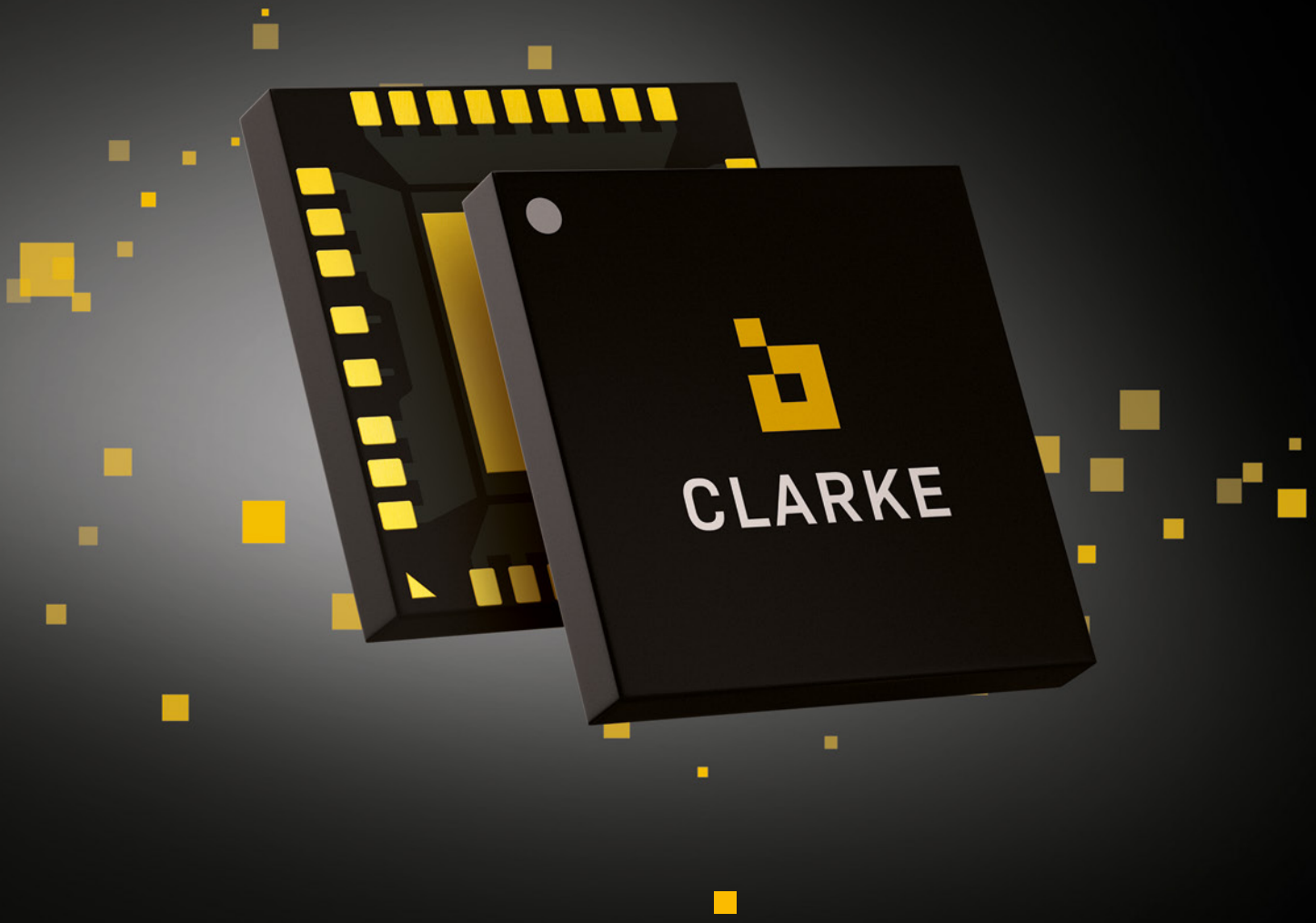


BITFURY CLARKE B1549493-010 Datasheet

1 October 2018
Revision: 0.5

BITFURY

SUMMARY

Bitfury Clarke is a double SHA256 ASIC designed for Bitcoin mining with optimized package size for smaller PCB designs. It provides the following features:

