Hewlett-Packard Company
8050 Foothills Boulevard
Roseville, California 95747
916 / 786-8000

HEWLETT PACKARD

TITLE: Hardware Internal Design Specification For the hp-71

PART NUMBER: 00071-90071

MICROFICHE: N/A

PRINT DATE: N/A

UPDATE:

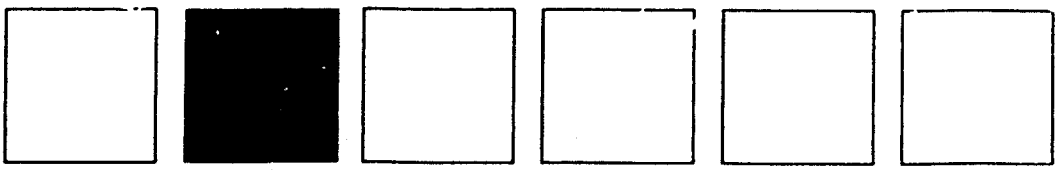
PRINTED IN THE U.S.A.

The product related to this manual is no longer in production at the Hewlett-Packard Corporation. The manual is maintained on a microfiche master by Hewlett-Packard Company. As a service to our customers we are providing a hardcopy print of the microfiche. The print is produced using a Microfiche Printing System. In addition, we are providing a duplicate of the microfiche to provide maximum flexibility for our customers.

HEWLETT PACKARD

Hardware Internal Design Specification

For the HP-71



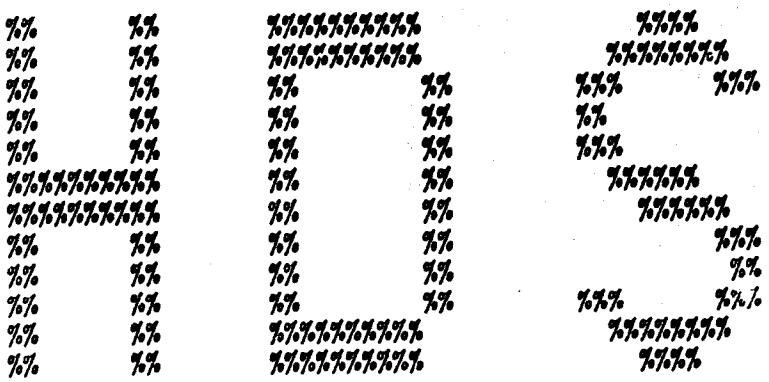
Hewlett-Packard -- Portable Computer Division

Corvallis, Oregon

HP-71 HARDWARE

INTERNAL DESIGN SPECIFICATION

Detailed Design Description



SEPTEMBER 1984

HP Part No. 00071-90071

Table of Contents

SYSTEM OVERVIEW — CHAPTER 1

The HP-71 mainframe chip set is composed of the 4 custom CMOS integrated circuits shown below:

CHIP SET:

1	1LF2 CPU	
3	1LF3 Display Driver	(@ .5K bytes RAM/chip)
1	1LG7 ROM hybrid	(@ 4 chips/hybrid and 16K bytes ROM/chip)
4	1LG8 RAM hybrids	(@ 4 chips/hybrid and 1K bytes RAM/chip)

The CPU and display drivers are mounted on the underside of the keyboard/LCD PC board. The ROM hybrid and 4 RAM hybrids are mounted on the I/O PC board. Connection to the 4 RAM/ROM ports in the front of the HP-71 and the HP-IL port in the back of the machine is made via the I/O PCB. The card reader port connects to the keyboard PC board.

The CPU (1LF2) is an arithmetic oriented microprocessor capable of both HEX and BCD arithmetic. The 1LF2 operates on a fully multiplexed 4-bit bus with a 512K byte address space and an internal word size of 64 bits. The CPU has a general purpose Input Register (16 lines) with interrupt capability and a general purpose Output Register (12 lines). The Input and Output Registers are used to form the HP-71 keyboard matrix. The Output Register is also used to drive the daisy-chain signal to each port and to drive the piezo-electric beeper used in the system. External circuitry provides the capability of loud or soft beep.

