWDN	Включение (1) и выключение (0) всего АЦП за исключением опоры.
4	ADCCLKPS3	Делитель входной частоты (для АЦП). Частота процессора HSPCLK делится вначале на величину $2 * ADCCLKPS[3:0]$, за исключением $ADCCLKPS[3:0]=0$ – в этом случае HSPCLK проходит напрямую. Далее этот сигнал делится на 2 если установлен бит ADCTRL1[7] и уже получившийся сигнал является тактовым для АЦП – ADCLK.
3	ADCCLKPS2	
2	ADCCLKPS1	
1	ADCCLKPS0	
0	SMODE_SEL	Режим выборки: 0 – последовательная выборка; 1 – одновременная выборка.

г) Регистр, определяющий число преобразований – ADCMAX-CONV

Бит	Название	Описание
15-7	(Зарезерв.)	
6	MAXCONV2_2	Это поле определяет максимальное число преобразований выполняемых за одну сессию автопреобразований. MAXCONV2[2:0] – SEQ2
5	MAXCONV2_1	
4	MAXCONV2_0	
3	MAXCONV1_3	MAXCONV1[3:0] – SEQ1
2	MAXCONV1_2	
1	MAXCONV1_1	
0	MAXCONV1_0	

д) Статусный регистр – ADCASEQSR

Бит	Название	Описание
15-12	(Зарезерв.)	
11-8	SEQCNTR[3:0]	В начальный момент (SOC) в это поле загружается значение MAXCONV и последовательно уменьшается по мере выполнения преобразований в последовательности.
7	(Зарезерв.)	
6-4	SEQ2STATE[2:0]	Указатели для SEQ2/1, зарезервированы для тестовых целей и не должны использоваться пользователем.
3-0	SEQ1STATE[3:0]	

е) Регистр флагов АЦП – ADCST

Бит	Название	Описание
-----	----------	----------

15-8	(Зарезерв.)	
7	EOSBUF2	Флаг конца буфера SEQ2 (конец последовательности). Этот бит не используется и остается нулевым в режиме прерывания 0 (ADCTRL2[2]=0). В режиме прерывания 1 (ADCTRL2[2]=1), он изменяется на каждое окончание последовательности преобразований (EOS) SEQ2.
6	EOSBUF1	Флаг конца буфера SEQ1 (конец последовательности). Этот бит не используется и остается нулевым в режиме прерывания 0 (ADCTRL2[10]=0). В режиме прерывания 1 (ADCTRL2[10]=1), он изменяется на каждое окончание последовательности преобразований (EOS) SEQ1.
5	INTSEQ2CLR	Бит очистки флага прерывания INTSEQ2. Нужно записать 1 для того чтобы очистить этот бит. Всегда читается как 0.
4	INTSEQ1CLR	Бит очистки флага прерывания INTSEQ1. Нужно записать 1 для того чтобы очистить этот бит. Всегда читается как 0.
3	SEQ2BSY	Бит занятого состояния SEQ2. 0 – SEQ2 в состоянии ожидания, 1 – SEQ2 занят преобразованием.
2	SEQ1BSY	Бит занятого состояния SEQ1. 0 – SEQ1 в состоянии ожидания, 1 – SEQ1 занят преобразованием.
1	INTSEQ2	Флаг прерывания блока SEQ2.
0	INTSEQ1	Флаг прерывания блока SEQ1.

ж) Регистры выбора канала преобразования –ADCCHSELSEQn

Биты 15-12	Биты 11-8	Биты 7-4	Биты 3-0	
(CONV03)	(CONV02)	(CONV01)	(CONV00)	ADCCHSELSEQ1
X	X	X	X	
(CONV07)	(CONV06)	(CONV05)	(CONV04)	ADCCHSELSEQ2
X	X	X	X	
(CONV11)	(CONV10)	(CONV09)	(CONV08)	ADCCHSELSEQ3
X	X	X	X	
(CONV15)	(CONV14)	(CONV13)	(CONV12)	ADCCHSELSEQ4
X	X	X	X	

x – номер вывода, 0000 – ADCINA0, ..., 0111 – ADCINA7; 1000 – ADCINB0, ..., 1111 – ADCINB7.

