

BITFURY CLARKE

B1549493-010

Техническое описание

 **BITFURY**

КРАТКАЯ ИНФОРМАЦИЯ

Bitfury Clarke — это ASIC, в котором используется двойной алгоритм SHA256. Чип разработан для биткоин-майнинга и имеет корпус оптимизированного размера, подходящий для установки на компактные печатные платы. Чип обладает указанными ниже характеристиками.

Производительность всего 55 мДж/ГХ

- Скорость выполнения вычислений до 120 ГХ/с
- 8154 ядра кольцевого хеша
- Простой двухпроводной синхронный последовательный интерфейс управления со скоростью до 8 Мбит/с
- Два буфера задач для самой высокой производительности
- Полностью интегрированный управляемый генератор тактовых импульсов
- Интегрированная схема со сбросом при включении питания
- Работает при напряжении от 0,3 В
- Компактный бессвинцовый корпус FCLGA 35L или FCLGA 4L размером 6 × 6 мм

Рабочие характеристики чипа Bitfury Clarke

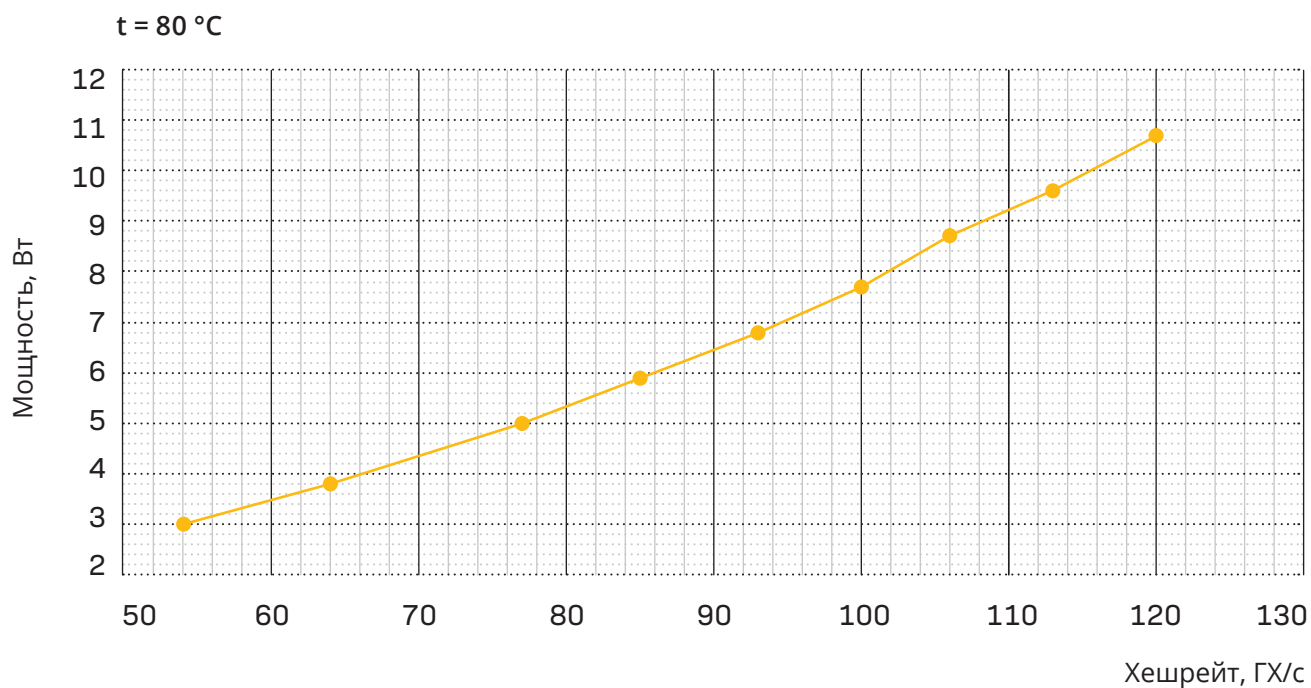
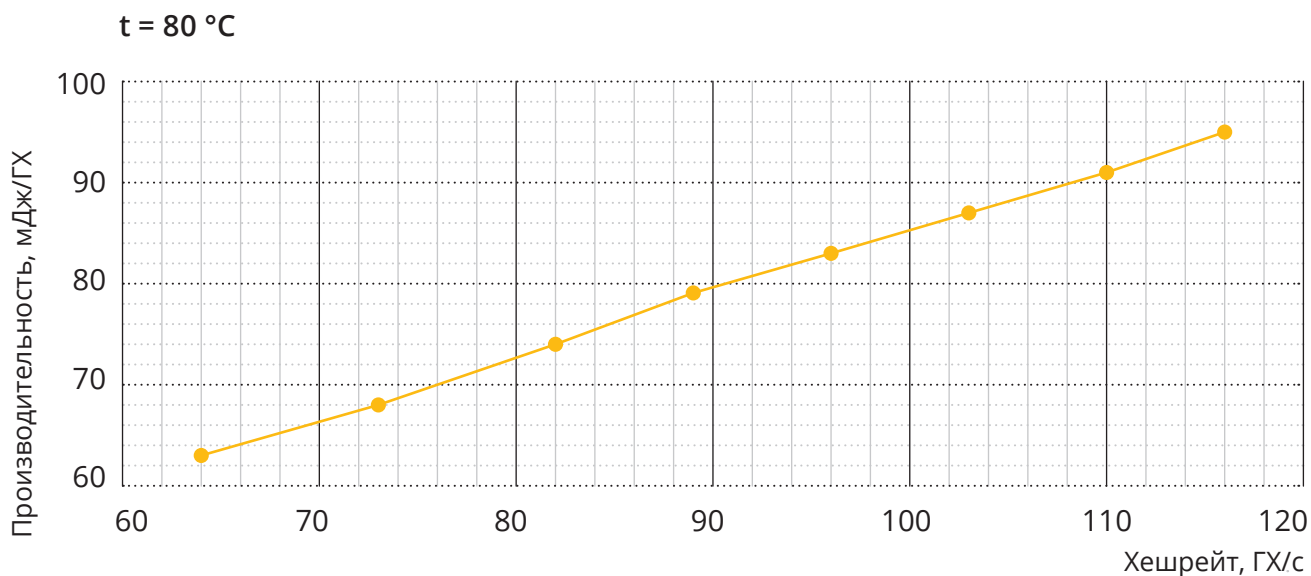
В перечисленных ниже характеристиках указана средняя производительность чипов. Производительность отдельных чипов или их партий может отличаться.