Efficiency as low as 55 mJ/GH

- Calculation speed up to 120 GH/s
- 8154 rolled hashing cores
- Simple two wire synchronous serial control interface with speed up to 8 Mbit/s
- Task double buffer for highest performance
- Fully integrated controllable clock generation
- Integrated power-on-reset circuit
- Operates from voltages as low as 0.3 V
- Compact Pb-free 6 × 6 mm FCLGA 35L package

Bitfury Clarke performance characteristics

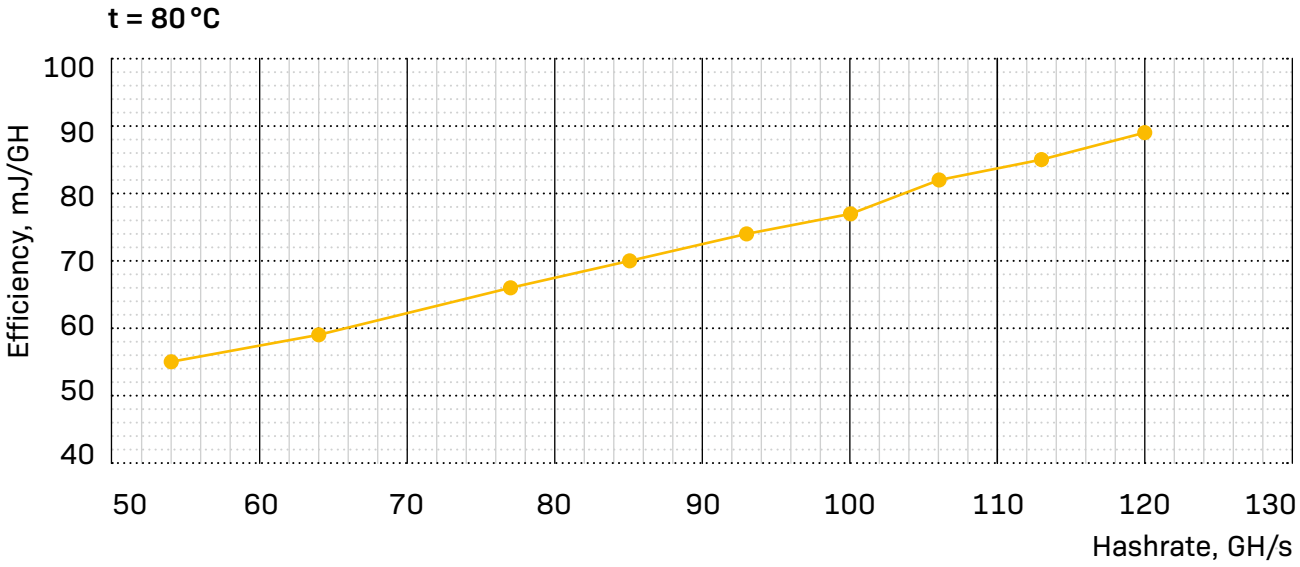
The below characteristics represent an average performance of the chips. Performance of individual chips or batch/lot of chips may vary.

