GA144A12 Multicomputer Array

Document No.:	GA144A12-DS-A12-PC
Revision:	A12-PC-2026.03
Date:	06 March 2026
Status:	Controlled Production Candidate

Contents

1 Document Control and Compliance

This datasheet is issued under controlled document revision practices intended to align with semiconductor industry publication norms and ISO-oriented quality management.

1.1 Document Identification

Field	Value
Document title	GA144A12 Datasheet
Document type	Product datasheet (controlled production candidate release)
Document owner	Product engineering and quality
Primary audience	Hardware, firmware, manufacturing, and quality teams
Controlled revision date	06 March 2026
Distribution class	External technical document

1.2 Revision History

Revision	Date	Change Summary	Approval
A12-PC-2026.03	06 March 2026	Formalized document-control front matter, standards framing, publication notes, and release checklist alignment.	Engineering + Quality

1.3 Normative References for Compliance Context

The following standards are referenced in this document where applicable to handling, packaging, assembly, and quality control practices:

- IPC/JEDEC J-STD-020 (moisture/reflow classification context)

- EIA/JEDEC J-STD-033 (moisture sensitive handling/storage context)

- JESD47 (stress test driven qualification methodology context)

- ISO 9001 principles (document control, traceability, corrective action discipline)

Conformance claims are limited to the specific statements and measurements in this revision and are not blanket certifications unless separately published by formal certificate.

2 Description

The GA144A12 is an array of 144 **_F18A computers_**, capable of a peak aggregate performance of 96 billion operations per second, conserving energy by varying its power level dynamically over a range of 14 microwatts to 650 milliwatts as the application demands. 22 of these computers are equipped with one or more I/O pins. The chip is designed for use in demanding applications yet its cost is low enough to consider using it as a coprocessor or even as a simple I/O expander, comparing favorably in cost and power consumption with an FPGA.

2.1 Ordering Information

Please refer to Section 8, Part Identification, for part numbering and ordering information.

2.2   Package and Mechanical Summary

Mechanical and package details are commonly the first design gate for PCB footprint, thermal path, and assembly-process decisions. The key packaging data are summarized below for quick review; full drawings and handling notes are provided in Section 7.

Parameter	Quick Data
Package family	Plastic-molded QFN, 88 pins (22 pins per side)
Body size	10 mm x 10 mm nominal
Pin pitch	0.4 mm
Grounding/thermal pad	Exposed center DAP used as the common ground reference and primary thermal path
Recommended footprint strategy	Constrain DAP metal to a 6 mm x 6 mm core area to preserve forward package compatibility (see Migration Path, Section 7.1.1)
Thermal indicator	$R_{\theta JA}$ 60 °C/W (still air), $R_{\theta JC}$ 0.5 °C/W (see Section 5.2)
Package moisture handling	MSL 3 per supplier data; follow J-STD-033 handling/storage controls (see Section 7.2.3)
Detailed dimensions and views	See Section 7: bottom, top, and side package drawings

2.3   Related Documents

This book describes the GA144A12, including its array and I/O configuration, pin-out, ROM contents, packaging, and its electrical and physical characteristics. In the interest of avoiding needless and often confusing redundancy, it is designed to be used in combination with other documents describing standard architecture and other components of the chip.

The general characteristics and programming details for the F18A computers and I/O used in this chip are described in a separate document; refer to **_F18A Technology Reference_**. The boot protocols supported by this chip are detailed in **_Boot Protocols for the GA144A Chip._** Current editions of these documents, together with the current edition of this datasheet, are published at https://cartheur.com. It is always advisable to ensure that you are using the latest documents before starting work.

2.4   Status of Data Given

The information given herein is PRODUCTION CANDIDATE data. Measurements have been made under the conditions noted in each section and are maintained under document revision control. Performance data, including speed and power consumption across temperature and V_DD ranges, are updated through controlled revisions when additional characterization data are published.

Table entries explicitly marked as unavailable indicate data for which measurements or estimates are presently incomplete. Qualified characterization results are identified in the corresponding table notes. Values identified as engineering estimates are provisional and may change as additional vendor and internal characterization data become available.

2.5   Documentation Conventions

This section defines notation and naming conventions used throughout this datasheet. Applying these conventions consistently helps prevent ambiguity when interpreting registers, node identifiers, and numeric values.

2.5.1   Numbers

Numbers are written in decimal unless otherwise indicated. Hexadecimal values are indicated by explicitly writing "hex" or by preceding the number with the lowercase letter "x". In colorForth coding examples, hexadecimal values are italicized and darkened.

2.5.2   Node coordinates

Each GA144A12 chip is a rectangular array of ***nodes,*** each of which is an F18 computer. By convention these arrays are visualized as seen from the top of the silicon die, which is normally the top of the chip package, oriented such that pin 1 is in the upper left corner. Within the array, each node is identified by a three or four digit number denoting its Cartesian coordinates within the array as *yxx* or *yyxx* with the lower left corner node always being designated as node 000. Thus, for a GA144 chip whose computers are configured in an array of 18 columns and 8 rows, the numbers of the nodes in the lower right, upper left, and upper right corners are 017, 700, and 717 respectively.

2.5.3   Register names

Register names in prose may be used with or without the word "register" and are usually shown in a bold font and capitalized where necessary to avoid ambiguity, such as for example the registers **T S R I A B** and **IO** or **io** .

2.5.4   Bit numbering

Binary numbers are visualized as a horizontal row of bits, numbered consecutively right to left in ascending significance with the least significant bit numbered zero. Thus bit n has the binary value 2^n. The notation P9 means bit 9 of register **P**, whose binary value is x200, and T17 means the sign (high order) bit of 18-bit register **T**.

3   Chip Configuration

This section identifies the F18A computer and I/O components included in the chip, their arrangement and pin-out, and their ROM software and reset behavior along with any variations from the documented norms for F18A technology.