Представленные характеристики производительности достигнуты в реальной рабочей среде на хеш-плате эталонной конструкции компании Bitfury, используемой в майнерах (серверах) Bitfury B8 эталонной конструкции с воздушным охлаждением.

При использовании чипов на хеш-платах и/или в майнерах (серверах) других конструкций компания Bitfury не гарантирует указанные ниже характеристики и не несет ответственность за несоответствие чипов этим характеристикам.

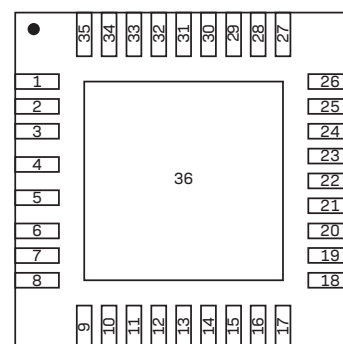
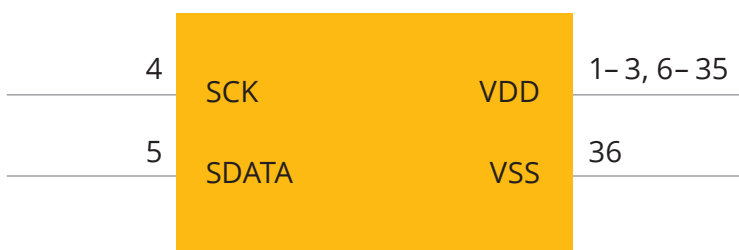
Средняя производительность и энергоэффективность чипов зависит от мощности (в ваттах) и диапазона рабочих температур.

Испытания проводились при температуре чипов около 80 °C.



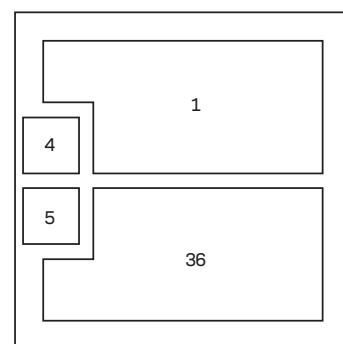
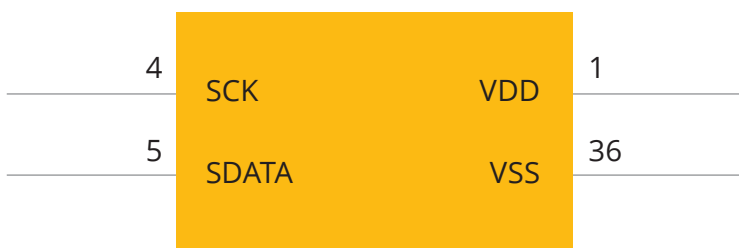
1. Расположение контактов устройства и описание сигналов

