МИНЕСТЕРСТВО ПО РАЗВИТИЮ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И КОММУНИКАЦИЙ РЕСПУБЛИКИ УЗБЕКИСТАН
НУКУССКИЙ ФИЛИАЛ ТАШКЕНТСКОГО УНИВЕРСИТЕТА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ИМЕНИ МУХАММАДА АЛЬ-ХОРЕЗМИ

Направление:_____

Курсовая Работа

По предмету_____
На тему _____

Выполнил(а): **Сапаров Ф.**
Принял(а): _____

НУКУС 2018

ПРОЕКТИРОВАНИЕ МИКРОКОНТРОЛЛЕРНЫХ СИСТЕМ

Содержание:

Введение

1. Критерии выбора микроконтроллера
2. Структура микроконтроллеров семейства MCS-96
3. Структура кристалла
4. Ядро 8XC196KC/KD
5. Регистровый файл
6. Техника кадрирования МК 8XC196KC/KD

Заключение

Список литературы

Введение

При проектировании микропроцессорной системы (МПС) необходимо четко представлять все задачи, которые данная система должна решать. Исходные данные для проектирования должны содержать требования к основным технико-экономическим показателям: производительности, энергопотреблению, стоимости, надежности, конструктивным и другим параметрам.

Разработка архитектуры системы подразумевает определение оптимального состава аппаратных и программных средств для решения поставленных задач. При этом разработчик решает какие функции системы будут реализованы аппаратными средствами, а какие - программным обеспечением.

В процессе разработки МПС удобно переходить от одного уровня ее представления к другому, более детальному. Каждая абстракция несет в себе только информацию, которая соответствует данному уровню, и не содержит каких-либо сведений относительно более низких уровней.

Микропроцессорная система может быть описана, например, на одном из следующих уровней абстрактного представления:
1) "черный ящик",
2) структурный;
3) программный;
4) логический;
5) схемный.

На уровне **"черного ящика"** МПС описывается внешними спецификациями; перечисляются внешние характеристики.

Структурный уровень создается компонентами микропроцессорной системы - микропроцессорами, запоминающими устройствами, устройствами ввода-вывода, внешними запоминающими устройствами, каналами связи. МПС описывается функциями отдельных устройств и их взаимосвязью, информационными потоками.

Программный уровень разделяется на два подуровня: команд процессора и языковой. Микропроцессорная система интерпретируется как последовательность операторов или команд, вызывающих то или иное действие над некоторой структурой данных. Кроме того, предварительно для управляющих систем должны быть определены реализуемые алгоритмы управления, для универсальных систем - классы выполняемых задач.

Логический уровень присущ исключительно дискретным системам. На этом уровне выделяются два подуровня: переключательных схем и регистровых пересылок. Подуровень переключательных схем, образуется вентилями и построенными на их основе операторами обработки данных. Подуровень регистровых пересылок характеризуется более высокой степенью абстрагирования и представляет собой

описание регистров и передачу данных между ними. Он включает в себя две части: информационную и управляющую. Информационная часть образуется регистрами, операторами и путями передачи данных. Управляющая часть обеспечивает зависящие от времени сигналы, инициирующие пересылку данных между регистрами.

Схемный уровень образуется резисторами и конденсаторами. Показателями поведения системы на этом уровне служат напряжение и ток, представляемые в функции времени или частоты.

Основные этапы проектирования микропроцессорных систем, их характеристики, возможные ошибки на каждом из этапов и способы их устранения рассмотрены в учебнике "Микропроцессоры".

Одним из важнейших вопросов при проектировании микропроцессорной системы является выбор микроконтроллера (МК). Разнообразие МК обеспечивает **модульный принцип построения.** Суть данного принципа состоит в том, что все микроконтроллеры одного семейства содержат в себе **базовый функциональный блок,** одинаковый для всех МК семейства и **изменяемый функциональный блок,** который отличает МК разных моделей в пределах одного семейства.

Базовый функциональный блок включает: центральный процессор (ЦП), внутренние шины адреса, данных и управления, схему формирования многофазной импульсной последовательности для тактирования ЦП и других модулей, устройство управления режимами работы МК.

Изменяемый функциональный блок содержит модули различных типов памяти, модули периферийных устройств и модули генераторов синхронизации и некоторых других специальных режимов работы МК.

В последнее время на российском рынке цифровых интегральных схем можно найти практически любые микроконтроллеры. Мы остановимся на шестнадцатиразрядных микроконтроллерах семейства MCS-96 фирмы Intel. МК этого семейства предназначены для использования в управляющих и контрольноизмерительных системах и приборах различного назначения в качестве встраиваемых микропроцессорных устройств.

Основными достоинствами МК семейства MCS-96, отличающими их от микроконтроллеров других семейств, являются:

- расширенная разрядная сетка, позволяющая выполнять операции с данными, представленными в форматах "байт" и "слово", а некоторые операции - в формате "двойное слово";

- улучшенная система операций, содержащая операции умножения и деления для чисел со знаком и без знака при разных форматах представления данных, операции сдвига на заданное число разрядов, операции групповой пересылки и др.;

- совершенная система команд, имеющая в своем составе двухадресные и трехадресные команды арифметических и логических операций с различными способами адресации, что позволяет создавать компактные и быстродействующие программы;

- многообразие расположенных на кристалле периферийных устройств, выполняющих функции ввода и вывода данных, событий и аналоговых сигналов, обслуживания запросов прерывания без прерывания текущей программы, контроля правильности функционирования МК, что позволяет разрабатывать малогабаритные и надежные устройства с минимальным числом дополнительных микросхем;

- наличие большого числа программных и программно-аппаратных средств поддержки разработки аппаратуры на базе МК семейства MCS-96 и в том числе программно-аппаратного комплекса Project BUILDER, предназначенного для первоначального освоения МК этого семейства [6].

Таким образом, микроконтроллеры семейства MCS-96 - это достаточно сложные многофункциональные устройства, изучение которых поможет быстрее освоить МК других фирм. Ведь при проектировании конкретной микропроцессорной системы нужно выбирать оптимально подходящий МК, критерии выбора которого приведены ниже.

Критерии выбора микроконтроллера

Основные критерии выбора микроконтроллера представлены ниже в порядке их значимости.

-Пригодность для прикладной системы. Может ли система быть реализована на микроконтроллере, или надо использовать какую либо специализированную микросхему.

-Имеет ли микроконтроллер требуемое число контактов портов ввода/ вывода, т.к. в случае их недостатка, он не сможет выполнить работу, а в случае избытка цена будет слишком высокой.

-Имеет ли он все требуемые встроенные периферийные устройства, такие как последовательные порты ввода/вывода, таймеры, АЦП, ЦАП, и т.д., внутреннюю память - RAM, ROM и какого типа.

-Имеет ли контроллер другие периферийные устройства, которые не потребуются в системе.

-Обеспечивает ли ядро процессора необходимую производительность, т.е. вычислительную мощность, позволяющую обрабатывать системные запросы в течение всей жизни системы на выбранном прикладном языке. Выделено ли в бюджете проекта достаточно средств, чтобы позволить себе использовать данный микроконтроллер. Для ответа на этот вопрос, обычно требуются расценки поставщика. Если данный микроконтроллер не приемлем для проекта, все

остальные вопросы становятся несущественными, и вы должны начать поиски другого микроконтроллера.

-Возможность покупки данного микроконтроллера.

-Доступность программного обеспечения (ассемблеры, компиляторы), отладчики программ в исходных текстах.

-Наличие средств отладки: оценочные модули (EVM), внутрисхемные эмуляторы, насадки для логических анализаторов, отладочные мониторы.

-Информационная поддержка.

-Примеры применения выбранного микроконтроллера.

Структура микроконтроллеров семейства MCS-96

Все микроконтроллеры семейства MCS-96 имеют единую базовую структуру, включающую (см. рис. 1.1):

- центральное процессорное устройство (Central Processor Unit, CPU);

- внутреннюю память (Internal Memory, IM);

- набор периферийных устройств (Peripherals);

- контроллер памяти (Memory Controller, MC).

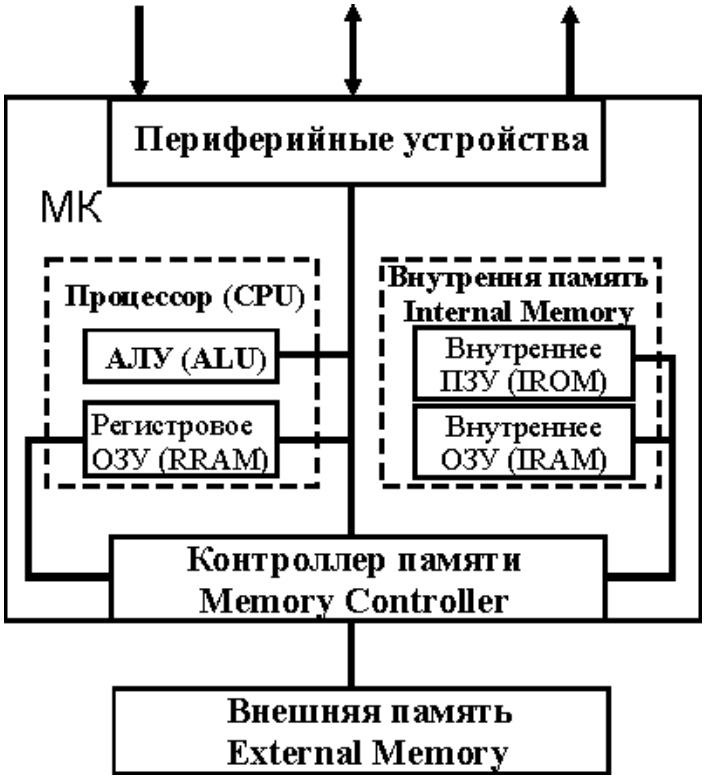


Рис. 1.1. Структура МК семейства MCS-96

К микроконтроллеру может быть подключена внешняя память (External Memory).

