INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
I.C.T.P., P.O. BOX 586, 34100 TRIESTE, ITALY, CABLE CENTRATOM TRIESTE

H4.SMR. 403/28

FIFTH COLLEGE ON MICROPROCESSORS: TECHNOLOGY AND APPLICATIONS IN PHYSICS

2 - 27 October 1989

# Other Microprocessors

**D. CROSETTO**
**CERN, SPS Division, Geneva, Switzerland**

## Avant propos:

Possibilities for these lectures:

-> Say nothing about all
(Give a list and short description of many types of micros and give a brief description of their features)

8 Bits: 8080,8086,Z80,M6800,M6809,68008.....
(You surely know some more)

16 Bits: [illegible],Z8000,M68000....

32 Bits: 80386,M68020,N

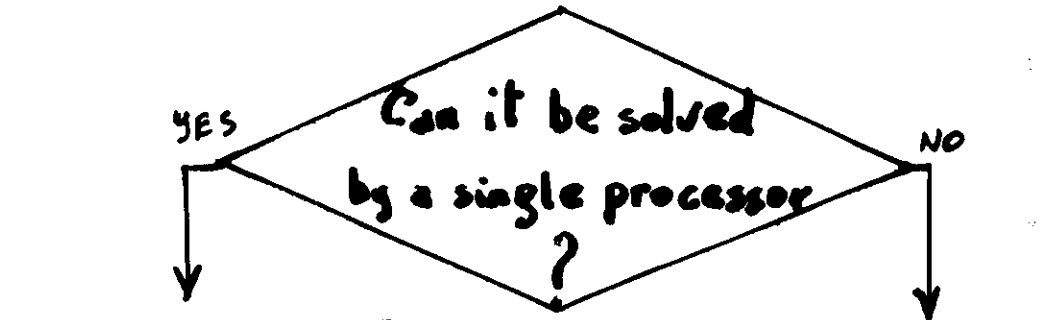
Specials: Transputer,Bit-Slices...

or . . .

-> Say all about almost nothing
(Select 1 or 2 micros other than the M6809 and explain their features in more detail)

**Decided: Describe M68000 and M68020 and compare them to the M6809. Where do we need 16/32 bit machines and why**

# PROBLEM DEFINITION




Which microprocessor is the best?

- MICROCONTROLLER
- CISC ← 6809
- RISC
- CRISP
- DSP
- TRANSPUTER
- BIT SLICE
- GATE ARRAY
- CUSTOM VLSI

Which architecture

- CONCEPTS
- PARALLEL COMPUTER STRUCTURE
- METHODS OF CLASSIFICATIO
- INTERCONNEC SYSTEM

# CONCEPTS:

Parallel processing is concurrent execution of multiple functions.

Concurrency implies concepts of:

- Parallelism : parallel events may occur in multiple resources
- Simultaneity: [illegible]

[redacted]

## COMPUTER STRUCTURES

Parallel computer systems may be divided into three architectural configurations:

**Pipeline computers**
Perform overlapped computations to exploit temporal parallelism

**Array processors**
Uses multiple, synchronized arithmetic logic units to achive spatial parallelism

**Multiprocessor system**
Achive asynchronous parallelism through a set [illegible] with shared resources such a

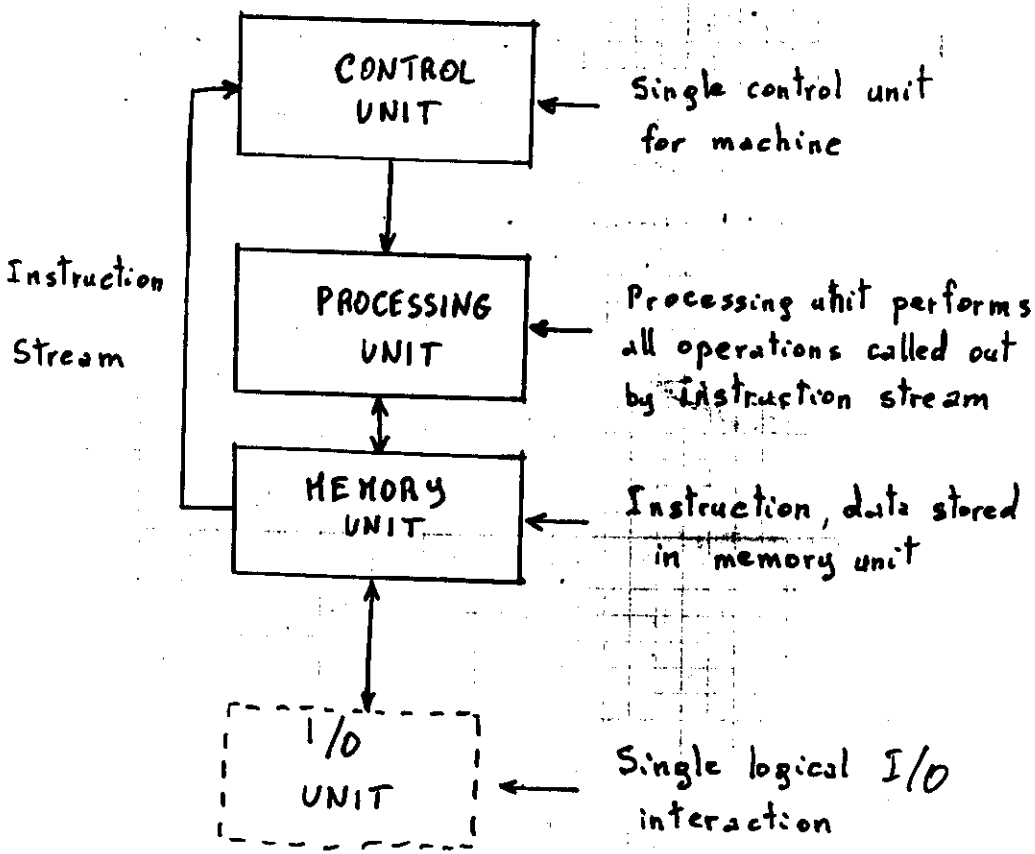


# METHODS OF CLASSIFICATION

- Performance metrics
- Technological contributions
- Architectural methods
- Generation of computers (1-5) 1953 - tubes-ri
- Architectural descriptions
  - SISD : Single Instruction Stream, Single Data Stream
  - SIMD: Single Instruction Stream, Multiple Data Stream
  - MIMD: Multiple Instruction Stream, Multiple Data Stream
  - MISD : Multiple Instruction Stream, Single Data Stream
- Interconnection techniques

# BASIC "SISD" ARCHITECTURE



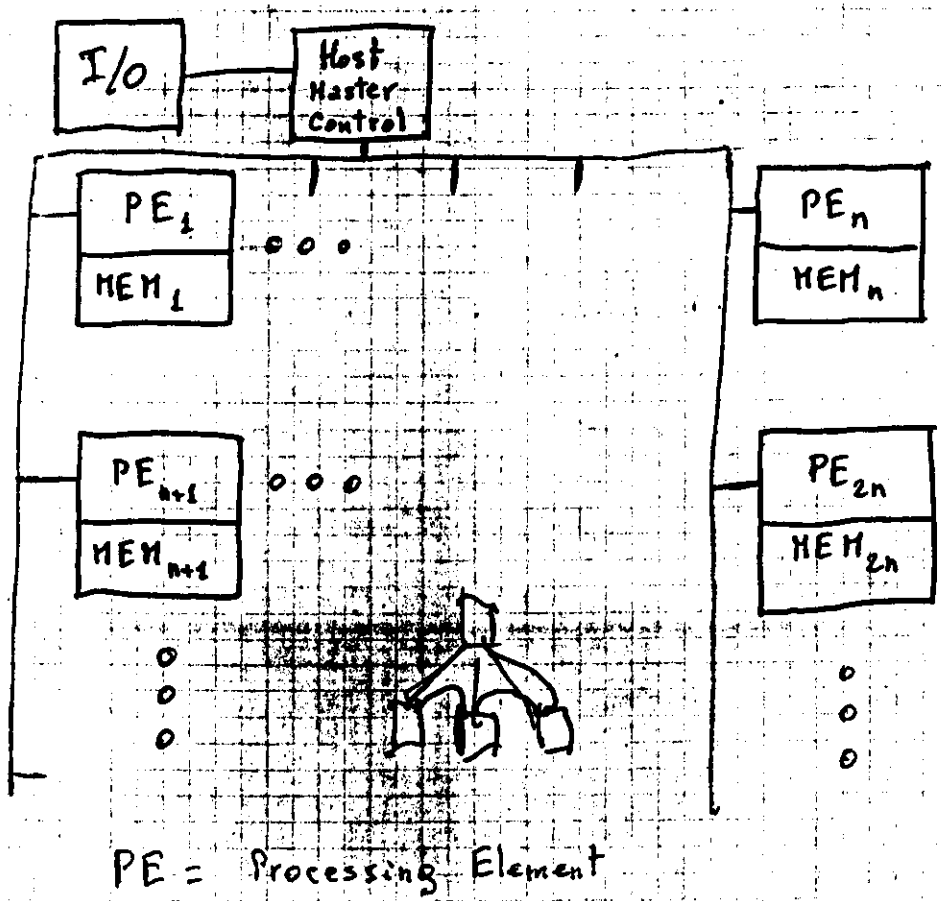



- One stream of instructions, one stream of data
- Most "normal" machines includes pipelined implem.
- Single control unit