3.1 Array Layout and Pin Designation
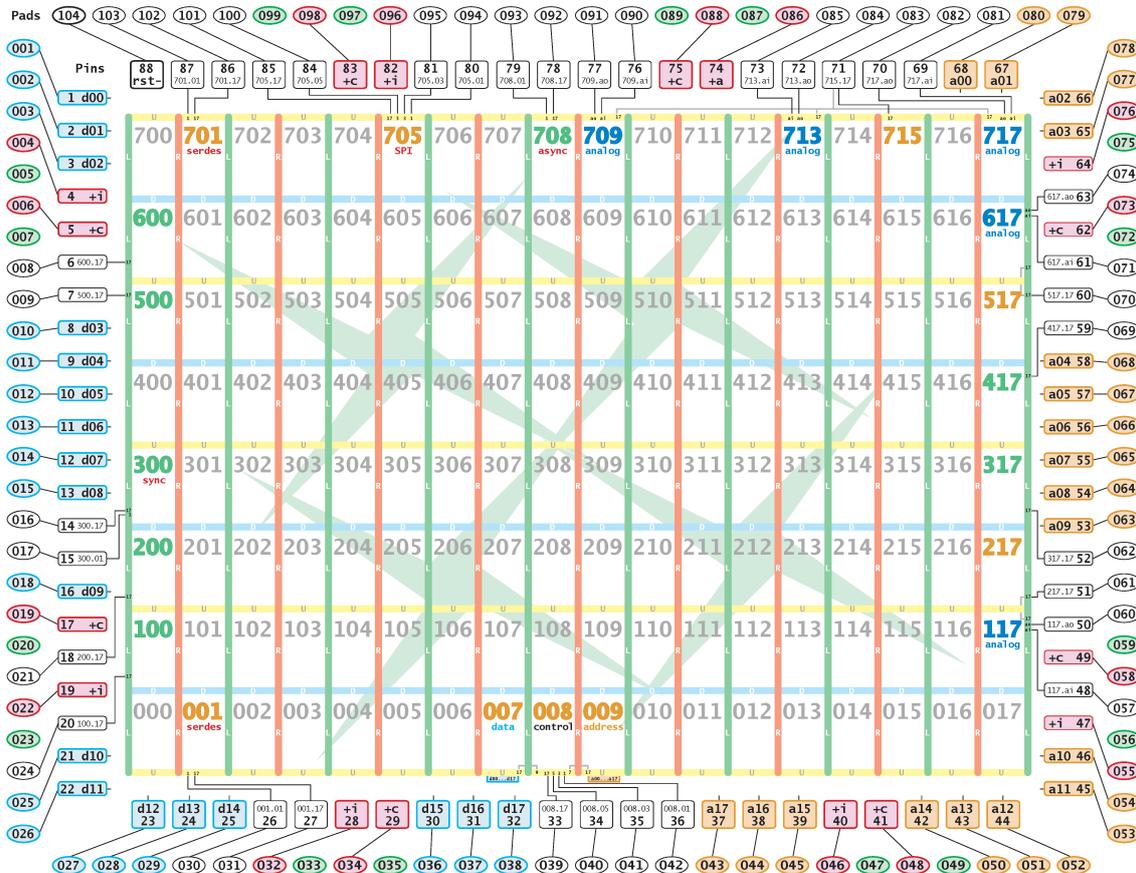
Figure 1: Array layout and pin designation

The above diagram depicts the layout of the chip, including 8 rows of 18 nodes and its I/O and power connections, as viewed from the top of the package. The outer ring shows internal die pads; the 88 exposed pins are shown on the inner ring. The ground pads are all connected internally to the exposed die attach paddle underneath the chip; this paddle serves both as a heat sink and as a single, common ground for all circuits on the chip.

Nodes numbered in green have one or more general purpose I/O pins. Those in yellow have digital I/O pin(s) with specialized configurations, which may include shared pins and/or phantom wakeup signals. Nodes numbered in blue are equipped with analog I/O. Captions underneath the node numbers indicate ROM specialization; red captions are reserved for the five nodes that are capable of booting the chip after reset. An exception is node 200 which has special 1-wire serial receive code in ROM but is not armed for booting.

Comm port directions are shown in the colored bars separating nodes. Small numbers in the outer bars represent I/O signal designations; for example, signal 100.17 (node 100 GPIO 17) is connected with chip pin 20.

3.2 Pin Functions

This section identifies each external contact on the chip.

Name	Pin	Type	Description
d00	1		
d01	2		
d02	3		
d03	8		
d04	9		
d05	10		
d06	11		
d07	12		
d08	13	Bus I/O	Bits 0 through 17 of node 007 UP port.
d09	16		General purpose bidirectional parallel bus, such as external memory data.
d10	21		
d11	22		
d12	23		
d13	24		
d14	25		
d15	30		
d16	31		
d17	32		

Name	Pin	Type	Description
008.17	33		
008.5	34	GPIO	General purpose 4-pin node.
008.3	35		Might be used for memory or bus control and handshake lines.
008.1	36		

Name	Pin	Type	Description
a17	37		
a16	38		
a15	39		
a14	42		
a13	43		
a12	44		
a11	45		
a10	46		
a09	53	Bus I/O	Bits 17 through 0 of node 009 UP port.
a08	54		General purpose bidirectional parallel bus, such as external memory address.
a07	55		
a06	56		
a05	57		
a04	58		
a03	65		
a02	66		
a01	67		
a00	68		

Name	Pin	Type	Description	
001.17	27	SERDES	Node 001 Clock	
001.1	26		Node 001 Data	Dedicated purpose pins.
701.17	86	SERDES	Node 701 Clock	ROM supports SERDES boot in each node.
701.1	87		Node 701 Data	

Name	Pin	Type	Description	
300.17	14	GPIO	Sync clock	General purpose 2-pin node. ROM supports synchronous boot
300.1	15		Sync data	with the signal assignments shown.

Name	Pin	Type	Description	
708.17	78	GPIO	Rx Input	General purpose 2-pin node. ROM supports asynchronous boot
708.1	79		Tx Output	with signal assignments shown.

Name	Pin	Type	Description	
705.17	85		Data In	General purpose 4-pin node. If 705.17 is low on reset, ROM will
705.5	84	GPIO	Data Out	attempt SPI memory boot using signal assignments shown, driving
705.3	81		Chip Enable-	signals on 705.5, 3, 1, and will leave these in output mode unless
705.1	80		Clock	programmed otherwise.

Name	Pin	Type	Description
100.17	20		
200.17	18		
500.17	7	GPIO	General purpose 1-pin nodes.
600.17	6		No special ROM or interconnections.
317.17	52		
417.17	59		

Name	Pin	Type	Description
709.ai	76	Analog In	
709.ao	77	Analog Out	
713.ai	73	Analog In	Analog nodes whose I/O is powered by separate $V_{DD}A$ bus.
713.ao	72	Analog Out	
717.ai	69	Analog In	
717.ao	70	Analog Out	

Name	Pin	Type	Description
715.17	71	GPIO	General purpose 1-pin node whose pin is shared (read only) by the above analog nodes and may be used by them for timing or other purposes.

Name	Pin	Type	Description
617.ai	61	Analog In	Analog node whose I/O is powered by $V_{DD}I$ bus.
617.ao	63	Analog Out	

Name	Pin	Type	Description
517.17	60	GPIO	General purpose 1-pin node whose pin is shared (read only) by Analog 617.

Name	Pin	Type	Description
117.ai	48	Analog In	Analog node whose I/O is powered by $V_{DD}I$ bus.
117.ao	50	Analog Out	

Name	Pin	Type	Description
217.17	51	GPIO	General purpose 1-pin node whose pin is shared (read only) by Analog 117.
RESET-	88	Input	Reset signal, active low.

Name	Pin	Type	Description
$V_{DD}C$	5	Power	Core power bus. Powers F18A computers, and parts of I/O circuitry (such as registers) that are internal to them.
	17		
	29		
	41		
	49		
	62		
	75		
	83		

Name	Pin	Type	Description
$V_{DD}I$	4	Power	I/O power bus. Powers I/O pads including the parts of the I/O circuitry collo-cated with the pads. Includes analog pads for nodes 117 and 617.
	19		
	28		
	40		
	47		
	64		
	82		

Name	Pin	Type	Description
$V_{DD}A$	74	Power	Analog power bus for pads of nodes 709, 713 and 717.
GND	DAP	Ground	Common ground and heat sink.

3.3   Configuration of each node

This table shows configuration details of each node in the array. The following section shows all specially configured nodes and groups of nodes.

Computers / Nodes	Comm Ports Present	I/O Addressing & Type		Ops with I/O or pin wakeup	Reset	ROM Type	Special Config
001	RDL-	U	SERDES	R/W	COLD	SERDES Boot	
100	RD-U	L	GPIOx1	Read	rd-u	Basic	
200	RD-U	L	GPIOx1	Read	rd-u	1-wire	
300	RD-U	L	GPIOx2	Read	COLD	Synchronous Boot	
500	RD-U	L	GPIOx1	Read	rd-u	Basic	
600	RD-U	L	GPIOx1	Read	rd-u	Basic	
701	RDL-	U	SERDES	R/W	COLD	SERDES Boot	
705	RDL-	U	GPIOx4	Read	COLD	SPI Boot	
708	RDL-	U	GPIOx2	Read	COLD	Asynchronous Boot	
709	RDL-	U	Analog	Write*	rdl-	Analog	
713	RDL-	U	Analog	Write*	rdl-	Analog	Shared pin
715	RDL-	U	GPIOx1	Read	rdl-	Basic	
717	RD--	U	Analog	Write*	rd--	Analog	
617	RD-U	L	Analog	Write*	rd-u	Analog	Shared pin
517	RD-U	L	GPIOx1	Read	rd-u	Basic	
417	RD-U	L	GPIOx1	Read	rd-u	Basic	
317	RD-U	L	GPIOx1	Read	rd-u	Basic	
217	RD-U	L	GPIOx1	Read	rd-u	Basic	Shared pin
117	RD-U	L	Analog	Write*	rd-u	Analog	
009	RDL-	U	Parallel Bus	R/W	rdl-	SDRAM Addr	
008	RDL-	U	GPIOx4	Read	rdl-	SDRAM Ctl	Phantom pin group, special ROM
007	RDL-	U	Parallel Bus	R/W	rdl-	SDRAM Data	
105	RDLU			---	rdlu	eForth Bitsy	
106	RDLU			---	rdlu	eForth stack	Special ROM
107	RDLU			---	rdlu	SDRAM mux	
108	RDLU			---	rdlu	SDRAM idle	
000, 017, 700	RD--			---	rd--	Basic	
002, 003, 004, 005, 006, 010, 011, 012, 013, 014, 015, 016, 702, 703, 704, 706, 707, 710, 711, 712, 714, 716	RDL-			---	rdl-	Basic	
400	RD-U			---	rd-u	Basic	
101-04, 109-16, 201-16, 301-16, 401-16, 501-16, 601-16	RDLU			---	rdlu	Basic	

"Read" means that only reading is defined for the I/O direction; *writing is undefined* and should not be attempted.