В состав процессора входят арифметико-логическое устройство (Arithmetic-Logic Unit, ALU) и регистровое оперативное запоминающее устройство (Register RAM - RRAM).

В ALU по командам программы выполняются арифметические и логические операции. В систему команд кроме арифметических и логических операций входят команды пересылочных операций и команды операций управления. Общее число команд у МК разных типов может быть равным 100, 106, 112 или 120.

Отличительной особенностью ALU микроконтроллеров семейства MCS-96 является отсутствие регистра-аккумулятора. В качестве источника первого операнда и приемника результата может использоваться любой регистр в RRAM, при этом операнд и результат могут иметь разные адреса. В состав ALU входят регистры временного хранения данных, главный счетчик команд (Master Program Counter), регистр команд (Instruction Register) и другие схемы, обеспечивающие выполнение операций и ход программы.

Регистровое оперативное запоминающее устройство используется для хранения данных. В состав RRAM входит массив восьмиразрядных регистров. Число регистров в RRAM у МК разных типов может быть равным 232, 360, 488 или 1000. ALU может обращаться к регистрам RRAM непосредственно или через контроллер памяти.

Внутренняя память микроконтроллера содержит внутреннее постоянное запоминающее устройство (Internal ROM - IROM) и внутреннее оперативное запоминающее устройство (IRAM).

IROM используется для хранения команд программы, констант и специальных данных. Выпускаются МК, у которых IROM отсутствует. В этом случае функции IROM выполняет запоминающее устройство, входящее в состав внешней памяти. Обращение к ячейкам памяти в IROM выполняется только через контроллер памяти.

IRAM используется для хранения данных и команд программы. У некоторых МК IRAM отсутствует. Обращение к ячейкам памяти в IRAM выполняется только через контроллер памяти.

Контроллер памяти управляет процессом обращения к IROM, IRAM и к внешней памяти, а в некоторых случаях - к RRAM. В контроллере памяти расположены вспомогательный счетчик команд (Slave Program Counter), стек типа FIFO, имеющий четыре восьмиразрядные ячейки, и другое оборудование, управляющее процессом обращения к памяти. Вспомогательный счетчик команд обеспечивает опережающую выборку кодов команд с их запоминанием в стеке FIFO.

Подключение внешней памяти позволяет расширить объем памяти МК. При отсутствии IROM использование внешней памяти обязательно. Общее число адресов, по которым производится обращение к внутренним регистрам и ячейкам во внутренней и внешней памяти, составляет 64 Кбайта, а у некоторых типов МК может быть увеличено до 1 Мбайта.

Периферийные устройства (ПУ) используются для приема и выдачи данных, событий и аналоговых сигналов, для обслуживания запросов прерываний и контроля правильности работы МК. Микроконтроллеры разных подсемейств имеют разный состав периферийных устройств. В таблице 1.1 перечислены все периферийные устройства, которые используются в МК семейства MCS-96, при этом ПУ объединены в пять функциональных групп.

Таблица 1.1

1	Устройство ввода/вывода данных
1.1	Параллельный порт Port (P)
1.2	Процессорный порт Slave Port (SLP)
1.3	Последовательный порт Serial Port (SP)
1.4	Синхронный последовательный порт Sincronous serial Port
2	Устройства ввода/вывода событий
2.1	Блок быстрого ввода/вывода Hight Speed Input/Output (HSIO)
2.2	Блок процессоров событий Event Processor Array (EPA)
3	Устройства ввода/ввывода аналоговых сигналов
3.1	Аналого-цифровой преобразователь Analog to Digital Converter
3.2	Широтно-импульсный модулятор Pulse Width Modulator (PWM)
3.3	Трехфазный генератор Waveform Generator (WG)
3.4	Генератор меандра freguency Generator (FG)
4	Устройство обслуживания запросов прерывания
4.1	Схема управления прерываниями Interrupt Control Circuitry (ICC)
4.2	Сервер переферийных транзакций Pheripheral Transaction Server (PTS)
5	Устройство контроля
5.1	Сторожевой таймер Watchdog Timer (WDT)
5.2	Детектор падения частоты Oscillator Fail Detect (OFD)

МК семейства MCS-96 являются синхронными цифровыми устройствами. В состав микроконтроллера входит генератор тактовых импульсов. Частота следования так товых импульсов определяется резонансной частотой внешнего кварцевого резонатора Fosc. Частота Fosc является обобщенной характеристикой быстродействия МК. У микроконтроллеров разных типов максимальное значение частоты может быть равным 10, 12, 16 или 20 МГц.

Состав семейства

В семейство MCS-96 входят микроконтроллеры, имеющие разные структурные характеристики (состав внутренних запоминающих устройств и периферийных устройств) и разные количественные характеристики (емкость запоминающих устройств, максимальная частота кварцевого резонатора и др.). МК изготавливаются по двум разным технологиям - высококачественной НМДП технологии (HMOS) и комбинированной КМДП технологии (CHMOS).

Микроконтроллеры, изготавливаемые по одной технологии и имеющие одинаковые или близкие структурные характеристики, но различающиеся количественными характеристиками, объединяются в подсемейство МК. В семействе MCS-96 могут быть выделены шесть подсемейств, которые обозначаются двумя буквами латинского алфавита: BH, KB, KC, KR, NT и MC. Микроконтроллеры подсемейства BH изготавливаются по HMOS технологии, МК остальных семейств - по CHMOS технологии.

Подсемейства делятся на типы. К одному типу принадлежат МК данного подсемейства, имеющие одинаковые количественные характеристики. Микроконтроллеры, различающиеся только наличием или отсутствием IROM, относятся к одному типу. В этом случае при наличии IROM это запоминающее устройство может быть масочным (MROM), или программируемым пользователем однократно (OTPROM), или многократно со стиранием ранее сделанной записи путем ультрафиолетового облучения кристалла через окно в корпусе микросхемы (EPROM).

Тип МК и некоторые его характеристики приводятся на корпусе микросхемы с использованием определенной системы обозначений, например, LS87C196KC20. В таблице 1.2 приведено обозначение на корпусе МК и показано деление на поля, в скобках указаны обзначения полей для приведенного примера.

Таблица 1.2

Номер поля	Обозначение
1 (L)	Указывается температурный диапазон работы микросхемы и наличие или отсутствие термотренировки перед поступлением микросхемы в продажу: L - микросхема может работать в диапазоне температур от -40 до +85 С и прошла термотренировку;

	T - тот же температурный диапазон, но микросхема не подвергалась термотренировке. При отсутствии буквы микросхема имеет рабочий диапазон температуры от 0 до +70 С и не подвергалась термотренировке.
2 (S)	Указывается тип корпуса микросхемы (одна или две буквы латинского алфавита: A, B, C, N, P, R, S, SB, U)
3 (87)	Указывается наличие или отсутствие IROM и тип IROM при его наличии: 80 - МК не имеет IROM; 83 - МК имеет IROM масочного типа; 87 - МК имеет EPROM.
4 (C196)	Используется для обозначения технологии изготовления микросхемы: 95, 96, 97, 98 - HMOS технология; C194, C196, C198, L196 - CHMOS технология. При этом буква "C" означает, что для питания микросхемы должен использоваться источник с напряжением 5 В, буква "L" - 3,3 В.
5 (KC)	Используется для обозначения типа МК. Здесь могут быть записаны две буквы: BH, JF, KB, KC, KR, JR, KQ, JQ, KT, KS, NT, NQ, MC, MD или буквы отсутствуют.
6 (20)	Используется для обозначения максимальной частоты кварцевого резонатора для данной микросхемы (в МГц). Могут быть записаны числа 10, 12, 16, 20 или число отсутствует.

Обозначения в полях N 4 и 5 в совокупности образуют обозначение типа микроконтроллера.

В таблице 1.3 приведены основные структурные и количественные характеристики микроконтроллеров всех типов семейства MCS-96. В таблице указаны:
- число команд в системе команд (NI);
- емкость RRAM, IROM, IRAM (в числе восьмиразрядных регистров или ячеек памяти);
- объем адресного пространства (AS) (в числе восьмиразрядных элементов);
- число параллельных портов (P);
- наличие (+) или отсутствие (-):
1) процессорного порта (SLP);
2) последовательного порта (SP);
3) синхронного последовательного порта (SSIO);
4) блока быстрого ввода/вывода (HSIO);
5) блока процессоров событий (EPA);
6) аналого-цифрового преобразователя (ADC) и числа входных каналов при его наличии;
7) широтно-импульсного модулятора (PWM) и числа выходных каналов при его наличии;

8) трехфазного генератора (WG);
9) генератора меандра (FG);
10) сервера периферийных транзакций (PTS);
11) детектора падения частоты (OFD);
- максимальная частота кварцевого резонатора (Fcr max) (в Мгц).