# BASIC "SIMD" ARCHITECTURE



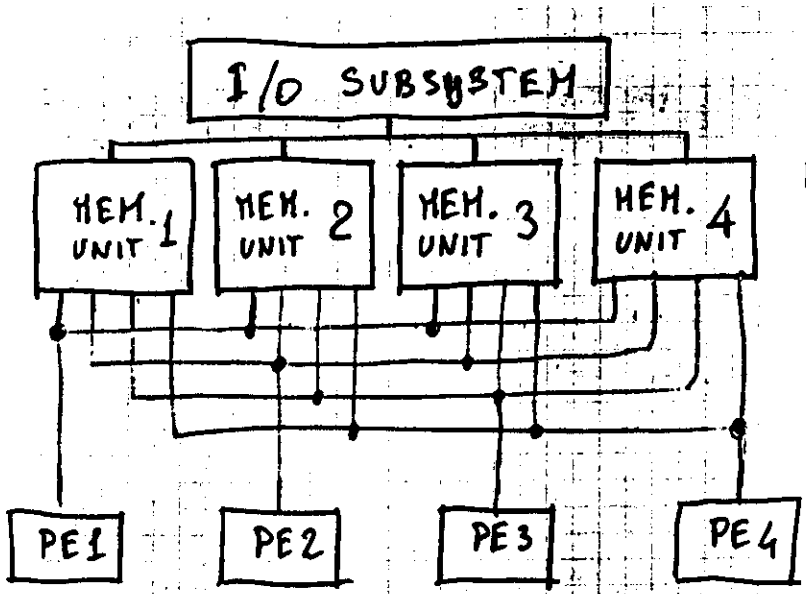



PE = Processing Element

All "PE" Execute Instructions on data in own memory

- Early attempt at parallel processing - ARRAY PROCESSING
- One instruction stream for all operations
- Multiple data stream: one for each PE
- Serial portion of instruction stream executed only in master

# BASIC "MIMD" ARCHITECTURE
(Multiport)



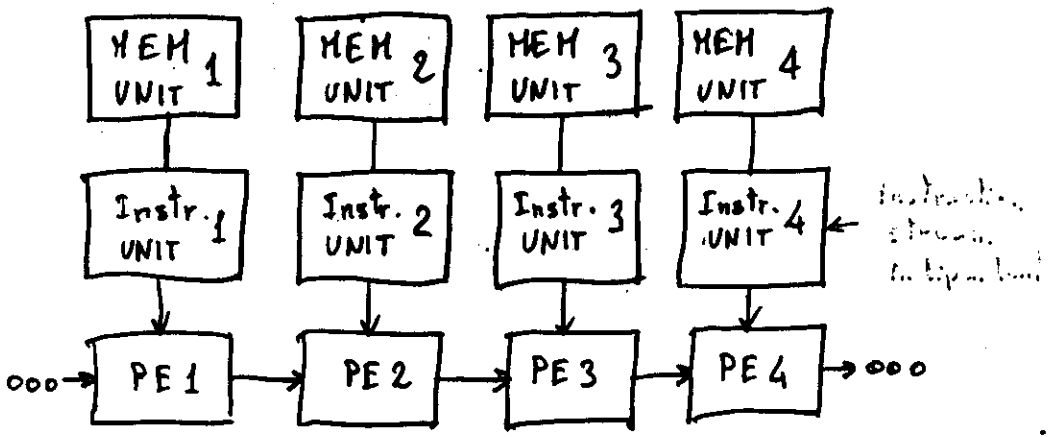

Memory contains multiple or single program(s)

each PE operates independent from others Communication through memory

- Instruction partitioned into independent streams
- Each instruction stream has own data stream
- Model fit multiprocessor, multicomputer organizations
- Memory can be shared or independent
- Tightly coupled (special purpose) and loosely coupled (general purpose machines)
- Intercomection methods:
  - Multiport memory
  - Busing schemes
  - Regular conn. system
  - Cross bar switch



# HYPOTHETICAL "MISD" ARCHITECTURE




Each PE operates on data and passes it on

- Multiple sets of instructions to execute
- Single stream of data
- No real processor of this type Known

# INTERCONNECTION SYSTEMS

## Switching systems: multiple simultaneous transfer

- BUS system
- Crossbar switch
- Omega network (Multiport)

## Shared resources: OS driven

- Shared memory

## Regular, structured interconnection: message passing sys

- Hypercube structures
- Tree connected machines
- 2-D grid



- Assuming to have a problem to solve that requires more computing power and I/O then any $\mu P$ or $\mu C$ existing on the market.

## Which Parallel Processing Architecture do we choose ?

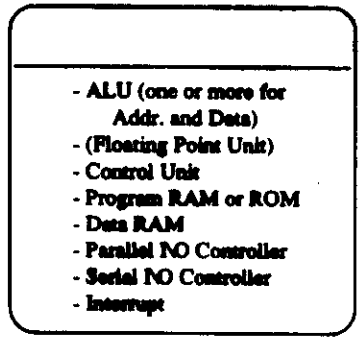
Is there the possibility to design an architecture flexible enough that can solve efficiently all types of applications ? (I/O, matrix calculation, Filters, Real-time, vector proc., many data - little process, a lot of process on few data, programmable interconnection configurable as three, matrix, etc. tightly or loosely coupled...

If such a super computer that will solve all types of problems will ever be build, the ratio cost/performance will certainly be very high and for some specific applications the efficiency and the facility to use it could not be the best

- Start from the Problem Definition
- Define the best Architecture
- Select the tools (~~Opera~~ BUS, Operating System, Languages, Debug
- Select the best and (more suitable to the applicatio most advanced technology (μP, ASIC, VLSI, μC, etc)

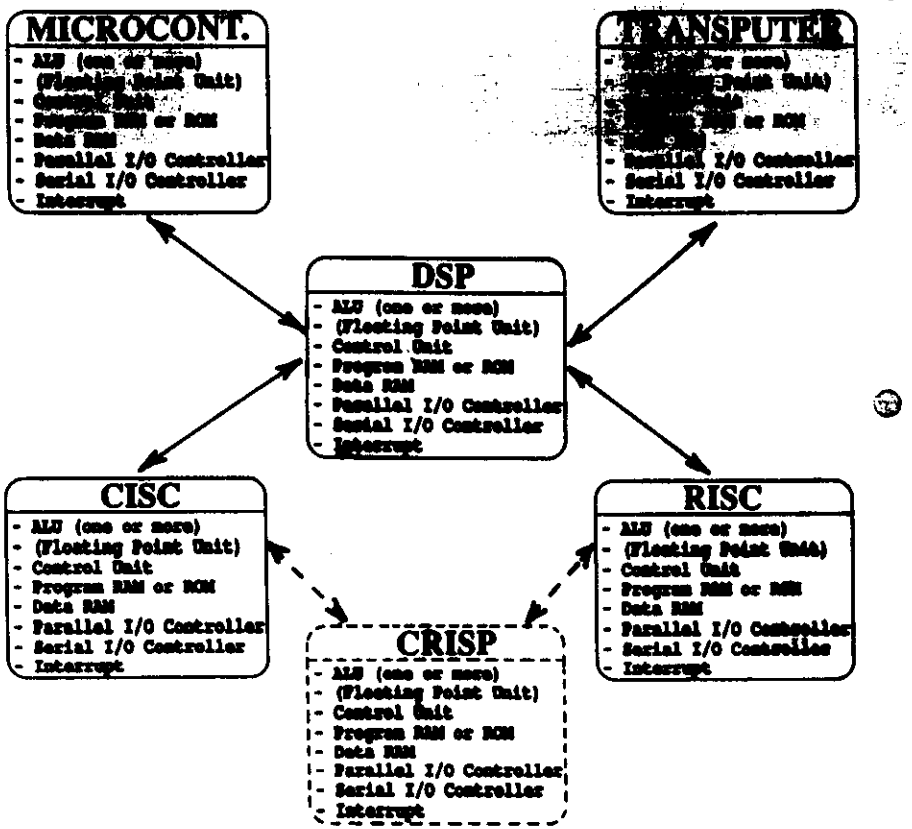
**After performance comparison with other components it will become clearer why the DSP is more suitable for several types of applications.**

**The basic elements of a real-time processor are:**



There are several ways to realize a concurrent system of many basic element described above, each one having a different throughput, privileging in one case one aspect respect to another

Differences on performance respect to a basic system that make use of DSP




## MICROCONTROLLERS < > DSP

- A "microcontroller" contains all the necessary components of a complete system on one piece of silicon (E.g. Intel 8051, Motorola MC6804, MC6805, MC68HC11, etc.)