з) Регистры результата – ADCRESULTn, n=0...15

Результат в 16-ти разрядном регистре сдвинут к левому краю. В каскадном режиме регистры ADCRESULT8-15 содержат результаты с 9-го по 16-й.

Прерывания ADC

Возможно два режима прерываний, которые определяются битом INTMODSEQ1 в регистре ADCTRL2. В режиме 0 прерывания вырабатываются на каждое окончание последовательности преобразований, а в режиме 1 через раз. Последний режим позволяет уменьшить загрузку процессора в случае, когда сбор данных подвергающихся обработке

определяется разными (как правило, идущими друг за другом) событиями.

Например: Необходимо по сигналу 1 (underflow – спад до нуля) автоматически измерить сигналы I1, I2, I3, а по сигналу 2 (период) автоматически измерить сигналы V1, V2, V3. Сигналы 1 и 2 разделены по времени на 25мкс и обеспечиваются менеджером событий EVA рис.22

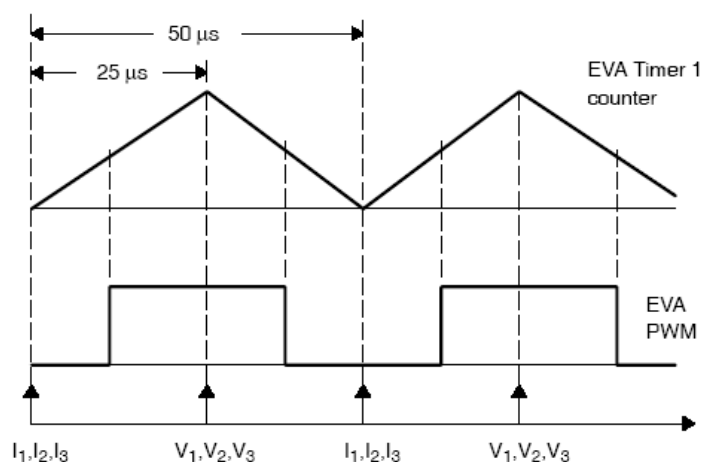


Рис. 22

Чтобы обеспечить решение поставленной задачи задействуем блок преобразования SEQ1. Количество преобразований устанавливаем равным трем (AdcRegs.ADCMAXCONV.MAXCONV1=2), а в регистры, определяющие последовательность выборки и преобразования загружаем значения согласно табл.5. Режим прерывания – INTMODSEQ1=1 (прерывание вырабатывается через раз). После включения и настройки блок преобразования ожидает запускающего сигнала (SOC) на начало преобразования от менеджера событий. После первого запуска происходит выборка значений I1, I2, I3. После второго запуска преобразуются сигналы V1, V2, V3 и вырабатывается прерывание. В обработчике прерывания происходит чтение регистров результата, обработка полученных результатов и сброс указателя (режим – CONTRUN=0).

Таблица 5

Биты 15-12	Биты 11-8	Биты 7-4	Биты 3-0	
(CONV03) V1	(CONV02) I3	(CONV01) I2	(CONV00) I1	ADCCHSELSEQ1
(CONV07) x	(CONV06) x	(CONV05) V3	(CONV04) V2	ADCCHSELSEQ2
(CONV11) x	(CONV10) x	(CONV09) x	(CONV08) x	ADCCHSELSEQ3
(CONV15) x	(CONV14) x	(CONV13) x	(CONV12) x	ADCCHSELSEQ4