Таблица 1.3

Тип МК	NI	Память				Периферийные устройства												Fcr(max)
		RRAM	IROM	IRAM	AS	P	1	2	3	4	5	6	7	8	9	10	11	
95BH	100	232	8K	-	64K	4	-	+	-	+	-	4	1	-	-	-	-	12
96BH	100	232	8K	-	64K	5	-	+	-	+	-	-	1	-	-	-	-	12
97BH	100	232	8K	-	64K	5	-	+	-	+	-	8	1	-	-	-	-	12
97JF	100	232	16K	256	64K	5	-	+	-	+	-	8	1	-	-	-	-	12
98	100	232	8K	-	64K	4	-	+	-	+	-	4	1	-	-	-	-	12
C194	106	232	8K	-	64K	4	-	+	-	+	-	-	1	-	-	-	-	12
C196KB	106	232	8K	-	64K	5	-	+	-	+	-	8	1	-	-	-	-	16
C198	106	232	8K	-	64K	4	-	+	-	+	-	4	1	-	-	-	-	16
C196KC	112	488	16K	-	64K	5	-	+	-	+	-	8	3	-	-	+	-	20
C196KD	112	1000	32K	-	64K	5	-	+	-	+	-	8	3	-	-	+	-	20
L196KD	112	1000	32K	-	64K	5	-	+	-	+	-	8	3	-	-	+	-	16
C196KR	112	488	16K	256	64K	7	+	+	+	-	+	8	-	-	-	+	+	16
C196KQ	112	360	12K	128	64K	7	+	+	+	-	+	8	-	-	-	+	+	16
C196JR	112	488	16K	256	64K	7	-	+	+	-	+	6	-	-	-	+	+	16
C196JQ	112	360	12K	128	64K	7	-	+	+	-	+	6	-	-	-	+	+	16
C196KT	112	1000	32K	512	64K	7	+	+	+	-	+	8	-	-	-	+	+	16
C196KS	112	1000	24K	256	64K	7	+	+	+	-	+	8	-	-	-	+	+	16
C196NT	120	1000	32K	512	1M	8	+	+	+	-	+	4	-	-	-	+	+	20
C196NQ	120	360	12K	128	1M	8	+	+	+	-	+	4	-	-	-	+	+	16
C196MC	112	488	16K	-	64K	7	-	-	-	-	+	13	2	+	-	+	-	16
C196MD	112	488	16K	-	64K	8	-	-	-	-	+	14	2	+	+	+	-	16

II. ПРЕДСТАВЛЕНИЕ ДАННЫХ И КОМАНД ПРОГРАММЫ

Команды программы и данные MCS-96 хранятся во внутреннем постоянном запоминающем устройстве (ЗУ) (IROM), внутреннем оперативном ЗУ (IRAM) и внешних ЗУ (EROM, ERAM). Данные, кроме того, хранятся в регистровом оперативном ЗУ (RRAM).

Адресуемым элементом IROM, IRAM, EROM и ERAM является восьмиразрядная ячейка памяти. Адресуемым элементом RRAM является

восьмиразрядный регистр. Номера регистров и ячеек памяти образуют единое адресное пространство МК.

Данные в памяти МК могут быть представлены в двух форматах: "байт" и "слово". Для хранения "байта" используется ячейка памяти, для хранения "слова" - пара соседних ячеек. Адрес "слова" должен быть четным.

Данные в RRAM могут быть представлены в трех форматах: "байт", "слово" и "двойное слово".

Множество команд, входящих в систему команд МК семейства MCS-96 (табл. 2.1), по виду операции, выполняемой по команде, делится на две группы:
1) команды операций с данными;
2) команды операций управления.

Группа команд операций с данными делится на две подгруппы:
1) команды арифметических и логических операций;
2) команды пересылочных операций.

Таблица 2.1

Подсемейство	Всего команд	Арифметических и логических команд	Команд пересылки	Команд управления
BH, JF	100	56	10	34
KB	106	57	13	36
KC, KR, KT, MC	112	57	16	39
NT	120	57	21	42

Арифметические операции могут выполняться с числами без знака в двоичном коде и с числами со знаком в дополнительном двоичном коде.

Коды команд могут иметь длину от одного (например, команды CLRC, DI, EI, NOP) до шести байтов (например, ADD, SUB, AND, CMP). Первый байт является кодом операции. В командах операций умножения и деления для чисел со знаком перед кодом операции размещается дополнительный байт-префикс. При наличии префикса максимальное число байтов в команде возрастает до семи (например, команды MUL, MULB).

В памяти МК коды команд хранятся в группах соседних ячеек памяти, при этом по младшему адресу располагается код операции, а при наличии префикса - префикс. Младший адрес в группе является адресом команды.

По числу адресов, используемых при выполнении арифметических и логических операций, команды бывают одноадресными, двухадресными и трехадесными.

Способы адресации данных

1. Непосредственная адресация. При непосредственной адресации операнд (байт или слово) размещается непосредственно в коде команды.

ADD AX,#220	К содержимому регистра AX прибавляется 220 и результат сохраняется в AX
PUSH #1232h	В стек помещается число 1232h

2. Прямая регистровая адресация. Прямая регистровая адресация используется в тех случаях, когда операнд или результат размещается в RRAM и его адрес находится в пределах от 0000H до 00FFH. В этом случае для представления адреса может быть использован один байт в коде команды.

ADD AX,BX,CX	Складываются значения в регистрах BX и CX, результат сохраняется в регистре AX
INCB CL	Содержимое счетчика CL увеличивается на 1

Остальные способы адресации используются при обращении к ячейкам памяти и регистрам RRAM, когда для указания адреса операнда требуется использовать два байта (адрес в формате "слово").

3. Косвенная адресация. При косвенной адресации адрес операнда, имеющий формат "слово", записывается в пару регистров в RRAM в область с адресами от 1AH до FFH, а адрес адреса операнда, имеющий формат "байт", входит в состав кода команды.

LD AX,[AX]	В регистр AX пересылается значение из ячейки памяти с адресом, на который ссылается AX {AX MEM_WORD (AX)}
POP [AX]	Из стека пересылается значение из ячейки памяти с адресом, на который ссылается AX; {AX <- MEM_WORD MEM_WORD (AX) <- MEM_WORD (SP)}; SP <- SP+2

4. Косвенная адресация с автоинкрементом. При косвенной адресации с автоинкрементом после выполнения операции с данными адрес операнда увеличивается на 1, если операнд имеет формат "байт", или на 2, если операнд имеет формат "слово".

LD AX,[BX]+	В регистр AX пересылается значение из ячейки памяти с адресом, на который ссылается BX {BX <- MEM_WORD (BX)} и адрес увеличивается на два { BX <- BX+2}
PUSH [AX]+	Из стека пересылается значение из ячейки памяти с адресом, на который ссылается AX; {AX <- MEM_WORD MEM_WORD (AX) <-

MEM_WORD (SP)};
SP <- SP+2

5. Индексная короткая адресация. При индексной адресации для задания адреса операнда A используют базовый адрес B и смещение (Offset) W. Адрес операнда образуется путем сложения базового адреса и смещения: A=D+W, где W - целое число со знаком в дополнительном коде.

Базовый адрес B, имеющий формат "слово", записывается в пару регистров в RRAM в область с адресами от 1AH до FFH, а адрес базового адреса, имеющего формат "байт", входит в состав кода команды. Смещение может быть представлено в форматах "байт" или "слово".

Если смещение имеет формат "байт" (-128<=W<=+127), индексная адресация является короткой.

LD AX,12[BX] В регистр AX пересылается значение из ячейки памяти с адресом, который определяется как сумма (BX+12);
{AX <- MEM_WORD (BX+12)}

MULB AX,BL,3[CX] В регистр AX пересылается результат умножения содержимого регистра BL и содержимого ячейки памяти с адресом, который определяется как сумма (CX+3)
{AX <- BL * MEM_BYTE (CX+3)}

6. Индексная длинная адресация. Если смещение имеет формат "слово" (-32768<=W<=+32767), индексная адресация является длинной.

AND AX,BX,TABLE[CX] В регистр AX пересылается результат логического умножения содержимого регистра BX и содержимого ячейки памяти с адресом, который определяется как сумма слова из таблицы и значения в регистре CX;
{AX <- BX and MEM_WORD(TABLE+CX]}

ST AX, TABLE[BX] Содержимое регистра AX пересылается в ячейку памяти с адресом, который определяется как сумма слова из таблицы и значения в регистре BX;
{MEM_WORD (TABLE+BX) <- AX}

Признаки результата операции

В МК семейства MCS-96 используются шесть признаков результата:

1. Нулевой результат (Z). Признак Z принимает единичное значение при получении нулевого результата операции (все биты кода результата имеют нулевое значение).