The three display drivers (1LF3) drive the 8 row by 136 column (including annunciators) display. One of these 3 1LF3s acts as the master. The master generates the display clock, the voltage reference, and the display-on signal used by the other two display chips (slaves). There is a 4-bit writeable register on the master that controls the display contrast by adjusting the value of the voltage reference signal. Each display chip supplies .5K bytes of hard-configured RAM and a 24-bit timer (crystal-oscillator controlled) with 1/512 second resolution. The timers are used by the operating system to implement the clock system. The master display chip also provides a low supply voltage sensor that is used by the operating system to indicate low-battery.

Each 1LG8 chip provides 1K bytes of soft configured RAM. The four 4-chip RAM hybrids in combination with the 1.5K bytes of display driver RAM provide a system total of 17.5K bytes. Plug-in RAM modules contain one 4-chip hybrid totalling 4K bytes of RAM. Each 1LG7 chip provides 16K bytes of ROM. The 64K operating system is stored in one 4-chip ROM hybrid that is hard addressed beginning at 00000. Plug-in ROM modules contain 1 to 4 hard or soft configured ROMs. For information on hard and soft address configuration see section 2.2.

The HP-71 system supports a low-power shutdown mode. The system can be shutdown with the display either on or off. For information on power down or wake up see section 2.4.

The system is powered from either the unregulated battery voltage or the regulated AC adapter voltage. A 470uF capacitor supplies "keep-alive" power to the system while batteries are being replaced with the system in shutdown mode.

BUS COMMUNICATION — CHAPTER 2

This section describes the HP-71 system bus structure, protocol, and timing. Also included is a description of the bus commands, device addressing, and the power down and wake up characteristics.

2.1 Bus Structure

The HP-71 bus consists of 8 lines including:

- 4 BUS[0:3] - data lines, driven by the CPU or system devices.
- 1 *STROBE (*STR) - driven by the CPU.
- 1 *COMMAND-DATA (*CD) - usually driven by the CPU.
- 1 VDD
- 1 GROUND

Plug-ins may have two additional lines:

- 1 DAISYIN - (DIN) input to device
- 1 DAISYOUT - (DOUT) output from device, may be tied to next device's DIN

NOTE : A '*' before a signal name denotes negative true logic.

2.1.1 General Protocol

*STR is driven by the CPU and serves to synchronize all bus transfers. It is not a true system clock since it can remain inactive for several cycles while the CPU is performing internal manipulations. All address, data, and bus commands are transferred on the BUS[0:3]. The BUS[0:3] is driven during *STR low by either the CPU or the device the CPU is accessing. This data is latched either by the CPU during *STR low or by the receiving device on the rising edge of *STR.

All bus operations are initiated by the CPU. The CPU starts a specific transfer on the bus by driving the *CD line low before *STR goes low. While *CD and *STR are low the CPU drives a bus command on the BUS[0:3] and all devices in the system latch the command on the rising edge of *STR. This strobe is referred to as

a command strobe. The bus command issued during a command strobe specifies the operation that is to be performed on each succeeding *STR until another bus command is issued. At all times when data or address is being transferred *CD is held high. A strobe issued while *CD is high is referred to as a a data strobe.

2.1.2 Bus C mands

The bus commands are:

0	NOP	All devices ignore *STR until a new command is loaded.
1	ID	The unconfigured device that has its DAISY-IN high sends its 5-nibble ID on the following data strobes, starting with the low- order nibble of the ID.
2	PC READ	(PC)->BUS or read using the Program Counter (PC). The device addressed by its program counter sends data pointed to by its local program counter on each following data strobe and all devices increment their local program counters once each data strobe. A dummy strobe immediately follows the issuance of this bus command (see subsection 2.1.4).
3	DP READ	(DP)->BUS or read using the Data Pointer (DP). The device addressed by its data pointer sends data pointed to by its local data pointer on each following data strobe and all devices increment their local data pointers once each data strobe. A dummy strobe immediately follows the issuance of this bus command (see subsection 2.1.4).
4	PC WRITE	BUS->(PC) or write using PC. The device addressed by its program counter loads the data on the following data strobes into the location pointed to by its local program counter and all devices increment their local program counter once each data strobe.
5	DP WRITE	BUS->(DP) or write using DP. The device addressed by its data pointer loads the data on the following data strobes into the location pointed to by its local data pointer and all devices increment their local data pointer once each data strobe.

6	LOAD PC	BUS->PC or load PC. All devices load the data on following 5 data strobes into their local program counter, starting with the low-order nibble. After all 5 nibbles are transferred the command code is automatically changed to a 2, PC READ (see subsection 2.1.3).
7	LOAD DP	BUS->DP or load DP. All devices load the data on following 5 data strobes into their local data pointer, starting with the low-order nibble. After all 5 nibbles are transferred the command code is automatically changed to a 3, DP READ (see subsection 2.1.3).
8	CONFIGURE	The unconfigured device that has its DAISY-IN high loads the following 5 data nibbles into its configuration register starting with the low-order nibble.
9	UNCONFIGURE	The device currently addressed by its data pointer unconfigures itself. The device then responds to CONFIGURE and ID bus commands only. The local data pointers must be loaded immediately preceding an unconfigure command.
A	POLL	All chips that require service pull one data line high during the next *STR low (see section 2.5).
B	Reserved	
C	BUSCC	The device currently addressed by its local data pointer performs a specific operation as defined by the individual device.
D	Reserved	
E	SHUTDOWN	When the CPU has received a SHUTDN instruction it issues this command and turns off its oscillator. Each device responds based on its own special requirements to this command (see section 2.4).
F	RESET	All devices reset their configuration flags (if applicable) and perform other local resets based on their own special requirements.

2.1.3 Command Auto-Switch

There exists one special case in which all devices change their current bus command. This is called 'auto-switch' and occurs following the load of either the PC or the DP. On the rising edge of *STR after the 5th nibble of address has been loaded all devices clear bit 2 of their command latch changing the bus command from either a LOAD PC to PC READ or LOAD DP to DP READ.

2.1.4 Dummy Strobe

Immediately following a PC READ bus command, a DP READ bus command, and a command auto-switch the CPU issues a 'dummy strobe'. This dummy strobe appears as a data strobe except that no data is transferred during this period and devices do not increment their local address registers. The dummy strobe provides memory devices a full strobe cycle for the first access and therefore allows data pipelining.

2.2 Addressing

Each device on the HP-71 bus has two 20-bit address registers; a local program counter (PC) and a local data pointer (DP). Each device is also either hard addressed at a specific address (hard configured) or capable of being dynamically located within the address space (soft configured). A device only responds to data reads and writes if its local address register (PC or DP depending on the read or write command) is within its configured address space.

The HP-71 operating system allows soft configured devices to have address spaces ranging in size from 8 bytes to 128K bytes. All devices are configured such that the upper-order bits of the local address register can be compared with the upper-order bits of the device's configuration register (hard or soft). If these bits are identical, the device has an address match and will respond to read and write commands (and the unconfigure and BUSCC commands if applicable). Each device with a given address space size compares a given number of the upper-order bits of address. For example, a device with an address space size of 1K bytes or 2K nibbles requires 11 bits of address leaving the upper 9 bits for its configuration register.

2.2.1 Soft Configuration

A soft configured device powers up unconfigured. When unconfigured a device responds only to the ID and CONFIGURE commands and drives its DAISYOUT low. A device's ID code is used to identify the device before it is configured. If a soft configured device is unconfigured and has its DAISYIN line high, it sources its 5-nibble ID code starting with the low-order nibble on the 5 data strobes immediately following the issuance of an ID command (no dummy strobe is issued).