Пример настройки АЦП

Допустим, мы хотим измерять напряжения с каналов ADCINA1, ADCINA2 и ADCINB0, причем последний канал два раза. Для этого настроим соответствующим образом делители частоты, и режим преобразования, например, настроим каскадный режим с ручным запуском. Схема тактирования АЦП показана на рис.8.

```
void InitAdc(void)
{
//Включение модуля
  AdcRegs.ADCCTRL1.bit.RESET = 1;           // Сброс модуля АЦП
  asm(" RPT #10 || NOP");                   // Пауза 12-cycles
//Вкл. ref and bandgap, установка делителя
  AdcRegs.ADCCTRL3.bit.ADCBGRFDN=3;
  AdcRegs.ADCCTRL3.bit.ADCCLKPS=6;         // Fclk=HSPCLK/2*PS
  DelayUs(7000);                            // Пауза 7ms
  AdcRegs.ADCCTRL3.bit.ADCPWDN = 1;        // Вкл. Основной модуль
  DelayUs(20);                               // Пауза 20us

//Настройка регистров
  AdcRegs.ADCMAXCONV.bit.MAX_CONV1=3; // 4 преобразования
  AdcRegs.ADCCHSELSEQ1.bit.CONV00=1; // Преобразуем INA1
  AdcRegs.ADCCHSELSEQ1.bit.CONV01=2; // Преобразуем INA2
  AdcRegs.ADCCHSELSEQ1.bit.CONV02=8; // Преобразуем INB0
  AdcRegs.ADCCHSELSEQ1.bit.CONV03=8; // Преобразуем INB0

  AdcRegs.ADCCTRL1.bit.SEQ_CASC=1;         // Каскадный режим

  AdcRegs.ADCCTRL2.bit.INT_ENA_SEQ1=1;    // Разрешим прерывание
  PieCtrlRegs.PIEIER1.bit.INTx6 = 1;      // Группа 1, номер 6
  IER |= 0x0001;                          // Включим INT1
} // end AdcInit()

Обработчик прерывания

interrupt void adc_isr()
{
  PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;
  AdcRegs.ADCCTRL2.bit.RST_SEQ1 = 1;      // Сброс SEQ1 (CONV00)
  AdcRegs.ADCST.bit.INT_SEQ1_CLR = 1;     // Очистка прерывания
  ...
  AdcRegs.ADCCTRL2.bit.SOC_SEQ1 = 1; //Запуск нового преобразования
//Первый запуск АЦП должен произойти в основной программе
}
```

7. ПРЕРЫВАНИЯ

Процессоры (ядро) серии 28xx имеют один вход немаскируемого прерывания (NMI) и 16 входов маскируемых прерываний с приоритетом выполнения (INT1 – INT14, RTOSIN и DLOGINT). С другой стороны, приборы этой серии обладают достаточно развитой периферией, которая вырабатывает гораздо большее число запросов не прерывание по различным событиям. Для коммутации этих запросов используется контроллер прерываний (PIE), который использует 12 входов прерывания ядра процессора INT1 – INT12. На рис.23 показана структурная схема формирования запроса на прерывание.