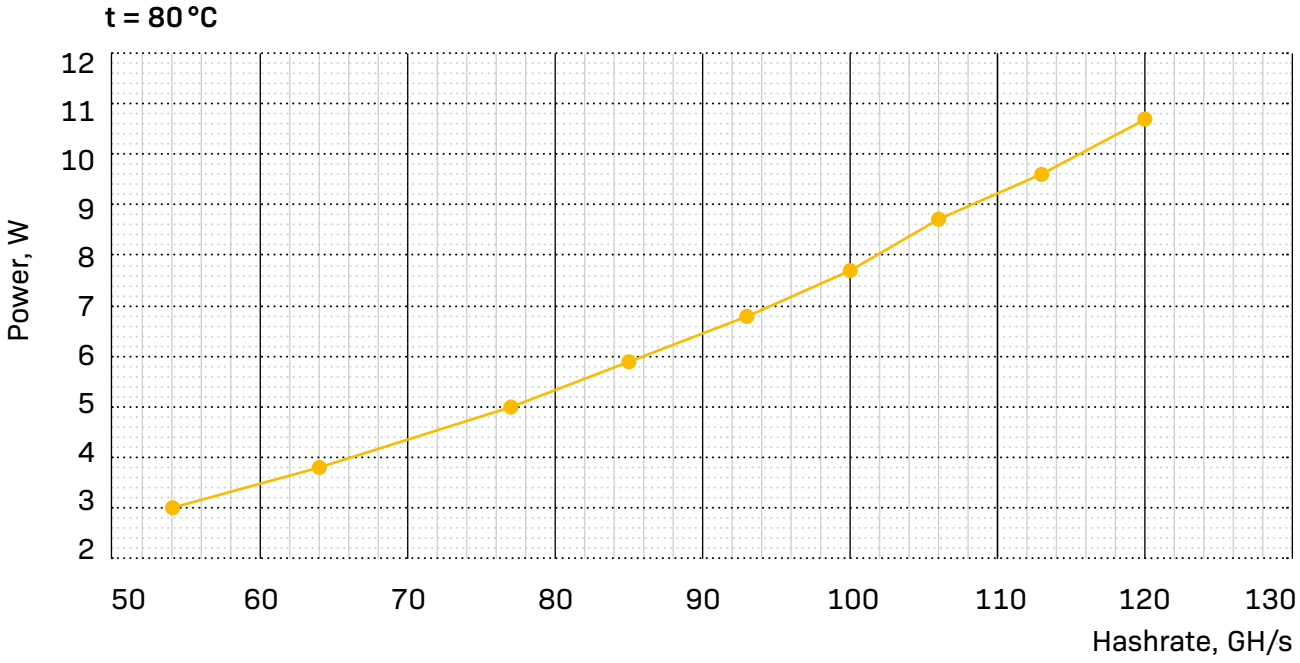
The represented performance characteristics were achieved in real working environment and in Bitfury reference design hash boards applied at Bitfury reference design air cooling miners (servers) Bitfury B8.

If applied in different/custom application designs both for hash boards and/or miners (servers) Bitfury cannot guarantee or be responsible for characteristics mentioned below.

Average performance and energy efficiency of the chips depends on power (in W) and working temperatures range.

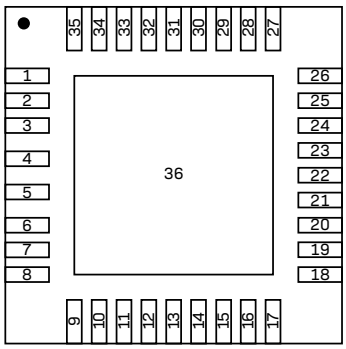
The tests were conducted at ~80 °C on chip.


t = 80 °C
Efficiency, mJ/GH
100
90
80
70
60
50
40
50
60
70
80
90
100
110
120
130
Hashrate, GH/s


t = 80 °C
Power, W
12
11
10
9
8
7
6
5
4
3
2
50
60
70
80
90
100
110
120
130
Hashrate, GH/s