1.1. КОРПУС FCLGA 35L



Вид сверху

1.2. КОРПУС FCLGA 4L



Вид сверху

1.3. ОПИСАНИЕ КОНТАКТОВ

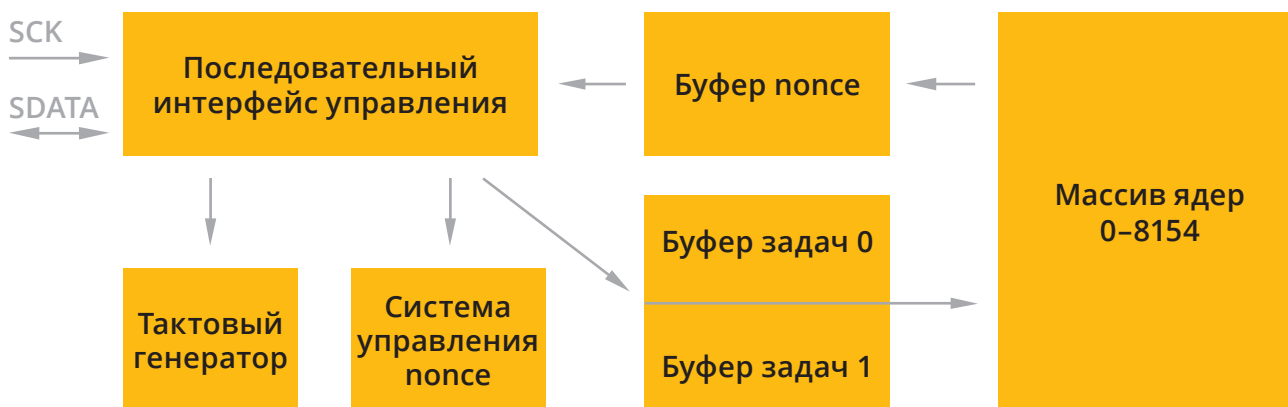
Табл. 1.1. Контакты питания, земли и функций

Номер контакта	Название	Тип	Описание
4	SCK	ВХОД	Вход линии синхронизации последовательной шины
5	SDATA	ВХОД/ВЫХОД	Вход/выход данных последовательной шины
1-3, 6-35	VDD	ПИТАНИЕ	Вход питания

2. Блок-схема

Блок последовательного интерфейса управления выполняет весь обмен данными с внешним контроллером через последовательную шину. Тактовый генератор создает тактовые импульсы, необходимые для работы чипа, и его параметры можно изменить с помощью команды `set clock` («Задать параметры тактового генератора»). Блок управления значениями `nonce` управляет вычислениями с учетом маски. Маску можно запрограммировать с помощью команд `set mask` («Задать маску») или `task write` («Запись задачи»). Маска размещается в буфере задач. Поэтому необходимо задать маски для обоих буферов задач. Буферы задач 0 и 1 представляют собой буферы на 20 значений типа `dword`, используемые для текущих и будущих вычислений. Для заполнения буфера задач данными используется команда `task write` («Запись задачи»). Буфер `nonce` представляет собой циклический массив на 12 значений типа `dword`. Он содержит значения `nonce` и маркеры переключения задач. Содержимое буфера `nonce` можно получить из чипа с помощью команды `read nonces` («Прочитать значения `nonce`»). Массив ядер представляет собой массив из 8154 ядер, в которых используется двойной алгоритм SHA256.

Рис. 2.1. Блок-схема



3. Описание функций

Для быстрого переключения между задачами чип оснащен двумя буферами задач. Один буфер задач используется для вычислений, связанных с алгоритмом SHA256, а второй буфер можно заполнить командой task write («Запись задачи»). По завершении текущей вычислительной задачи чип выбирает следующий буфер задач и начинает новый цикл вычислений.

Переключение задач может происходить в конце цикла вычислений либо выполняться принудительно с помощью команды force task switch («Принудительно переключить задачу»). Буфер попсе представляет собой кольцевой буфер на 12 значений типа dword. Во время цикла вычислений чип может найти решение (попсе) и сохранить его в буфер попсе. В конце цикла вычислений в буфер попсе записывается маркер конца задачи. Для маркера конца задачи используется 16 различных шаблонов (0x0FFFFFFC, 0x1FFFFFFC и т. д. до 0xFFFFFCC). При каждом переключении задачи происходит увеличение значения счетчика в старшем значащем ниббле маркера конца задачи на 1. Содержимое буфера попсе можно считать из чипа с помощью команды read popses («Прочитать значения попсе»).

4. Последовательный интерфейс управления

4.1. ОСНОВНЫЕ ХАРАКТЕРИСТИКИ

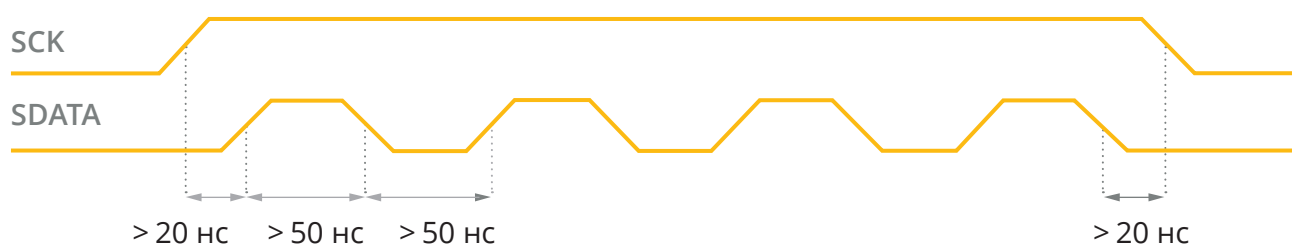
Последовательный интерфейс управления похож на интерфейс SPI с контактами MOSI и MISO, объединенными с линией SDATA. Из-за отсутствия контакта CS необходимо использовать особую последовательность для сброса. Чип делает выборку всех разрядов данных по переднему фронту сигнала SCK. Изменение состояния линии SDATA, управляемой чипом, происходит по заднему фронту сигнала SCK. В последовательности разрядов первым является старший значащий разряд. В последовательности байтов в значении типа dword первым является старший значащий байт. Уровни напряжения: 0 — VSS, 1 — VDD.

