

USB-МИКРОКОНТРОЛЛЕР ДЛЯ ИЗМЕРИТЕЛЬНОГО ОБОРУДОВАНИЯ: "СИСТЕМА НА КРИСТАЛЛЕ"

Для того чтобы соответствовать современным требованиям, измерительная аппаратура становится все более многофункциональной, а значит, и более сложной. Это в свою очередь повышает требования к применяемой элементной базе. Разработчики идут по одному из двух путей: берут за основу мощный процессор, а функциональность реализуют программно, или же применяют специализированные СБИС типа "системы на кристалле". Последний вариант выглядит предпочтительнее. Системы на кристалле объединяют в себе микропроцессорное ядро, аналоговые элементы и блоки аппаратной поддержки различных функций. Следовательно, уменьшаются габариты и энергопотребление изделий, число используемых компонентов (а значит и себестоимость производства). В данной статье рассмотрены результаты подобной реализации USB-микроконтроллера.

ЗАО "ПКК Миландр" – отечественный дизайн-центр, который специализируется на разработке и производстве заказных ИС. Компания ведет проектирование цифровых, аналоговых, цифроаналоговых микросхем и схем памяти различного применения. Одним из основных направлений является проектирование "систем на кристалле" на базе микропроцессорных ядер. Например, созданы микроконтроллеры 1886BE1 и 1886BE2 – функциональные аналоги микроконтроллера PIC17C756 от Microchip. Использованное в них 8-битное микропроцессорное ядро стало основой нового микроконтроллера 1886BE4U (рис.1). Он предназначен для соединения различных интерфейсов и интеграции оборудования в измерительных системах, при этом обладая производи-

М. Какоулин,
С. Шумилин,
А. Алексеев

тельностью, достаточной для сбора и первичной обработки информации. В настоящее время произведены работающие экспериментальные образцы кристаллов.

1886BE4U содержит USB-контроллер, интерфейсы SPI и USART, 256 байт энергонезависимой памяти данных (табл.1). Основное его применение – высокоскоростная передача данных в различной аппаратуре. USB-контроллер может оперировать в режимах Full Speed (до 12 Мбит/с) и Low Speed (до 1,5 Мбит/с).

При передаче данных по протоколу USB хост взаимодействует с адресами так называемых конечных точек (endpoints) пользовательского устройства. Каждая из них имеет собственный буфер очереди (FIFO) и может обрабатывать данные нескольких типов: объемные (Bulk), данные прерываний (Interrupt), и изохронные (Isochronous). 1886BE4U содержит четыре конечных точки с глубиной FIFO 64 байт, а также 128 байт памяти для хранения пользовательского дескриптора USB-устройства. В нем описаны характеристики устройства, в том числе наименование производителя (Vendor ID) и продукта (Product ID).

Основной задачей при разработке USB-контроллера было обеспечить максимальную скорость передачи данных, но минимально задействовать микропроцессорное ядро на обслуживание интерфейса. Также USB-блок должен быть удобным в использовании и не требующим от разработчиков аппаратуры глубокого знания самого интерфейса. Поэтому работа контроллера максимально автоматизирована (рис.2). Он автоматически принимает и разбирает пакеты от хоста, проверяет их целостность, при необходимости подтверждает успешность приема пакета. При передаче данных хосту USB-контроллер формирует пакеты и проверяет подтверждение их доставки. Таким образом, обмен данными между микроконтроллером и хостом не затрагивает микропроцессорное ядро, что значительно упрощает работу с интерфейсом.

При тестировании образцов микросхем были достигнуты результаты, близкие к максимально возможным. При частоте микроконтроллера 32 МГц (производительность – 8 mips),