"Write*", applicable to Analog nodes, means that the operation of writing to the register (which stops counter and does not change it) suspends and awakens on shared pin state, but that reading the register to get the counter value does not care about pin state.

3.4   Specially Configured Nodes

This section summarizes node groups with non-default interconnect, boot, or I/O behavior that may affect board design and firmware initialization. Review these group-specific notes before assigning pins or relying on reset-time behavior.

3.4.1   Bottom Parallel Bus Group

Nodes 007 and 009 can work with node 008 to support parallel I/O including parallel pipelines. 007 has a register at UP connected to 18 pins D00 through D17. 009 has a similar register connected to pins A00 through A17. Each register is configured for output or high impedance input using IO bit 12 which is written 1 for output mode. This is also (unfortunately) the reset state of that bit.

Node 008 has four GPIO pins as well as driving phantom pins. IO bit 7 is the phantom to node 009, and bit 9 is the phantom to node 007. Write 1 to these bits to make the phantom pin appear high to the receiving node. Nodes 7 and 9 are configured to suspend on write to UP. Unlike all other node types, reading UP does not suspend regardless of pin state. Neither suspends in any manner on DATA.

4 ROM Software

The ROM supplied in standard, off-the-shelf chips is documented but is subject to change in future revisions of this chip. When that occurs, the part number will be changed and inventories of the previous parts will not necessarily be maintained. The most likely changes will be improvements in boot code for efficiency or robustness, and revision of the external memory support code in the nodes near the bottom of the chip. The common ROM routines will not be changed gratuitously, in recognition of the fact that an application which must use them from ROM due to memory constraints will not be upward compatible if for example such a routine is removed from ROM in a later chip version.

The source code for these routines is available in the arrayForth distribution at the block numbers shown. The boot protocols are documented separately; only implementation specific details are documented here.

The following table lists common ROM functions and which are present in each ROM type as listed in 2.3.

			ROM Type						
Word	**Function**	**Block**	**Basic**	**Analog**	**SERDES Boot**	**Sync Boot**	**Async Boot**	**SPI Boot**	**1-wire**
warm	ROM a9 in all nodes contains jump to the appropriate multi-port execution addr		Yes	Yes	Yes	Yes	Yes	Yes	Yes
relay	Message passing tool	1388	Yes	Yes	Yes	Yes	Yes	Yes	
*.17	17 bit frac multiply	1390	Yes	Yes	Yes	Yes			Yes
*.	16 bit frac multiply	1396	Yes	Yes	Yes	Yes			Yes
triangle	Rev to triangle wave	1394	Yes	Yes	Yes	Yes			Yes
clc	Clear carry latch	1398	Yes	Yes	Yes				Yes
--u/mod	clc before –u/mod	1398	Yes	Yes	Yes				Yes
-u/mod	Uns div, divisor neg	1398	Yes	Yes	Yes				Yes
interp	Table interpolation	1384	Yes	Yes	Yes				
taps	IIR/FIR filter	1386	Yes		Yes	Yes			
poly	Polynomial approx	1382	Yes	Yes					
lsh	Variable left shift	1392					Yes		
rsh	Variable right shift	1392					Yes		
-dac	High res DAC drive	1434		Yes					
cold	Reset entry to boot routine	Defined in:			1420	1422	1424 1426	1428 1430	1436

4.1 Common ROM Definitions

The following common utility routines are present in the nodes indicated in the above table. If a definition uses more stack space than the greater of its arguments and results, the notation **-n-** indicates the peak number **n** of stack elements used.

warm Warm start. Placed at the same ROM address in all nodes, it jumps to the appropriate multiport execution address for each node, awaiting instructions from any adjacent neighbor node.

relay (a -4-) Moves a port executable packet down a sequence of nodes linked by their **b** registers. If **relay** is called from a port, the packet is read from the same port; if it is called from memory, the packet immediately follows that call. The packet

consists of a 1-cell index, a 1-cell count (less one) of body size, and the body cells. Relay assumes that all nodes in the chain have been set up with **b** pointing to the next node in the chain. Uses one return stack and four data stack elements. This word must be at the same address in every node that supports it.

*.17 (x y -3- x x*y) "star dot seventeen" Fractional multiply with 17-bit fractional part, bit 17 to left of binary point is simply twos complement sign. Thus the range of values supported in this format is [-1..1[although if it is essential to represent +1 this can be achieved at some expense with scaling. Multiplies two fractions, returning a fraction, or a value in some other scale including integer by a fraction, returning a result in that other scale.

*. (x y -3- x x*y) "star dot" Fractional multiply with 16-bit fractional part, so that there is one bit of integer part as well as a twos complement sign to the left of the binary point. In this scale the range of values supported is [-2..2[and +1 can be represented exactly in its natural form (x10000). Other than scale, usage is same as above for ***.17** .

triangle (n -3- f) This is a factor for approximating trig functions. Given an angle in revolutions that has been scaled to an 18 bit unsigned fraction, returns a 16-bit fraction representing a triangle wave whose value is in [-1..1] and whose phase is that of the cosine of the angle. This is suitable for use as an argument to approximations whose range is 1/2 cycle.

clc (-2-) "clear carry" Clears extended arithmetic mode (EAM) carry latch to zero. The address of this word has the EAM bit set.

--u/mod (h l d - r q) "full negative u-slash-mod" Unsigned division. **h** is the high 18 bits of the dividend. **l** is the low part. **d** is the negated divisor. **r** is the 18-bit remainder. **q** is the 18-bit quotient. Since **d** is negated, the actual divisor must be 17 rather than 18 bits. If the divisor is known at compile time then negate it and call **--u/mod** . If the divisor is variable, then first negate it, for example with **- 1 . +** before calling **--u/mod** . The address of this word has the EAM bit set.

-u/mod (h l d - r q) "negative u-slash-mod" An alternate entry point for unsigned division which may be used when the EAM carry latch is known to be zero. The address of this word has the EAM bit set.

interp (i m s -4- v) Performs linear interpolation using a table located at 0. The table contains 2n+1 entries (3, 5, 9, 17, 33, or 65.) **i** is the raw input to be scaled. **m** and **s** can be calculated from the number of meaningful bits in **i**, called L, and the power of 2 which gets you the number of entries in the table, called n. Set s=L-n-1 and m=$2^{(L-n)}$-1 . For example, say your ADC gives you an eight bit raw value and the scaled value ranges from 0 to 1800 mV. Let n=2 and the table has five entries: **0 org 0 , 450 , 900 , 1350 , 1800 ,**

and you could define mv as **63 5 interp ;** where m=2^{8-2}-1=2^6-1=63 and s=8-2-1=5 .

taps (y x c -4– y' x') Pushes a new sample through a table of **c**-1 entries which immediately follow the call to **taps** in memory, each entry consisting of a one word coefficient and a one word temporary storage, implementing a tapped delay line with summation of scaled taps. This may be used to implement various sorts of IIR or FIR filters. **x** is the new value coming into this section of the tapped delay line. The temp storage cells represent a simple delay line for the unaltered values of **x** so that after each cycle the new **x** is in the first temp, the first has been moved into the second, and so on. On exit, **x'** is the oldest raw value from the last temp. On entry, **y** may be zero unless there is feedback or multiple delay lines in series. On exit, **y'**

is the sum of products of the coefficient at each tap with the new data going into the temp at each tap. The incoming value **x** is multiplied by the first coefficient; the next older value is multiplied by the second coefficient, and so on.

poly (x n –3- x y) Evaluates a Chebyshev polynomial where x is the argument, a table of $n+2$ coefficients immediately follows the call to **poly** in memory, and y is the result.

lsh (x ns – x') "left shift" Shifts x left by $ns+1$ bits.

rsh (n ns – n') "right shift" Shifts n right arithmetically by $ns+1$ bits.

cold Cold start. Used as reset value for P on any node (such as boot nodes) that must execute some code for any reason immediately on reset.

