HOLTEK

HT1337
8-bit Microcontroller

Features

- Operating voltage: 2.4V~3.3V
- 12 input lines
- Four output lines
- Five working registers
- Sound effect circuit
- 2K 8 program ROM
- 4K 8 bits data memory RAM size
- 32 4 segment LCD driver
- RC oscillator for system clock
- Halt feature reduces power consumption
- Internal timer overflow interrupt
- External interrupt
- One level subroutine nesting
- 8-bit timer with internal or external clock source
- 8-bit table read instruction
- Up to 4ms instruction cycle with 1MHz system clock at V_{DD}=3V
- All instructions in 1 or 2 machine cycles

General Description

The HT1337 is a processor from Holtek s 8-bit stand alone single chip microcontroller range specifically designed for LCD product applications. The device is ideally suited for multiple LCD low power applications among which are calculators, scales and hand-held LCD products.

Functional Description

Program counter - PC

This counter addresses the program ROM which contents specify a maximum of 2048 addresses. The program counter counts with an increment of 1, 2 or 3 with each execution of an instruction.

When executing the jump instruction (JMP, JNZ, JC, JTMR,...), a subroutine call, initial reset, internal interrupt, external interrupt or returning from a subroutine, the program counter is loaded with the corresponding instruction data as shown in the table.

Program memory - ROM

The program memory is the executable memory and is arranged in a 2048 8 bit format.

Data memory – RAM

The static data memory (RAM) is arranged in 4096 8 bit format and is used to store data. All of the data memory locations are indirectly addressable through the register pair R1,R0 or R3,R2; for example MOV A,[R3R2] or MOV [R3R2],A.

Working registers - R0, R1, R2, R3, R4, R5

There are five 8 bit working registers (R0, R1, R2, R3, R4) and one 16 bit register (R5) used to store the frequently accessed intermediate results. Using the instructions INC Rn and DEC Rn, the working registers can increment (+1) or decrement (-1). The JNZ Rn (n=0,…,5) instruction makes efficient use of the working registers as a program loop counter. The register pairs R0,R1 and R2,R3 are also used as a data memory pointer when the memory transfer instruction is executed.

Accumulator - ACC

The accumulator is the most important data 16 bit register in the processor. It is one of the sources of input to the ALU and the destination of the results of the operations performed in the ALU. Data to and from the I/O ports and memory also pass through the accumulator.

Arithmetic and logic unit - ALU

This circuit performs the following arithmetic and logic operations ...

Add

Subtract

AND, OR, Exclusive-OR

Instruction Definitions

ADD A,XXH Add immediate data to the accumulator

Machine code 0 1 0 0 0 0 0 0 d d d d d d d d d d d d d d d d

Description The specified data is added to the accumulator.

Operation ACC ACC+XXH

ADD A,[R1R0] Add data memory contents to the accumulator

Machine code 0 0 0 0 1 0 0 1

Description The contents of the data memory addressed by the register pair "R1,R0" is added to the accumulator.

Operation ACC ACC+M(R1,R0)

AND A,XXH Logical AND immediate data to accumulator

Machine code 0 1 0 0 0 0 1 0 d d d d d d d d d d d d d d d d

Description Data in the accumulator is logical AND with the immediate data specified by the code.

Operation ACC ACC "AND" XXH

AND A,[R1R0] Logical AND accumulator with data memory

Machine code 0 0 0 1 1 0 1 0

Description Data in the accumulator is logical AND with the data memory addressed by the register pair "R1,R0" Operation

ACC ACC "AND" M(R1,R0)

AND [R1R0],A Logical AND data memory with accumulator

Machine code 0 0 0 1 1 1 0 1

Description Data in the data memory addressed by the register pair "R1,R0" is logical AND with the accumulator

Operation M(R1,R0) M(R1,R0) "AND" ACC

DEC A	Decrement accumulator
Machine code	0 0 1 1 1 1 1 1
Description	Data in the accumulator is decremented by 1.
Operation	ACC ACC-1

DEC [R1R0]	Decrement data memory
Machine code	0 0 0 0 1 1 0 1
Description	Data in the data memory specified by the register pair "R1,R0" is decremented by 1.
Operation	M(R1,R0) M(R1,R0)-1

DEC [R3R2]	Decrement data memory
Machine code	0 0 0 0 1 1 1 1
Description	Data in the data memory specified by the register pair "R3,R2" is decremented by 1.
Operation	M(R3,R2) M(R3,R2)-1

DEC Rn	Decrement register
Machine code	0 0 1 1 1 0 0 0 0 0 0 0 0 n n n
Description	Data in the working register "Rn" is decremented by 1.
Operation	Rn Rn 1; Rn=R0,R1,R2,R3,R4,R5, for n=0, 1, 2, 3, 4, 5

INC A	Increment accumulator
Machine code	0 0 1 1 0 0 0 1
Description	Data in the accumulator is incremented by 1.
Operation	ACC ACC+1

INC [R1R0]	Increment data memory
Machine code	0 0 0 0 1 1 0 0
Description	Data in the data memory specified by the register pair "R1,R0" is incremented by 1.
Operation	M(R1,R0) M(R1,R0)+1

INC [R3R2]	Increment data memory
Machine code	0 0 0 0 1 1 1 0
Description	Data memory specified by the register pair "R3,R2" is incremented by 1.
Operation	M(R3,R2) M(R3,R2)+1

INC Rn	Increment register
Machine code	0 0 0 1 1 0 0 0 0 0 0 0 0 n n n
Description	Data in the working register "Rn" is incremented by 1.
Operation	Rn Rn+1; Rn=R0~R5 for n=0~5

JMP address Direct jump

Machine code 1 1 1 0 0 a a a a a a a a a a a

Description Bits 0~11 of the program counter are replaced with the directly-specified address.