- Less performance than a DSP

- 4, 8, 16-bit

- instruction set more like CISC processor (using more then one cycle per instruction)

- some extra programmable peripherals on chip, like A/D converters are not available on DSP.

- Is not designed to build concurrent systems but for economical applications in embedded systems where is necessary only to have the capability of one of the most common 8-bit or 16-bit microprocessor instruction sets.

## Applications:

- industrial control
- device controller (printers, plotters, etc.)
- in an array of front end processors in a High Energy Physics Experiment for slow calculations

- DSP is replacing to this component in the most sophisticated applications where speed is an important factor.

# TRANSPUTER < > DSP

- A Transputer contains in a single chip:
  - an integer processor
  - a Floating Point Unit
  - 4 Kbyte of memory
  - 4 high speed serial links (20 Mbit/sec)

- Transputer is designed as a programmable component to implement a system with much higher degree of concurrency then is currently common.

- The Transputer, together with the formal rules of Occam, provides the design methodology for this family of concurrent systems.

- Special instructions divide the processor time between the concurrent processes and perform interprocess communication

- In addition the transputer is designed so that its standard behavior corresponds to the formal model of a process. As a consequence it is possible to program systems containing multiple interconnected transputers in which each transputer implements a set of processes.

- Since a program is defined as a set of processes, it can be mapped onto such a system in a variety of ways, for example to minimize cost or to optimize throughput, or to maximize the responsiveness to specific events.

- The architecture should give the possibility to span the range of application from microcontrollers to supercomputers

# ADVANTAGES AND DISADVANTAGES:

- It is easier to build concurrent systems because of the good coordination between hardware and software (Occam)

- Easier to transport software on different concurrent systems with different number of transputers.

- Good concurrency, and good flexibility, but the throughput respect to another architecture that makes use of DSP for a more specialized application has to be verified.

- The time required for a multiplication is 500 nsec average for T800 (~ 2 usec for a T414). Most DSP's do it in one cycle (75 to 200 nsec), the same is true for the division and for the floating point operations.

- The performance of a Transputer begins to drop noticeably as soon as the on-chip memory is too small to hold all the frequently accessed data. And its premise that the world is process-shaped rather than procedure-shaped may well be true, but the majority of available software doesn't reflect that belief.

- on the contrary the DSP don't have special signals or instructions foreseen to implement a concurrent system with message passing.

# RISC < > DSP

- Deeper investigation is merited by the RISC architecture because it is the most innovative and is based on concepts that are attractive for applications in: workstations, superminis and also as embedded controllers.

- From the initial simple concepts of a register-intensive cpu design from Seymour Cray in 1960 to the modern notion of RISC architectures emerged from John Cocke's project at IBM in 1970.

- Cocke's team goal was to design the best CPU architecture for an optimizing compiler

- the machine should be register-to-register with only load and store accessing the memory.

- the architecture eliminated microcode and microsequencers in favor of simple, hardwired, pipelined, one-instruction-per cycle CPU design.

- RISC technology created an almost insatiable demand for memory speed. The answer to the problem come with high performance memory hierarchy, including general purpose registers and cache memories.

- the instruction set is regular and simple with few addressing modes: indexed and PC-relative.

There are then some RISC variations from these common theme.

- IBM 1975 with 801 minicomputer

- BERKELEY 1980 with RISC I and RISC II

- STANFORD 1981 with MIPS (Microprocessor Without Interlocked Pipeline Stages)

- IBM and Stanford pushed the state of art in Compiler Technology to maximize the use of registers.

  - the key idea is to expose in the instruction set all the processor activity that could effect performance. This philosophy, coupled with the concept of a streamlined instruction set, allows a shift of functions from hardware to software.

  - Hennessy's team at Stanford recognized that with a clever compiler, interlocking the pipeline wasn't necessary. The compiler simply had to make sure that the instruction directly after the LOAD didn't use the new data.

- The BERKELEY team did not include compiler experts, so a hardware solution was implemented to keep operand in registers.

  - to optimize the task switching time they have defined many sets or windows of registers (global and local) so that registers would not have to be saved on every procedure call. The disadvantage of register windows is that they use more chip area.

- As Compiler Technology improves and sufficiently fast processors become available, there should be decreasing necessity to program in assembly language.

## Each vendor has had to improve some characteristics to translate the University design into workstations products.

- SUN Microsystems adopted some ideas of Patterson's work at Berkeley and designed a system that should be portable between implementation technologies (CMOS, ECL, etc.) calling it SPARC (Scalable Processor Architecture).

  Customers are looking at SPARC as an instruction-set definition, with many implementations available.

- MIPS Computer System turned Stanford team's effort into a product. They retained the Stanford design's delayed load and branches, and focused on a single high-speed implementation rather than scalability.

- The MIPS designers felt strongly that the key to performance was the ability of the compiler to manage CPU pipeline during floating point as well as integer operations. The floating point unit must understand the state of the integer unit's pipeline at all times (R3000, R3010)

- The Fairchild team did not attempt to reimplement either research chip.

  The Fairchild Clipper now available from Intergraph Advanced Processor Division, was the first microprocessor design to recognize the growing memory bandwidth.

  Their solution was to separate the relentless demand of instruction fetches Load/Store activity by providing separate instruction and data buses.

  The Clipper supports the dual busses with a pair of integrated caches and memory management chips.

- The MC88000 from Motorola appears to be a blend of the purity of MIPS' CPU concepts and the innovations of Clipper's bus architecture.

  It follows the dogma of simple, one-cycle, fixed-length instructions and load/store architecture.

  Like Clipper, the MC88000 is a dual bus, three-chip layout (HCMOS). All the execution units work over the same two source buses and a single destination bus.

  The most important characteristics of the MC88000 may be the system's ability to incorporate new, specialized execution units. That starts to make room for some really interesting special purpose units, like vector processors or graphic engines.

- MIPS, Intergraph and Motorola all have taken different architectural approaches to a common goal:

## A FAST UNIX WORKSTATION

- Chip vendors, with characteristic optimism, are already discussing coprocessors for signal processing, message handling and graphics.

**This time the breakthrough could come in multiprocessing, and again it could be led by software.**

**Just as RISC Technology will let Compilers make CPU pipeline more efficient, perhaps a new technology will let compilers make a cluster of CPU more efficient.**

### FIRST 88000-BASED UNIX WORKSTATION APPEARS

The first Unix workstation based on the Motorola 88000 RISC processor has arrived from Opus Systems. The Personal Mainframe Series 8000 is a 17-MIPS machine featuring a dual-processor architecture that pairs the RISC chip with an 80386 I/O subsystem.

The machine brings mainframe technology to the workstation environment, thanks to its I/O subsystem designed by Everex Systems. The subsystem is a dedicated I/O processor capable of simultaneously running Unix and MS-DOS. Opus Systems' 88000-based CPU board handles all system functions.

An important benefit of using the 88000 processor is full binary compatibility with other 88000 Unix systems and products. Motorola's Binary Compatibility Standard provides a low-level specification for interaction with the Unix kernel for all applications running 88000-based Unix. Many vendors, including Opus Systems, have agreed to support this standard. As a result, applications running on the Opus machine will be assured of running on any BCS-compliant 88000-based Unix system, and vice-versa.

The company's port of Unix includes all commands, utilities, and other programs that are part of the standard AT & T release, as well as the portable C compiler and an ANSI-standard Fortran 77.

The system, which is the first product of a strategic alliance between Opus Systems and Everex Systems, will be marketed separately by the two companies. Prices for Opus' Personal Mainframe Series 8000 machine start at $8995, and shipments begin during the second quarter. Everex Systems has yet to announce its pricing structure for its machine.

*Opus Systems, 30863 Stevens Creek, Building 400, Cupertino, CA 95014; (408) 446-8110.* CIRCLE 223

*Everex Systems Inc., 48431 Milmont Dr., Fremont, CA 94538; (415) 498-1111.* CIRCLE 224

## Other RISC's vendors:

-ACORN VL86C010
tech. VLSI Technology - Sanyo
- INTEL: 80960 (for embedded applications)
- AMD: 29000 Family
- HARRIS: TRX2000 (high integrated FORTH executing microcontroller

| Univ: | BERKLEY | STANFORD |
|---|---|---|
| Vendor: | SUN | MIPS Comp. Sys. |
| Type: | SPARC | MIPS (Microp. Without Int. Pip.) |
| - | | |

### Second Sources

- FUJITSU 1.3um 25MHz
- Bipolar Integrated Technology
- Cypress .8um 33MHz

- LSI Logic .9um 25 MHz
- Performance Semicond. .8um 25MHz
- Device Technology

Competitors recognize as long-term microprocessors innovators:

**MOTOROLA, INTEL, AMD**

# CISC < > DSP

- CISC (Complex Instruction Set Computer) architecture use a large amount of hardware complexity to provide high degree of instruction set capability.