4.2   Analog ROM

-dac (p a w -4- p) "hybrid DAC" Sources current onto a load controlling both duty cycle and maximum current. Each call to **-dac** results in a stimulus cycle whose period is proportional to $p+1$ during which current proportional to the value in register **A** is sourced for a high-time duration proportional to $w+1$. In addition, current proportional to a is sourced for a period of time on the order of 4 nS. The period and high-time are in units of **begin unext** cycles and the current sourced in both instances is determined by the DAC transfer function into the effective load resistance. −**dac** is designed to drive relatively high resolution signals onto a lossy integrator and must be tuned for the electrical characteristics, including time constant, of the load.

4.3   Boot ROMs

This section provides details about the implementation by each boot ROM of its protocol, *supplementing* the information in the boot protocol document identified in 1.2, Related Documents. The functions named and documented here are supported for upward source language compatibility. You are invited to study the code and its shadows in the arrayForth source, but dependency on the internals of boot code other than the word names and interfaces described below is strongly discouraged since internal improvements including redesign may appear in future chip revisions. In all cases, **cold** is the address to which **P** is reset for boot nodes.

4.3.1   SERDES Boot ROM

cold puts the SERDES hardware in read mode by storing x3141 (the address of **data**) into **A**, storing x3fffe to **data** , putting a copy of x3fffe in **T**, and executing the four-port address **rdlu** . This will execute incoming code on the right, down, and left neighbor ports or will execute an incoming SERDES stream. SERDES streams have special timing constraints because there is no flow control on this interface.

4.3.2   Synchronous Boot ROM

GPIO 17 is clock and GPIO 1 is data. **cold** reads data from the four-port address including wake-up on GPIO 17 high. If the clock pin is seen going high, the F18 makes a reasonableness test by measuring how long it stays high. If the line remains high for a period longer than 261510 cycles of a timing loop (≈4.1 mS), the booting effort is abandoned and the F18 reverts to **warm** , ignoring further pin activity. Otherwise it reads and processes a boot frame. **ser-exec** is the concatenation address for processing additional boot frames.

The present code will be changed in the next revision of this chip to use less power while booting and to be interruptible by com port activity. In addition, the present line protocol is

unsuitable for bidirectional communication and will be altered so that the clock line is low while the interface is at rest. A specification of the new protocol will be added to the published boot protocols manual once it has been tested.

4.3.3   Asynchronous Boot ROM

GPIO 17 is input and GPIO 1 is output. **cold** reads data from the four-port address including wake-up on GPIO 17 high. If the input pin is seen going high, signaling the leading edge of a possible start bit, the F18 starts timing its duration. If the line remains high for 262144 cycles of a timing loop ($\approx$4.1 mS), the booting effort is abandoned and the F18 reverts to **warm**. Otherwise it reads and processes a boot frame. **ser-exec** is the concatenation address for processing additional boot frames. Baud rates used should be from 9600bps to 1Mbps. The F18 ignores its comm ports throughout this process.

4.3.4   SPI Boot ROM

cold in the SPI boot ROM first checks if an SPI FLASH might be present by reading its GPIO 17. If this pin is strapped high it aborts this procedure and reverts to **warm** , leaving all four GPIO pins in weak pull-down mode.

If GPIO 17 is low, the F18 tries to read the first 18 bits at location zero in an external SPI flash device using a half clock delay constant of 497 ($\approx$5.1 μS/bit). This process takes 41 clock cycles or about 260 μS. The reasonableness test described in the **BOOT-xx** document is applied and if it fails the F18 reverts to **warm** with pins in the states shown in the table below.

When the ROM code operates on external flash, the signal assignments for its GPIO pins are defined as follows:

GPIO Pin	Signal	Meaning	Mode	State when read completed
705.17	DI	Data in from SPI device	Input	High impedance
705.5	DO	Data out to SPI device	Output	Driven low
705.3	CS-	Chip select (Enable-) to SPI device	Output	Driven low
705.1	SCK	Serial clock to SPI device	Output	Driven high

Note that if an SPI device is actually present, it will under these conditions be driving the DI line unless specific actions are taken by the boot stream as described in following paragraphs.

If the first word appears reasonable, indicating that a non-erased device is present, the F18 reads and processes a boot frame. **spi-exec** is the concatenation address for processing additional boot frames. The F18 ignores its comm ports throughout this process.

A boot stream may interact with the ROM by depositing code in RAM and executing it, for example by using a boot frame with transfer and completion addresses of zero with an appropriate transfer size. Here are some relevant examples:

<n> dup spi-exec ; changes the half clock delay constant to <n> for subsequent operations. On the f18a, each iteration is $\approx$5.1 nS so that with <n> of 0 each bit time is $\approx$10.2 nS (<98 MHz) and with a value of 1 each bit time is $\approx$20.4 nS for a clock frequency <49 MHz.

<n> dup select ...   terminates any current SPI operation by de-asserting chip select briefly using the delay constant <n> (497 for example). Lines are left in the completion state shown in the above table, however the SPI device should no longer be driving the DI line.

io b! 2002a !b 15555 !b warm ; may be used immediately after the above **select** phrase to drive all of the SPI node's pins to ground and then leave them in weak pull-down state. This is a recommended procedure for completing an SPI boot

sequence so that the SPI device is idle and all of the SPI pins are set correctly for a subsequent reset of the GA144 chip.

4.4 Other Special ROMs

Experimental code is supplied in several ROMs. This code is described here for information, but the code itself is subject to change without regard for upward compatibility.

4.4.1 1-wire ROM

Node 200 resets directly to **warm** rather than to the ROM. The 1-wire asynchronous boot protocol in ROM will load and execute standard boot frames. Words are encoded as eighteen serial bits where one bits are ≈11 nS pulses and zero bits are ≈4 nS pulses. A program can call **rcv** to read an 18-bit word sent in 1-wire format. A program to transmit 1-wire format may be executed from RAM after being loaded.

4.4.2 SDRAM and eForth ROMs

Nodes 007, 008, 009, 107 and 108 have ROM support for an external SDRAM using 18-bit data. Nodes 105 and 106 have ROM support for an 18-bit eForth virtual machine.

5 Operating Conditions

This section defines environmental and electrical limits for safe handling, qualification, and normal operation of the device. Use these limits together with the recommended operating ranges to set design margins and production test criteria.

5.1 Absolute Maximum Ratings[1]

Ratings apply over operating free-air temperature range (unless otherwise noted)

Voltage applied at any V_{DD} pin relative to V_{SS}	0 V to +2.3 V
Voltage applied to any I/O pin	-0.5 V to $V_{DD}IO$ + 0.5 V
Protection diode current at any device pin	±4 mA
Punch-through voltage defined by fabrication process	3.6 V
Storage temperature range, $T_{STORAGE}$[2]	-55°C to 150°C
Maximum junction temperature, T_J	150°C

(1)   Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability. It must be noted that in some cases these limits have not yet been explored empirically and that considering their sources they may be excessively conservative.

(2)   Higher temperatures may be applied during board soldering according to the current JEDEC J-STD-020 specification with peak reflow temperatures not higher than classified in section 6.2.1.

5.2 Thermal Packaging Characteristics

		Value	Units
Thermal resistance, junction-to-ambient, still air[1]	$R_{\theta JA}$	60	
	$R_{\theta JMA}$	20	°C/W
Thermal resistance, junction-to-case ($R_{\theta JC}$)		0.5	

(1) $36mm^2$ of thermal/ground pad area on PCB soldered to DAP.