1. Device pinout and signal description

1.1. FCLGA 35L PACKAGE



Top view

1.2. PINOUT DESCRIPTION

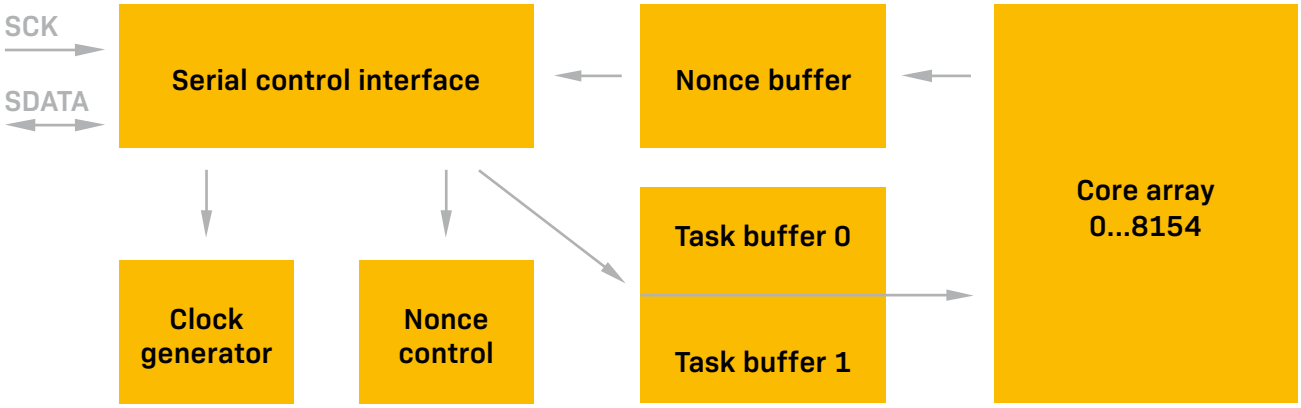
Table 1.1. Power, ground and function pins

PIN #	Name	Type	Description
4	SCK	INPUT	Serial clock input
5	SDATA	I/O	Serial data input/output
1...3, 6...35	VDD	POWER	Power input

2. Block diagram

Serial control interface block does all communications with external controller via serial bus. Clock generator provides master clock for chip operation and can be changed by command “set clock”. Nonce control block does calculation control and takes in account mask. The mask can be programmed by “set mask” or “task write” commands. Mask located in task buffer. So need set masks for both task buffers. Task buffer 0 and 1 are 20 dwords buffers used for current and next calculations. Data fills to task buffer by “task write” command. Nonce buffer is 12 dwords cyclic array. It contains nonces and task switch markers. Nonce buffer can be received from chip by “read nonces” command. Core array is 8154 double SHA-256 array of kernels.

Figure 2.1. Block Diagram



3. Function description

For fast switching between tasks the chip has two task buffers. The chip uses one task buffer for SHA256 calculations, the second one can be filled by “task write” command. As soon as current calculation task is finished the chip selects next task buffer and starts new calculation cycle. Task switching may happen at the end of calculation cycle or may be forced by “force task switch” command. Nonce buffer is a twelve dword ring buffer. During the calculation cycle the chip may find a solution (nonce) and store it to the nonce buffer. End task marker writes to the nonce buffer at the end of calculation cycle. End task marker has 16 different patterns (0x0FFFFFFC, 0x1FFFFFFC, ..., 0xFFFFFFFC). Each task switch stores is increased by 1 most significant nibble in end task marker. Nonce buffer can be read from chip by “read nonces” command.

4. Serial control interface

4.1. KEY FEATURES

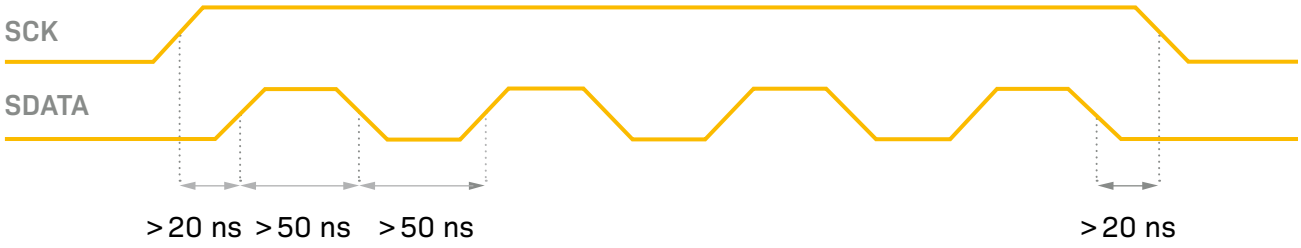
The serial control interface looks like SPI interface with combined MOSI and MISO pins to SDATA. Special reset sequence is necessary due to absence CS pin. All data bits sampled by chip on the rising SCK edge. SDATA line controlled by chip changes on the falling SCK edge. Bit sequence is MSB first. Byte sequence in dword is MSB first. Voltage levels: 0 is VSS and 1 is VDD.