  - They are characterized by a large instruction set with some very complex instructions.

  - The length and execution time of instruction is different from one another. Instructions can manipulate bit, byte, word and long word

  - The dynamic bus interface allows for simple, highly efficient access to devices of different data bus width.

  - The latest components of this technology support, directly via BUS Monitoring, Multimaster and Multiprocessor applications.

## Advantages and Disadvantages

- Some instructions needs more then 50 cycles to be executed. However it does have control lines to support a multiprocessing environment and is connectable to different bus width devices. On the other hand, DSP executes most instructions in one cycle.

- Benchmark analysis of CISC, RISC, DSP and Transputer instruction sets measured in MIPS cannot be used to make a meaningful comparison of performance, particularly between machines with different architectures.

  Each MIPS unit should be multiplied by a normalizing factor that reflects overall system performance.

  The real comparison between one component to the other in a particular task, is the time required to execute that particular task.

- RISC and CISC may become more alike in the future. RISC is a technology, a philosophy of design, not a product.

  Some design techniques that have been applied to RISC machine can be applied to CISC architecture to improve performance.

- An example is the National Semiconductor 32532 general purpose processor which incorporates many RISC features. It has:

  - on-chip data and instruction caches
  - direct-mapped caches for stack access
  - pipelining and branch-prediction logic
  - it uses microcode (not used in RISC) for only the most complex instructions and hardwired logic elsewhere.

- Processors like the 32532 with Intel 80486 and Motorola 68040 (expected next year), incorporate more RISC-like features to push the number of cycles for most of the instructions below 2.

  These new features will probably characterize the new type of processor as:

## CRISP

(Complexity-Reduced Instruction Set Processor)

to think how one component will fit better than another.

- An investigation must certainly include the component instruction set, speed and internal architecture that is best suitable for the application algorithms or for the needs of the more general project.

- More decisive in determining the overall throughput in a project, can be the harmonious interaction and inter-communication of the various components rather than the speed of any one single component.

- Thus the real effort in designing a specific application or project should be based on defining

## "THE MULTIPROCESSING ARCHITECTURE"

for the best performance solution to the application or project.

1B - CMOS / HCMOS overview

Currently available products;

| Device | RAM Bytes | ROM K bytes | Pins | I/O | Special features |
|---|---|---|---|---|---|
| MC68HC04J2 | 32 | 1.0 | 20 | 12 | Low cost, 2V operation |
| MC68HC04J3 | 124 | 1.7 | 20 | 12 | Low cost, 2V operation |
| MC68HC04P3 | 124 | 1.7 | 28 | 20 | Low cost, 2V operation |
| MC68HC04P4 | 156 | 3.7 | 28 | 20 | Low cost, 2V operation |
| MC146805E2 | 112 | 0.0 | 40 | 16 | 8K external address range |
| MC146805G2 | 112 | 2.0 | 40 | 32 | |
| MC146705G2 | 112 | 2.0 | 40 | 32 | EPROM |
| [illegible] | 64 | 1.1 | [illegible] | 20 | |
| [illegible] | 64 | 1.1 | [illegible] | 20 | EPROM |
| [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
| MC68HC805C4 | 176 | 4.1 | 40/44 | 31 | [illegible], SCI, SPI, 16-bit timer with 2 I/P cap., 1 O/P comp. |
| MC68HC05C8 | 176 | 7.7 | 40/44 | 31 | SCI, SPI, 16-bit timer with 1 I/P cap., 1 O/P comp. |
| MC68HC805B6 | 176 | 5.9 | 52 | 48 | EEPROM (8K + 256B), SCI, 8 ch. 8-bit A/D, 8 ch. 8-bit PLM, watchdog, 16-bit timer with 2 I/P cap., 2 O/P comp. |
| MC68HC11A8 | 256 | 8.0 | 52/48 | 38/34 | 512B EEPROM, SCI, SPI, 8 ch. 8-bit A/D, 16-bit timer with 5 I/P cap., 5 O/P comp., watchdog, periodic int., 8-bit pulse accumulator |
| MC68HC11A1 | 256 | 0.0 | 52/48 | 20/16 | As 68HC11A8 but without ROM |
| MC68HC11A0 | 256 | 0.0 | 52/48 | 20/16 | As 68HC11A8 but without ROM and EEPROM |
| MC68HC811A2 | 256 | 2.0 | 52/48 | 38/34 | EEPROM, features as 68HC11A8 |

COP = Computer Operating Properly

SPI - serial peripheral interface (synchronous)
SCI - serial communications interface (asynchronous)
I/P cap. - input capture register
O/P comp. - output compare register
PLM - pulse length modulation

## 1C - HCMOS plans

Products due to be introduced during 1988;

| Device | RAM Bytes | ROM K bytes | Pins | I/O | Special features |
|---|---|---|---|---|---|
| MC68HC704P4 | 156 | 3.7 | 28 | 20 | EPROM, low cost, 2V operation |
| MC68HC705C8 | 176 | 7.7 | 40/44 | 31 | EPROM, SCI, SPI, 16-bit timer with 1 I/P cap., 1 O/P comp. |
| MC68HC05C9 | 384 | 15.7 | 40/44 | 31 | SCI, SPI, 16-bit timer with 1 I/P cap., 1 O/P comp. |
| MC68HC05L6 | 176 | 6.0 | 56 | 56 | 96-segment LCD driver, SCI, 16-bit timer with 1 I/P cap., 1 O/P [illegible] |
| [illegible] | 128 | 4.1 | 52/44 | 40/36 | SPI, [illegible] |
| [illegible] | [illegible] | 8.1 | 40/44 | 31 | A/D [illegible] |
| [illegible] | [illegible] | 8.0 | 40/44 | 31 | [illegible] |
| [illegible] | 176 | 8.0 | 52 | 42 | [illegible] 8 ch. [illegible] 2 I/P cap., [illegible] comp. |
| MC68HC05B4 | 176 | 4.0 | 52 | 42 | SCI, 8 ch. 8-bit A/D, 3 ch. 8-bit PLM, [illegible], 16-bit timer with 2 I/P cap., 2 O/P comp. |
| MC68HC11E9 | 512 | 12.0 | 52/48 | 38/34 | 512B EEPROM, SCI, SPI, 8 ch. 8-bit A/D, 16-bit timer with 3/4 I/P cap., 5/6 O/P comp., watchdog, periodic int., 8-bit pulse accumulator |
| MC68HC11E1 | 512 | 0.0 | 52/48 | 20/16 | As 68HC11E9 but without ROM |
| MC68HC11E0 | 512 | 0.0 | 52/48 | 20/16 | As 68HC11E9 but without [illegible] and EEPROM |
| MC68HC811E2 | 512 | 2.0 | 52/48 | 20/16 | As 68HC11E9 but with 2K EEPROM, no ROM |

As the MCU guide is only updated once per year it is not possible to include the exact introduction dates. These can be obtained from your product marketer or a separate product introduction list.

SPI - serial peripheral interface (synchroneus)
SCI - serial communications interface (asynchronous)
I/P cap. - input capture register
O/P comp. - output compare register
PLM - pulse length modulation

## MOTOROLA

| | |
|---|---|
| MC28HC14 | 256-BYTE EEPROM IN 8-PIN DIP |
| MC14021 | 8-BIT INPUT PORT |
| MC74HC165 | 8-BIT INPUT PORT |
| MC74LS165 | 8-BIT INPUT PORT |
| MC74HC595 | 8-BIT OUTPUT PORT |
| MC74LS673 | 16-BIT OUTPUT PORT |
| MC144115 | 16 SEGMENT NON-MUXED LCD DRIVER |
| MC144117 | 4 DIGIT LCD DRIVER (X2 MUX) |
| MC14549 | A/D CONVERTER SUCCESSIVE APPROXIMATION REGISTER |
| MC14559 | A/D CONVERTER SUCCESSIVE APPROXIMATION REGISTER |
| MC14489 | [illegible] |
| MC144116 | [illegible] |
| MC144111 | [illegible] |
| [illegible] | [illegible] |
| [illegible] | [illegible] |
| [illegible] | [illegible] |
| MC145040 | A/D [illegible] |
| MC145041 | A/D CONVERTER [illegible] |
| MC145155 | PLL 14-BIT PROG. DIV-BY-R COUNTER (RANGE = 3 TO 16383) |
| MC145156 | PLL 7-BIT/10-BIT PROG. DIV-BY-N COUNTERS (RANGE = 8 TO 1023 AND 0 TO 127) |
| MC145157 | PLL DUAL 14 BIT PROG. DIV-BY-N COUNTERS (RANGE = 8 TO 16383 AND 3 TO 16383) |
| MC145158 | PLL 14-BIT/10-BIT PROG. DIV-BY-N COUNTERS (RANGE = 8 TO 16383 AND 3 TO 1023) |
| MC145159 | PLL 14-BIT/10BIT/7-BIT PROG. DIV-BY-N COUNTERS (RANGE 3 TO16383 AND 16 TO 1023 AND 1 TO 127) |
| MC145453 | LCD DRIVER (39 NON-MUXED SEGMENTS) (4 1/2 DIGITS BY 7-SEGMENTS MAY BE PARALLED FOR MORE DIGITS) |