2. Отрицательный результат (N). Значение признака N совпадает со значением старшего бита кода результата. Если результат является числом со знаком, старший бит кода результата имеет единичное значение при отрицательном результате.

3. Переполнение разрядной сетки (V). Признак V принимает единичное значение, если при выполнении операций сложения и вычитания чисел без знака и со знаком возникает комбинация значений старших битов операндов и результата, соответствующая переполнению разрядной сетки. Комбинация значений старших битов кодов первого операнда (x), второго операнда (y) и результата (z), при которых признак V принимает единичное значение, приведены в таблице 2.2.

Таблица 2.2

Старший бит	Операция			
	Сложение		Вычитание	
X	0	1	0	1
Y	0	1	1	0
Z	1	0	1	0

Признак V, кроме того, принимает единичное значение, если возникает переполнение разрядной сетки при выполнении операции деления и операции сдвига влево. Признак V принимает соответствующее значение при выполнении каждой очередной операции, при которой он формируется.

4. Сохраняемый признак переполнения разрядной сетки (VT). Признак VT принимает единичное значение в тех же случаях, что и признак V, но единичное значение этого признака сохраняется до выполнения команды условного перехода, в которой используется этот признак, или до выполнения команды операции управления, по которой этот признак принимает нулевое значение (CLRVT).

5. Перенос (C). Значение признака C:
- совпадает со значением переноса из старшего разряда при выполнении операции сложения;
- противоположно значению переноса из старшего разряда при выполнении операции вычитания;
- совпадает со значением бита кода, выходящего за пределы разрядной сетки при выполнении операций сдвига кода влево или вправо.

6. Сохраняемый перенос при сдвиге кода вправо (ST). Признак ST имеет единичное значение, если при выполнении операции сдвига кода вправо за пределы разрядной сетки вышла хотя бы одна единица.

Значение признаков результата представляется состоянием разрядов регистра слова состояния программы (Program Status Word, PSW).

Одноадресные команды

При выполнении одноадресной команды используется один адрес, по которому до выполнения операции находится операнд, а после выполнения операции записывается результат (см. табл. 2.3).

Таблица 2.3

No п/п	Операция	Формат данных		
		Байт	Слово	Двойное слово
1	Сложение с единицей	INCB	INC	-
2	Вычитание с единицей	DECB	DEC	-
3	Обнуление	CLRB	CLR	-
4	Инверсия	NOTB	NOT	-
5	Изменение знака	NEGB	NEG	-
6	Рсширение со знаком	EXBT		EXT

В одноадресных командах используется прямая регистровая адресация.

INCB CL; CL <- CL+1

DEC AX; AX <- AX-1

Двухадресные команды

При выполнении операции с двумя операндами (двухадресная команда) по первому адресу до выполнения операции находится первый операнд, а после выполнения операции записывается результат.

Для указания местоположений первого операнда и результата используется прямая регистровая адресация. По другому адресу находится второй операнд, для указания его местоположения могут быть использованы все шесть способов адресации (см. табл. 2.4).

Таблица 2.4

No п/п	Операция	Формат данных		
		Байт	Слово	Двойное слово
1	Сложение	ADDB	ADD	-
2	Сложение с переносом	ADDCB	ADDC	-
3	Вычитание	SUBB	SUB	-
4	Вычитание с займом	SUBCB	SUBC	-
5	Логическое умножение	ANDB	AND	-
6	Логическое сложение	ORB	OR	-
7	Исключающее ИЛИ	XORB	XOR	-

8	Сравнение	CMPB	CMP	CMPL
9	Сдвиг влево	SHLB	SHL	SHLL
10	Сдвиг вправо	SHRB	SHR	SHRL
11	Арифметический сдвиг вправо	SHRAB	SHRA	SHRAL
12	Умножение числа без знака	MULUB		MULU
13	Умножение числа со знаком	MULB		MUL
14	Деление числа без знака	DIVUB		DIVU
15	Деление числа со знаком	DIVB		NORML DIV
16	Нормализация	-	-	NORML

ADD AX,BX; AX <- (AX+BX)

ST AX, TABLE[BX]; MEM_WORD (TABLE+BX) <- AX

Трехадресные команды

С помощью трехадресных команд выполняются арифметические и логические операции с двумя операндами (см. табл. 2.5). При этом используется три адреса. По первому адресу записывается результат операции, по второму и третьему - первый и второй операнд соответственно.

В трехадресных командах для указания местоположений первого операнда и результата, имеющих разные адреса, используется прямая регистровая адресация, а для указания местоположения второго операнда могут быть использованы все шесть способов адресации.

Таблица 2.5

No п/п	Операция	Формат данных		
		Байт	Слово	Двойное слово
1	Сложение	ADDB	ADD	-
2	Вычитание	SUBB	SUB	-
3	Логическое умножение	ANDB	AND	-
4	Умножение числа без знаком	MULUB		MULU
5	Умножение числа со знаком	MULB		MUL

MULB AX,BL,3[CX] AX <- BL * MEM_BYTE (CX+3);

	(MULB_3op)
AND AX,BX,TABLE[CX]	AX <- BX and MEM_WORD(TABLE +CX]; (AND_3op)

Двухадресные команды

При выполнении операции с двумя операндами (двухадресная команда) по первому адресу до выполнения операции находится первый операнд, а после выполнения операции записывается результат.

Для указания местоположений первого операнда и результата используется прямая регистровая адресация. По другому адресу находится второй операнд, для указания его местоположения могут быть использованы все шесть способов адресации (см. табл. 2.4).

Таблица 2.4

No п/п	Операция	Формат данных		
		Байт	Слово	Двойное слово
1	Сложение	ADDB	ADD	-
2	Сложение с переносом	ADDCB	ADDC	-
3	Вычитание	SUBB	SUB	-
4	Вычитание с займом	SUBCB	SUBC	-
5	Логическое умножение	ANDB	AND	-
6	Логическое сложение	ORB	OR	-
7	Исключающее ИЛИ	XORB	XOR	-
8	Сравнение	CMPB	CMP	CMPL
9	Сдвиг влево	SHLB	SHL	SHLL
10	Сдвиг вправо	SHRB	SHR	SHRL
11	Арифметический сдвиг вправо	SHRAB	SHRA	SHRAL
12	Умножение числа без знака	MULUB		MULU
13	Умножение числа со знаком	MULB		MUL
14	Деление числа без знака	DIVUB		DIVU
15	Деление числа со знаком	DIVB		NORML DIV
16	Нормализация	-	-	NORML

ADD AX,BX; AX <- (AX+BX)

ST AX, TABLE[BX]; MEM_WORD (TABLE+BX) <- AX

Трехадресные команды

С помощью трехадресных команд выполняются арифметические и логические операции с двумя операндами (см. табл. 2.5). При этом используется три адреса. По первому адресу записывается результат операции, по второму и третьему - первый и второй операнд соответственно.

В трехадресных командах для указания местоположений первого операнда и результата, имеющих разные адреса, используется прямая регистровая адресация, а для указания местоположения второго операнда могут быть использованы все шесть способов адресации.

Таблица 2.5

No п/п	Операция	Формат данных		
		Байт	Слово	Двойное слово
1	Сложение	ADDB	ADD	-
2	Вычитание	SUBB	SUB	-
3	Логическое умножение	ANDB	AND	-
4	Умножение числа без знаком	MULUB		MULU
5	Умножение числа со знаком	MULB		MUL

MULB AX,BL,3[CX]	AX <- BL * MEM_BYTE (CX+3); (MULB_3op)
AND AX,BX,TABLE[CX]	AX <- BX and MEM_WORD(TABLE +CX]; (AND_3op)

Комады пересылочных операций

При выполнении пересылочной операции код, хранящийся в одном месте памяти по адресу источника, записывается в другую ячейку памяти по адресу приемника (см. табл. 2.6). Для задания адресов операндов могут использоваться любые способы адресации.

Таблица 2.6

No п/п	Операция	Формат данных		
		Байт	Слово	Двойное слово
1	Пересылка в RRAM	LDB	LD	-

2	Пересылка в RRAM с расширением без знака	LDBZE		-
3	Пересылка в RRAM с расширением со знаком	LDBSE		-
4	Пересылка из RRAM	STB	ST	-
5	Обмен	XCHB	XCH	-
6	Групповая пересылка без прерывания	-	BMOV	-
7	Групповая пересылка с возможностью прерывания	BMOVI		-
8	Пересылка в стек	PUSH		-
9	Пересылка из стека	POP		-
10	Специальная пересылка в стек	PUSHF		PUSHA
11	Специальная пересылка из стека	POPF		POPA

При выполнении операций No 2,3 данные, хранящиеся по адресу источника в формате "байт", записываются по адресу приемника в формате "слово".

Команды операций управления

Группа команд операций управления делится на две подгруппы - команды управления ходом программы и команды управления состоянием элементов и систем микроконтроллера. В первую подгруппу входят команды безусловного и условного переходов (см. табл. 2.7).