Operation PC address

JNZ A,address Jump if accumulator is not 0

Machine code 1 0 1 1 1 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address but bit 11 of the program counter is unaffected, if the accumulator is not 0.

Operation PC (bit 0~10) address, if ACC=0
PC PC+2, if ACC=0

JZ A,address Jump if accumulator is 0

Machine code 1 0 1 1 0 a a a a a a a a a a a

Description Bits 0~10 of the program counter are replaced with the directly-specified address but bit 11 of the program counter is unaffected, if the accumulator is 0.

Operation PC (bit 0~10) address, if ACC=0
PC PC+2, if ACC=0

MOV A,Rn Move register to accumulator

Machine code 0 0 1 0 1 1 0 1 0 0 0 0 0 n n n

Description Data in the working register "Rn" is moved to the accumulator.

Operation ACC Rn; Rn=R0~R4, for n=0~4

MOV A,XXH Move immediate data to accumulator

Machine code 0 1 1 1 0 0 0 0 d d d d d d d d d d d d d d d d

Description The 4-bit data specified by the code is loaded to the accumulator.

Operation ACC XXH

MOV A,[R1R0] Move data memory to accumulator

Machine code 0 0 0 0 0 1 0 0

Description Data in the data memory specified by the register pair "R1,R0" is moved to the accumulator.

Operation ACC M(R1,R0)

MOV A,[R3R2] Move data memory to accumulator

Machine code 0 0 0 0 0 1 1 0

Description Data in the data memory specified by the register pair "R3,R2" is moved to the accumulator.

Operation ACC M(R3,R2)

MOV Rn,A Move accumulator to register
Machine code 0 0 1 0 1 1 0 0 0 0 0 0 0 n n n
Description Data in the accumulator is moved to the working register "Rn".
Operation Rn ACC; Rn=R0~R5, for n=0~5

MOV [R1R0],A Move accumulator to data memory
Machine code 0 0 0 0 0 1 0 1
Description Data in the accumulator is moved to the data memory specified by the register pair "R1,R0".
Operation M(R1,R0) ACC

MOV [R3R2],A Move accumulator to data memory
Machine code 0 0 0 0 0 1 1 1
Description Data in the accumulator is moved to the data memory specified by the register pair "R3,R2".
Operation M(R3,R2) ACC

NOP No operation
Machine code 0 0 1 1 1 1 1 0
Description Do nothing, but one instruction cycle is delayed.

SUB A,XXH Subtract immediate data from accumulator
Machine code 0 1 0 0 0 0 0 1 d d d d d d d d d d d d d d d d
Description The specified data is subtracted from the accumulator.
Operation ACC ACC-XXH

SUB A,[R1R0] Subtract data memory contents from accumulator
Machine code 0 0 0 0 1 0 1 1
Description The contents of the data memory addressed by the register pair "R1,R0" is subtracted from the accumulator.
Operation ACC ACC-M(R1,R0)

OR A,XXH	Logical OR immediate data to accumulator
Machine code	0 1 0 0 0 1 0 0 d d d d d d d d d d d d d d d d
Description	Data in the accumulator is logical OR with the immediate data specified by the code.
Operation	ACC ACC "OR" XXH

OR A,[R1R0]	Logical OR accumulator with data memory
Machine code	0 0 0 1 1 1 0 0
Description	Data in the accumulator is logical OR with the data memory addressed by the register pair "R1,R0".
Operation	ACC ACC "OR" M(R1,R0)

OR [R1R0],A	Logically OR data memory with accumulator
Machine code	0 0 0 1 1 1 1 1
Description	Data in the data memory addressed by the register pair "R1,R0" is logical OR with the accumulator.
Operation	M(R1,R0) M(R1,R0) "OR" ACC

XOR A,XXH	Logical XOR immediate data to accumulator
Machine code	0 1 0 0 0 0 1 1 d d d d d d d d d d d d d d d d
Description	Data in the accumulator is Exclusive-OR with the immediate data specified by the code.
Operation	ACC ACC "XOR" XXH

XOR A,[R1R0]	Logical XOR accumulator with data memory
Machine code	0 0 0 1 1 0 1 1
Description	Data in the accumulator is Exclusive-OR with the data memory addressed by the register pair "R1,R0".
Operation	ACC ACC "XOR" M(R1,R0)

XOR [R1R0],A	Logical XOR data memory with accumulator
Machine code	0 0 0 1 1 1 1 0
Description	Data in the data memory addressed by the register pair "R1,R0" is logically Exclusive-OR with the accumulator.
Operation	M(R1,R0) M(R1,R0) "XOR" ACC

RET	Return from subroutine or interrupt
Machine code	0 0 1 0 1 1 1 0
Description	The program counter bits 0~11 are restored from the stack.
Operation	PC Stack