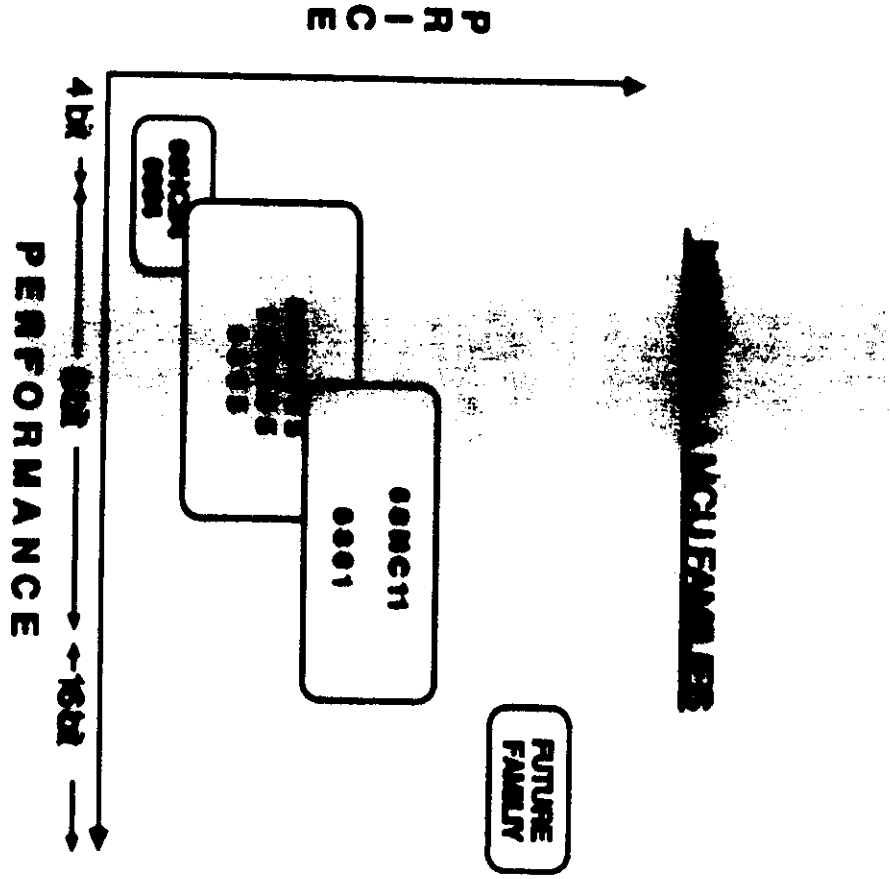
## RCA

| | |
|---|---|
| CDP68HC68A1 | 10-BIT A/D CONVERTER |
| CDP68HC68R1 | 128 X 8-BIT STATIC RAM |
| CDP68HC68R2 | 256 X 8-BIT STATIC RAM |
| CDP68HC68T1 | REAL TIME CLOCK |

## SIGNETICS

| | | |
|---|---|---|
| * | PCx2100 | 40 SEGMENT LCD DUPLEX DRIVER |
| * | PCx2110 | 60 SEGMENT LCD DUPLEX DRIVER |
| * | PCx2111 | 64 SEGMENT LCD DUPLEX DRIVER |
| * | PCx2112 | 32 SEGMENT LCD STATIC DRIVER |
| | PCx3311 | DTMF GENERATOR WITH PARALLEL INPUTS |
| | PCx3312 | DTMF GENERATOR WITH I²C BUS INPUTS |
| | PCx8570A | 256 X 8 STATRRIC RAM |
| | PCx8571 | 128 X 8 STATRRIC RAM |
| | PCx8573 | CLOCK / TIMER |
| | PCx8574 | 8-BIT REMOTE I/O EXPANDER |
| | PCx8576 | 1 : 4 MUX LCD DRIVER |
| | PCx8591 | 8-BIT, 4-CHANNEL A/D CONVERTER AND A D/A CONVERTER |
| * | SAA[illegible] | PLL TUNING [illegible] |
| * | SA[illegible] | 32 [illegible] |
| * | SA[illegible] | 16 [illegible] |
| * | SA[illegible]A | 20 [illegible] |
| * | SAA[illegible] | FLOURESCENT [illegible] DRIVER |
| | SAA1250 | SWITCHING CIRCUIT |
| * | SAA3019 | CLOCK / TIMER |
| | SAA3028 | I/R TRANSCODER |
| * | SAB3013 | HEX 6-BIT D/A CONVERTER |
| | SAB3035 | PLL DIGITAL TUNING CIRCUIT WITH 8 D/A CONVERTERS |
| | SAB3036 | PLL DIGITAL TUNING CIRCUIT |
| | SAB3037 | PLL DIGITAL TUNING CIRCUIT WITH 4 D/A CONVERTERS |
| | SAA5240 | TELETEXT CONTROLLER CHIP - 625 LINE SYSTEM |
| | TDA3820 | DIGITAL STEREO SOUND CONTROL IC |
| | TEA6000 | MUSTI: FM/IF SYSTEM |
| * | TDA1534A | 14-BIT A/D CONVERTER - SERIAL OUTPUT |
| * | TDA1540P, D | 14-BIT A/D CONVERTER - SERIAL INPUT |
| * | NE5036 | 8-BIT A/D CONVERTER - SERIAL OUTPUT |

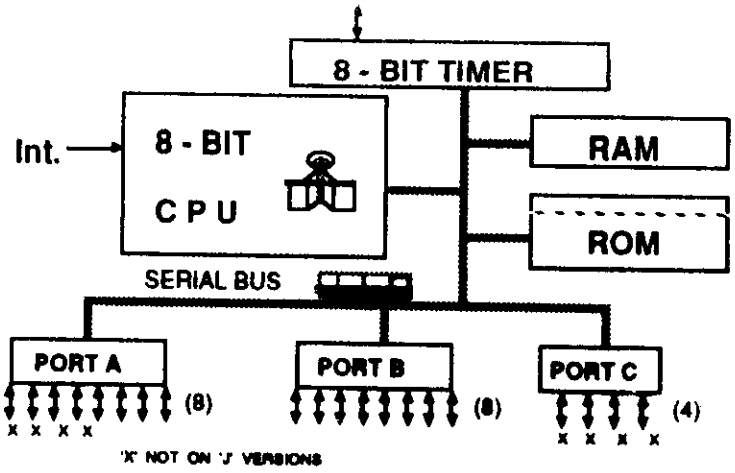
* = REQUIRES AN EXTRA ENABLE LINE
ABOVE PARTS, EXCEPT THE PCx8591 AND TEA6000, ARE AVAILABLE TODAY




## SPRAGUE

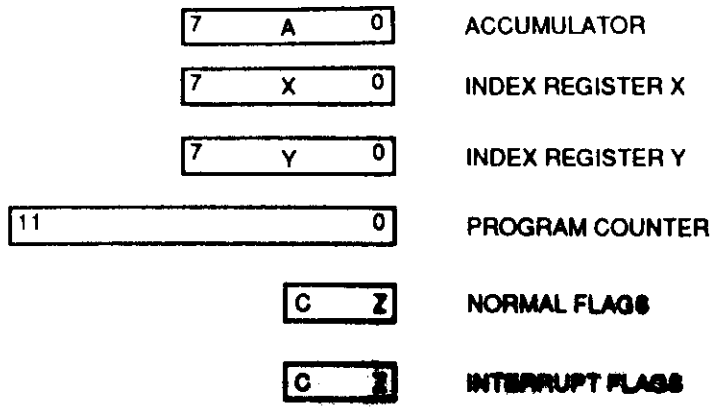
| | |
|---|---|
| UCN4810/5810 SERIES | POWER DRIVING OUTPUT PORTS |
| UAA2022/2023 | 16-BIT POWER OUTPUT PORT |