5.3 Recommended Operating Conditions

		Min	Nom	Max	Units
V_{DD}	Supply voltage during operation including RAM writing ($V_{DD}C = V_{DD}IO = V_{DD}A$)[3]	1.62		2.0	V
V_{SS}	Supply ground		0		V
T_A	Operating free-air temperature[3]	-40		125	°C
T_J	Operating junction temperature[3]	-40		125	°C
T_{BURN}	Burn-in temperature			150	°C
$I_{SRC}IO$	I/O current sourced for sum of all I/O pins except nodes 709..717[1]			1680	mA
$I_{SINK}IO$	I/O current sunk for sum of all I/O pins	At 110°C		1680	mA
$I_{SRC}A$	Analog current sourced for sum of nodes 709..717			240	mA
$I_{TOTAL}C$	Continuous current used from sum of all VDDC power pins[1,2]			240	mA

(1)   These recommendations assume that all power pins are connected to suitable power sources. Note that driving all I/O pins to short circuit exceeds this value. Loads and duty cycles should be balanced to conform with this recommendation. Note also that max currents are specified at a die/junction temperature of 110°C. Multiply these currents by 2 at 85°C, and by 0.67 at 125°C.

(2)   Power during execution depends on instruction mixes and is therefore application dependent. If it is necessary to run large numbers of nodes at high duty cycles in high temperatures, we recommend measuring average power and junction temperature during operation to check conformity with this recommendation. See Application notes for measurement techniques.

(3)   Operating temperature range has not been qualified as of this revision. The values shown in orange are subject to change or qualification based on the results of that testing.

6 Electrical Characteristics

All measurements at $V_{DD} = 1.8V$ at nominal ambient free air temperature 22°C and operating die temperature <25°C unless otherwise noted.

In interpreting these data, it is important to remember that this is an asynchronous device. The speed at which circuits operate varies directly with supply voltage and inversely with temperature. The strengths, and therefore saturation currents, of individual transistors vary similarly. Thus at high temperatures the maximum drive current available from a pad driver will be less than it is at normal temperatures.

6.1 General

	Parameter	Test Conditions		Min	Typ	Max	Units
C_{CORE}	$V_{DD}C$ capacitance	$V_{DD}C$ to V_{SS}. $V_{DD} = 0V$			9722		pF
C_{IO}	$V_{DD}I$ capacitance	$V_{DD}I$ to V_{SS}. $V_{DD} = 0V$			336		pF
V_{CLAMP}[1]	Negative protection diode clamp voltage	$0.1\ \mu A$ across $10K\Omega$			-100		mV
			V_{DD}=2.0V		2.18		V
	Positive protection diode clamp voltage	$0.1\ \mu A$	V_{DD}=1.8V		1.98		V
			V_{DD}=1.6V		1.78		V
T_{RESET}	Duration of RESET- input low[2]				500		nS

(1)   Clamp voltage is measured as the protection diodes just begin to conduct, defined for the purpose of these measurements as the indicated current. Currents then increase rapidly as voltages depart further from the norm. Current limiting resistors are recommended when intentionally exposing pads to voltage sources above V_{DD} or below V_{SS}.

(2)   The times shown for RESET- reflects not only the propagation of the reset signal within the chip and the actions needed to restore the internals of the computers and of the I/O circuitry to their defined reset states, which is very short, but also reflects the time required to bleed the charge on an *unconnected* output pin to a safely low logic state. In practice this should generally be longer as discussed in later sections.

6.1.1 Power Supply Decoupling

Each power supply that is laid out distinctly should be given wideband decoupling treatment. A good place to start is multiple low-inductance $0.1\ \mu F$ capacitors in parallel to maximize C and minimize L. For systems whose primary noise source originates with the computer or I/O clock, the time honored practice of mixing capacitor values to create impedance nulls at the fundamental and harmonic frequencies of the highest energy fixed frequency noise sources was an effective method for power supply decoupling. However, while there may be noise at certain frequencies defined by the application, in asynchronous systems the character of the noise is wideband and calls for different practices. Multiple capacitor values create impedance peaks as well as nulls in the frequency domain and should be avoided.

6.2 Computers

	Parameter	Test Conditions	Min	Typ	Max	Units
$I_{TOTAL}C$	Core current for entire chip	All nodes running[1]	468	540	612	mA
		All nodes suspended	2	7	31	μA
$I_{YXX}C$	Core current for a single F18A	Running	3.25	3.75	4.25	mA
		Suspended	15	50	210	nA
$P_{TOTAL}C$	Core power for entire chip	All nodes running	842	972	1102	mW
		All nodes suspended	4	13	56	μW
$P_{YXX}C$	Core power for a single F18A	Running	5.9	6.8	7.7	mW
		Suspended	27	90	378	nW

(1)   Power used when running depends on instruction and data sequence and is therefore dependent on the application. Energy used depends on program duty cycle and is also application-dependent. The values shown are intended to represent a high estimate of typical programming at 100% duty cycle. Real applications typically use considerably less. Please see Application Notes for more information on this topic.

6.3   Typical Instruction Timings

Each F18 is an asynchronous computer running its proper, natural speed. This speed automatically and instantly adapts to the characteristics of each computer (such as fabrication process variation and accumulated stress "aging"), as well as the environmental variables to which that computer is being subjected (such as supply voltage and junction temperature). As these conditions vary, each computer runs at the best speed the conditions permit.

This table expresses a simplified timing model based on empirical data at VDD 1.8v and ambient temperature 22°C across the extremes of process variation. The model is intended to be adequate for estimating software performance. Effects of variables such as voltage and temperature may be predicted using the relationships shown in the following sections.

	Parameter	Test Conditions	Min	Typ	Max	Units
T_{SLOT}	Basic opcode execution time	+* 2* 2/ - + and or drop dup pop over a . push b! a!	1300	1400	1650	pS
T_{IOREG}	IO register read/write opcode time	@+ @b @ !+ !b !	3250	3500	4100	pS
T_{MEM}	RAM/ROM read/write opcode time	@+ @b @ !+ !b ! @p !p	4500	5000	5700	pS
T_{UNEXT}	Empty micronext time	begin unext	2150	2400	2750	pS
T_{NEXT}	Empty next time	begin next	4500	5000	5800	pS
T_{JUMP}	Non-empty next or jump time	next if -if (call) (jump) ; ex	4800	5200	6200	pS
$T_{PREFETCH}$	Minimum instruction fetch adder	Best case of prefetch	880	1200	1520	pS
T_{FETCH}	Maximum instruction fetch adder	Worst case, no prefetch	4000	4300	5000	pS

To estimate the time required to execute a routine, start at its first word of object code and add the timings of the opcodes in that word. If the word contains a jump, continue accumulating time at its destination word. If it does not contain a jump, estimate the time to fetch the next instruction and add it before accumulating the opcodes in the next word. Use $T_{PREFETCH}$ if there are no @ or ! opcodes in slots 1, 2 or 3; use T_{FETCH} if there is a @ or ! in slot 3; interpolate in 1000 pS steps in between as in this example:

@p . . . 1200 pS for next instruction fetch

. @p . . 2200 pS

. . @p . 3200 pS

. . . @p 4300 pS

Here is another example of timing estimation using this model (comment slashes indicate word boundaries):

mul nn-n a! 17 dup dup / or for / +* unext drop / a ;

.. 17 mul

The second line is a code fragment starting on a word boundary. The literal in slot 0 costs 5000 pS and the call to mul costs 5200 (slot 1 call) for a total so far of 10200 pS. The first word of mul has three basic opcodes (4200 pS) and a literal (5000) plus a fetch adder of 2200 for a total of 11400 pS. The second word has four basic opcodes (5600) and a prefetch adder of 1200 for a total of 6800 pS, 18200 pS so far for the definition. +* unext takes 1400+2400 per iteration (3800) times 18 iterations (68400 pS) and when it completes we have two opcodes (2800 pS) and a 2200 pS prefetch adder for a total of 73500 pS in that word and 91700 pS so far in the definition. In the final word of mul we have one basic opcode and one return (1400+5200=6600), giving a total of 98300 pS in mul. Adding the 10200 pS in the phrase that called mul we obtain a total time of 108500 pS, or 108.5 nanoseconds.

6.4 Variables Affecting Performance and Efficiency

The performance of asynchronous devices varies as a function of supply voltage, temperature and wafer fabrication process variables, as well as instruction sequences. This section documents the relationships based on measured data. ***While relationships are measured for micronext, they apply in the same ratios to the timings of all F18 instructions.***

6.4.1 Fabrication Process

Process variables are accounted for in the Min/Typical/Max values shown for measured parameters in all tables of this document. Measurements were taken on a sampling of chips from two wafers run at normal process settings and are the source for the "Typical" values in the tables.