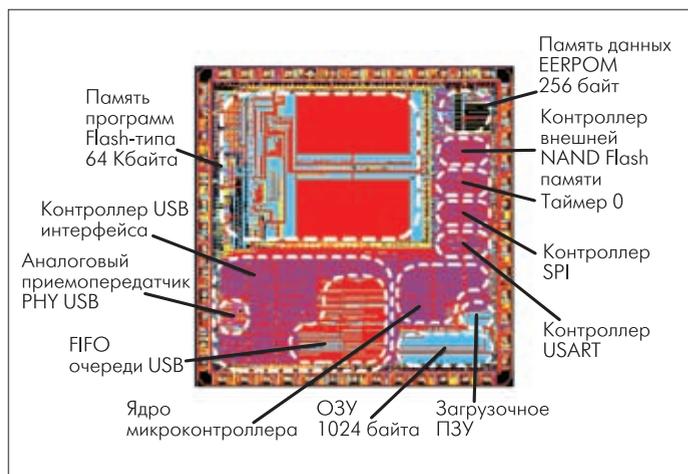


Рис.1. Схема кристалла 1886BE4Y

USB-интерфейс работал в режиме Full Speed (12 Мбит/с) с максимальным размером пакета в 64 байта при работе с конечными точками типа Bulk. При этом на обслуживание USB-блока тратилось не более 20% процессорного времени, которое практически полностью занимали операции чтения/записи из буферов очередей конечных точек. Взаимодействие ядра с FIFO может происходить одновременно с приемом или передачей пакетов. Состояние FIFO можно определить с помощью значений флагов Empty и Full в соответствующих регистрах состояния контроллера. Очевидно, что постоянный опрос флагов снижает скорость работы с буферами (рис. 3).

Вместо этого можно считать число слов в очереди из регистра соответствующей конечной точки и произвести безусловные считывания заданного числа байт из буфера (рис. 4).

Это значительно улучшает производительность микроконтроллера (табл. 2). Однако, по мере усложнения обработки,

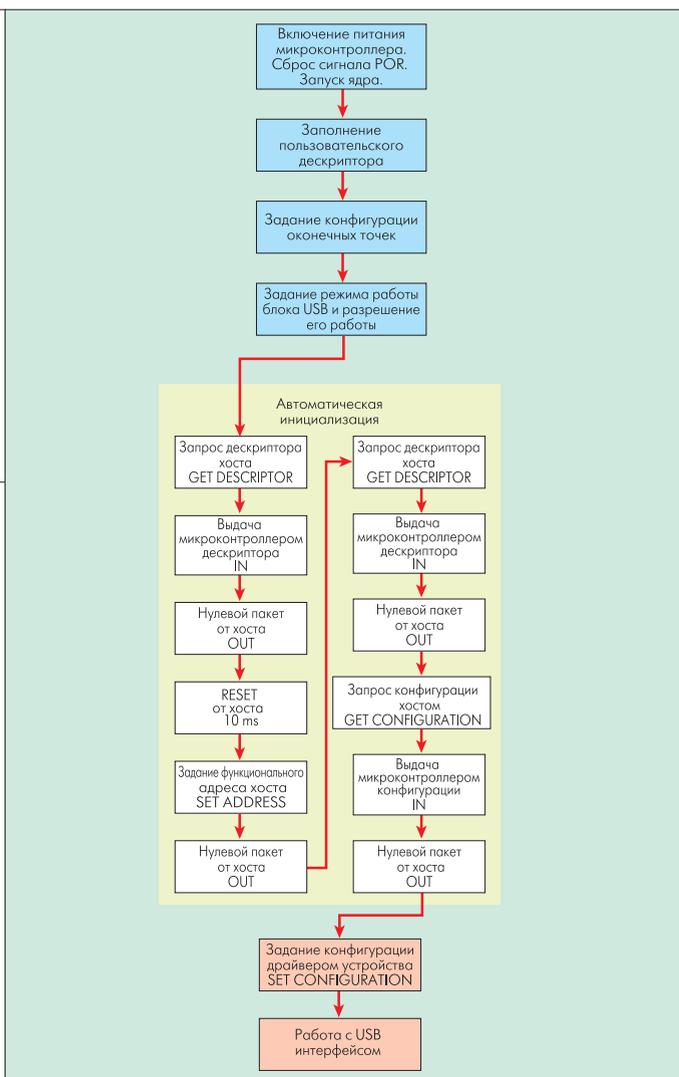


Рис.2. Процесс автоматической инициализации USB интерфейса

Таблица 1. Сравнительные характеристики микроконтроллеров ЗАО "ПКК Миландр"