4.2. RESET SEQUENCE

Special reset sequence resets the communication block of chip only and initialize his state machine to initial state. It must be used just before each command send to the chip.

At start of reset sequence SDATA line is 0 and SDA line should go from 0 to 1. After small time interval SDATA line should produce 4 pulses. And finally SCK should go to 0. The number of SDATA pulses may be more than 4, but 4 pulses is minimum.

Figure 4.1. Reset sequence



4.3. COMMON COMMAND PATTERN

All commands must have same command pattern. The first data byte of command pattern is command code. The second data byte is length in bytes minus 1. If you need to use command without data you should set length to zero and that will result in one dummy data byte in the data field. The third element is the data field.

For different commands this field has different length. The longest command allowed in data field is “task write”—80 bytes. The shortest is “force task switch”—1 dummy byte.

The command code, length and data fields must be sent by controller to the chip. As soon as last data byte has been sent to the chip, the driver SDATA line at controller side should be switched off. At falling edge of LSB bit of last data byte SDATA line will result in command being run by the chip.

The fourth element is chip status byte. The fifth is command checksum. The sixth element is data from nonce buffer. Nonce buffer length is 48 bytes, so this field has fixed length. The seventh field is nonce checksum. Apart from “read nonces” command all other commands may use only 5 first fields—from command code to command checksum.

Figure 4.2. Common command pattern

Reset	Command Code	Length	Data	Status	Command Checksum	Nonces	Nonce Checksum
	1 byte	1 byte	1...80 bytes	1 byte	1 byte	48 bytes	1 byte

4.4. STATUS

Status byte has three fields. The most significant nibble contains the MSB nonce counter. It may help to know how much time remained to finish current task. Bits 3 and 2 are equal and contain the number of receiving task buffer at start of current command on serial bus. Bits 1 and 0 contain the number of receiving task buffer just at time of receiving them on serial bus. As usual bit 1 and bit 0 are equal. They may be different only if task switch happens between sending bit 1 and bit 0. The difference between bit 2 and 1 may help to find split task write sending. If bit 2 and 1 are different it means this command was started at one receiving task buffer but finished on other. So for all commands excluding “task write” this situation is not critical. For “task write” it means the “task write” is unsuccessful.

Figure 4.3. Status byte

MSB nonce counter		Start buffer		End buffer	
7	4	3	2	1	0

Last two bits in status byte have one interesting feature. At times one of two LSB data bits on SDATA line controlled directly from the buffer switch schematic it is possible to freeze SCK line to 0 between bit 1 and bit 0 and has changing SDATA line just at buffer switch time. If controller needs read only status without any active command, command code should be set to zero.

4.5. CHECKSUM

Chip sends to controller command checksum and nonce checksum. Both checksums calculate as arithmetic sum of bytes starting from 0. Command checksum consists of sum of all bytes from command code up to last data byte inclusively. Nonce checksum consists of sum of all bytes from first nonce byte up to last nonce byte inclusively.

For example: command 04 03 03 8C 18 00 has checksum = AE.

4.6. MULTI COMMANDS

Command in command byte marks as appropriate bit set to 1. So possible number of commands is 8. But chip serial control block supports only 6. Two most significant bits in command byte will be ignored. This approach allows set several bits in command byte. In this case chip will execute commands in some order.