4.2. ПОСЛЕДОВАТЕЛЬНОСТЬ СБРОСА

Специальная последовательность сброса служит только для сброса коммуникационного блока чипа, сбрасывая до первоначального состояния машину состояний этого блока. Эту последовательность необходимо использовать непосредственно перед отправкой каждой команды на чип.

В начале последовательности сброса на линии SDATA присутствует значение 0, а линия SDA должна перейти из состояния 0 в состояние 1. Вскоре на линии SDATA должны появиться 4 импульса. И, наконец, линия SCK должна перейти в состояние 0. Количество импульсов на линии SDATA может быть больше 4, но не меньше этого количества.

Рис. 4.1. Последовательность сброса



4.3. СТАНДАРТНЫЙ ШАБЛОН КОМАНДЫ

Все команды должны соответствовать одному и тому же шаблону. Первый байт данных в шаблоне команды представляет собой код команды. Второй байт данных — длина в байтах минус 1. Если необходимо использовать команду без данных, следует задать нулевую длину. В результате в поле данных появится один фиктивный байт данных. Третий элемент — поле данных.

Для различных команд это поле имеет разную длину. Самая длинная команда, разрешенная в поле данных — `task write` («Запись задачи»). Она имеет длину 80 байтов. Самая короткая команда — `force task switch` («Принудительно переключить задачу») длиной в 1 фиктивный байт.

Контроллер должен отправлять код, длину и поля данных команды на чип. После отправки последнего байта данных на чип необходимо отключить линию SDATA драйвера на стороне контроллера. По заднему фронту младшего значащего разряда последнего байта данных на линии SDATA чип начнет выполнять команду.

Четвертый элемент — байт состояния чипа. Пятый элемент — контрольная сумма команды. Шестой элемент — данные из буфера `nonce`. Длина буфера `nonce` составляет 48 байтов, поэтому это поле имеет фиксированную длину. Седьмое поле — контрольная сумма значения `nonce`. Помимо команды `read nonces` («Прочитать значения `nonce`»), для всех остальных команд можно использовать только 5 первых полей: от кода команды до контрольной суммы команды.

Рис. 4.2. Стандартный шаблон команды

Сброс	Код команды	Длина	Данные	Состояние	Контрольная сумма команды	Значения попсе	Контрольная сумма значения попсе
	1 байт	1 байт	1-80 байтов	1 байт	1 байт	48 байтов	1 байт

4.4. СОСТОЯНИЕ

Байт состояния имеет три поля. Старший значащий ниббл содержит старший значащий разряд счетчика попсе. Эту информацию можно использовать для определения времени, необходимого на завершение текущей задачи. Разряды 3 и 2 имеют одинаковое значение и содержат номер принимающего буфера задач в начале текущей команды на последовательной шине. Разряды 1 и 0 содержат номер принимающего буфера задач на момент их приема на последовательной шине. Обычно разряды 1 и 0 имеют одинаковое значение. Они могут иметь различные значения, только если между событиями отправки разрядов 1 и 0 происходит переключение задач. Различие между разрядами 2 и 1 позволяет обнаружить отправку разделенной команды task write («Запись задачи»). Если разряды 2 и 1 имеют различные значения, это означает, что выполнение этой команды начато в одном принимающем буфере задач, а закончено в другом. Таким образом, для всех команд, за исключением команды task write («Запись задачи»), эта ситуация не критическая. Для команды task write («Запись задачи») такая ситуация означает, что команда выполнена неуспешно.

Рис. 4.3. Байт состояния

Старший значащий разряд счетчика попсе	Начало буфера		Конец буфера		
7	4	3	2	1	0

Два последних разряда байта состояния обладают одной интересной особенностью. Когда одним из двух младших значащих разрядов данных на линии SDATA управляет непосредственно схема переключения буферов, можно «заморозить» линию SCK в состоянии 0 между разрядами 1 и 0 и заставить изменить состояние линии SDATA во время переключения

буфера. Если контроллеру необходимо считать только состояние в отсутствие какой-либо активной команды, для кода команды следует задать значение 0.

4.5. КОНТРОЛЬНАЯ СУММА

Чип отправляет контрольные суммы команды и значения nonce в контроллер. Обе контрольные суммы представляют собой арифметическую сумму байтов, начиная с нулевого байта. Контрольная сумма команды состоит из суммы всех байтов от кода команды до последнего байта данных включительно. Контрольная сумма значения nonce состоит из суммы всех байтов от первого до последнего байта nonce включительно.

Пример: контрольная сумма команды 04 03 03 8c 18 00 равна AE.

4.6. МУЛЬТИКОМАНДЫ

Команда в байте команды помечается как соответствующий разряд, которому присвоено значение 1. Таким образом, возможное количество команд — 8. Но блок последовательного интерфейса управления чипа поддерживает только 6 команд. Два старших значащих разряда в байте команды будут проигнорированы. Такой подход позволяет задавать несколько разрядов в байте команды. В этом случае чип будет выполнять команды в определенном порядке.