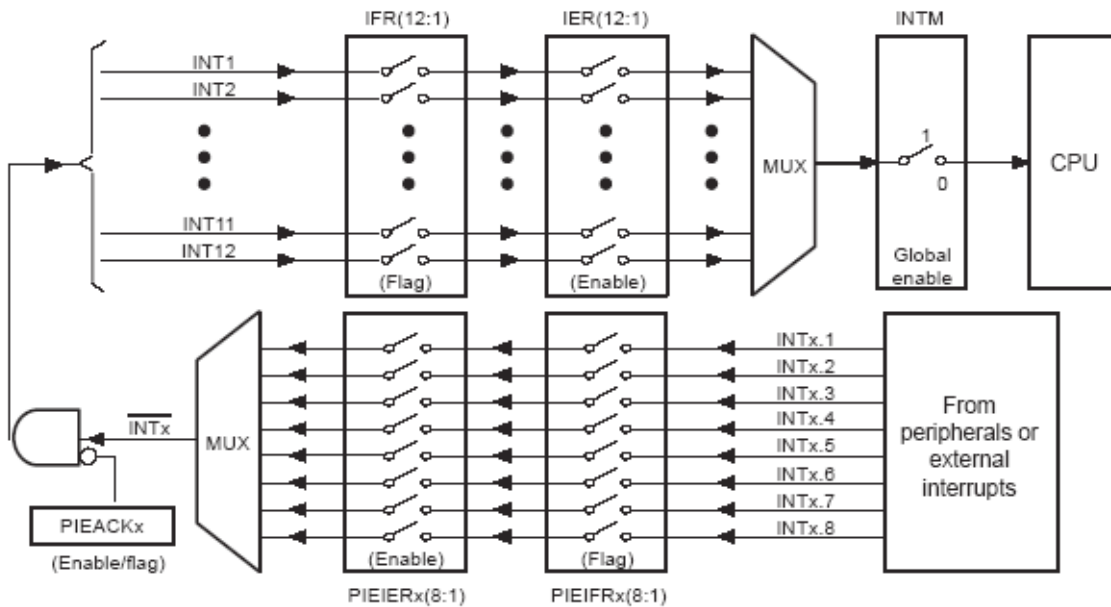


Рис.23

В ее состав входят регистры флагов и блокировок. Допустим, произошло некое событие (на уровне периферии), которое вызвало запрос на прерывание (например, АЦП закончил преобразование). Соответствующий флаг прерывания (IF) устанавливается в регистре этого периферийного устройства. Если соответствующий этому прерыванию бит разрешения (IE) установлен, т.е. на уровне периферийного устройства разрешены прерывания, то вырабатывается запрос на прерывание контроллеру PIE. (Флаг прерывания в периферийном регистре должен быть вручную очищен).

В блоке PIE все прерывания сгруппированы по восемь (INTx.1-8, здесь x – номер группы). Каждая группа подключена к своему процессорному входу прерывания, т.е. группа 1 к прерыванию INT1, группа 2 к прерыванию INT2, и так до INT12. Каждый вход прерывания в блоке

PIE имеет соответствующий флаг прерывания – PIEIFR_{x.y} и бит разрешения PIEIER_{x.y}. Кроме того, для каждой группы есть подтверждающий бит – PIEACK_x. Если этот бит установлен, то контроллер ждет его очищения, чтобы подать следующий запрос на прерывание.

Дальше, уже процессорный уровень. Блок PIE вырабатывает соответствующие запросы на прерывание (INT1 – INT12). Устанавливается соответствующий флаг прерывания (IFR_x) и если соответствующий бит разрешения установлен (IER_x), а также бит общего разрешения очищен (INTM), то выполняется запрос на прерывание по соответствующему адресу. (Регистры IER и IFR.)

!!! Флаги PIEIFR_x очищаются аппаратно, а подтверждающий бит – PIEACK_x, который устанавливается в момент прерывания нужно очищать вручную в подпрограмме прерывания, чтобы разрешить другие прерывания от данной группы. В табл.6 и табл.7 приведено распределение прерываний периферийных устройств по группам соответственно для семейств F281x и F280x.

Table 6–5. 281x PIE Peripheral Interrupts[†]

CPU Interrupts	PIE Interrupts							
	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1
INT1.y	WAKEINT (LPM/WD)	TINT0 (TIMER 0)	ADCINT (ADC)	XINT2	XINT1	Reserved	PDPINTB (EV-B)	PDPINTA (EV-A)
INT2.y	Reserved	T1OFINT (EV-A)	T1UFINT (EV-A)	T1CINT (EV-A)	T1PINT (EV-A)	CMP3INT (EV-A)	CMP2INT (EV-A)	CMP1INT (EV-A)
INT3.y	Reserved	CAPINT3 (EV-A)	CAPINT2 (EV-A)	CAPINT1 (EV-A)	T2OFINT (EV-A)	T2UFINT (EV-A)	T2CINT (EV-A)	T2PINT (EV-A)
INT4.y	Reserved	T3OFINT (EV-B)	T3UFINT (EV-B)	T3CINT (EV-B)	T3PINT (EV-B)	CMP6INT (EV-B)	CMP5INT (EV-B)	CMP4INT (EV-B)
INT5.y	Reserved	CAPINT6 (EV-B)	CAPINT5 (EV-B)	CAPINT4 (EV-B)	T4OFINT (EV-B)	T4UFINT (EV-B)	T4CINT (EV-B)	T4PINT (EV-B)
INT6.y	Reserved	Reserved	MXINT (McBSP)	MRINT (McBSP)	Reserved	Reserved	SPITXINTA (SPI)	SPIRXINTA (SPI)
INT7.y	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
INT8.y	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
INT9.y	Reserved	Reserved	ECAN1INT (CAN)	ECAN0INT (CAN)	SCITXINTB (SCI-B)	SCIRXINTB (SCI-B)	SCITXINTA (SCI-A)	SCIRXINTA (SCI-A)
INT10.y	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
INT11.y	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
INT12.y	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