The order of multi command is:

1. **Task write**
2. **Force task switch**
3. **Read nonces**

All others commands should not be used in multi command. Chip behavioral for others commands into multi command is not specified.

5. Command set

5.1. TASK WRITE

“Task write” command is used to send new job to a chip. Command pattern is shown below.

Reset	0x01	0x4F	Data	Status	Command Checksum
	1 byte	1 byte	1...80 bytes	1 byte	1 byte

Figure 5.1. Task write command pattern

Code of “task write” command is 0x01. Length element may variate from 0 up to 0x4F (79 dec). If full task should be updated the length should be 0x4F. Data dwords sequence is shown below in Table 5.1.

Table 5.1. Task write data sequence

Seq. number	Name	Description
1	MS0_A	Initial value of SHA-2 word ^ 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	Midstate value after 3 rounds ^ 0xAAAAAAAA
10	MS3_G	
11	MS3_F	
12	MS3_E	
13	W0	SHA-2W[0] word value ^ 0xAAAAAAAA
14	W1	SHA-2W[1] word value ^ 0xAAAAAAAA
15	W2	SHA-2W[2] word value ^ 0xAAAAAAAA
16	MS3_D	Midstate value after 3 rounds ^ 0xAAAAAAAA
17	MS3_C	
18	MS3_B	
19	MS3_A	
20	MASK	Mask value, see “set mask” command

5.2 FORCE TASK SWITCH

"Force task switch" command is used to change current task buffer. It may be used if calculations for current buffer don't need to finish. Command pattern is shown in Figure 5.2.

Figure 5.2. Force task switch command pattern

Reset	0x02	0x00	0x00	Status	0x02
	1 byte	1 byte	1 byte	1 byte	1 byte

5.3 READ NONCES

"Read nonces" command is used to extract last nonce data from a chip. Command pattern is shown below in Figure 5.3.

Figure 5.3. Read nonces command pattern

Reset	0x04	0x00	0x00	Status	0x04	Nonces	Nonce Checksum
	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte

Nonces field contains nonce data filled by chip. This field contains 12 dwords. Chip fills nonce buffer from last dword (12) to first one (1). Nonce buffer is cyclic buffer, so as soon first dword (1) is filled the next one filled will be the last dword (12). Nonce buffer also contains end task markers. End task marker writes to nonce buffer at the end of current task calculation and may be used to understand this task has been finished and no more nonces are expected. Please note that nonces values should be xored by 0xAAAAAAAA constant.

5.4 SET CLOCK

"Set clock" command is used to set new control value for clock generator. Command pattern is show below in Figure 5.4. The Clock value element contains several fields and shown in Table 5.2.

Figure 5.4. Set clock command pattern

Reset	0x08	0x03	Clock value	Status	Nonce Checksum
	1 byte	1 byte	4 byte	1 byte	1 byte

Table 5.2. Clock value

Bits	Description
31...20	Magic const = 0x038
19	Prescaler disabled — 1, enabled — 0
18...13	Clock generator control code (0...0x3F)
12	Prescaler disable — 1, enabled — 0
11...6	Clock generator control code (0...0x3F)
5...0	Should be 0

Bits 19 and 12 must have same value. Bits 18..13 and bits 11..6 also must have same value. In any other case the clock value will be ignored by chip.

For example if clock code = 0x1F and prescaler disabled the clock value = 0x038BF7C0.

5.5 SET MASK

"Set mask" command is used to control range of calculations. Command pattern is shown in Figure 5.6.

Figure 5.6. Set mask command pattern

Reset	0x20	0x03	Mask value	Status	Command Checksum
	1 byte	1 byte	4 byte	1 byte	1 byte

Table 5.3. Mask value

Bits	Description
31...16	Mask code—integer from 0x0000 to 0x000F
15...0	Nonce value—lowest 16 bits of nonce ^ 0xAAAA

For normal routine operation with full range nonce search Mask value should be set to 0x00000000. Mask code integer define numbers of bits in nonce (lowest first) that will be fixed during Nonce search.