Вот порядок выполнения мультикоманды:



Task write
(«Запись задачи»)



Force task switch
(«Принудительно переключить задачу»)



Read nonces
(«Прочитать значения nonce»)

Все остальные команды не следует использовать в мультикоманде. Поведение чипа при включении других команд в мультикоманду не оговорено.

5. Набор команд

5.1. TASK WRITE («ЗАПИСЬ ЗАДАЧИ»)

Команда task write («Запись задачи») используется для отправки нового задания на чип. Шаблон команды показан ниже.

Сброс	0x01	0x4F	Данные	Состояние	Контрольная сумма команды
	1 байт	1 байт	1-80 байтов	1 байт	1 байт

Рис. 5.1. Шаблон команды task write («Запись задачи»)

Код команды task write («Запись задачи»): 0x01. Элемент может иметь длину от 0 до 0x4F (79 в десятичной системе счисления). Если необходимо обновить всю задачу, следует указать длину 0x4F. Последовательность данных типа dword показана ниже в табл. 5.1.

Табл. 5.1. Последовательность данных команды task write («Запись задачи»)

Номер последовательности	Название	Описание
1	MS0_A	Начальное значение слова SHA-2 ^ 0xAAAAAAAA
2	MS0_B	
3	MS0_C	
4	MS0_D	
5	MS0_E	
6	MS0_F	
7	MS0_G	
8	MS0_H	
9	MS3_H	Значение промежуточного состояния после 3 раундов ^ 0xAAAAAAAA
10	MS3_G	
11	MS3_F	
12	MS3_E	Значение слова SHA-2W[0] ^ 0xAAAAAAAA
13	W0	
14	W1	
15	W2	Значение слова SHA-2W[2] ^ 0xAAAAAAAA
16	MS3_D	Значение промежуточного состояния после 3 раундов ^ 0xAAAAAAAA
17	MS3_C	
18	MS3_B	
19	MS3_A	
20	MASK	Значение Mask («Маска»), см. описание команды set mask («Задать маску»)

5.2. КОМАНДА FORCE TASK SWITCH («ПРИНУДИТЕЛЬНО ПЕРЕКЛЮЧИТЬ ЗАДАЧУ»)

Команда force task switch («Принудительно переключить задачу») используется для изменения текущего буфера задачи. Ее можно использовать, если не нужно завершать вычисления для текущего буфера. Шаблон команды показан на рис. 5.2.

Рис. 5.2. Шаблон команды force task switch («Принудительно переключить задачу»)

Сброс	0x02	0x00	0x00	Состояние	0x02
	1 байт	1 байт	1 байт	1 байт	1 байт

5.3. КОМАНДА READ NONCES («ПРОЧИТАТЬ ЗНАЧЕНИЯ NONCE»)

Команда read nonces («Прочитать значения nonce») используется для получения последних данных nonce из чипа. Шаблон команды показан ниже на рис. 5.3.

Рис. 5.3. Шаблон команды read nonces («Прочитать значения nonce»)

Сброс	0x04	0x00	0x00	Состояние	0x04	Значения nonce	Контрольная сумма значения nonce
	1 байт	1 байт	1 байт	1 байт	1 байт	1 байт	1 байт

Поле значений nonce содержит данные nonce, внесенные чипом. Это поле содержит 12 значений типа dword. Чип заполняет буфер nonce от последнего значения типа dword (12) до первого (1). Буфер nonce — циклический, и после внесения первого значения типа dword (1) следующим будет внесено последнее значение dword (12). Кроме того, в буфере nonce содержатся маркеры окончания задачи. Маркер окончания задачи записывается в буфер nonce в конце вычисления текущей задачи. С его помощью можно узнать, что задача завершена и больше не ожидается новых значений nonce. Обратите внимание, что для значений nonce необходимо выполнить операцию XOR (исключающее ИЛИ) с константой 0хАААААААА.

5.4. SET CLOCK («ЗАДАТЬ ПА НЕРАТОРА»)

Команда set clock («Задать параметры тактового генератора») используется для задания нового управляющего значения для тактового генератора. Шаблон команды показан ниже на рис. 5.4. Элемент значения Clock («Тактовый сигнал») содержит несколько полей и показан в табл. 5.2.

Рис. 5.4. Шаблон команды set clock («Задать параметры тактового генератора»)

Сброс	0x08	0x03	Значение Clock («Тактовый сигнал»)	Состояние	Контрольная сумма значения поспе
	1 байт	1 байт	4 байт	1 байт	1 байт

Табл. 5.2. Значение Clock («Тактовый сигнал»)

Разряды	Описание
31-20	Константа Magic = 0x038
19	Предварительный делитель частоты отключен: 1, включен: 0
18-13	Управляющий код генератора тактовой частоты (от 0 до 0x3F)
12	Предварительный делитель частоты отключен: 1, включен: 0
11-6	Управляющий код генератора тактовой частоты (от 0 до 0x3F)
5-0	Должно быть равно 0

Разряды 19 и 12 должны иметь одинаковое значение. Разряды 18-13 и разряды 11-6 также должны иметь одинаковое значение. В любом другом случае чип проигнорирует значение Clock («Тактовый сигнал»).