[†] Out of the 96 possible interrupts, 45 interrupts are currently used. the remaining interrupts are reserved for future devices. However, these interrupts can be used as software interrupts if they are enabled at the PIEIFR_x level.

Регистры, контролирующие работу PIE-блока

а) Регистр управления – PIECTRL

№Бит	Название	Описание
15-1	PIEVECT	Это поле содержит адрес текущего вектора прерывания в таблице контроллера (PIE). Младший бит игнорируется.
0	ENPIE	Этот бит разрешает выборку адреса векторов прерывания из таблицы контроллера PIE. Если этот бит равен 0, то PIE блок выключен, и адреса векторов прерывания берутся из таблицы CPU (boot ROM или интерфейс внешней памяти Zone 7).

б) Регистр группового разрешения – PIEACK

в) Регистры флагов прерывания – PIEIFRx (x=1-12)

г) Регистр разрешения прерываний внутри группы – PIEIERx (x=1-12)

Примеры настройки прерываний

1. Допустим мы хотим организовать прерывание от менеджера событий EVA в момент равенства значения таймера 1 периоду (period).

а. При настройке работы менеджера событий разрешим прерывания

```
void init_eva()
{
...
EvaRegs.EVAIMRA.bit.T1PINT = 1; //Enable period int
PieCtrlRegs.PIEIER2.bit.INTx4 = 1; //Enable T1PINT in PIE group 2
IER |= M_INT2;
}
```

б. Запуск программы может быть таким

```
void main(void)
{
InitSysCtrl(); //PLL, WatchDog, enable Peripheral Clocks
InitPieCtrl(); //Запрещаем прерывания, выключаем PIE, выключаем прерывания и очищаем флаги, рег-ры PIEIERx=0 и PIEIFRx=0
IER = 0x0000; //выключаем прерывания и очищаем флаги на уровне CPU
IFR = 0x0000;
InitPieVectTable(); //Инициализируем таблицу векторов прерывания и включаем PIE
init_eva(); //Настраиваем менеджер событий EVA
EINT; //разрешаем прерывания
for{;;}
{основная часть программы}
}
```

- с. В обработчике прерывания необходимо разрешать групповое прерывание и очистить флаг прерывания (сам обработчик находится в файле DSP281x_DefaultIsr.c)

```
interrupt void T1PINT_ISR(void)
{
    ...
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP2;
    EvaRegs.EVAIFRA.bit = BIT7; //Clear flag
}
```

Можно переопределить вектор прерывания на ваш обработчик. Для этого необходимо объявить соответствующую функцию прерывания (interrupt) и после инициализации таблицы векторов прерывания сделать переопределение.

2. Допустим необходимо реализовать прерывания АЦП.