6. Clarke difference to BF8162B

6.1. “TOGGLE” COMMAND

Clarke chip have not “Toggle” command. Now not necessary to write predefined constant by “Toggle” command. If “Toggle” command apply to Clarke chip than correct check sum will be generated by chip but inside of chip nothing happens. It means the BF8162B control software can be optimized but may be leave as is and it works correctly.

6.2. TASK SWITCH MARKER

Clarke chip have new task switch marker. The new marker dword has the least significant nibble set to 0xC value (BF8162B has 0xF marker). It may help the control software identify Clarke chip.

6.3. OPERATION VOLTAGE

The operation voltage of Clarke chip is lower than BF8162B chip. It means the Clarke chip power chain should be changed for fixed voltage power supply.

6.4. INTEGRATED OSCILLATOR

Clarke chip have more internal oscillator variation than BF8162B chip. The optimal oscillator range may be from 40 till 63.

6.5. BETTER MV/GH

Clarke chip have better mV/GH than BF8162B chip. The optimal operation points also changed (Recommended operation current 18 A).

6.6. NEW ADDITIONAL PACKAGE

New package FCLGA 4L added. This package optimized to be comfortable for PCB power chain layout. The chip size 6x6mm is same as FCLGA 35L package to be compatible with existing heatsinks.

7. Device characteristics and ratings

7.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Description	Min.	Typical	Max.	Unit
V_{VDD}	Power supply maximum voltage	—	—	0.8	V

7.2 DC CHARACTERISTICS

Ratings for V_{VDD} = 0.4 V

Symbol	Description	Min.	Typical	Max.	Unit
I_{OL}	SDATA pin output current drive	—	1.7	—	mA
I_{OH}	SDATA pin output current drive	—	–1.7	—	mA

7.3 AC CHARACTERISTICS

Ratings for V_{VDD} = 0.38 V

Symbol	Description	Min.	Typical	Max.	F
$F_{SCK-MAX}$	Maximum operating frequency on SCK pin	—	—	8	MHz
$F_{SDATA-RISE}$	SDATA pin output rise time	—	13	—	nS
$F_{SDATA-FALL}$	SDATA pin output fall time	—	13	—	nS

8. Communication example

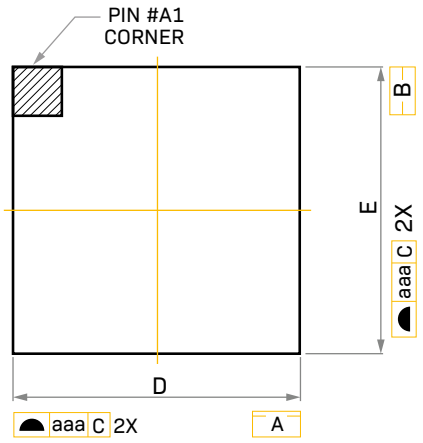
```
// Program clock generator to value 0x20 and prescaler=1 (disabled)
Send: 0803038c1800
Recv: f0b200b2
// Set nonce mask to 0x00000000
Send: 200300000000
Recv: f0230023
// Force task switch
Send: 020000
Recv: f0020002
// Send task
Wi 0=CD3F992C 1=037F8197 2=A58E091A
MS0 A=0CAD7CD1 B=CBE38FD9 C=D14DC164 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
Send:_014fa607d67b614925737be76bce53a41ba12b3c8b659f27ef6726be6049f924522d2b3c8b659c78616c779
a13c81a609
35067953386a9d52b3d0f24a3b0a607d67bb2a8cc5326708e0ef55b267700000000
Recv: 0fb200b2
// Force task switch
Send: 020000
Recv: 0f020002
// Read nonce buffer
Send: 040000
Recv:_0f040000000000000000000000000000000000000000000000000000000003ffffffc7f7a42132ffffffc
1ffffffc0001ffbf8c
FIFO[0] = aaaaaaaa
FIFO[1] = aaaaaaaa
FIFO[2] = aaaaaaaa
FIFO[3] = aaaaaaaa
FIFO[4] = aaaaaaaa
FIFO[5] = aaaaaaaa
FIFO[6] = aaaaaaaa
FIFO[7] = 3ffffffc <- task switch marker
FIFO[8] = d5d0e8b9 <- nonce value
FIFO[9] = 2ffffffc <- task switch marker
FIFO[A] = 1ffffffc <- task switch marker
FIFO[B] = aaab5515
```