The 5-nibble ID code contains information on the device type and the address space required by the device as defined below:

NIBBLE 0 : Determines the size of the address space and is interpreted differently for memory and memory mapped I/O.

Nib 0	Memory Size	MM I/O space
F	1K nibble	16 nibbles
E	2	32
D	4	64
C	8	128
B	16	256
A	32	512
9	64 (max RAM)	1K
8	128	2K
7	256 (max)	4K
6	-	8K
7	-	16K (max)

NIBBLE 1 : Reserved for future use.

NIBBLE 2 : Device type--

0 : RAM
1 : ROM
2-E : Assorted memory types
F : Memory-mapped I/O

NIBBLE 3 : Device class--

Memory : unassigned

Memory-mapped I/O - 0 : HP-IL mailbox
1-F : unassigned

NIBBLE 4 : bits 0-1 : unassigned
bit 2 : Last chip in sequence.
Always assumed high for MM I/O.
bit 3 : Last chip in module.

Since the BUS[0:3] is precharged low before each strobe, the CPU will read an ID of all zeros if all devices are configured (or are unconfigured but have DAISYIN low). For more information on how the operating system handles configuration see the HP-71 Software IDS Volume 1.

A soft configured device is assigned its address configuration by the CONFIGURE command. If an unconfigured device has its

DAISYIN line high, it loads the configuration address that is issued on the 5 data strobes immediately following the CONFIGURE command (low-order nibble first) into its configuration register. A device may actually latch only the number of high-order bits it requires as determined by its address space size.

After being configured a device no longer responds to either an ID or CONFIGURE command. A configured device drives DAISYOUT to the same logic level as DAISYIN. The DAISYOUT of one device may be tied to the DAISYIN of second device. In this way many devices may be daisy-chained together in a way that they can be configured one at a time to different addresses. After being configured a device waits until the next command strobe to set its configuration flag in order to delay DAISYOUT so that the next device on the daisy-chain will not be configured simultaneously.

A device may be unconfigured by either a RESET or UNCONFIGURE bus command. The bus RESET command unconfigures all soft configured devices in the system. A device responds to an UNCONFIGURE command by clearing its configuration flag if the DP is within its address configuration.

2.2.2 Hard Configuration

A hard configured device powers-up configured to a specific address. It will not respond to an ID, CONFIGURE, or UNCONFIGURE command and a bus RESET will not affect its configuration. If the device has a DAISYOUT, it is always driven to the same logic level as its DAISYIN.

2.3 Data Transfer

All information that is transferred from the CPU to other devices in the system (commands, addresses, and data) is latched from the BUS[0:3] on the rising edge of *STR. Data that is transferred from system devices to the CPU is latched off the BUS[0:3] after the falling edge of *STR (timed internally on the CPU).

The CPU loads all devices' local address registers by issuing a LOAD PC or LOAD DP bus command followed by 5 data strobes of address, least significant nibble first. After the last nibble of address has been loaded all devices auto-switch to a PC READ or DP READ bus command. The CPU may then read the contents of that address location by issuing one dummy strobe followed by 1 to 16 data strobes during which the CPU latches the BUS[0:3] data. The

CPU may read without first loading the local address registers by issuing a PC READ or DP READ, followed by a dummy strobe, followed by 1 to 16 data strobes. The CPU precharges the BUS[0:3] low each cycle before *STR goes low. Therefore if no device responds the CPU reads zeros.

The CPU writes the contents of a specific addressed location similarly. It is not required to load the local address registers immediately before issuing a PC WRITE or DP WRITE command. The write command is followed by 1 to 16 data strobes during which the addressed device latches the BUS[0:3] data.

All devices increment their local address registers once each data strobe during read and write operations. It is possible for a read or write operation to begin in one device and cross the address boundary into another device. Future controllers on the HP-71 bus may read and write more than 16 nibbles at a time.