Например, если код тактового сигнала равен 0x1F, а предварительный делитель частоты отключен, значение Clock («Тактовый сигнал») будет равно 0x038BF7C0.

5.5. SET MASK («ЗАДАТЬ МАСКУ»)

Команда set mask («Задать маску») используется для управления диапазоном вычислений. Шаблон команды показан на рис. 5.6.

Рис. 5.6. Шаблон команды set mask («Задать маску»)

Сброс	0x20	0x03	Значение маски	Состояние	Контрольная сумма команды
	1 байт	1 байт	4 байт	1 байт	1 байт

Табл. 5.3. Значение маски

Разряды	Описание
31-16	Код маски — число типа integer в диапазоне от 0x0000 до 0x000F
15-0	Значение попсе — 16 младших разрядов значения попсе ^ 0xAAAA

Для обычной работы с поиском попсе во всем диапазоне значение Mask («Маска») должно быть равно 0x00000000. Код Mask («Маска») типа integer определяет количества разрядов в значении попсе (первым является младший разряд), которые будут зафиксированы в процессе поиска значения попсе.

6. Отличия чипа Clarke от чипа BF8162B

6.1. КОМАНДА TOGGLE («ПЕРЕКЛЮЧИТЬ»)

В чипе Clarke нет команды Toggle («Переключить»). Теперь не нужно записывать предопределенную константу с помощью команды Toggle («Переключить»). Если к чипу Clarke будет применена команда Toggle («Переключить»), он создаст правильную контрольную сумму, но внутри чипа не будет выполнено никакой операции. Это означает, что можно оптимизировать управляющее программное обеспечение BF8162B, но можно и оставить его без изменений, так как ПО будет работать правильно.

6.2. МАРКЕР TASK SWITCH («ПЕРЕКЛЮЧЕНИЕ ЗАДАЧИ»)

В чипе Clarke используется новый маркер task switch («Переключение задачи»). У нового маркера типа dword младший значащий бит имеет значение 0xC (в чипе BF8162B использовался маркер 0xF). Это позволяет управляющему программному обеспечению определить чип Clarke.

6.3. РАБОЧЕЕ НАПРЯЖЕНИЕ

Рабочее напряжение чипа Clarke меньше, чем у чипа BF8162B. Соответственно, при использовании чипа Clarke следует изменить цепь его питания, перенастроив ее на необходимое напряжение.

6.4. ВСТРОЕННЫЙ ГЕНЕРАТОР

Чип Clarke допускает изменение параметров внутреннего генератора в большем диапазоне, чем чип BF8162B. Оптимальный диапазон генератора может составлять от 40 до 63.

6.5. ЛУЧШИЙ ПОКАЗАТЕЛЬ МВ/ГХ

У чипа Clarke показатель мВ/ГХ лучше, чем у чипа BF8162B. Кроме того, изменились оптимальные рабочие точки (рекомендуемый рабочий ток 18 А).

6.6. НОВЫЙ ДОПОЛНИТЕЛЬНЫЙ КОРПУС

Добавлен корпус FCLGA 4L. Этот корпус оптимизирован для удобной разводки цепи питания на печатной плате. Размер чипа (6 x 6 мм) такой же, как у корпуса FCLGA 35L. Это сделано для того чтобы можно было использовать существующие радиаторы.

7. Характеристики и номинальные параметры устройства

7.1. АБСОЛЮТНЫЕ МАКСИМАЛЬНЫЕ ПАРАМЕТРЫ

Обозначение	Описание	Мин.	Типовое	Макс.	Единица измерения
V _{VDD}	Максимальное напряжение питания	—	—	0,8	В

7.2. ХАРАКТЕРИСТИКИ ПО ПОСТОЯННОМУ ТОКУ

Номинальные значения при V_{VDD} = 0,4 В

Обозначение	Описание	Мин.	Типовое	Макс.	Единица измерения
I _{OL}	Поддержание выходного тока на контакте SDATA	—	1,7	—	мА
I _{OH}	Поддержание выходного тока на контакте SDATA	—	-1,7	—	мА

7.3. ХАРАКТЕРИСТИКИ ПО ПЕРЕМЕННОМУ ТОКУ

Номинальные значения при V_{VDD} = 0,38 В

Обозначение	Описание	Мин.	Типовое	Макс.	F
F _{SCK-MAX}	Максимальная рабочая частота на контакте SCK	—	—	8	МГц
F _{SDATA-RISE}	Время нарастания выходного сигнала на контакте SDATA	—	13	—	нс
F _{SDATA-FALL}	Время спада выходного сигнала на контакте SDATA	—	13	—	нс