Таблица 2.7

No п/п	Операция	Мнемокод
1	Длинный безусловный переход	LJMP
2	"Короткий" безусловный переход	SJMP
3	Косвенный безусловный переход	BR
4	Табличный безусловный переход	TIJMP
5	"Длинный" безусловный переход с возвратом	LCALL
6	"Короткий" безусловный переход с возвратом	SCALL
7	Возврат	RET
8	Переход через один адрес	NOP
9	Переход через два адреса	SKIP
10	"Длинный" безусловный переход в расширенном пространстве	EJMP

11	Косвенный безусловный переход в расширенном пространстве	ERB
12	"Длинный" безусловный переход с возвратом в расширенном пространстве	ECALL

В таблице 2.8 приведены мнемокоды команд условного перехода и сами условия, при выполнении которых совершается переход.

Таблица 2.8

No п/п	Мнемокод	Условие
1	JE	Z=1
2	JNE	Z=0
3	JLT	N=1
4	JGE	N=0
5	JV	V=1
6	JNV	V=0
7	JVT	VT=1
8	JNVT	VT=0
9	JC	C=1
10	JC	C=0
11	JST	ST=1
12	JNST	ST=0
13	JGT	N=0 Z=0
14	JLE	N=1 Z=1
15	JH	C=1 Z=0
16	JNH	C=0 Z=1
17	JBS	Бит=1
18	JBC	Бит=0
19	DJNZ	Переход, пока регистр не равен нулю
20	DJNZW	Переход, пока слово не равно нулю

В командах условного перехода No 1-12 в качестве условия перехода используется наличие определенного значения одного из признаков результата в слове состояния программы. В командах No 13-16 требуется наличие определенного значения одновременно у двух признаков результата. Команды No 19,20 используются для организации цикла в программе. При каждом выполнении этой команды код числа, хранящийся соответственно в регистре или паре регистров (слово), уменьшается на единицу. Переход осуществляется, пока результат не равен нулю.

В подгруппу команд управления состоянием элементов и систем МК входят команды установки значений для некоторых разрядов регистра слова состояния программы (PSW) и команды программного сброса, программного прерывания и перевода микроконтроллера в энергосберегающие режимы (см. табл. 2.9).

Таблица 2.9

No п/п	Мнемокод	Действие
1	CLRTVT	VT=PSW.4:=0
2	SETC	C=PSW.3:=1
3	CLRC	C=PSW.3:=0
4	EI	I=PSW.1:=1
5	DI	I=PSW.1:=0
6	EPTS	PSE=PSW.2:=1
7	DPTS	PSE=PSW.2:=0
8	RST	2080 -> PC, 00H -> reg PSW
9	TRAP	(2 reg 18H)-2->2 reg 18H, (PC) 2 loc (2 reg 18H) (2 loc 2010H) -> PC
10	IDLPD#N	N=01H -> режим IDL N=02H -> режим PD

Все команды данной подгруппы, кроме IDLPD # N, однобайтовые.

Команда RST переводит МК в состояние сброса, после чего начинается выполнение программы с первой команды, хранящейся по адресу 2080H.

Команда с мнемокодом TRAP вызывает прерывание текущей программы с переходом к выполнению прерывающей подпрограммы с начальным адресом, хранящимся в ячейке 2010H. Эта команда используется при работе МК в процессе отладки совместно с внешними отладочными средствами. В рабочих программах команда TRAP не используется.

Команда IDLPD при N=01H переводит МК в режим холостого хода (Idle mode), а при N=02H - в режим пониженного энергопотребления (Powerdown mode). При других значениях константы N эта команда выполняется как команда RST.

Далее мы подробно рассмотрим структуру, функционирование и программирование основных периферийных устройств микроконтроллера 8XC196KC/KD.

ФУНКЦИОНАЛЬНОЕ ОПИСАНИЕ МИКРОКОНТРОЛЛЕРА 8XC196KC/KD

Структура кристалла

Микроконтроллер 8XC196KC/KD является 16-разрядным микропроцессорным устройством, разработанным специально для высокоскоростного управления событиями, что важно применительно к моторам, электрическим машинам,

подвескам, устройствам антиблокировочного контроля и приводам жестких дисков. Блок-схема кристалла МК приведена на рис. 3.1.

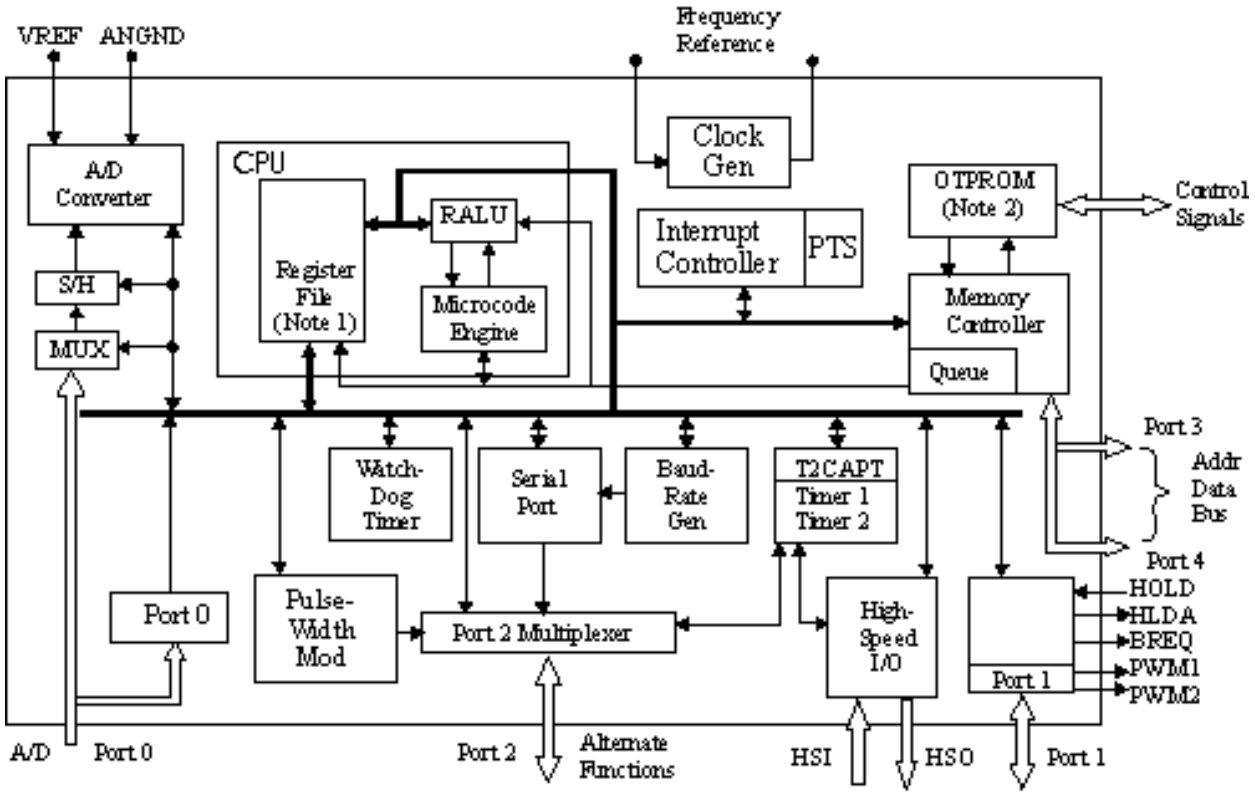


Рис. 3.1. Структурная схема микроконтроллера 8XC196KC/KD

МК имеет 16-разрядный центральный процессор (Central Processing Unit - CPU), который соединяется с контроллером прерываний (Interrupt Controller) и с контроллером памяти (Memory Controller) через шину CPU. Эта шина соединяет CPU с внутренними периферийными устройствами. Кроме того, по 8-битовой шине центрального процессорного устройства передаются байты команд из контроллера памяти в регистр команд в регистровом арифметико-логическом устройстве (RALU). Другие компоненты, входящие в схему - несколько типов памяти, 5 параллельных портов ввода/вывода, последовательный порт (Serial Port) и расположенные непосредственно на том же кристалле периферийные устройства: блок быстрого ввода/вывода (High Speed Input/Output - HSIO), аналого-цифровой преобразователь (Analog-to-Didital Converter - ADC), широтно-импульсный модулятор (Pulse Width Modulator - PWM), программируемый контроллер прерываний (Programming Interrupt Controller - PIC) и сервер периферийных транзакций (Peripheral Transaction Server - PTS), а также устройство контроля - сторожевой таймер (Watchdog Timer).

8XC196KC/KD имеет адресное пространство 64 Кбайта. В таблице 3.1 сравниваются МК 8XC196KC и 8XC196KD.

Таблица 3.1

Черты	8XC196KC	XC196KD
Адресное пространство	64 Кбайта	64 Кбайта

Внутренне ОЗУ (включая SFR)	512 байтов	1024 байта
Однократно программируемое пользователем ПЗУ (OTPROM)	16 Кбайтов	32 Кбайта
Максимальная тактовая частота	16 МГц	20 МГц

Арифметические операции могут выполняться с числами без знака в двоичном коде и с числами со знаком в дополнительном двоичном коде.

Коды команд могут иметь длину от одного (например, команды CLRC, DI, EI, NOP) до шести байтов (например, ADD, SUB, AND, CMP). Первый байт является кодом операции. В командах операций умножения и деления для чисел со знаком перед кодом операции размещается дополнительный байт-префикс. При наличии префикса максимальное число байтов в команде возрастает до семи (например, команды MUL, MULB).