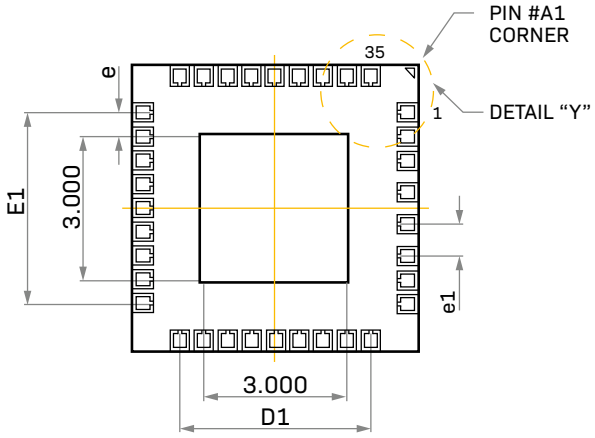
9. Package dimensions

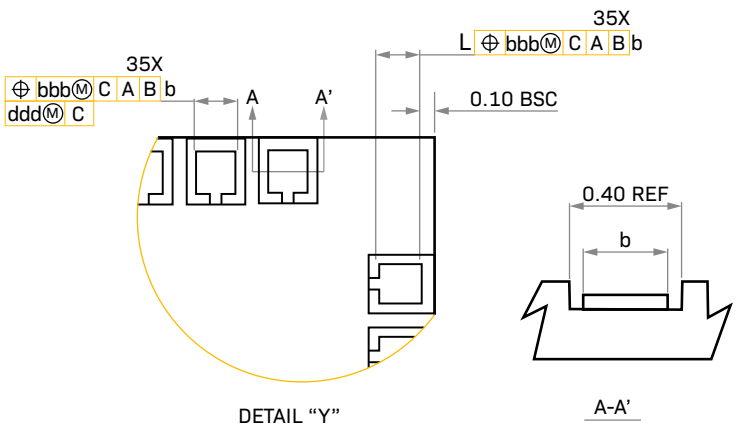
9.1. FCLGA 35L

- FCLGA 35L 6 × 6 mm
- Total package thickness — max 0.81 mm

Symbols	Min.	Nom.	Max.
A			0.81
A1	0.23	0.27	0.31
A2		0.48 REF	
C		0.03 REF.	
D	5.90	6.00	6.10
E	5.90	6.00	6.10
D1		4.00 BSC.	
E1		4.00 BSC.	
b	0.25	0.30	0.35
e		0.50 BSC.	
e1		0.667 BSC.	
L	0.25	0.30	0.35
aaa	—	—	0.10
bbb	—	—	0.10
ccc	—	—	0.10
ddd	—	—	0.08
eee	—	—	0.08







CONTACT

sales@bitfury.com

AMSTERDAM OFFICE

Herengracht 168,
1016 BP,
Amsterdam,
The Netherlands

SAN FRANCISCO OFFICE

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

WASHINGTON, DC OFFICE

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

LONDON OFFICE

Level 39,
One Canada Square,
Canary Wharf,
London, E14 5AB,
UK

HONG KONG OFFICE

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

TOKYO OFFICE

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

MOSCOW OFFICE

6 Novaya Square,
2nd floor,
Moscow,
Russia

www.bitfury.com

BITFURY