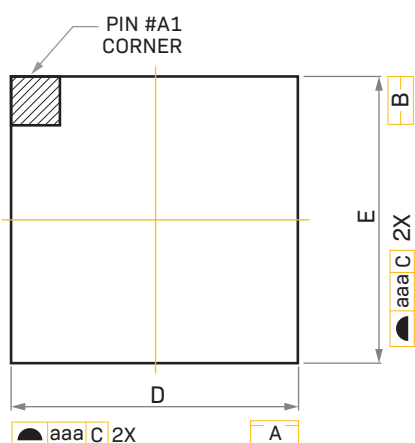
8. Пример обмена данными

```
// программное присвоение тактовому генератору значения 0x20 и предварительному делителю значения 1
(отключен)
Отправлено: 0803038с1800
получено: f0b200b2
// Задание для маски поппе значения 0x00000000
Отправлено: 200300000000
получено: f0230023
// Принудительное переключение задачи
Отправлено: 020000
получено: f0020002
// Отправка задачи
wi 0=CD3F992C 1=037F8197 2=A58E091A
MS0 A=0CAD7CD1 B=CBE38FD9 C=D14DC164 D=F90EB10B E=819621CF F=358D45CD G=8C14CAE3 H=538EF887 MS3
A=5FF18CDD B=8CDA24A4 C=180266F9 D=0CAD7CD1 E=B0CA39FA F=DD30B962 G=36D2CBC6 H=819621CF NONCE=D5D0E8B9
Отправлено: _014fa607d67b614925737be76bce53a41ba12b3c8b659f27ef6726be6049f924522d2b3c8b659c78616c779
a13c81a609
35067953386a9d52b3d0f24a3b0a607d67bb2a8cc5326708e0ef55b2677000000000
получено: 0fb200b2
// Принудительное переключение задачи
Отправлено: 020000
получено: 0f020002
// чтение из буфера поппе
Отправлено: 040000
получено: _0f040000000000000000000000000000000000000000000000000000000000000000000003ffffffc7f7a42132ffffffc
1fffffffc0001ffbfb8c
FIFO[0] = aaaaaaaaa
FIFO[1] = aaaaaaaaa
FIFO[2] = aaaaaaaaa
FIFO[3] = aaaaaaaaa
FIFO[4] = aaaaaaaaa
FIFO[5] = aaaaaaaaa
FIFO[6] = aaaaaaaaa
FIFO[7] = 3ffffffc <- маркер task switch (переключение задачи)
FIFO[8] = d5d0e8b9 <- значение nonce
FIFO[9] = 2ffffffc <- маркер task switch (переключение задачи)
FIFO[A] = 1ffffffc <- маркер task switch (переключение задачи)
FIFO[B] = aaab5515
```

