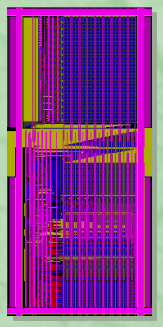


Actual size

241 µm x 523 µm

Green Arrays™ F18A

18-bit Computer

FEATURES

- Small size, low cost
- Fast, low energy design consumes only as much energy as is needed to do the programmed work. There is no cost per MIP, only per unit work
- Self-contained memory and stacks
- Port and pin wakeup
- Simple, compact instruction set
- Memory size balanced for high speed
- Customizable ROM
- Executable communication ports
- Multiport operations & port execution

SIGNATURE F18 PROPERTIES

- Dual stack (Forth) architecture
- Fetches up to 4 instructions per word
- Tight loops without instruction fetches
- Bidirectional data during port execution
- Inherent cooperation between nodes
- Roles of nodes determined by software

APPLICATIONS

- Software defined I/O controllers
- Simple MAC and PHY components
- Node groups control complex devices such as SDRAM or implement complex algorithms such as cryptography
- Networks of nodes cooperating to perform parallel and/or pipelined data or signal processing
- Networks of nodes cooperating to simulate analog function blocks for control or DSP applications
- Simulation, singly or in networks
- Interpretation of higher level languages for applications or parts of applications where bulk is justified
- May be provided in hard IP macros for incorporation in customers' ASIC, SOC, or wafer scale integrated designs.
- Featured in GreenArrays G144A12

OVERVIEW

The F18A is a stable, mature design for a computer and its I/O whose robustness has been proven in many chip configurations. It has been proven in 180nm geometry, and a prototype in 130 nm has also performed well. The computer is small; eight fit in roughly a square millimeter. Depending on chip configurations, this yields between 100,000 and 200,000 computers per 8 inch wafer, contributing to the low cost of our chips.

FAST, LOW-ENERGY DESIGN: Our ruthless commitment to simplicity is reinforced by design tools that force our engineers to confront and deal with the speed and energy costs of each design feature at every stage of the layout and simulation process. We continually strive to minimize internal loading and circuitry, always willing to reinvent not only inefficient conventional designs, but also any part of our own. The F18A is an unclocked, fully asynchronous computer which can execute basic instructions in ≈1.5 nanoseconds without instruction fetches, and sustained rates of ≈1.8 nanoseconds when instruction words are fetched inline from RAM, with Vdd at 1.8V. The energy required to execute a basic instruction is on the order of 7 picojoules (pJ).

SELF CONTAINED MEMORY AND STACKS: Each F18A contains 128 words of memory (up to 512 instructions) plus 20 words of stacks and registers; no memory bottlenecks. 18-bit registers are S and T, the top two elements of the data stack; R, the top element of the return stack; and A, a read/write register. Address register B has 9 bits, and the program counter P has 10.

PORT AND PIN WAKEUP: An F18A suspends execution in mid-instruction, using only leakage (≈100 nW) when waiting to move data to or from another computer or when waiting for a pin to change state. Suspension and resumption take no time or energy.

INSTRUCTION SET: An 18-bit word is subdivided into four slots, each of which may contain a five-bit operation code (only codes banded in green may be used in slot 3):

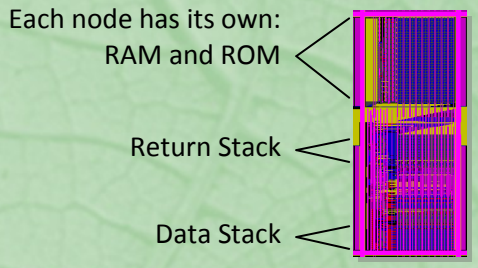
SLOT 0					SLOT 1					SLOT 2					SLOT 3		
17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Opcode	Notes	Opcode	Notes
;	return	+*	multiply step
ex	execute (swap P and R)	2*	left shift
name ;	jump to name	2/	right shift (signed)
name	call to name	-	invert all bits
unext	loop in I (decr R)	+	add (or add with carry)
next	loop (decr R)	and	
if	jump if T=0	or	exclusive or
-if	jump if T>=0	drop	
@p	literal (auto incr)	dup	
@+	fetch via A (auto incr)	pop	from R to T
@b	fetch via B	over	
@	fetch via A	a	fetch from register A
!p	store via P (auto incr)	.	nop
!+	store via A (auto incr)	push	from T to R
!b	store via B	b!	store to register B
!	store via A	a!	store to register A

Add (+) becomes add with carry when bit 9 of program counter P is set. The destinations of jump, next and call in slots 0, 1, or 2 are combinations of existing P value with the bits (13, 8, or 3) in the remaining slots of the instruction word.