В памяти МК коды команд хранятся в группах соседних ячеек памяти, при этом по младшему адресу располагается код операции, а при наличии префикса - префикс. Младший адрес в группе является адресом команды.

По числу адресов, используемых при выполнении арифметических и логических операций, команды бывают одноадресными, двухадресными и трехадесными

Ядро 8XC196KC/KD

Ядро 8XC196KC/KD (см. рис. 3.2) состоит из центрального процессорного устройства, контроллера памяти и контроллера прерываний. Центральное процессорное устройство содержит регистровое арифметико-логическое устройство и регистровый файл (Register File - RF). Микропрограммное управление CPU инструктирует, чтобы RALU выполняло операции, используя байты, слова или двойные слова из 256-байтового нижнего регистрового файла или из верхнего регистрового файла через вертикальные окна. Команды центрального процессора поступают из четырехбайтовой очереди команд (4-Byte Queue), расположенной в контроллере памяти, в регистр команд RALU (Instruction Register). Дешифратор команд (Microcode Engine) декодирует команды и затем генерирует управляющие сигналы для осуществления действий, заданных командой.

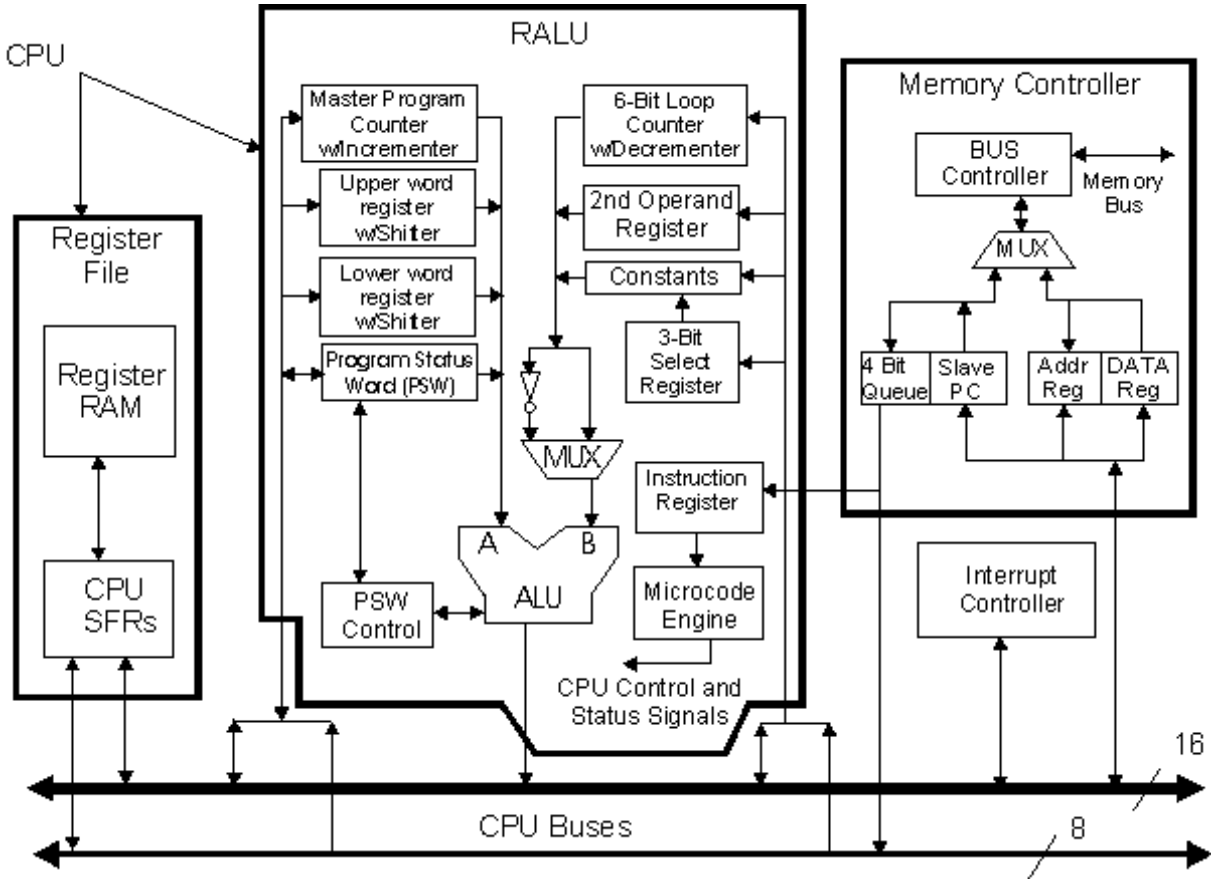


Рис. 3.2. Ядро микроконтроллера 8XC196KC/KD

Регистровый файл

Регистровый файл разделен на верхний и нижний файлы. Нижний регистровый файл содержит 24 байта для регистров специальных функций (Special Function Registers - SFR) и 232 байта для регистров общего назначения ОЗУ (Register RAM). Верхний регистровый файл содержит только регистры общего назначения ОЗУ. Регистры общего назначения могут рассматриваться как байты, слова или двойные слова. RALU обращается к верхнему и нижнему регистровым файлам по-разному. Нижний регистровый файл всегда непосредственно доступен через режим прямой адресации. Верхний регистровый файл доступен через режим прямой адресации, только когда разрешена работа с вертикальными окнами. Работа с вертикальными окнами - методика, которая отображает блоки верхнего регистрового файла в окно в нижнем регистровом файле.

Регистровое арифметико-логическое устройство

RALU содержит 17-битовое арифметико-логическое устройство (ALU), слово состояния программы (Program Status Word - PSW), главный программный счетчик (Master Program Counter - PC), регистр команд (Instruction Register), дешифратор команд (Microcode Engine), регистр констант (Constants), 3-битовый регистр выбора константы (3-Bit Select Register), счетчик циклов (6-Bit Loop Counter) и три временных регистра: старшего слова (Upper Word Register), младшего слова (Lower Word Register) и регистр второго операнда (2nd Operand Register). Все регистры, за исключением 3-битового регистра выбора, являются 16- или 17-битовыми (16 битов плюс расширение знака).

ALU ускоряет вычисления, сохраняя константы (0,1,2) в регистре констант, поэтому они легко доступны и эффективно используются при дополнении, приращении, уменьшении байтов или слов.

Слово состояния программы состоит из 2-х байтов. Младший байт - это маска прерываний INT_MASK (см. Приложение). Старший байт - это сборка булевых флагов (ST, I, PSE, C, VT, V, N, Z, с восьмого по пятнадцатый бит сответственно), которые сохраняют информацию относительно состояния программы пользователя. Только байт прерываний INT_MASK может быть доступен прямой адресации. Не существует SFR для флагов PSW.

Программный счетчик содержит адрес следующей команды и имеет встроенный инкремент адреса, который автоматически загружает адрес следующей выполняемой команды. Если переход, прерывание, вызов подпрограммы или возврат из подпрограммы изменяют последовательность адресов, ALU загружает соответствующий адрес в PC.

Регистры старшего и младшего слова используются совместно для 32-битовых команд и как временные регистры для большого количества команд. Так как они имеют собственную логику сдвига, RALU также использует их для операций, которые требуют логических сдвигов (например, нормализация, умножение и деление). Регистр младшего слова используется только тогда, когда сдвигаются двойные слова. Регистр старшего слова используется при любом типе сдвига.

Циклические сдвиги считаются 6-битовым счетчиком циклов.

В регистре второго операнда хранится второй операнд при выполнении команд с двумя операндами. Например, он хранит множитель во время выполнения умножения или делитель во время выполнения деления.

RALU связывается со всей памятью, за исключением регистрового файла, через контроллер памяти (он связывается с верхним регистровым файлом через контроллер памяти, за исключением того случая, когда используются вертикальные окна). Контроллер памяти содержит регистры адреса (Address Register) и данных (Data Register), 4-байтовую очередь команд (4-Byte Queue), подчиненный программный счетчик (Slave PC) и контроллер шины (Bus Controller). Контроллер шины управляет доступом к 4 областям памяти:
- 256-байтовому внутреннему ОЗУ (IRAM) (для кода или данных);
- 16 килобайтам ПЗУ (IROM/EPROM), расположенным на кристалле;
- регистрам спецфункций периферийных устройств на кристалле;
- внешней памяти.

Контроллер шины получает запросы доступа к памяти или из RALU, или из 4-байтовой очереди команд с упреждением; запросы очереди имеют приоритет. Очередь прозрачна для RALU и пользователя. Когда контроллер шины получает запрос из очереди, он выбирает код адреса из подчиненного PC. Это увеличивает быстродействие выполнения, так как следующий байт команды доступен немедленно и процессор может не ждать пересылки адреса из главного PC в

контроллер памяти. Если переход, прерывание, вызов подпрограммы или возврат из подпрограммы изменяют последовательность адресов, главный PC загружает новый адрес в подчиненный PC, очередь обновляется, и обработка продолжается.

