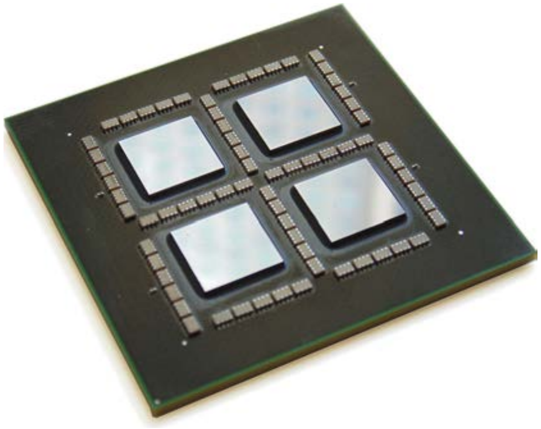


Summary

The HashFast Golden Nonce GN is an ultra high performance multi-SHA engine able to produce 768 double SHA256 results per clock cycle. The ASIC is implemented in a 28nm silicon process. The highest frequency of operation is typically around 1GHz, depending on silicon process, for a peak performance of around 768GHash/Sec. The Golden Nonce is a carefully balanced design; the objective being to achieve peak performance given the device is sufficiently cooled. In other words an application will be thermally limited rather than frequency limited, thus allowing from simple to extreme cooling solutions in order to achieve a specific hash rate. To allow maximum cooling efficiency, the ASIC substrate contains four bare dies spaced apart, which may be placed in direct contact with a cold plate; air cooled for medium performance applications, or fluid cooled for maximum performance. Hashing efficiency has been measured at approximately 0.62 Joules per gigahash in a typical application operating at 420GH/s.

Each die of the four die on the substrate are completely independent with a simple serial interface to the outside world. At a minimum the die require a reference clock (typically 25MHz), RESET, and asynchronous serial data in and out, IO power (1.8v) and core power (.56-1.00v). Each die also provides eight general purpose inputs and eight general purpose outputs

Each die contains 96 addressable cores. Each core contains two complete double hash engines which share work across one job. (96 cores) x (2 engines) x (4 die) = 768 double SHA256 engines per substrate. Each SHA engine is pipelined to accept input and produce output on every clock.

A job queue front ends each core and allows for one job in operation and one job pending. The pending queue gives the host processor an indication to feed another job to a core without ever letting that core go idle.

Pin Operating Condition and Characteristics

Signal	min	typ	max	units
VDD	0.6*	0.81	1	V
IO_VDD	1.62	1.8	1.98	V
T_j	-40	25	125	C
V input max			1.98	V
Input Pull-dn	49k	85k	159k	ohms
V_{il} input low V	-0.3		0.63	V
V_{ih} input high V	1.17		1.98	V
V_t threshold	0.81	0.89	0.97	V
C_{il} input load cap			4	pf
V_{ol} output low			0.45	V
V_{oh} output high	1.35			V
I_{ol} low output I	5.5	9.1	12.9	ma
I_{oh} High output I	4.4	9.6	17.1	ma
C_{ol} output load			4	pf

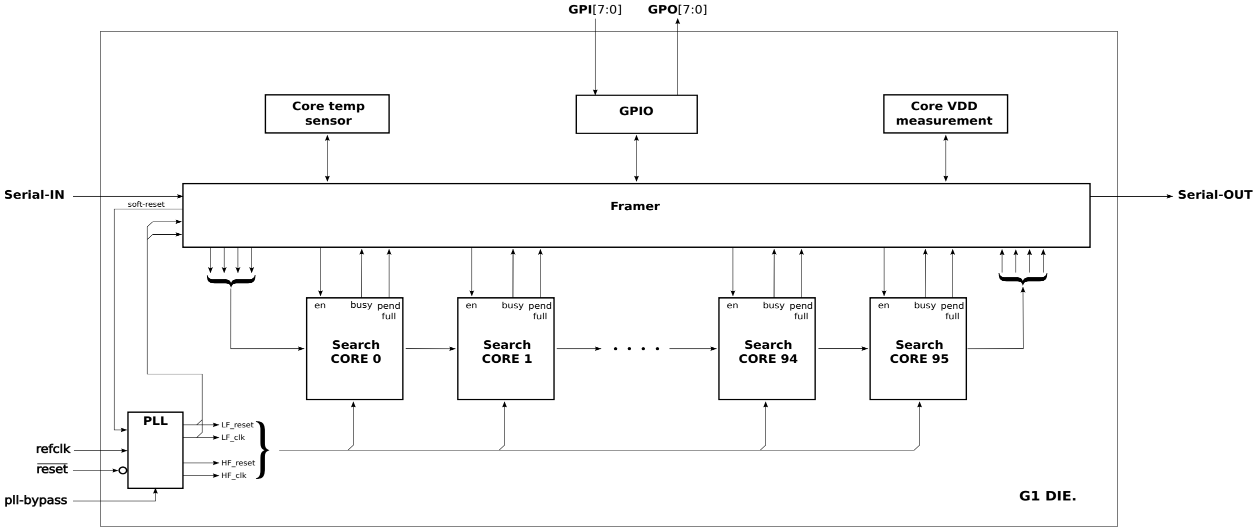
* Note – the PLL may not operate correctly below 0.7v.