- а. При настройке работы АЦП разрешим прерывание

```
void init_adc(void)
{
    ...
    /*Enable the ADC interrupt */
    AdcRegs.ADCTRL2.bit.INT_ENA_SEQ1=1;
    PieCtrlRegs.PIEIER1.bit.INTx6 = 1;// Enable ADCINT in PIE group 1
    IER |= M_INT1;
}
```

- б. Запуск программы может быть таким

```
void main(void)
{
    InitSysCtrl();//PLL, WatchDog, enable Peripheral Clocks
    InitPieCtrl(); //Запрещаем прерывания, выключаем PIE, выключаем прерывания и очищаем флаги, рег-ры PIEIERx=0 и PIEIFRx=0
    IER = 0x0000; //выключаем прерывания и очищаем флаги на уровне CPU
    IFR = 0x0000;
    InitPieVectTable(); //Инициализируем таблицу векторов прерывания и включаем PIE
    //Переопределяем вектор прерывания
    EALLOW;
    PieVectTable.ADCINT = &adc_isr; //Прерывание АЦП, функция interrupt void adc_isr(void) должна быть объявлена
    EDIS;
    init_adc(); //Настраиваем АЦП
    EINT; //разрешаем прерывания
    for{;;}
        {основная часть программы}
}
```

- с. В обработчике прерывания необходимо разрешать групповое прерывание и очистить флаг прерывания (сам обработчик находится в файле DSP281x_DefaultIsr.c)

```
interrupt void adc_isr(void)
{
    ...
    PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;
    AdcRegs.ADCST.bit.INT_SEQ1_CLR = 1;
}
```

ПРЕДВАРИТЕЛЬНОЕ ЗАДАНИЕ

1. Освоение среды CCS

Запустите программу CCS и, следуя Tutorial (меню Help) проделайте первую часть обучающей программы, которая посвящена основам работы со средой Code Composer Studio.

2. Простые программы

В отладочном комплекте встроен светодиод, которым и будем управлять (порт GPIOF14).

Задание 1)

Написать и загрузить простую программу, которая бы циклически зажигала светодиод без использования таймера (обычная задержка ~ 30т. – 60т. циклов).

Методические указания.

Ознакомьтесь с работой портов и их управлением (раздел 2). Создать проект (раздел 1). Для освоения работы портов напишите несколько вариантов функций: с использованием TOGGLE, с использованием SET и CLEAR. Краткая информация по основам C изложена в приложении 1.

Задание 2)

Написать и загрузить простую программу, которая бы циклически зажигала светодиод с использованием таймера 1, менеджера событий EVA (период 250 мсек.).

Ознакомьтесь с работой менеджера событий и системы прерываний (разделы 3 и 4). Создать или изменить проект (раздел 1). Для освоения работы менеджера событий напишите несколько вариантов запуска и работы таймера с использованием прерывания.

КРАТКАЯ ИНФОРМАЦИЯ ПО С.

1. Типы данных

Целые: int, short, long – знаковые; unsigned int, unsigned short, unsigned long – беззнаковые.

С плавающей точкой: float, double.

Символьный (собственно целый) char.

2. Операторы:

*, /, +, - – умножение, деление, сложение, вычитание соответственно;

% – деление по модулю, для целых чисел (напр. 13%5 даст 3).

++, -- – инкремент, декремент (увеличение\уменьшение на 1), напр. ++I, или -I, может быть как префикс, так и постфикс, напр. Q=2*++x – вначале увеличится x, а затем произойдет умножение, если Q=2*x++, то x увеличится на 1 уже после умножения. Работает этот оператор только с переменными, например ошибкой будет запись вида (x*y)++.

3. Дополнительные операции присвоения

A+= 20; то же что и A=A+20; По аналогии: -=; *=; /=; %=.

4. Условные операторы

< – меньше; > – больше; <= – меньше или равно; >= – больше или равно; == – равно; != – не равно.

5. Логические операции

&& – логическое И; || – логическое ИЛИ; ! – логическое НЕ.

6. Побитовые логические операции

~ – НЕ; << – сдвиг влево; >> сдвиг вправо; & – И; | – ИЛИ; ^ – исключающее ИЛИ. С присвоением тоже работают: &= – И с присвоением, например A&=B; аналогично A=A&B;. Остальные логические действуют аналогично. Пример: A&=~(1<<27); – очищает 27 разряд A, A|=(1<<3); – устанавливает 3 разряд A.