Two other types of data transfers are ID, which is simply a 5-nibble read with no address load or dummy strobe, and CONFIGURE, which is a 5-nibble write with no address load. Both these data transfers require that a device be unconfigured and that the DAISYIN line be high. POLL is a unique read and is discussed in section 2.5.

2.4 Power down; Wake up

The HP-71 system can be shutdown under software control. The CPU executes a SHUTDN instruction by issuing a SHUTDOWN bus command and on the ensuing cycle stopping the system clock (*STR) and its own oscillator. While in shutdown mode all data stored in RAM and CPU resident memory is preserved. The CPU is brought out of shutdown mode by either pulling an Input Register line high, or by driving *CD low.

*CD is driven low to wake up the CPU primarily by a device in the system that needs service while the system is in shutdown mode. If a device wakes up the CPU and the CPU shuts down without satisfying its service request the device will not wake up the CPU again until its service request has been satisfied and it needs service again. This avoids a situation where the operating system does not know how to handle a device's service request and cannot shutdown. For more information on power down and wake up see section 3.5.

2.5 Service Poll

If a device needs service while the CPU is operating it must wait until the CPU executes a service request instruction (SREQ), or, if it has the capability, interrupt the CPU using IR14 (available at all ports). The SREQ instruction causes the CPU to issue a POLL bus command followed by one data strobe during which the CPU latches the BUS[0:3] data in the manner of a usual read. A device may respond to the service POLL by pulling one of the BUS[0:3] lines high. Since the CPU precharges the BUS[0:3] low every cycle before *STR goes low the data read by the CPU is a binary OR of all devices' responses.

The following HP-71 chips can wake-up the CPU and can respond to a service POLL on the BUS[0:3] line shown:

Device	Bus line	Reason for Service Request
Display Driver	BUS[0]	Timer underflow.
HP-IL chip	BUS[1]	Data Avail; Interrupt; Pwr-on Reset; Loop Service Request.
Card Reader	BUS[2]	FIFO servicing; Error condition.

PROCESSOR — CHAPTER 3

3.1 CPU Overview

The 1LF2 CPU is a proprietary CPU optimized for high-accuracy BCD math and low power consumption. The CPU's principle function is to fetch and execute micro-instructions in proper sequence. Memory is accessed in 4-bit quantities called "nibbles". Addresses are 20 bits, providing a physical address space of 512K bytes. The CPU operates internally on four 4-bit busses with one or two busses acting as the source and one or two busses acting as the destination. Data for either source bus can originate from the system Bus, CPU resident memory, or other CPU register. Operations performed by the ALU include "add", "subtract", "and", "or", "1's complement", "2's complement", and in combination with the shifter "bit-shift", "digit-shift", and "rotate", on up to 64-bit operands.

For information regarding the full CPU instruction set see Chapter 4, entitled "HP-71 ASSEMBLER INSTRUCTION SET".

3.2 Pin Designations

The 1LF2 CPU's external pins are as follows:

PIN	FUNCTION
VDD	Power supply.
GND	System ground potential.
BUS[0:3]	System bus.
*STR	*STROBE
*CD	*COMMAND-DATA

IR[0:15]	INPUT REGISTER - For keyboard input and general input.
OR[0:11]	OUTPUT REGISTER - For keyboard output and general output.
HALT	HALT - When asserted high, the CPU completes the current instruction, tristates the Bus, *STR, and *CD lines and waits in a loop until HALT goes low again. The Bus is held passively low, *STR and *CD passively high.
NIF	NEXT INSTRUCTION FETCH - Goes high to indicate occurrence of the next instruction following the current *STR.
*INT	INTERRUPT - When pulled low the CPU completes the current instruction, pushes the PC onto the top of the subroutine return stack, and initiates the interrupt routine.
OSC1,OSC2	Oscillator input/output used for LC connection.
ECE	EXTERNAL CLOCK ENABLE - tied low on PC board.
CIO	CLOCK I/O - The primary oscillator frequency is driven out on this pin.
DRI	DRIVE - High when the CPU is driving the Bus, low if the CPU is sensing or tristated.
ENP	ENABLE POWER SUPPLY - Tied low on PC board.
VCO	VOLTAGE CONTROL OUT - Not used.
VCI	VOLTAGE CONTROL IN - Not used.