## M68HC04 BLOCK DIAGRAM






## CPU

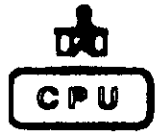





## 68HC04

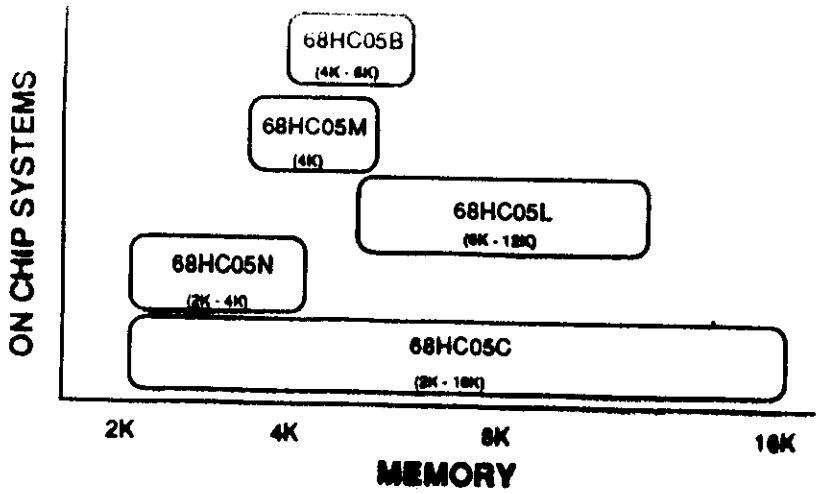
- HCMOS Technology
- 64-156 Bytes RAM
- 1-4K Bytes ROM
- 4K Bytes EPROM Version
- 20 & 28 PIN Packages - DIL & PLCC
- Low-Power Stop & Wait Modes
- Self-Test
- Serial Address & Data Buses (for small die size)
- Parallel 8-Bit Peripheral Interface
- Similarity to 6800/6805 Software






## INSTRUCTION SET

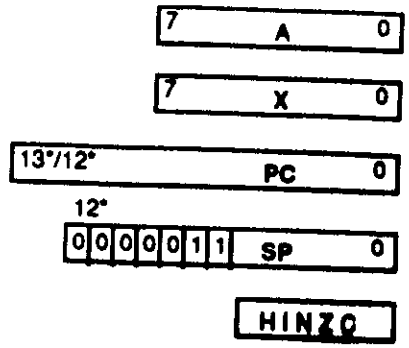
- Indexed addressing
- Power saving - stop and wait
- True bit manipulation - bit set, clear, test
- Move immediate
- Arithmetic and logical intructions