Green Arrays™ F18A

18-bit Computer



ADDRESS MAP: Memory and registers are addressed with nine bits in the F18A, according to the map shown here. When auto incrementing reads or writes are done using P (instruction fetch @p !p) or A (@+ !+), incrementation of an address in RAM wraps to zero after 07F; of an address in ROM wraps to 080 after 0FF; and addresses in the I/O port and register area *are not incremented at all*. Address decoding in the latter area has a value for the register **io** (see F18A I/O documentation), for each of five communication ports (one of which may refer to an I/O pin or special register), and for all possible combinations of those five ports as well as a bit to allow reading or writing a special I/O register without waiting for pin wakeup.

Start	End	Address Space
000	03F	RAM, 64 words
040	07F	RAM 000-03F repeated
080	0BF	ROM, 64 words
0C0	0FF	ROM 080-0BF repeated
100	1FF	I/O ports and registers

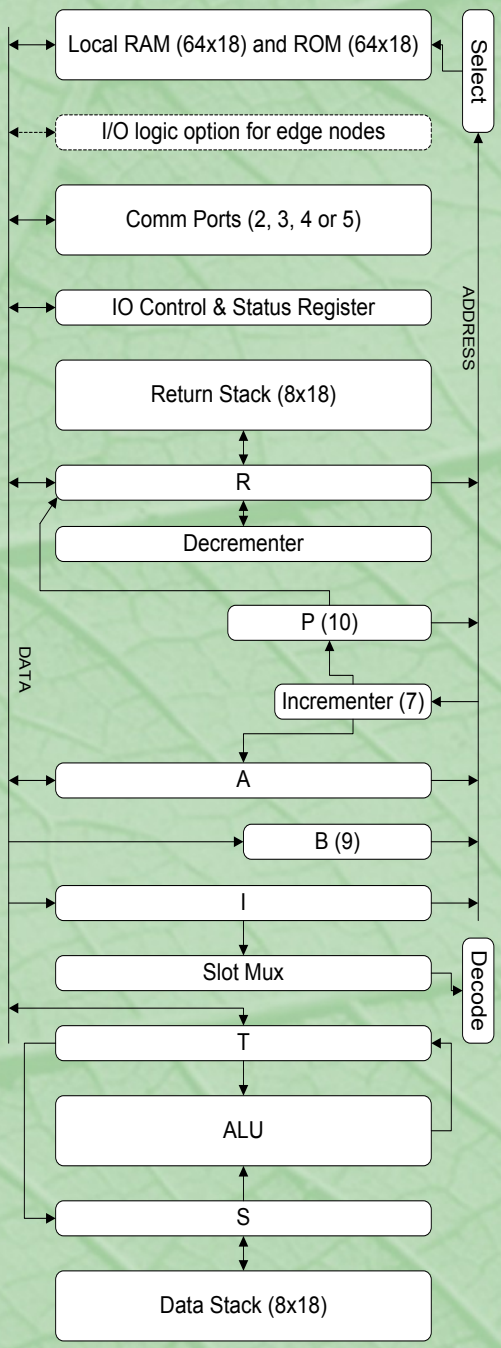
EXTENDED PRECISION INSTRUCTION: When executing code with bit 9 of P set to 1, the add instruction is interpreted as add with carry. In this case, and only in this case, a latched carry bit is added to the low order bit of the sum, and the carry out of the 18-bit ALU is latched at the end of the operation. The latch is neither used nor changed while P9 is zero. P9 has no effect on addressing as such.

ROM CONTENTS: Each F18A can be built with unique ROM contents, and code written for a particular GA part may use routines in ROM to conserve RAM space. We are able to build customers' application code into ROM for purposes such as creation of ASICs or barriers to competition.

COMMUNICATION PORTS: Data move between nodes across these 18-bit parallel ports. Handshaking is automatic, requiring no code and little time. When one node is writing a given port at the same time as the other node sharing that port is reading it, data move between the nodes. Until that occurs, the node which reads or writes first is suspended until the other writes or reads to satisfy the operation.

MULTIPORT OPERATIONS: Each port is addressed by a single bit. These can be combined in an address to read or write on multiple ports in a single operation. When the desired operation is satisfied by another node sharing any of the selected ports, the operation completes. Among many other things, this facilitates efficiently spreading a serial data stream among multiple paths in an array of nodes programmed for parallel, pipelined processing.

PORT EXECUTION: A definitive feature of the GreenArrays architecture is the ability of a computer to directly execute instructions "fed" to it via a communication port, with profound positive results on chip capability and communication efficiency. For communication, there is no need to decode or parse a message when the message itself may be directly executed. This saves space, time, and energy. A node may use the resources of a neighbor without requiring any code in its memory; thus for example a neighbor may be used as a 64 word RAM resource. Interactive debugging may be done without devoting any space in RAM or ROM for supporting that capability.



Each F18A COMPUTER

For more information, visit www.GreenArrayChips.com

GreenArrays, Inc.
774 Mays Blvd #10 PMB 320
Incline Village, NV 89451

(775) 298-4748 voice
(775) 548-8547 fax
sales@GreenArrayChips.com