Note that *INT is available at all ports except the Card Reader port; IR14 is available at all ports; and HALT is available at all ports except PORT1. The Output Register is not only used to form the keyboard matrix, but also drives the DAISYIN (DIN) signal to each port and the piezo-electric beeper.

3.3 Registers

There are two types of registers on the CPU; those used for data transfers and arithmetic operations, and those used for program and system control.

Arithmetic Registers:

A	64-bits	Working register - I/O register
B	64-bits	Working register
C	64-bits	Working register - I/O register
D	64-bits	Working register
R0	64-bits	Scratch register
R1	64-bits	Scratch register
R2	64-bits	Scratch register
R3	64-bits	Scratch register
R4	64-bits	Scratch register
CARRY	1-bit	Flag set by arithmetic operations and tests

Control Registers:

POINTER	4-bits	Pointer register
DPOINT0	20-bits	Address pointer register
DPOINT1	20-bits	Address pointer register
PC	20-bits	Program Counter
RETURN STACK	20-bits	8-level subroutine stack
STATUS	16-bits	Program status flags
HW. STATUS	4-bits	CPU/system status flags
OUTPUT	12-bits	Keyscan/write only Output Register
INPUT	16-bits	Keyscan/read only Input Register

3.3.1 Working and Scratch Registers

All arithmetic operations are performed using the 4 working registers: A, B, C, and D. Data transfers are performed principally with the A and C registers.

The scratch registers are used to temporarily hold the contents of the working registers. The lower 20 bits of scratch register R4 are reserved by the operating system for interrupt processing, and therefore are not normally available for data storage.

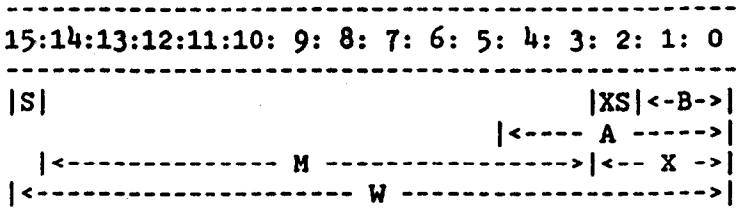
3.3.1.1 Field Selection

Subfields of the working registers may be manipulated using field selection. The possible field selections range from the entire register to any single nibble of the register. Certain subfields are designed for use in BCD calculations and others are designed for use in general data manipulation and data access.

FIELD SELECTION FIELDS

P	Digit pointed to by P register
WP	Digit 0 through digit pointed at by P
XS	Digit 2 - Exponent sign
X	Digits 0-2 - Exponent and exponent sign
S	Digit 15 - Mantissa sign
M	Digits 3-14 - Mantissa
B	Digits 0-1 - Exponent or byte field
W	Digits 0-15 - Whole word
A	Digits 0-4 - Address field

Nibbles of Register



3.3.2 Carry Bit

The Carry bit is adjusted when a arithmetic operation or test is performed. During a calculation, such as incrementing or decrementing a register, it is set if the calculation overflows or borrows and cleared if it does not. During a arithmetic test, such as comparing two registers for equality, it is set if the test is true and cleared if it is not.

3.3.3 Pointer Registers

The Data Pointer registers, D0 and D1, provide the source addresses for all external data transfers.

The P Pointer register is used in Field Selection operations with the working registers.

3.3.4 Program Counter and Return Stack

The program counter points to the next instruction to be executed by the CPU. It can be accessed only using jump, gosub, and return instructions.

The current value of the program counter is automatically pushed onto the 8-level subroutine return stack when a gosub instruction is executed or an interrupt occurs. A 20-bit value is automatically popped off the return stack into the program counter when a return instruction is executed. The return stack can also be manipulated through use of the push (RSTK=C) and pop (C=RSTK) instructions.