Pins Functions – Die 1

Signal	Pins	Function
Rx_Serial_D1,2,3,4	C12,C27,BB33,BB18	Die serial data input (see protocol specification), data valid for parallel mode
Tx Serial_D1,2,3,4	B12,B27,BC33,BC18	Die serial data output (see protocol specification), data valid for parallel mode
CLK_IN_P_D1,2,3,4	C10,C25,BB35,BB20	Die reference clock (25MHz)
PLL_AVSS_D1,2,3,4	A10,A25,BD35,BD20	Die Analog bypass (do not connect to gnd) bypass to PLL_AVDD_Dx
PLL_AVDD_D1,2,3,4	A9,A24,BD36,BD21	Die Analog 1.8v. Connect to 1.8v I/O VDD with series 100ohm resistors. Bypass to PLL_AVSS_Dx with .01uf and .1uf caps as close to pins as possible
Hard_Reset_D1,2,3,4	B11,B26,BC34,BC19	Die reset (asynchronous)
VTRIM_D1,2,3,4	C19,C34,BB26,BB11	Die temperature sensor trim pin (Connect to accurate 0.7v source for better accuracy on temperature readings)
MODE_D1,2,3,4	A11,A26,BD34,BD19	Die mode pin. Tie high (to I/O VDD) for serial mode input and output. Tie low (to I/O VSS) for parallel input and output
GPO[0]_D1,2,3,4	A17,A32,BD28,BD13	Die General purpose outputs. May be configured to produce PWM outputs, LED control, general host_data, or parallel output data in place of the serial mode of operation. The GPO function is part of the framer. See the Interface Protocol Guide for description.
GPO[1]_D1,2,3,4	B17,B32,BC28,BC13	
GPO[2]_D1,2,3,4	B18,B33,BC27,BC12	
GPO[3]_D1,2,3,4	C18,C33,BB27,BD12	
GPO[4]_D1,2,3,4	A19,A34,BD26,BD11	
GPO[5]_D1,2,3,4	A20,A35,BD25,BD10	
GPO[6]_D1,2,3,4	B20,B35,BC25,BC10	
GPO[7]_D1,2,3,4	A21,A36,BD24,BD9	
GPI[0]_D1,2,3,4	A13,A28,BD32,BD17	Die General purpose input, or parallel data input.
GPI[1]_D1,2,3,4	C13,C28,BB32,BB17	
GPI[2]_D1,2,3,4	A14,A29,BD31,BD16	
GPI[3]_D1,2,3,4	B14,B29,BC31,BC16	Die General purpose input, or parallel data, or tachometer

GPI[4]_D1,2,3,4	B15,B30,BC30,BC15	Die General purpose input, or parallel data input , Sampled at reset to loopback serial_in to serial_out
GPI[5]_D1,2,3,4	C15,C30,BB30,BB15	Die 1 General purpose input, or parallel data input , Sampled at reset to set Serial baud delay
GPI[6]_D1,2,3,4	A16,A31,BD29,BD14	
GPI[7]_D1,2,3,4	C16,C31,BB29,BB14	
VDD_D1,2,3,4	See package	Die 0.81v nominal core voltage. The cores will run at approximately 720MHz at nominal voltage. Higher frequencies require a higher core voltage. There are approximately 270 pins per die dedicated to supplying core power
VSS		Core power return (common to all four die)
IO_VDD_1P8V_D1,2,3,4	See Package	IO_VDD supplies power to nine low speed output pins, each with 4ma drive capability. 10ma of 1.8v capability will be adequate.
IO_VSS_1P8V_D1,2,3,4	See Package	IO_VSS is the return for the IO_VDD and the PLL. This ground should be common with external logic which drives and receives signals from the device such as the reference clock, serial_rx and serial_tx

Block Diagram



Operation

Dataflow

Each of the four die on the chip has it's own serial IN and OUT pins. Serial IN and OUT maybe daisy chained directly from one die to the next on the PCB. It is however wise to route through some simple routing logic should the ability to route around a bad die be required. Such logic may be purely combinatorial. Asynchronous serial data enters the device through these pins and travels to the framer. Alternatively, the die may be put in parallel mode with data entering on the GPI[7:0] pins and the rx_serial input pin serving as data_valid. GPO[7:0] and tx_serial serve the egress side. Parallel mode is expected to be a test feature. Regardless of mode, the framer handles all data to and from the die. The framer contains a standard UART function
which hunts for the serial preamble and then translates the subsequent serial stream into bytes making up a frame with a header and various fields. The header and data fields are CRC checked.