## CPU

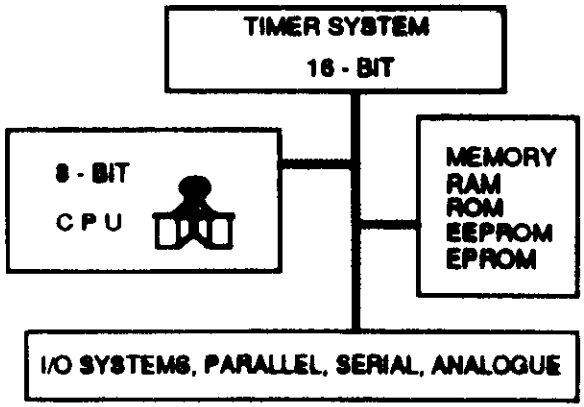
CPU




* may vary on some versions



## 68HC05 GENERAL BLOCK DIAGRAM






CPU

## INSTRUCTION SET

Multiply *

Indexed addressing

Power saving - stop and wait

True bit manipulation - bit set, clear, test

Int. pin test

Arithmetic and logical instructions

## ADDRESS AND DATA BUS

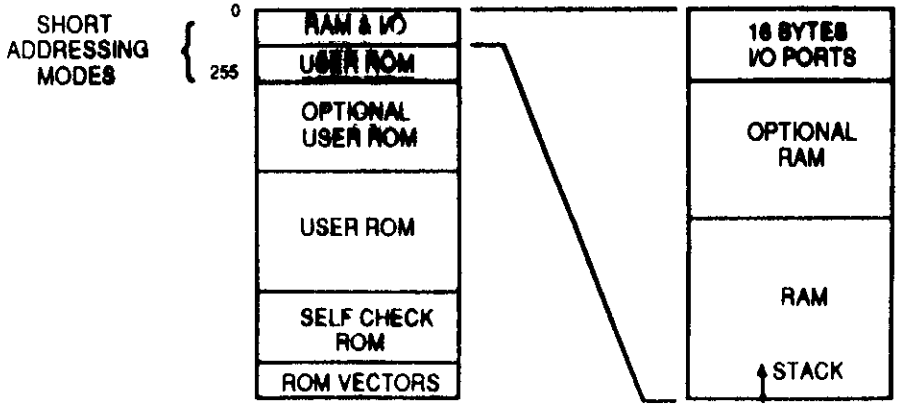
Internal only, NOT expandable

0 to 2.1 MHz bus speed, up to 4.2 MHz as option

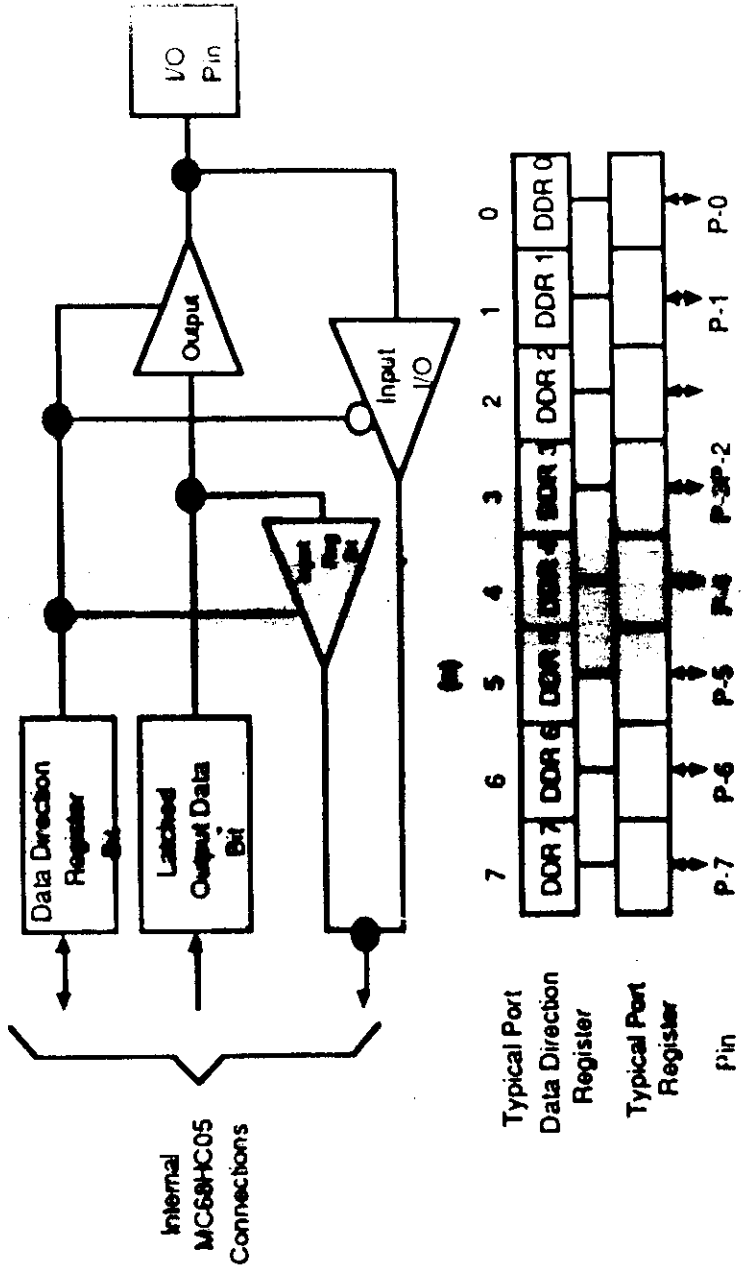
Address range up to 16K implemented,

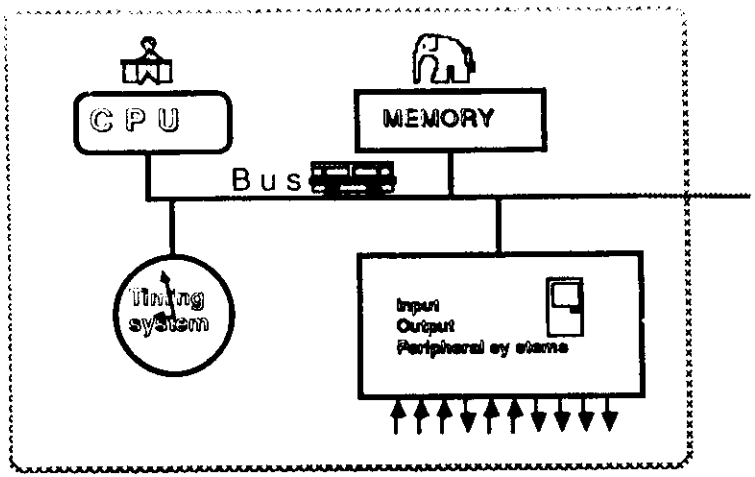
architectural limit 64K

## M68HC05 MEMORY MAP

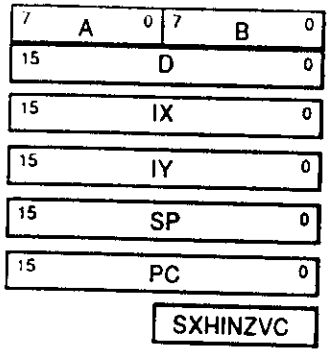



## PARALLEL PORT I/O CIRCUIT






RB & SH

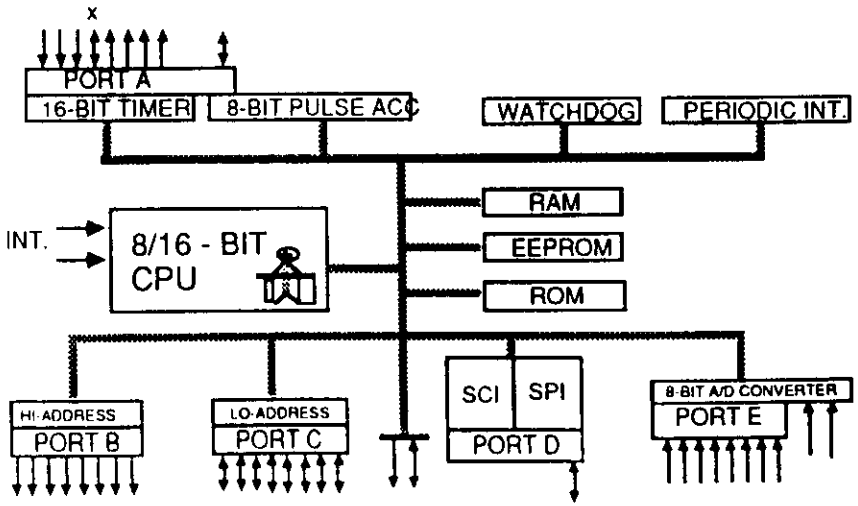



ACCUMULATOR
INDEX REGISTER X
INDEX REGISTER Y
STACK POINTER
PROGRAM COUNTER
CONDITION CODES

RB & SH



## 68HC11A/E BLOCK DIAGRAM






CPU

## INSTRUCTION SET

Divide and Multiply / *
16 bit compare, exchange, arithmetic, load/store
Power saving - stop and wait
Multiple bit manipulation - set, clear, test