3.3.5 Status Bits

Additional program control is provided by the 16-bit Program Status register and the 4-bit Hardware Status register. Each Status bit can be individually set, reset, and tested.

The operating system uses the upper 4 Program Status bits to indicate the state of the operating system. The remaining 12 Program Status bits are generally available to applications software, and may be manipulated collectively as the ST register.

The four Hardware Status bits are set (but not cleared) by hardware-related events, and must therefore be cleared beforehand in order to detect a particular occurrence. They are individually accessible by name. The Module Pulled bit (MP) is set when a module is pulled from or added to the machine. The Sticky Bit (SB) is set when a non-zero bit or digit shifts off the right end of a working register as the result of a shift right instruction, or the least significant nibble of a working register is non-zero prior to a shift right circular instruction. The Service Request (SR) bit is set as a result of a response to the SREQ? instruction (see section 2.5). The external Module Missing bit is set by execution

of a "00" opcode (RTNSXM instruction). Since the BUS[0:3] is precharged low, the CPU will receive a RTNSXM instruction if no device responds to a PC READ bus command.

PROGRAM STATUS: 16 bits

Bits	Usage
15 thru 12	Indicate state of operating system
11 thru 0	Available to programs, may be manipulated as the ST register

HARDWARE STATUS: 4 bits

Bit	Symbol	Name
3	MP	Module Pulled
2	SR	Service Request
1	SB	Sticky Bit
0	XM	External Module Missing

3.3.6 Input/Output Registers

The Input and Output Registers provide the CPU with general purpose I/O. This consists of a 16-bit Input Register with interrupt capability (see section 3.5) and a 12-bit Output Register.

Data read from the Input Register corresponds to the logic levels sensed on the 16 input lines, IR[0:15]. The Input Register lines are passively held low so that if an input line is not driven a zero level will be sensed. Data written to the Output Register is driven onto the 12 output lines OR[0:11].

The Input and Output Registers are used to form the keyboard matrix. The Output Register lines OR[0:3] drive the 4 key-rows. The Input Register lines IR[0:13] sense the 14 key-columns. The most significant bit of the Input Register (IR15) is dedicated to the On-key and has additional interrupt capability (see section 4.4.2). When a key is pushed its key-row is shorted to its key-column, and the logic level driven on the Output Register line can be read from the Input Register bit. If the Output Register line is high and interrupts are enabled, pushing the key will result in an interrupt.

When one of the 8 lower order bits of the Output Register is low, the Output Register line will be actively driven low briefly

each instruction cycle, then passively held low. This limits the current and provides a deterministic state when Output Register lines are driving different logic levels and are shorted. Output Register lines are shorted when 2 or more keys in the same column are depressed at the same time. This will result in a logic one level on the corresponding bit of the Input Register if one of the Output Register lines is high. The remaining 4 Output Register lines, OR[8:11] are actively driven at all times.

The 5 Output Register lines OR[0:4] also drive the DAISYIN lines of each I/O port. During configuration the operating system individually selects each I/O port by driving its DAISYIN line high (see section 8.3).

The 2 Output Register lines OR[10:11] drive the piezo-electric beeper. Two beeper loudness settings are obtained by using one of two Output Register lines.

3.3.7 Loading Data from Memory

When data is read from an external device into a register, the CPU places the lowest addressed nibble in the least significant nibble of the register. For example, if the data shown below in memory is read into the C register using the C=DAT1 4 instruction, the data in the register will be arranged as shown above.

Memory Location	Value
1000	0
1001	1
1002	2
1003	3
.	
.	
.	

C Register

15	. . .	3	2	1	0
		3	2	1	0

This principle also applies to loading constants into a CPU register such as C, D0,or D1, since the CPU must read the constant from the instruction opcode in memory. For example, the instruction LCHEX 3210 produces the opcode 330123 and the C register is loaded as shown above.