The header is parsed into an op_code, chip_address, core_address, sequence number and length of data to follow. If the header indicates the operation is *not* for this die, the data is placed in a forwarding queue and routed to the next device in the chain. Note: This same queue is used to collect results from this die to be forwarded through the chain and back to the host. If the header indicates the operation is for this die (or a broadcast), data emerges from the framer as a sequence of bytes and flags which are sent to the first hash core in the array of hash cores on the die. Each core inspects the data to see of the operation should be performed. Regardless of whether the operation is to be performed or not, the data is sent from one core to the next until the last core sends it back to the framer for forwarding to the next die or next substrate in the chain, eventually returning to the host.

If addressing information within the header indicates an operation is to be performed, the byte stream is parsed and collected into a wide parallel bus which crosses from the reference clock domain into the core's high frequency clock domain. Data in the high frequency domain is further sliced into fields for the core. One pending job may be held at this boundary.

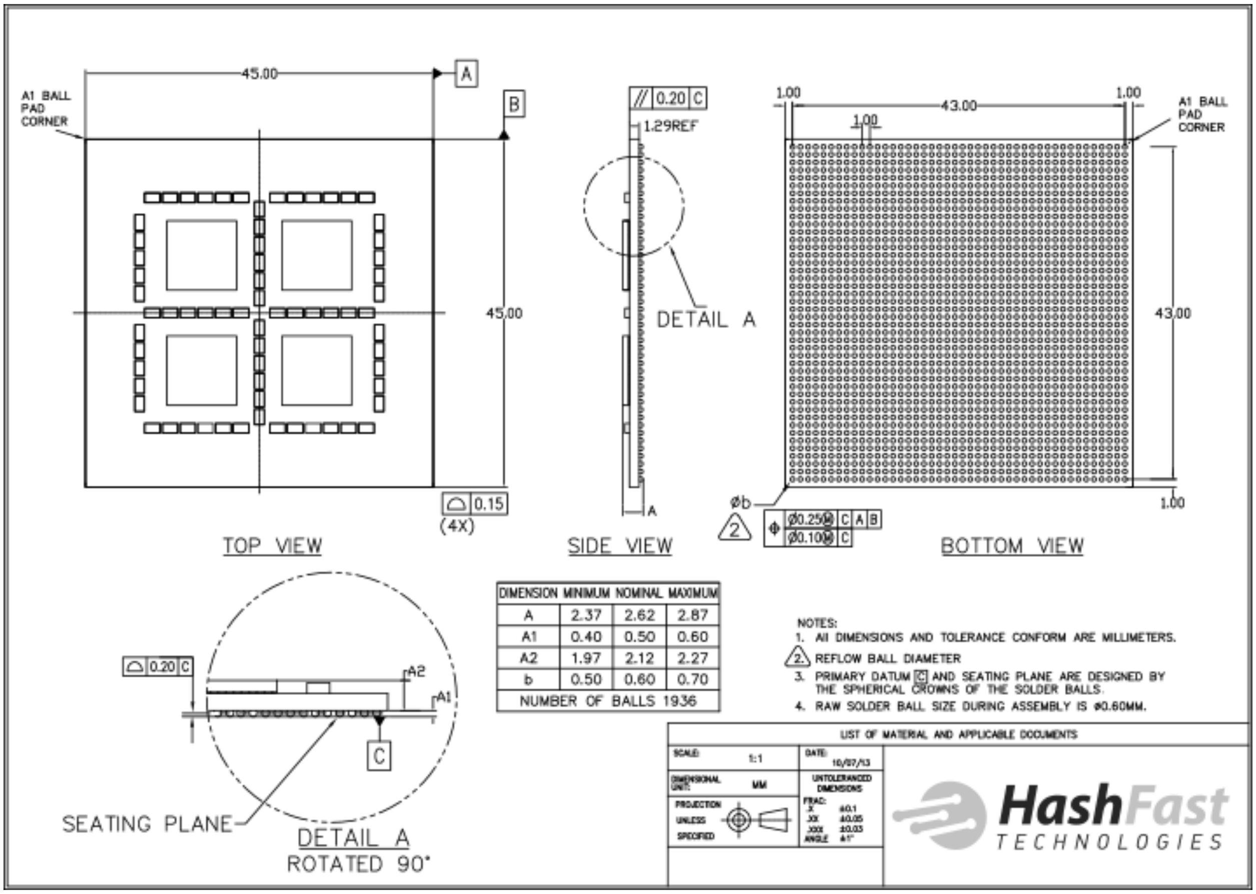
For a detailed description of the operations that can be sent to the ASIC, and the results that are returned, see the Interface Protocol Guide.