7. Условный оператор

if(выражение) {оператор1; оператор2; и т.д.} – операторы выполняются если выражение истинно (не равно 0). Варианты с «иначе» (else):

```
if(выр-е) {...}    if(выр-е) {...}
else {...}         else if(выр)
                   else {...}
```

8. Циклы

for(n=0; n < 10; n++) {...} – операторы выполняются пока условие в скобках (второй оператор) не станет ложным

`while(x<5) { ...; ++x;}` – операторы выполняются пока условие в скобках «правда» (TRUE, не равно нулю).

9. Массивы

`int A[20];` – массив из 20 элементов типа `int`, первый элемент – `A[0]`, последний `A[19]`.

Вариант с заполнением: `int days[]={31,28,31};`

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Севернс Р., Блум Г. Импульсные преобразователи постоянного напряжения для систем вторичного электропитания: Пер. с англ. под ред. Л.Е. Смольникова. – М.: Энергоатомиздат, 1988. – 294 с.: ил.
2. Четти П. Проектирование ключевых источников электропитания/ Пер. с англ. – М.: Энергоатомиздат, 1990. – 240 с.
3. TMS321x281x DSP System Control and Interrupts Reference Guide (spru078.pdf)
4. TMS320x280x DSP System Control and Interrupts Reference Guide (spru712.pdf).
5. TMS320x281x DSP Event Manager (EV) Reference Guide (spru065.pdf)
6. TMS320x280x Analog to Digital Converter (ADC) Module Reference Guide (spru716.pdf).
7. Моин В.С. Стабилизированные транзисторные преобразователи. – М.: Энергоатомиздат, 1986. – 376 с.: ил.
8. Изерман Р. Цифровые системы управления: Пер. с англ. – М.: Мир, 1984. – 541 с.
9. Бесекерский В.А. Цифровые автоматические системы. – М.: Наука, 1976.

1. ОГЛАВЛЕНИЕ

Введение	3
Описание лабораторного стенда	3
ЛАБОРАТОРНАЯ РАБОТА № 1 Исследование понижающего преобразователя постоянного напряжения с обратной связью по выходному напряжению	10
ЛАБОРАТОРНАЯ РАБОТА № 2 Исследование понижающего преобразователя постоянного напряжения с обратной связью по выходному напряжению и току дросселя	13
ЛАБОРАТОРНАЯ РАБОТА № 3 Исследование повышающего преобразователя постоянного напряжения с обратной связью по выходному напряжению	16
ЛАБОРАТОРНАЯ РАБОТА № 4 Исследование обратного преобразователя (инвертирующего) постоянного напряжения с обратной связью по выходному напряжению	18
ЛАБОРАТОРНАЯ РАБОТА № 5 Исследование мостового инвертора напряжения с выходным LC-фильтром и многократной широтно-импульсной модуляцией выходного напряжения	20
ЛАБОРАТОРНАЯ РАБОТА № 6 Исследование мостового резонансного инвертора напряжения с фазовой автоподстройкой частоты	24
ЛАБОРАТОРНАЯ РАБОТА № 7 Исследование трехфазного мостового инвертора напряжения	26
Создание проекта и описание основных функциональных блоков процессора TMS320F28xx	29
3. Менеджер событий (Event manager)	41
4. Модуль ШИМ для семейства F280x (ePWM)	52
5. Модуль захвата семейства F280x (eCAP)	58
6. Аналого-цифровой преобразователь (АЦП)	63
7. Прерывания	72
Предварительное Задание	76
Краткая информация по С	77
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	79

Буркин Евгений Юрьевич

**Цифровые системы управления
устройств силовой электроники**

Практикум

Редактор М.В. Сурикова

Подписано к печати 10.08.06.
Формат 60×84/16. Бумага офсетная.
Печать RISO. Усл. печ. л. 2,56. Уч.-изд. л. 2,32.
Тираж 100 экз. Заказ . Цена свободная.
Издательство ТПУ. 634050, Томск, пр. Ленина, 30.