3.3.8 Storing Data in Memory

When data is written from a register to an external device, the CPU places the least significant nibble of the register in the first addressed nibble of the memory location. For example, if the data shown above in the C register is written to memory using the DAT1=C 4 instruction, the data will be written to memory as shown.

3.4 HEX/DEC Modes

All arithmetic operations, except for those listed below, are performed according to the HEX or DEC mode setting. The mode is set using the SETHEX or SETDEC instruction. The following operations are performed in HEX regardless of the mode setting.

C+P+1	
D0=D0+ n	D0=D0- n
D1=D1+ n	D1=D1- n
P=P+1	P=P-1

3.5 Interrupt System

The 1LF2 CPU can be interrupted in the following manners:

1. Pulling the *INT line low (module interrupt).
2. Shorting the least significant bit of the Output Register to the most significant bit of the Input Register (hardware "ON"-key).
3. An Input Register line (bits 0-15) being pulled high.

If interrupted, the CPU completes the current instruction, pushes the PC onto the return stack and sets the PC to the interrupt routine starting address of 0000F (where F is in the least significant nibble of the PC). Interrupts are disabled upon entry into the interrupt routine and are not enabled until a Return From Interrupt (RTI) instruction is executed. Each interrupt type operates in a somewhat different manner and is described separately.

3.5.1 *INT Interrupt

Pulling the *INT line low causes a non-maskable interrupt. When *INT is asserted the CPU enters the interrupt routine and disables all interrupts. When a RTI instruction is issued all interrupts may be processed immediately, including the *INT line even if it remains low from the previous interrupt.

3.5.2 Hardware "On-key" Interrupt

During the last cycle of every instruction the least significant bit of the Output Register (OR0) is driven to a one. This does not modify the contents of the Output Register, and the previous value is driven after this cycle. If the MSB of the Input Register (IR15) is driven high (On-key down) during this period, an interrupt is executed. This interrupt cannot be disabled outside of the interrupt routine.

Once a RTI instruction has been issued the CPU cannot be interrupted again by this Input Register line or any others until all Input Register lines are low. The CPU can be interrupted by the *INT line regardless of the state of the Input Register lines as long as a RTI instruction has been issued.

3.5.3 Input Register Interrupt

At any time except the time period allocated to the hardware "On-key", the CPU may be interrupted by any Input Register line pulled high. This interrupt may be disabled outside the interrupt routine by using the Interrupt Off (INTOFF) instruction, and re-enabled using the Interrupt On (INTON) instruction.

The CPU cannot be interrupted again by any Input Register line until a RTI instruction is issued and all Input Register lines are again low.

3.6 CPU Power-down; Wake-up

The CPU supports a low-power standby mode. To place the system in standby, a SHUTDN instruction is issued. The CPU then issues the SHUTDOWN bus command and on the ensuing cycle, stops *STR and its own clock. All CPU resident memory is preserved during standby. In standby mode the CPU holds the Bus lines in the conditions as follows:

BUS[0:3]	Low - through a high impedance
*STR	High - through a high impedance
*CD	High - through a high impedance

The CPU is brought out of standby mode when either an Input Register line goes high or the *CD line is driven low. On wake-up the CPU starts it's clock and immediately drives *CD low with a NOP bus command. If the CPU detects that a severe low voltage condition has occurred while it was in standby, the PC is set to zero and hex mode arithmetic is asserted. A LOAD PC bus command is issued after the NOP and the current CPU PC is loaded into all devices' local PCs. At this point the standard instruction fetch sequence in initiated. If the CPU was awakened by the Input Register and interrupts are enabled a normal interrupt will occur prior to the first instruction fetch. If a SHUTDN instruction is executed with the least significant bit of the Output Register (OR0) set at a "0" the CPU will wake-up immediately, set its PC to zero, assert hex mode arithmetic, and send out the PC. This avoids a situation where the CPU is shutdown and the keyboard cannot wake it up.