Параметр	1886BE2	1886BE4Y
Микропроцессорное ядро	58 инструкций, (функциональный аналог Microchip PIC17C756)	58 инструкций (совместимое с Microchip PIC17C756)
Flash-память программ	64 Кбайт	64 Кбайт
Оперативная память данных	1024 байт	1024 байт
Энергонезависимая память данных	нет	256 байт
Тактовая частота	33 МГц	33 МГц
Напряжение питания ядра	4,5 – 5,5 В	4,5 – 5,5 В
Напряжение питания портов ввода-вывода	4,5 – 5,5 В	3,0 – 5,5 В
Интерфейс USB	нет	USB 1.1
Интерфейс USART	2	1
Интерфейс SPI	1	1
Интерфейс I2C	1	нет
АЦП	12 каналов, 10 разрядов	нет
Таймеры	4	1
Интерфейс NAND Flash	нет	1
Встроенный регулятор напряжения	нет	На 3,3В
Рабочий диапазон температур	-60 ... 85°C	-60 ... 85°C
Тип корпуса	H18.64 (64 вывода)	H16.48 или LQFP 64

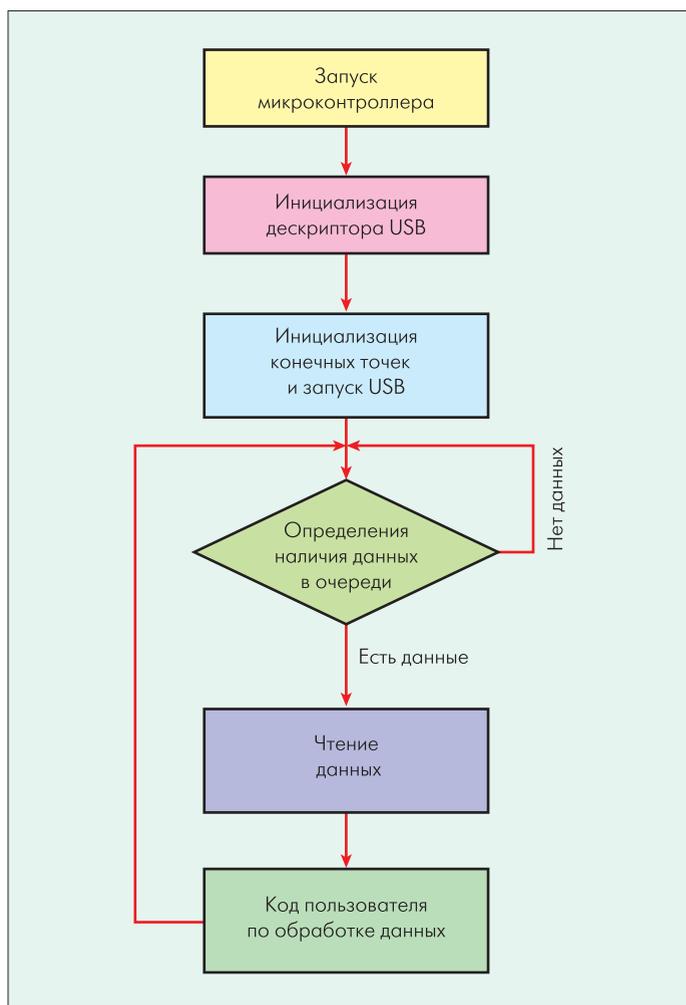


Рис.3. Блок-схема алгоритма передачи данных типа Bulk от хоста к микроконтроллеру

эффективность обоих алгоритмов становится примерно одинаковой (рис.5).

Результаты, достигнутые при передаче данных от микроконтроллера к хосту, аналогичны полученным при взаимодей-