The cores may be of several different architectures. The GN ASIC cores are made up of two double SHAs which further consist of a proprietary level of unrolling, pipelining and other optimizations, designed for high speed and high efficiency.

The underlying SHA engines within a core perform iterative SHA calculations, typically some portion of the 4 billion required, depending on work load spreading across the underlying architecture. Nonce results may range from none to a few and are dependent on the difficulty level set by the host. Results are collected and traverse a boundary for forwarding in the reference clock domain. The forwarding queue noted earlier is used to merge nonce results in with operations commands which may be in flight across the die. The forwarding queue feeds the transmit UART and serializes the data for egress on the tx_serial output pin (or parallel mode GPO).

The framer is central to device operations, in addition to parsing host commands and collecting results for the host, the framer receives status information from each of the cores on the die in the form of a 2 bit status field. One bit indicates the core is busy. The second bit indicates the presence of a pending job for the core. Status of each core may be collected for the host by polling, enabling a periodic status report, or by a status change. In addition to core status the framer receives die temperature and die voltages from sensors across the device. Temperature and voltage are included in the chip status sent back to the host.

Package



DIMENSION	MINIMUM	NOMINAL	MAXIMUM
A	2.37	2.62	2.87
A1	0.40	0.50	0.60
A2	1.97	2.12	2.27
b	0.50	0.60	0.70
NUMBER OF BALLS 1936			

NOTES:

1. All DIMENSIONS AND TOLERANCE CONFORM ARE MILLIMETERS.
2. REFLOW BALL DIAMETER
3. PRIMARY DATUM C AND SEATING PLANE ARE DESIGNED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.
4. RAW SOLDER BALL SIZE DURING ASSEMBLY IS ø0.60MM.

Pinout

Illustrations show the pinout as viewed down from above the chip.

Die #1 (quadrant) core power pins

V = VDD_D1
G = VSS
+ = IO_VDD
- = IO_VSS

	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2
A		v	g	v	g	v	g	v				-			+			-				v
B	v	g	v	g	v	g	v	g	-	-			+			-			+		v	g
C	g	v	g	v	g	v	g	v			+			-			+			-	g	v
D	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
E	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
F	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
G	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
H	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
J	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
K	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
L	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
M	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
N	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
P	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
R	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
T	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
U	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
V	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
W	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
Y	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AA	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AB	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g

Die #2 (quadrant) power pins

V = VDD_D2
G = VSS
+ = IO_VDD
- = IO_VSS

	23	4	5	6	7	8	9	30	1	2	3	4	5	6	7	8	9	40	1	2	3	4
A	v				-			+			-				v	g	v	g	v	g	v	g
B	g	-	-			+			-			+		v	g	v	g	v	g	v	g	v
C	v			+			-			+			-	g	v	g	v	g	v	g	v	g
D	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
E	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
F	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
G	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
H	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
J	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
K	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
L	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
M	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
N	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
P	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
R	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
T	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
U	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
V	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
W	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
Y	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AA	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AB	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v

Die #4 (quadrant) core power pins

V = VDD_D4
G = VSS
+ = IO_VDD
- = IO_VSS

	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2
AC	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AD	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AE	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AF	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AG	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AH	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AJ	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AK	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AL	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AM	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AN	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AP	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AR	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AT	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AU	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AV	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AW	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
AY	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v
AA	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g	v	g
BB	g	v	g	v	g	v	g	v		-			+			-				+		v
BC	v	g	v	g	v	g	v	g			+			-			+			-	-	g
BD	g	v	g	v	g	v	g					-			+			-				v

Known Issues

Occasionally cores may act as if an exceptionally low difficulty level has been set, and return a constant stream of OP_NONCE results, until the work has been completed. This stream of back to back results may cause issues further down the serial line, preventing new commands or results being sent on dies further down the serial chain. Sending an OP_ABORT to the core will stop the stream of results.

Document Revision Control

Version 1.0 – 4/8/2014