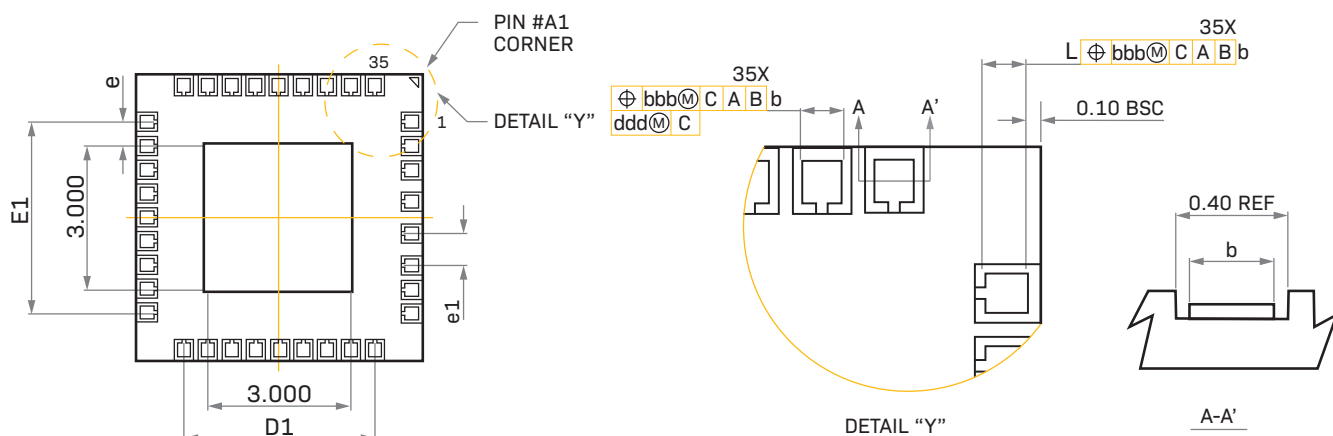
9. Габариты корпуса

9.1. КОРПУС FCLGA 35L

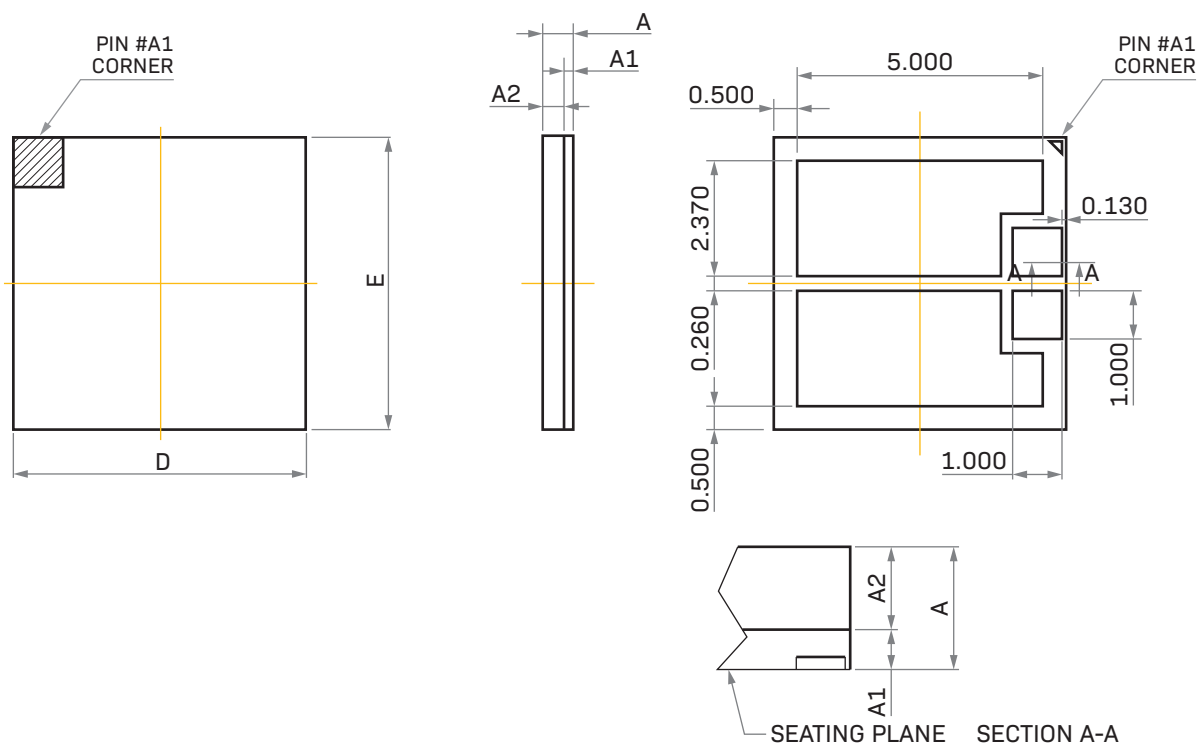
- FCLGA 35L 6 × 6 мм
- Общая толщина корпуса: не более 0,81 мм



Обозначения	Мин.	Номин.	Макс.
A			0,81
A1	0,23	0,27	0,31
A2	0,48 (справ. р.)		
C	0,03 (справ. р.)		
D	5,90	6,00	6,10
E	5,90	6,00	6,10
D1	4,00 (межд. центр.)		
E1	4,00 (межд. центр.)		
b	0,25	0,30	0,35
e	0,50 (межд. центр.)		
e1	0,667 (межд. центр.)		
L	0,25	0,30	0,35
aaa	—	—	0,10
bbb	—	—	0,10
ccc	—	—	0,10
ddd	—	—	0,08
eee	—	—	0,08



9.2. КОРПУС FCLGA 4L



	Ось	Обозначение	Общие габариты		
			Мин.	Номин.	Макс.
Общая толщина		A	—	—	0,81
Толщина подложки		A1	0,29 (справ. р.)		
Толщина отливки		A2	0,45 (справ. р.)		
Размеры корпуса	X	D	5,9	6,0	6,1
	Y	E	5,9	6,0	6,1
Количество выводов		n	4		

КОНТАКТНАЯ ИНФОРМАЦИЯ

sales@bitfury.com

ОФИС В АМСТЕРДАМЕ

Herengracht 168,
1016 BP,
Amsterdam,
The Netherlands
(Нидерланды)

ОФИС В САН-ФРАНЦИСКО

456 Montgomery St.,
Suite 1350,
San Francisco, CA
94104,
United States (США)

ОФИС В ВАШИНГТОНЕ (ОКРУГ КОЛУМБИЯ)

1440 G St.,
NW, Suite 900,
Washington, D.C.,
20001,
United States
(США)

ОФИС В ЛОНДОНЕ

Level 39,
One Canada Square,
Canary Wharf,
London, E14 5AB,
UK (Соединенное
Королевство)

ОФИС В ГОНКОНГЕ

Global Trade
Centre,
Units 305-307,
3/F 15 Wing Kin
Road Kwai Chung,
N.T. (Новые
Территории)

ОФИС В ТОКИО

6-5-1 Nishi Shinjuku,
Shinjuku-ku, Tokyo,
Shinjuku Island
Tower 5F, Japan
(Япония)

ОФИС В МОСКВЕ

Новая площадь,
6,
2-й этаж,
Москва,
Россия

Информация, содержащаяся в данном документе, отражает текущую точку зрения компании The Bitfury Group на обсуждаемые в документе вопросы на дату публикации. Из-за постоянно изменяющихся рыночных условий данный документ не следует считать каким-либо обязательством со стороны компании The Bitfury Group, и компания The Bitfury Group не гарантирует точность представленной в документе информации после даты публикации.

Данный документ предназначен только для информационных целей. В данном документе компания The Bitfury Group не предоставляет каких-либо явных или косвенных гарантий.

www.bitfury.com

© The Bitfury Group, 2018