Measurements were taken on a sampling of chips from each of eight wafers that were run in a qualification split lot. These eight wafers represented the extremes of process variables and in general supplied the min and max values for the tables. Transistor "aging" due to stresses is considered a process variable but is characterized separately later.

6.4.2 Supply Voltage Effects

Execution speed varies directly with supply voltage. Our benchmark measure for speed is the micronext loop (an instruction word with **unext** in slot 0.) A node is programmed to drive a pin high, execute 1000 cycles of micronext loop, and drive the pin low; the program generates only one pulse so that it has minimal effect on die temperature, a premise that has been validated by observation. We measure the width of this pulse in microseconds using a storage oscilloscope whose time base calibration has been checked against a precision crystal reference in an oven, giving agreement on time intervals within 0.1%. The resulting measurement gives micronext timing in nanoseconds, with a bias of less than 5 picoseconds for controlling the pin. The nominal width of this pulse is longer than 2 μS and the data are measured to 3 digits so that the timing measurements have a resolution of 10 pS and an accuracy better than 1%

The graph below shows micronext execution time as a function of V_{DD}. The data are normalized to a given node's performance at 1.8v. Our measurements show that this relationship holds across the investigated temperature range of -50 to +120°C.

T_{UNEXT} Execution Time vs V_{DD} any temperature

$y = -0.6775x^3 + 4.2904x^2 - 9.4878x + 8.1287$
$R^2 = 1$
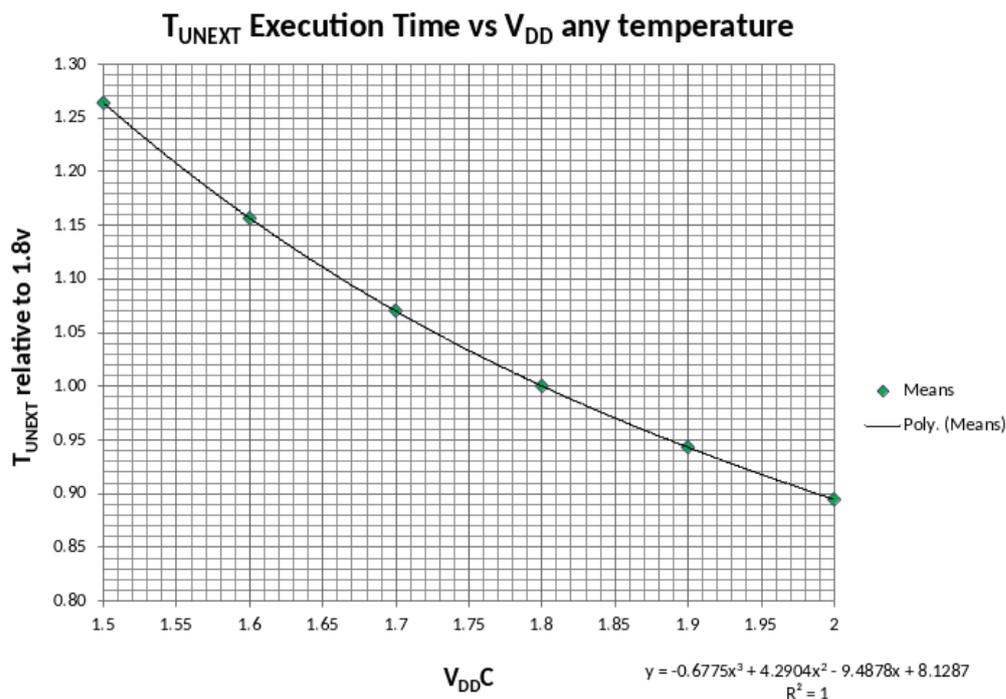
6.4.3   Temperature Effects

The benchmark pulse described above provides a very satisfactory and practical way to measure junction temperature under operational conditions. It is calibrated by completely suspending a chip so that it is dissipating only leakage power internally, taking an entire board to a given temperature in an environmental chamber, and generating a single pulse. We empirically verified the settling by checking that successive pulses did not change in width to the limit of our measurement resolution. The following graphs show the raw data for node 705 of a particular chip, and the same data normalized to 1.8v and 22°C yielding relationships that are independent of process variables.

To calibrate a particular node of a particular chip, let it reach ambient temperature while fully suspended and then generate and measure a benchmark pulse. Using the relationships above, interpolate the execution time back to what it would have been at 1.8v and 22°C. The pulse may subsequently be used on that chip to measure junction temperature. By measuring this temperature while the application is running and the chip soldered into your product, you may directly determine both T_J in operation and also, if your design permits measurement of power dissipation within the chip, you may directly determine thermal resistance. However since the goal of knowing thermal resistance is generally just to estimate T_J, it is simpler to measure it directly.

Efficiency Decreases with Temperature   Although execution time is basically a linear function of temperature, the amount of energy expended in executing a given instruction is not. The graph below shows data taken from node 705 of a single typical chip. The black trace shows execution time for micronext, normalized to 1.00 at this node's timing at 20°C (2.43 nS in this case.) The red trace shows power consumed during a continuous micronext loop on that same node, normalized to 1.00 at the same temperature (2.50 mW in this case.)

The blue trace shows energy required to execute a single micronext instruction, normalized to 1.00 at the same temperature; for this node, that energy is 6.08 picojoules (pJ) at 20°C. The data indicate that at high die temperatures the chip becomes less energy efficient. The message for programmers is that needless code execution not only wastes energy for the execution of that code, but also by increasing duty cycle increases power dissipated inside the chip; since the package thermal resistance is nonzero, any average power dissipated will translate into a temperature rise and thus decrease the efficiency of those instructions which are being executed.

The GreenArrays architecture makes it possible to dramatically reduce the energy required to perform useful work. Commitment, craftsmanship and attention to detail at all levels are rewarded by the realization of these economies.

Leakage Increases with Temperature   The low leakage of the GA144, typically on the order of 10 μA at room temperature when all nodes are suspended, varies with die temperature as well and can become much more significant when ambient temperature, internal duty cycle, and thermal resistance combine to heat up the die. The curve below was measured on one typical chip with all nodes suspended and I/O current minimized. The values for -40 and -50°C are shown as 100 nA; although the actual values are less than that, they were less than the reliable measurement range of the apparatus in use. "Initial" and "Later" refer to leakage measurements taken before and after checking performance; leakage keeps changing after performance has stabilized, suggesting that these differences might pertain to accumulation of charges in gate oxides rather than to continuing change of die temperature.

The relationship shown above is consistent with normal models for subthreshold leakage.

6.4.4 Instruction and Data Mix

The pay-off from careful programming can be substantial, and it can be very helpful to have equipment handy for measuring small currents during software development. In addition to the usual programming considerations for minimizing execution time and thus duty cycle, the energy consumed in executing a given instruction in the F18 is very sensitive to the data being processed. For example, changing T, or T and S, can expend considerable energy since there are bit buses to be charged and discharged as well as ALU gates which are exercised when these registers change. As one example, here is the loop in which one async receive algorithm waits for the leading edge of a start bit:

wait (x) begin . drop @b −until . drop ;

This loop reads io repeatedly until its high order bit is 1. On one chip this loop burns an average of 4.8 to 6.51 mW, depending on what's on the stack! If the stack has, on entry, three copies of a value very close to the rest state of io for that node, the least power is used. If these values are simply the inverse of io then the loop burns an additional milliamp. Attention to detail can pay off with 1.7 less milliwatt used, or one joule of battery charge saved every 10 minutes of loop operation.

6.5 I/O Circuits

Although the F18A I/O pads share a common design, there are some variations. This section characterizes the common GPIO pin and then shows details that differ from the GPIO model for each specialized type.

6.5.1 GPIO Pins

The following measurements apply to all I/O pins of all types, including RESET-, unless otherwise noted in later sections pertaining to more specialized types of pins. GPIO pins are reset to weak pull-down mode unless otherwise noted.