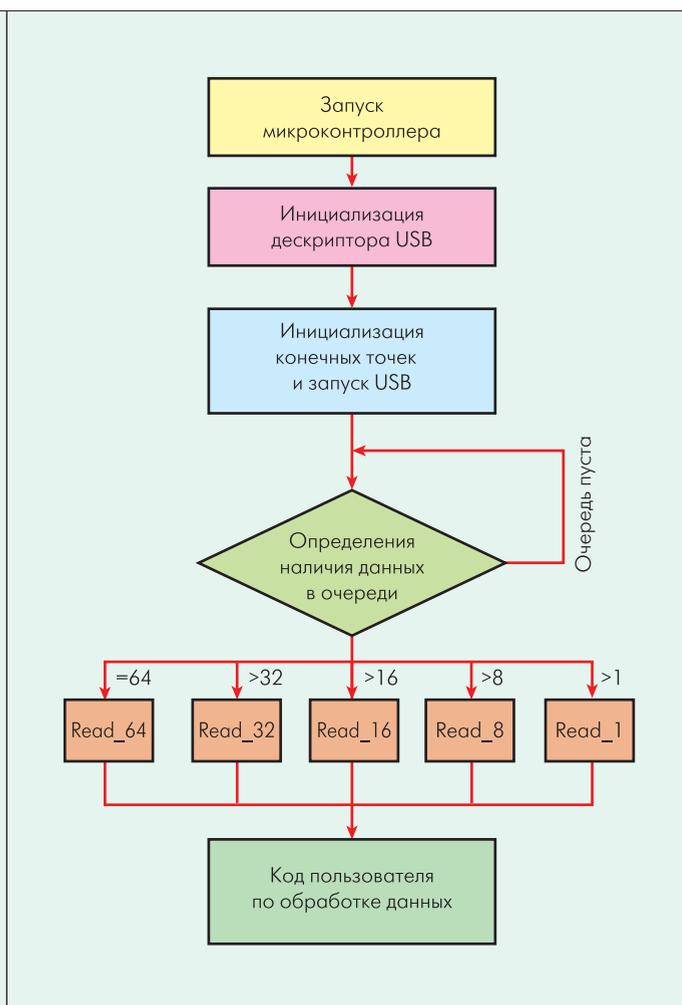


Рис.4. Блок-схема усложненного алгоритма передаче данных типа Bulk от хоста к микроконтроллеру

твии в обратном направлении (табл.3): обычный алгоритм, основанный на значении флагов, требует больше команд на передачу одного байта. Усовершенствованный метод использует функции безусловной записи 64, 32, 16, 8 и 1 байта.

Таблица 2. Характеристики скорости передаче данных типа Bulk от хоста к микроконтроллеру

Число команд на один получаемый байт	Средняя скорость при работе по флагам, Кбайт/с	Средняя скорость при работе по счетчику, Кбайт/с	Объем программного кода инициализации USB-устройства	Объем программного кода чтения одного байта
0	917	1048	350	При работе по флагам: 5. Для функций записи: 64 байт – 1.094 32 байт – 1.188 16 байт – 1.375 8 байт – 1.75 1 байт – 7
1	851	1048		
5	790	1048		
10	530	655		
15	327	458		
25	262	262		
35	196	196		
50	131	131		