Доступ к внешней памяти осуществляется через шину адреса/данных, которая может функционировать в 8- или 16-битовом режиме. Для поддержки многообразия конфигурации внешней памяти могут быть выбраны несколько сигналов управления шиной.

Регистры специальных функций

24 байта нижнего регистрового файла (00H-17H) обеспечивают доступ к регистрам специальных функций CPU (SFRs) через три горизонтальных окна (HWindows 0, 1 и 15). Через SFRs можно непосредственно управлять всеми периферийными устройствами, за исключением портов 3 и 4.

При использовании SFR в качестве базового или индексного регистра для косвенных или индексных операций следует сознавать, что содержимое SFRs не всегда предсказуемо. Внешние события могут изменять содержимое SFRs, а некоторые SFRs очищаются при чтении.

Функции большинства SFRs изменяются в зависимости от того, читают ли из них или записывают в них. По этой причине никогда не используйте SFR как операнд в командах типа чтение-модификация-запись (например, XORB AD_RESULT). Не используйте зарезервированные SFRs; запишите в них нули или оставьте их в состоянии по умолчанию. При чтении зарезервированные биты в SFRs будут возвращать неопределенные значения.

Техника кадрирования МК 8XC196KC/KD

Кадрирование - это техника для расширения суммарного количества памяти, доступного с помощью прямой регистровой адресации. Кадрирование допускает прямую регистровую адресацию некоторых участков памяти с адресом выше FFH. Когда кадрирование разрешено, посредством прямой адресации к области памяти в определенном блоке младшего регистрового файла доступно содержимое блока верхней памяти. Блок в младшем RF называется окном. Размер окна может быть 32, 64 и 128 байтов. Эти окна называют вертикальными (VWindow). МК 8XC196KC/KD имеет шестнадцать 32-байтовых, восемь 64-байтовых и четыре 128-байтовых вертикальных окна. Поскольку у МК 8XC196KD емкость ОЗУ в два раза больше, чем у 8XC196KC, то и вертикальных окон у него также в два раза больше.

Кроме вертикальных МК 8XC196KC/KD использует и горизонтальные окна, которые открывают три 24-байтовых блока (HWindows 0, 1 и 15) через самые низшие 24 байта младшего регистрового файла. Каждое окно обеспечивает чтение и запись уникальной комбинации регистров спецфункций.

Работа с горизонтальными окнами

8XC196KC/KD использует горизонтальную схему работы с окнами, которая меняет три 24-байтовых блока памяти (HWindows 0, 1 и 15) внутри младших 24 байтов нижнего регистрового файла (см. рис. 3.3). Каждое HWindow обеспечивает доступ для чтения или записи к уникальной комбинации SFRs. Одни регистры доступны как одиночный байт, другие - как слово (два байта).

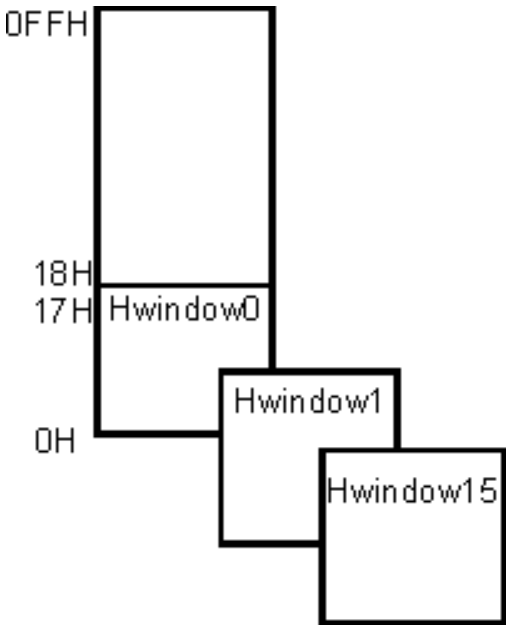


Рис. 3.3. Горизонтальные окна

Некоторые регистры, типа регистра выбора окна (WSR,14H), являются доступными во всех трех HWindows. Другие могут быть записываемы в одном HWindow, а читаемы в другом.

Для работы с регистрами через HWindow используются команды с прямой адресацией.

Выбор HWindow

Регистр выбора окна (Window Select Register - WSR,14H) обеспечивает доступ к HWindows и VWindows (см. "Работа с вертикальным окном"). Чтобы выбрать HWindow, запишите номер желаемого окна в WSR.0-WSR.3 и очистите WSR.4-WSR.6. Доступны только HWindows 0, 1 и 15. Все другие HWindows зарезервированы. Ниже приведено содержимое WSR для выбора каждого HWindow.

HWindow	Содержимое WSR
0	X000 0000B = 00H
1	X000 0001B = 01H
15	X000 1111B = 0FH

HWindow 0 - окно, устанавливаемое по умолчанию. Оно обеспечивает доступ для чтения к 19 регистрам и доступ для записи к 21 регистру. Некоторые регистры (например, INT_MASK1) могут и читаться, и записываться внутри HWindow 0. Другие (например, IOS1) могут или читаться, или записываться.

Для таких регистров выберите HWindow 15, чтобы выполнить обе функции.

	HWINDOW 0 (Read)	HWINDOW 0 (write)
17H	IOS2	PWM0_CONTROL
16H	IOS1	IOC1
15H	IOS0	IOC0
14H	WSR	WSR
13H	INT_MASK1	INT_MASK
12H	INT_PEND1	INT_PEND
11H	SP_STAT	SP_CON
10H	IOPORT2	IOPORT2
0FH	IOPORT1	IOPORT1
0EH	IOPORT0	BAUD_RATE
0DH	TIMER2 (HI)	TIMER2 (HI)
0CH	TIMER2 (LO)	TIMER2 (LO)
0BH	TIMER1 (HI)	IOC2
0AH	TIMER1 (LO)	WATCHDOG
09H	INT_PEND	INT_PEND
08H	INT_MASK	INT_MASK
07H	SBUF (RX)	SBUF (TX)
06H	HSI_STATUS	HSO_COMMAND
05H	HSI_TIME (HI)	HSO_TIME (HI)
04H	HSI_TIME (LO)	HSO_TIME (LO)
03H	AD_RESULT (HI)	HSI_MODE
02H	AD_RESULT (LO)	AD_COMMAND
01H	ZERO_REG (HI)	ZERO_REG (HI)
00H	ZERO_REG (LO)	ZERO_REG (LO)

HWindow 1 обеспечивает чтение/запись для 12 регистров. Некоторые из них также доступны и в HWindow 0, и в HWindow 15.

	HWINDOW 0 (Read/Write)
17H	PWM2_CONTROL
16H	PWM1_CONTROL
15H	Reserved
14H	WSR
13H	

12H	INT_MASK1
11H	INT_PEND1
10H	Reserved
0FH	Reserved
0EH	Reserved
0DH	Reserved
0CH	Reserved
0BH	IOC3 (LO)
0AH	Reserved
09H	Reserved
08H	INT_PEND
07H	INT_MASK
06H	PTSSRV (HI)
05H	PTSSRV (LO)
04H	PTSSEL (HI)
03H	PTSSEL (LO)
02H	AD_TIME
01H	Reserved
	ZERO_REG (HI)
00H	ZERO_REG (LO)

HWindow 15 обеспечивает доступ к тем же самым регистрам, что и HWindow 0, кроме байтов 0ch-10h. Те регистры, которые доступны только для чтения в HWindow 0, становятся доступными только для записи в HWindow 15, и наоборот.

	HWINDOW 15 (Read)	HWINDOW 15 (write)
17H		
16H	PWM0_CONTROL	IOS2
15H	IOC1	IOS1
14H	IOC0	IOS0
13H	WSR	WSR
12H	INT_MASK1	INT_MASK1
11H	INT_PEND1	INT_PEND1
10H	SP_CON	SP_STAT
0FH	Reserved	Reserved
0EH	Reserved	Reserved
0DH	Reserved	Reserved
0CH		

0BH	T2CAPTURE (HI)	T2CAPTURE (HI)
0AH	T2CAPTURE (LO)	T2CAPTURE (LO)
09H	IOC2	TIMER1(HI)
08H	WATCHDOG	TIMER1(LO)
07H	INT_PEND	INT_PEND
06H	INT_MASK	INT_MASK
05H	SBUF (TX)	SBUF (RX)
04H	HSO_COMMAND	HSI_STATUS
03H	HSO_TIME (HI)	HSI_TIME(HI)
02H	HSO_TIME (LO)	HSI_TIME(LO)
01H	HSI_MODE	AD_RESULT (HI)
	AD_COMMAND	AD_RESULT(LO)
00H	ZERO_REG (HI)	ZERO_REG (HI)
	ZERO_REG (LO)	ZERO_REG (LO)

Работа с вертикальными окнами

Для обращения к регистрам, адрес которых составляет 2 байта (0100H-01FFH), должны использоваться команды с косвенной или индексной адресацией.

Вместе с тем у микроконтроллеров 8XC196KC/KD есть возможность обращаться по адресам 0100H-01FFH, используя прямую регистровую адресацию, что позволяет сократить объем памяти для хранения команд и время их выполнения.

Обращение по адресу, представленному двумя байтами, с использованием прямой регистровой адресации осуществляется через вертикальное окно.