	Parameter	Test Conditions	Min	Typ	Max	Units
C_{IN}	Input capacitance	Pin to V_{SS}, V_{DD} = 0V		2.8		pF
I_{LKG}	High-impedance leakage current	Pin to V_{DD}, V_{DD} = 1.8V	2.4	3.5	5.5	nA
		Pin to V_{SS}, V_{DD} = 1.8V	1.1	1.5	5.6	
R_{LKG}	Effective high-impedance input resistance at maximum I_{LKG}	Same as I_{LKG}	320			MΩ
I_{WPD}	Weak pull-down current (in saturation)	Pin to V_{DD}	30	36	43	μA
R_{WPD}	Effective input resistance at I_{WPD}	Same as I_{WPD}	60	50	42	KΩ
V_{IT+}	Positive-going Schmitt Trigger input threshold voltage	V_{DD} = 1.6V	0.94	1.00	1.04	V
		V_{DD} = 1.8V	1.04	1.10	1.16	V
		V_{DD} = 2.0V	1.15	1.21	1.27	V
V_{IT-}	Negative-going Schmitt Trigger input threshold voltage	V_{DD} = 1.6V	0.55	0.61	0.65	V
		V_{DD} = 1.8V	0.62	0.68	0.74	V
		V_{DD} = 2.0V	0.70	0.76	0.83	V
V_{HYS}	Schmitt Trigger input voltage hysteresis (V_{IT+} - V_{IT-})	V_{DD} = 1.6V	350	390	420	mV
		V_{DD} = 1.8V	380	430	470	mV
		V_{DD} = 2.0V	400	460	500	mV
V_{IHOPT}	High- and low-level input voltages for minimal input circuit power[1]		V_{DD}/2+0.6			V
V_{ILOPT}			0.200			V
T_{INW}	Input pulse width for wakeup[2]	Threshold to threshold	1			nS
T_{INR}	Input pulse width for identification[3]		8			nS
$SLEW_{IN}$	Input signal slew rate[4]		1			V/S
V_{OH}	Output high at 10 mA source current	Pin to V_{SS}	V_{DD} = 1.6V	1.37	1.43	1.45	V
			V_{DD} = 1.8V	1.59	1.65	1.67	V
			V_{DD} = 2.0V	1.82	1.85	1.88	V
V_{OL}	Output low at 10 mA sink current	Pin to V_{DD}	V_{DD} = 1.6V	0.125	0.135	0.240	V
			V_{DD} = 1.8V	0.110	0.125	0.270	V
			V_{DD} = 2.0V	0.105	0.115	0.190	V
I_{SH}	Max current sourced (in saturation)	Pin to V_{SS}	36	42	49	mA
I_{SL}	Max current sunk (in saturation)	Pin to V_{DD}	36	40	44	mA

	Parameter	Test Conditions	Min	Typ	Max	Units
R_{OH}	Output source res. for 0.5 V_{DD} at pin	Pin to V_{SS}		23.5	25.1	Ω
R_{OL}	Output sink resistance for 0.5 V_{DD} at pin	Pin to V_{DD}		21.5		Ω

(1)   Input circuits begin consuming measurable power between $\approx$0.450 and $\approx$1.400 V at VDD=1.8V. At steady state input voltages near V_{IT+} and $V_{IT\text{-}}$ the worst case can be on the order of 120 μA per pin.

(2)   Wakeup state duration required for consistent wakeup behavior.

(3)   Wakeup state duration required if IO register read required immediately after wakeup to identify wakeup source.

(4)   Slow moving inputs are permitted on GPIO input pins. However, avoid lingering in the vicinity of V_{IT+} or $V_{IT\text{-}}$ particularly if high-frequency noise is superimposed on the input signal. The greater the noise, the more rapidly inputs should slew.

6.5.2   RESET- Pin

This pin is electrically identical with a GPIO pin except that it does not have output circuitry nor weak pull-down. It must be presented at all times with a suitable voltage source in order for the chip to operate.

6.5.3   Reset conditions and considerations

The reset conditions of GPIO and other pins reflect a judgment based on compromise. Systems should be designed to fit this compromise where possible. When there are exceptions, the most practical solution supported by this chip is to rapidly boot a stream from SPI memory that sets I/O circuits as desired as first thing before booting application code.

Note that the duration of RESET- assertion should include at least three time constants of the largest RC load that is attached to an output pin whose state should be low after reset, when it is being bled to ground by the weak pull-down devices that are enabled at the start of RESET- assertion.

6.5.4   SPI Node 705

As noted earlier, this node must be treated carefully in system design because, if it finds 705.17 low on reset, the boot ROM will go through the protocol for reading a word from SPI flash, at low speed, for a reasonableness check. If the value appears reasonable then at least one boot frame will be read and processed at low speed before there is any opportunity for that node to pay attention to any other node. If there is a flash in the circuit, a jumper or other provision for forcing 705.17 high is advisable to provide a recovery path for garbage in the flash; if there is no flash then it may be desirable to force 705.17 high both to avoid driving spurious signals onto the other three pins of 705 and to avoid having the node spend time reading the nonexistent device. A 10K pull-up resistor is not a bad compromise.

Note also that, if the flash is in the process of executing a read operation when the GA144 is reset, the read operation is not automatically terminated unless the flash is also reset. If the flash is driving its data line high at the time, this line will be seen high on 705.17 after reset and node 705 will not attempt to boot from the flash at all. If any of these cases will compromise your system's integrity, the system will need to sequence its reset signals so that the flash has been completely reset before the GA144 is taken out of reset. Note that the time required for some SPI devices to complete a reset can be significant.

6.5.5   Asynchronous Boot Node 708

After reset and until it has been programmed otherwise, node 708 will react to a high level on 708.17 by applying reasonableness checks for asynchronous boot input and attempting to boot if the signals appear reasonable. This process can consume considerable time during which the node will not respond to stimuli on its comm ports. Avoid connecting 708.17 to non-boot devices that are capable of raising this line before node 708 has been programmed.

6.5.6   Synchronous Boot Node 300

After reset and until it has been programmed otherwise, node 300 will react to a high level on 300.17 by applying reasonableness checks for synchronous boot input and attempting to boot if the signals appear reasonable. This process can consume considerable time during which the node will not respond to stimuli on its comm ports. Avoid connecting 300.17 to non-boot devices that are capable of raising this line before node 300 has been programmed.

6.5.7   Parallel (bus) Nodes

Parallel bus nodes have only two modes of operation: All 18 lines are driven, or all 18 lines are high impedance inputs. There are no weak pull-down devices. Parallel bus pins reset to output mode with unspecified data.

	Parameter	Test Conditions	Min	Typ	Max	Units
V_{IH}	Input high level[1]			1.1		V
V_{IL}	Input low level			0.650		V
V_{IHOPT}	High- and low-level input voltages for minimal input circuit power[2]	$V_{DD}/2+0.6$				V
V_{ILOPT}				0.200		V

(1)   Parallel bus inputs have no hysteresis. Signals should not be read during transition, and actual levels should leave adequate noise margins relative to V_{IH} and V_{IL}. These ports are designed for efficient transmission and reception of well synchronized data only, and not for the reception of asynchronous signals. The voltages shown have 50 mV margins across process variation and V_{DD} range.

(2)   These input circuits begin consuming measurable power between $\approx$0.450 and $\approx$1.400 V at VDD=1.8V. At steady state input voltage around $V_{DD}/2$ the worst case can be on the order of 150 μA per pin.

6.5.8   Analog Nodes

Each analog node has an input and an output pin. Their electrical characteristics differ significantly:

Analog Inputs Analog input pins are always high impedance; if not driven they can float and their input structures can consume measurable power when their voltages are not at VDD or VSS. To minimize power, terminate all unused analog inputs suitably.

	Parameter	Test Conditions	Min	Typ	Max	Units
C_{IN}	Input capacitance	Pin to V_{SS}, $V_{DD} = 0V$		2.8		pF
I_{LKG}	High-impedance leakage current	Pin to V_{DD}, $V_{DD} = 1.8V$	2.4	3.5	5.5	nA
		Pin to V_{SS}, $V_{DD} = 1.8V$	1.1	1.5	5.6	
R_{LKG}	Effective high-impedance input resistance at maximum I_{LKG}	Same as I_{LKG}	320			MΩ

Analog Outputs   Analog output pins are only capable of acting as current sources and are reset to high impedance (the equivalent of 0V). The Analog outputs are designed to be terminated by a resistance to ground.