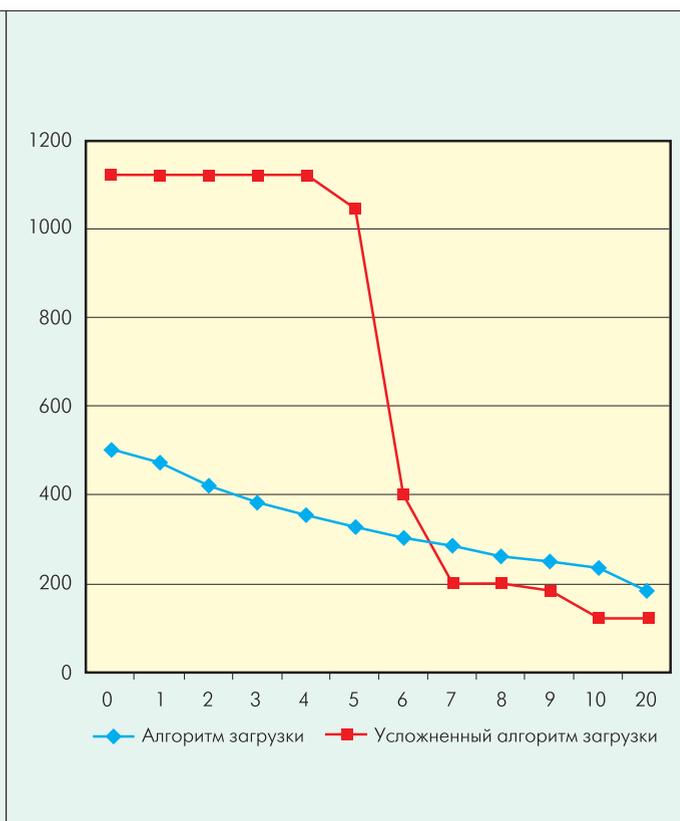
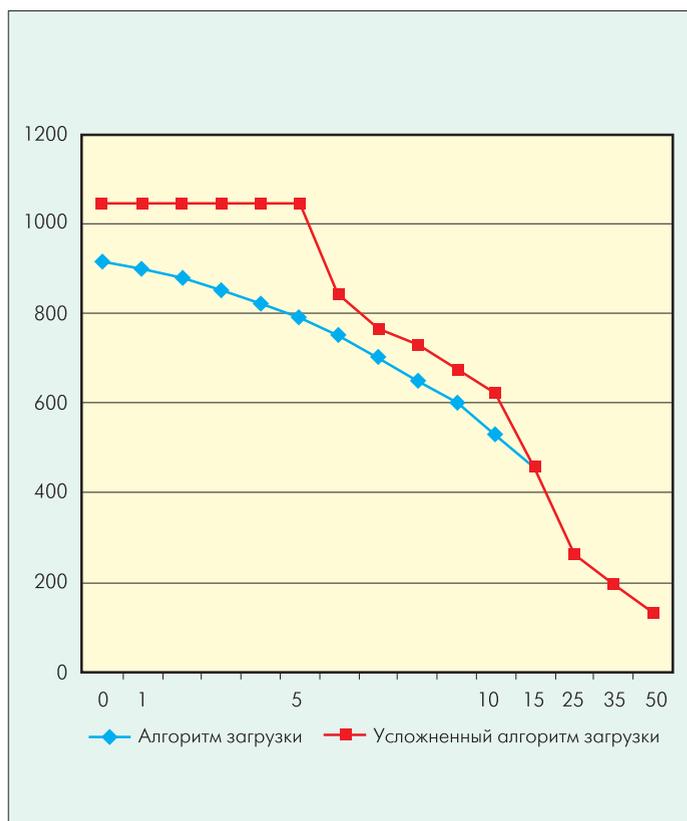


Рис.5. Диаграмма производительности при передаче данных типа Bulk от хоста к микроконтроллеру (по вертикальной оси отображается скорость передачи, по горизонтальной – число команд на обработку одного принятого байта)

Рис.6. Диаграмма производительности при передаче данных типа Bulk от микроконтроллера к хосту

При увеличении сложности обработки данных (более 5–6 команд ядра на один принимаемый или отправляемый байт) происходит резкое падение скорости обмена по шине USB (рис.6). Это обусловлено тем, что длительная обработка данных микроконтроллером приводит к уменьшению числа байт данных, передаваемых в одном пакете по шине USB, и, следовательно, увеличению доли служебной информации.

В настоящее время ЗАО "ПКК Миландр" продолжает работу по расширению серии микроконтроллеров 1886. В частности, разрабатываются новые интерфейсные блоки: CAN2.0В, Ethernet 10BaseT, LIN. Ведется разработка маломощного микроконтроллера с потреблением менее 0.5 мА на 1 МГц. Планируются создание микроконтроллера со встроенным 12-разрядным АЦП и др.

Таблица 3. Характеристики скорости передачи данных типа Bulk от микроконтроллера к хосту

Число команд на один отправляемый байт	Средняя скорость при работе по флагам, Кбайт/с	Средняя скорость при работе по счетчику, Кбайт/с	Объем программного кода инициализации USB-устройства	Объем программного кода чтения одного байта
0	500	1120	350	При работе по флагам: 5. Для функций записи: 64 байт – 1,094 32 байт – 1,188 16 байт – 1,375 8 байт – 1,75 1 байт – 7
1	470	1120		
2	417	1120		
3	380	1120		
4	351	1120		
5	324	1048		
6	300	400		
7	284	200		
8	257	200		
9	249	180		
10	235	120		
20	180	120		