Работа с вертикальными окнами отображает блоки верхнего регистрового файла в верхние области нижнего регистрового файла в 32-, 64- или 128-байтовые области, известные как VWindows. 8XC196KC имеет шестнадцать 32-байтовых, восемь 64-байтовых и четыре 128-байтовых VWindows (рис. 3.4). 8XC196KD имеет в два раза больше вертикальных окон соответствующих размеров.

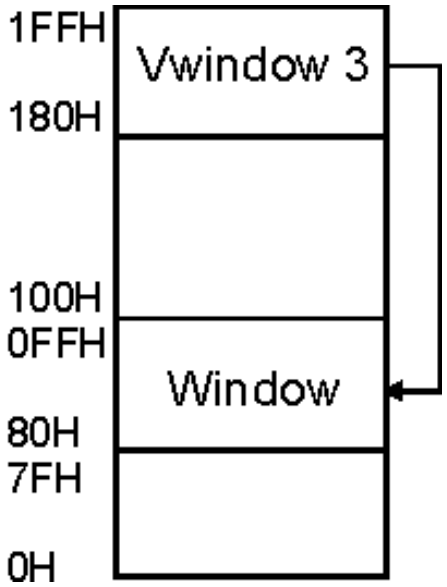


Рис. 3.4. Пример 128-байтового VWindow

Выбор VWindow

Регистр выбора окна (WSR, 14H) обеспечивает доступ к HWindows и VWindows. Установите WSR.4, WSR.5 или WSR.6, чтобы выбрать 128-, 64- или 32-байтовое VWindow соответственно. Запишите номер VWindow в младшие биты WSR. Например, чтобы выбрать вертикальное окно, показанное на рис. 3.4, загрузите 13H в WSR (LDB WSR,#13H для 8XC196KC).

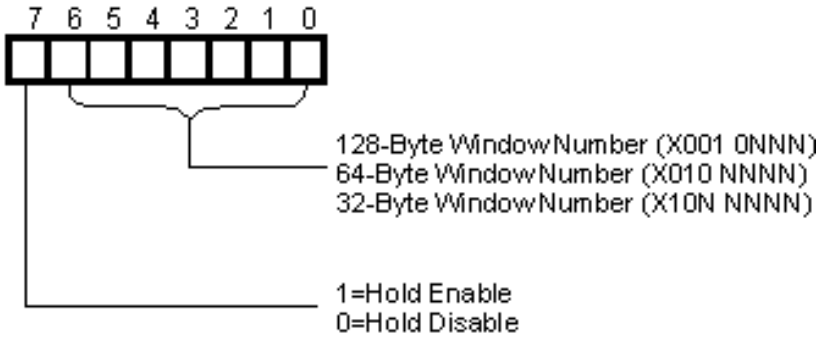
Бит WSR.7 включает (Enable) или отключает (Disable) протокол захвата шины.

Аббревиатура -WSR

Размер - 1 байт

Находится по адресу - 14h

Использовать для записи/чтения во всех окнах



Организация системы прерываний

Гибкая система управления прерываниями 8XC196KC/KD включает два основных устройства: программируемый контроллер прерываний (Program Interrupt Control - PIC) и сервер периферийных транзакций (Peripheral Transaction Server - PTS).

PIC имеет аппаратную схему приоритетов, которые могут быть изменены программой пользователя. Прерывания обслуживаются пользовательскими программами. Пользователь может выбрать наиболее важные прерывания, которые должны быть обслужены PTS вместо PIC. PTS имеет несколько микропрограмм обслуживания аппаратных прерываний с фиксированным набором действий типа: одиночные и пакетные передачи байтов или 16-разрядных слов между любой выбранной памятью, старт аналого-цифрового преобразователя, чтение FIFO очереди модуля быстрого ввода, загрузка событий в CAM модуля быстрого вывода, выполнение которых чередуется с выполнением нормальных инструкций. PTS выполняет эти задачи быстрее, чем стандартные сервисные программы обработки прерывания.

PTS может обслуживать все прерывания, за исключением следующих: немаскируемое прерывание (NMI), ловушка (Trap) и несуществующий код операции. PTS циклы имеют более высокий приоритет, чем стандартные прерывания, и могут временно приостанавливать программы обработки прерывания.

Параметры синхронизации

Тактовый генератор делит на два опорную частоту и вырабатывает два внутренних временных сигнала PH1 и PH2. Активный уровень сигналов высокий. Переход с низкого на высокий уровень сигналов PH1 и PH2 генерируется сигналом CLKOUT (см. рис. 3.5).

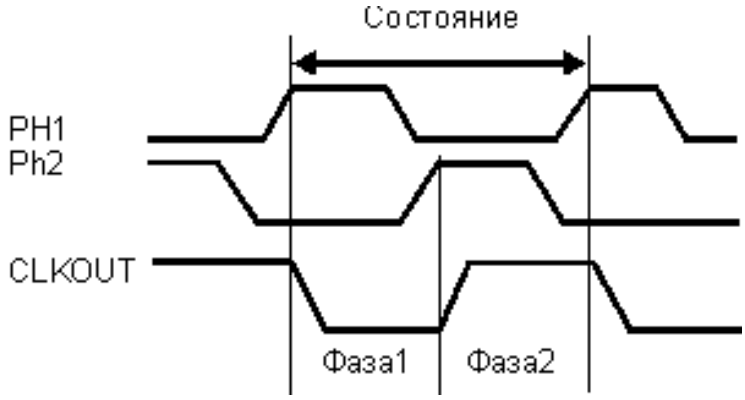


Рис. 3.5. Фазы внутренней синхронизации

Скомбинированный период сигналов PH1 и PH2 определяет основную временную единицу, известную как время состояния State Time или временное состояние. При максимальной частоте МК 8XC196KD 20 МГц одно временное состояние равняется 100 нс. При максимальной частоте МК 8XC196KC 16 МГц одно временное состояние - 125 нс. Поскольку МК 8XC196KC/KD могут работать

на многих частотах Fosc, все временные интервалы приводятся в единицах временных состояний.

Заключение

Мировой рынок однокристальных микроЭВМ постоянно расширяется и совершенствуется. Признанные фавориты в производстве микроконтроллеров - фирмы Intel, Motorola, Texas Instruments, Zilog и другие наряду с новыми семействами 16-, 32-разрядных, RISC процессоров продолжают развивать линию 8-разрядных МК, которые находят себе все новые и новые применения в современной индустрии.

Сегодня фирме Texas Instruments требуется 8-12 недель для создания МК.

В это же время наблюдается масштабный скачок в развитии нечетких контроллеров. Простота и дешевизна разработки нечетких систем управления способствует повсеместному внедрению этой технологии: в технических областях это автоматическое управление высокого уровня, интеллектуальные роботы, системы поддержания целостности баз данных и системы обеспечения безопасности, распознавание изображений и речи, автоматическое проектирование, поиск информации; в медицине - диагностика, искусственные встроенные органы; в сфере бизнеса - помощь в принятии экономических решений, различного рода управление и планирование.

К концу 80-х годов бесспорным лидером в производстве нечетких контроллеров, выполненных по технологии СБИС, стала Япония. Чтобы не терять драгоценного времени на разработку собственных нечетких устройств, фирма Intel предоставила пользователям систему fuzzy TECH - новое средство разработки приложений на базе 8- и 16-разрядных цифровых МК с использованием технологии нечеткости.

Таким образом, в настоящее время Вам предлагается широкий выбор конфигураций однокристальных микроЭВМ с разными типами и размерами памяти, с принципиально разным программным обеспечением, с различными таймерами и другими встроенными устройствами, обеспечивающими нужную функциональность системы.

Если Ваша деятельность в какой-либо степени касается информационных технологий, то знакомство с классом современных однокристальных микроЭВМ ведущих зарубежных фирм будет, безусловно, полезным и перспективным.

Автор благодарит коллектив центра "Микропроцессорные технологии" Санкт-Петербургского ГЭТУ им. Ульянова (Ленина) за предоставление технического перевода руководства User's Manuals и дружескую поддержку в процессе подготовки данного пособия, а также инженера И. М. Некрылову за помощь в оформлении текста.

Список литературы

1. Ангуло Хосе М. Микропроцессоры: Архитектура, программирование и проектирование систем. Тбилиси: Ганатлеба, 1989. 548 с.

2. Басманов А.С., Широков Ю.Ф. Микропроцессоры и однокристальные микроЭВМ: Номенклатура и функциональные возможности / Под ред. В. Г. Домрачева. М.: Энергоатомиздат, 1988. 128 с.

3. Нерода В.Я., Торбинский В.Э., Шлыков Е.Л. Однокристальные мироЭВМ MCS-51. М., 1995.

4. Григорьев В.Л. Программирование однокристальных микропроцессоров. М.: Энер-гоатомиздат, 1987.

5. Белоус А.И., Ковалевский В.К. Однокристальные микроЭВМ: Современное состояние и перспективы развития. Минск, 1991. 83 с.

6. Гребнев В.В. Однокристальные микроЭВМ (микроконтроллеры) семейства MCS-96. Псков: Псковская коммерческая палата, 1995.