	Parameter	Test Conditions	Min	Typ	Max	Units
C_{OUT}	Pin capacitance	Pin to V_{SS}, $V_{DD} = 0V$		2.8		pF
I_{LKG}	High-impedance leakage current	Pin to V_{DD}, output 0 Pin to V_{SS}, output 0	2.4 1.1	3.5 1.5	5.5 5.6	nA
R_{LKG}	Effective high-impedance pin resistance at maximum I_{LKG}	Same as I_{LKG}	320			MΩ
I_{SMAX}	Max current sourced (in saturation)	Pin to V_{SS}, output x1FF	20	24	28	mA
I_{SMIN}	Min current sourced (in saturation)	Pin to V_{SS}, output 1		46		μA

6.5.9   SERDES

|	Parameter	Test Conditions	Min	Typ	Max	Units

7 Mechanical Dimensions and Package Data

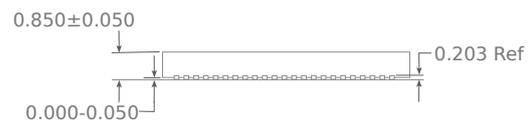## 7.1 Package Description (10 x 10 mm QFN-88)

This chip is available in a plastic-molded Quad Flat No-lead (QFN) package measuring 10x10 mm and with a total of 88 pins (22 per side) at 0.4mm pitch. The exposed central pad (DAP) on the bottom of the package is also used as the single grounding point for the entire chip. Dimensions, in mm, are as follows:
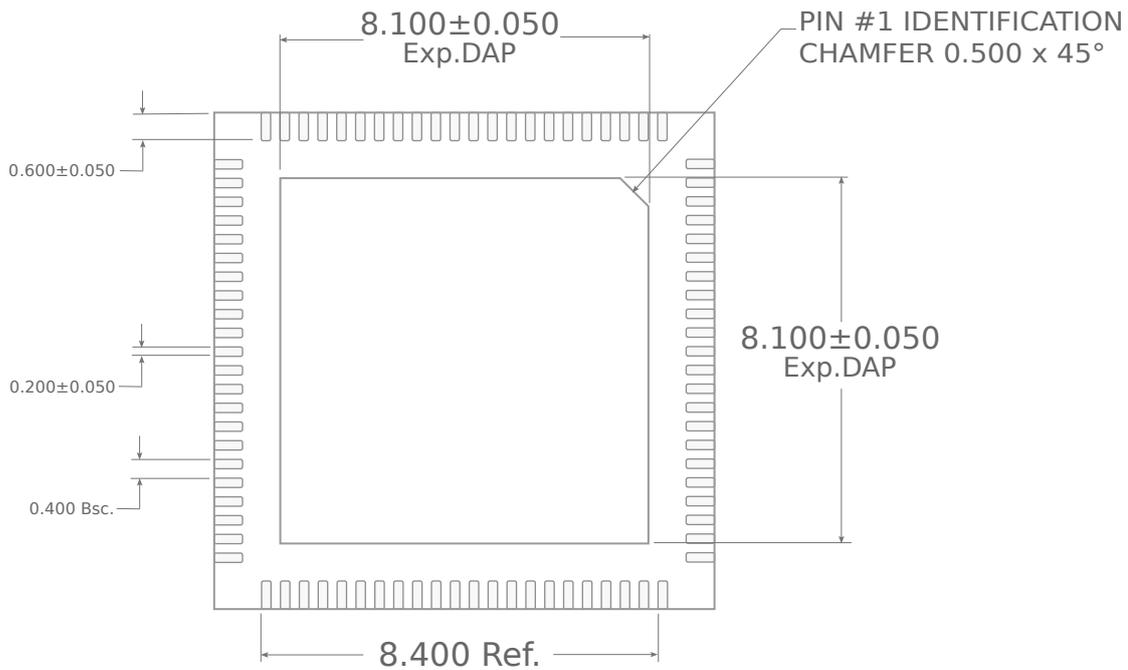
Bottom View
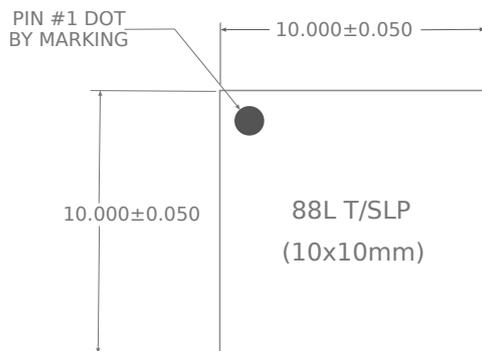
Top View

Side View

7.1.1 Migration Path

We expect to refine this package in the future, and that will most likely result in shrinkage of the DAP to approach 6x6 mm. It is recommended that PCB layout limit metal pads for soldering to the central 6x6 mm area of the chip footprint for maximum compatibility with future packages.

7.1.2 PCB Layout and Assembly Considerations

Assembly guidance published by the package manufacturer (Amkor) is applicable. For DAP soldering, control paste volume to avoid package floating and solder bleed.

Practical experience has shown that signal traces and vias may be placed beneath the DAP by providing a solder pad smaller than the DAP and solder masking the remainder of the DAP area under which signal traces may be routed.

7.2 Physical Data

A wafer of chips assembled as a qualification lot met the following criteria: wire pull strength >3g, ball shear >12g, and die shear >13.4 Kg. Intermetallic coverage is >60%. Solderability testing shows coverage >95% at solder temperature of 245±5°C with steam aging at 93±3°C for 8 hours. RoHS/WEEE status shall be based on current supplier declarations and internal compliance records for the released lot.

7.2.1 Peak Reflow Temperature

Peak package body temperature is nominally 260°C. Check "moisture sensitive devices" label on each shipment of devices for definitive data.

7.2.2 ESD Sensitivity

Prototype parts have been tested to 1 kV HBM. Handle devices using standard ESD controls and implement system-level ESD protection as required by the end application.

7.2.3 Moisture

The package is rated by its manufacturer at IPC/JEDEC J-STD-020 Moisture Sensitivity Level (MSL) 3, level defined in EIA JEDEC Standard JESC122-A112, indicating floor life out of the bag of 168 hours. Handling and storage controls shall follow current J-STD-033 practices. Parts ordered in trays sealed in moisture barrier bags (when available) are suitable for reflow without baking during the specified floor-life period. Parts exposed to room air during test/repack operations should be baked (48 hours at 125±5°C) before reflow soldering.

7.2.4 Pressure

This package has not been qualified for extreme pressure or vacuum operation. For vacuum exposure, moisture-control and bake procedures are required before assembly processing.

7.2.5 Vibration

Internal components and bondwires are embedded in the molded package body. Application-specific vibration and acceleration validation remains the responsibility of the end-product integrator.

7.2.6 Materials

The leadframe is plated with 300-800 μinches of tin. The package body is a compound designated G770HP. Bondwires are gold.

8 Part Identification

8.1 Part Numbers for Ordering

Order Number	# Chips	Description
G144A12-PAK10	10	Manufacture pack of 10 chips. Previously GA144120PAK10

8.2 Part Labeling and Key

The top of each chip is labeled as shown:

GA144f18a is the generic product name.

G144A12 is part number including revision and any options or customizations. **CUXQIG** is internal process ID.

SGA047058 is fab wafer lot number; **1** is sub-lot if any; 0**9** is wafer number. Sub-lot or wafer numbers may be omitted.

1106 is year and week numbers of assembly, and **CN** is the ISO abbreviation for the country in which the chip was assembled.

9 Publication Notes

9.1 Change Control

Each released revision of this datasheet shall include:

- updated revision date and revision identifier

- summary of technical changes affecting limits, timing, pin behavior, package data, or ordering codes

- impact statement for backward compatibility and qualification status

9.2 Legal and Application Disclaimer

This document is provided for engineering guidance. The purchaser is responsible for validating suitability in the intended end application, including safety, regulatory, and reliability requirements. Product behavior outside the stated operating conditions is not warranted by this datasheet.

9.3 Contact and Latest Revision

For latest revisions, errata, and application notes, refer to https://cartheur.com.