"C" compiler

M68HC11

Timing system

RB & SH

INPUT CAPTURE

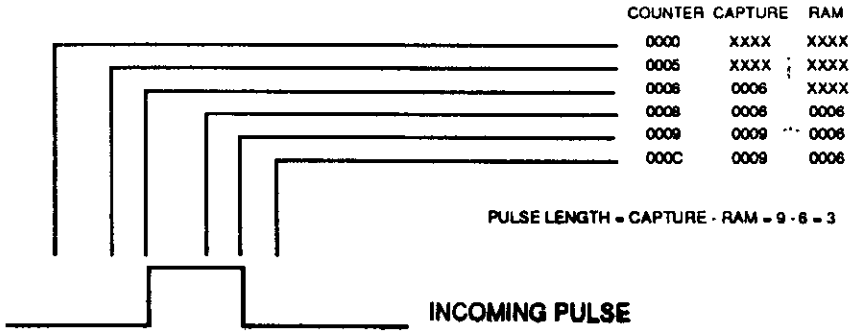



M68HC11

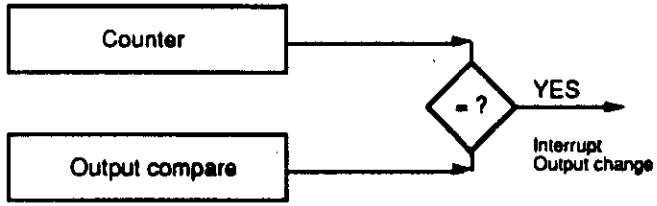
Timing system

## INPUT CAPTURE

# INPUT CAPTURE

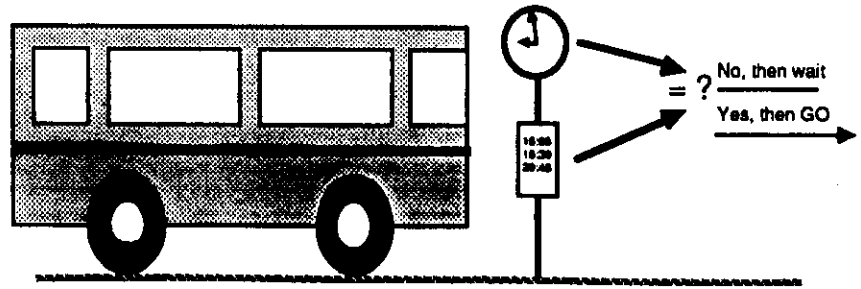



# OUTPUT COMPARE



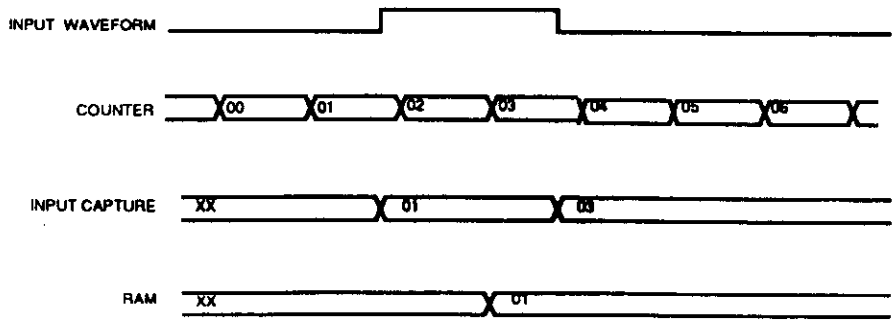



# OUTPUT COMPARE



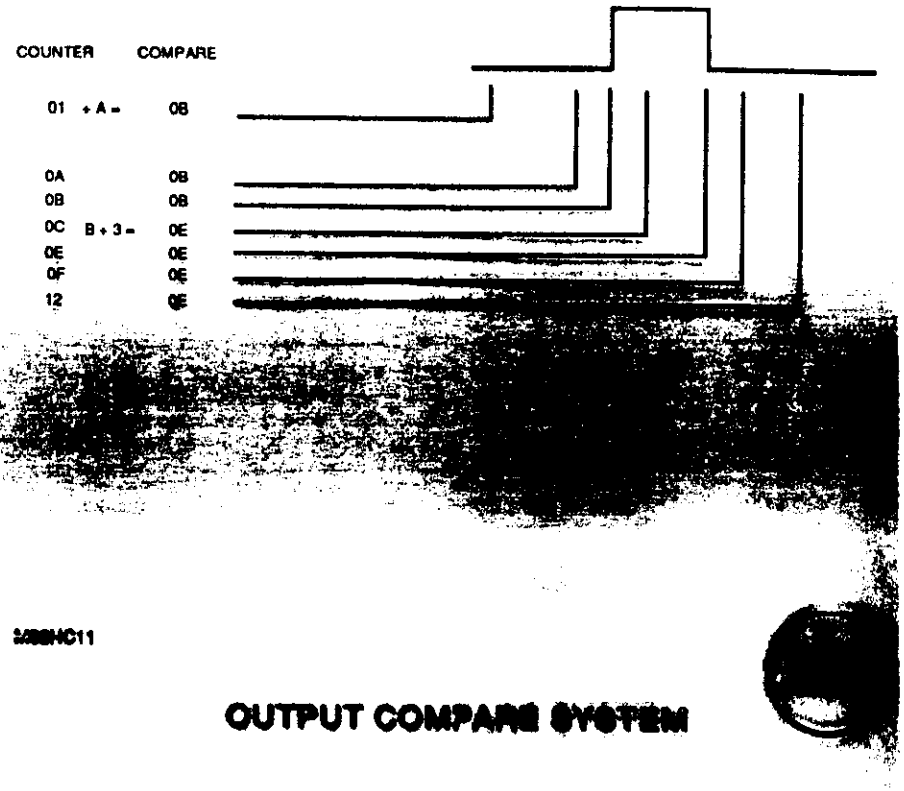



# OUTPUT COMPARE



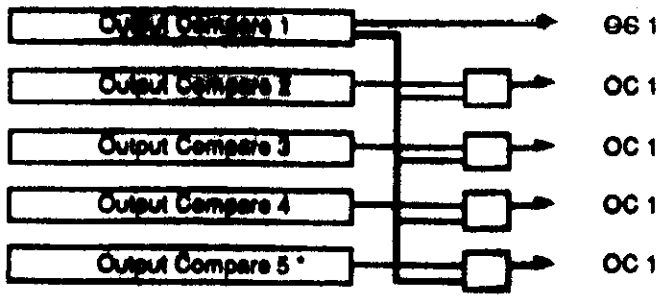
# OUTPUT COMPARE

OUTPUT A 3 µS PULSE IN 10 µS FROM NOW




M68HC11

## OUTPUT COMPARE SYSTEM



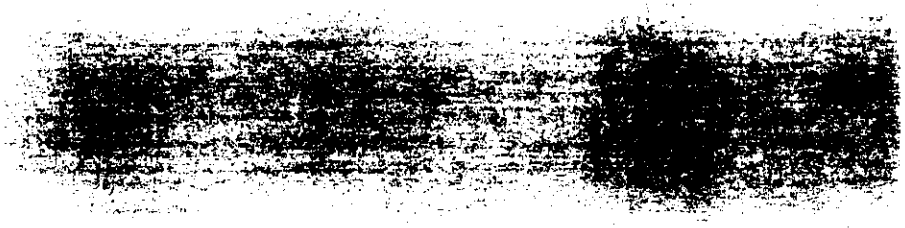

* Preprogrammable to Input Capture

PERIODIC INTERRUPT:

Software programmable for (@ 2 MHz bus);
4.10 mS, 8.19 mS, 16.38 mS, 32.77 mS int. rate

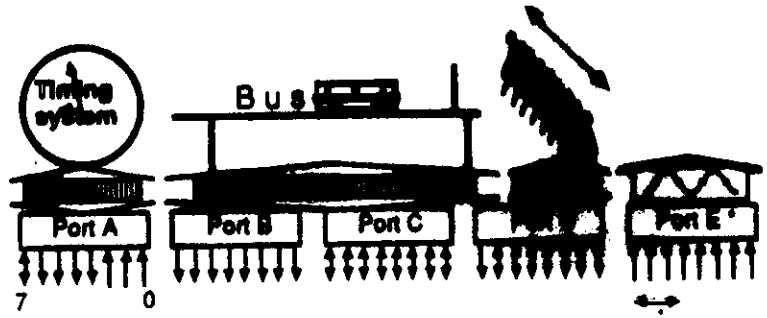
WATCHDOG:

Software programmable for (@ 2 MHz bus);
16.38 mS, 65.54 mS, 262.14 mS, 1.05 S int. rate



M68HC11

## INPUT / OUTPUT PORTS AND SPECIAL FUNCTIONS



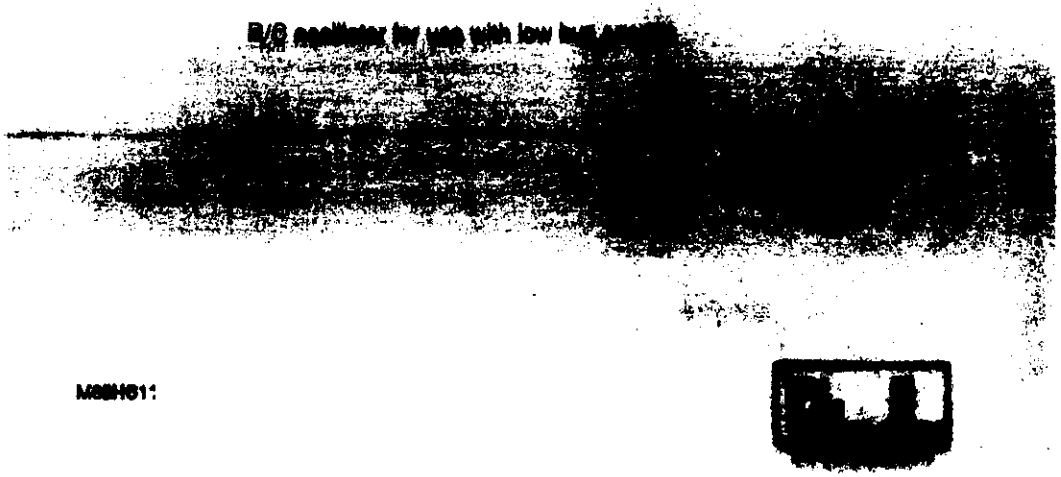
## ANALOGUE TO DIGITAL CONVERTER

Successive approximation with capacitive ladder

8 (4) analogue input channels

8 bit resolution, max. total error +/- 1 LSB

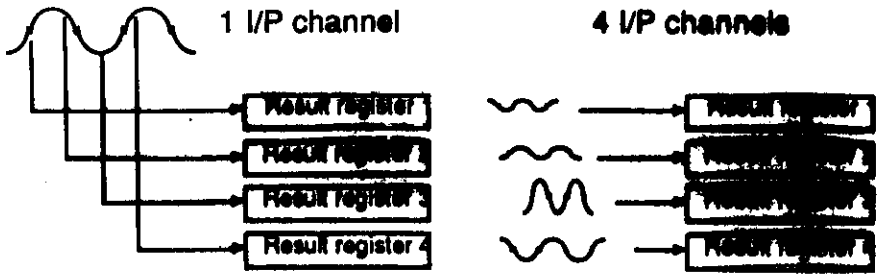
4 result registers







## ANALOGUE TO DIGITAL CONVERTER




## 68HC11 EMULATION SUPPORT

- Very low cost with evaluation board, EVB
- Low cost evaluation module, EVM
- Full feature emulation with HDS300 plus personality
- Third party emulation suppliers
- 68HC811 and 68HC711 versions



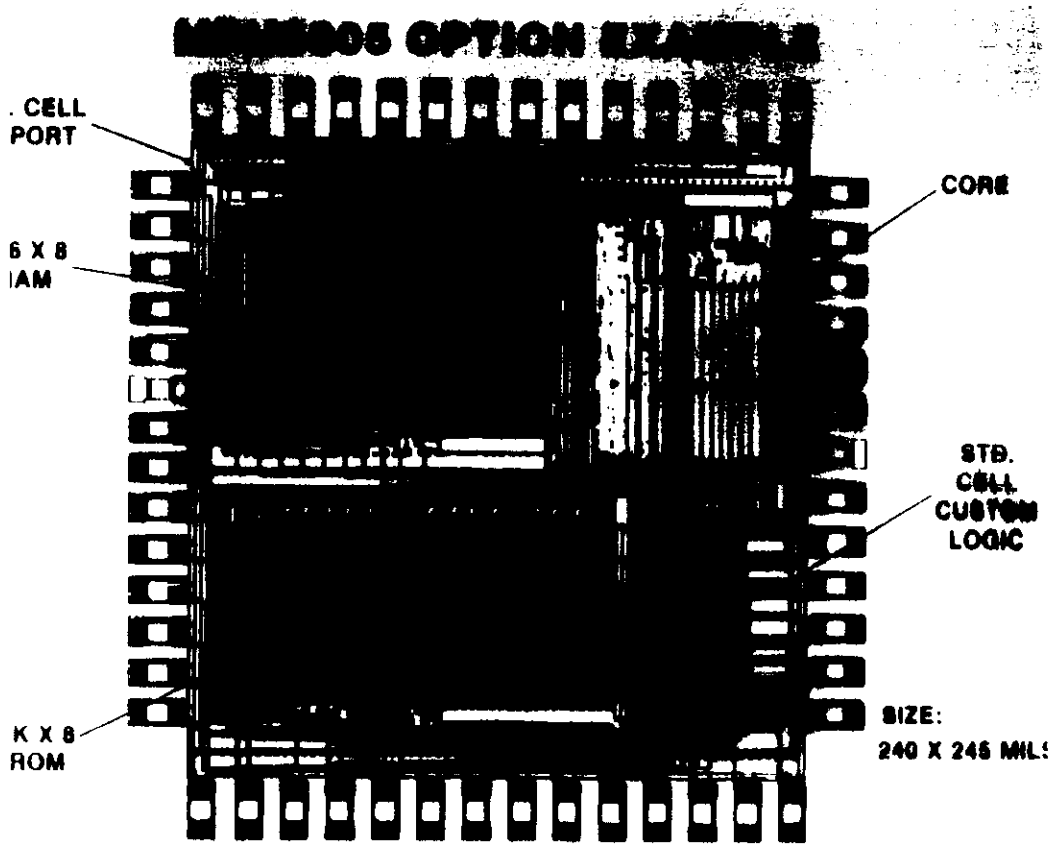
## LITERATURE SUPPORT

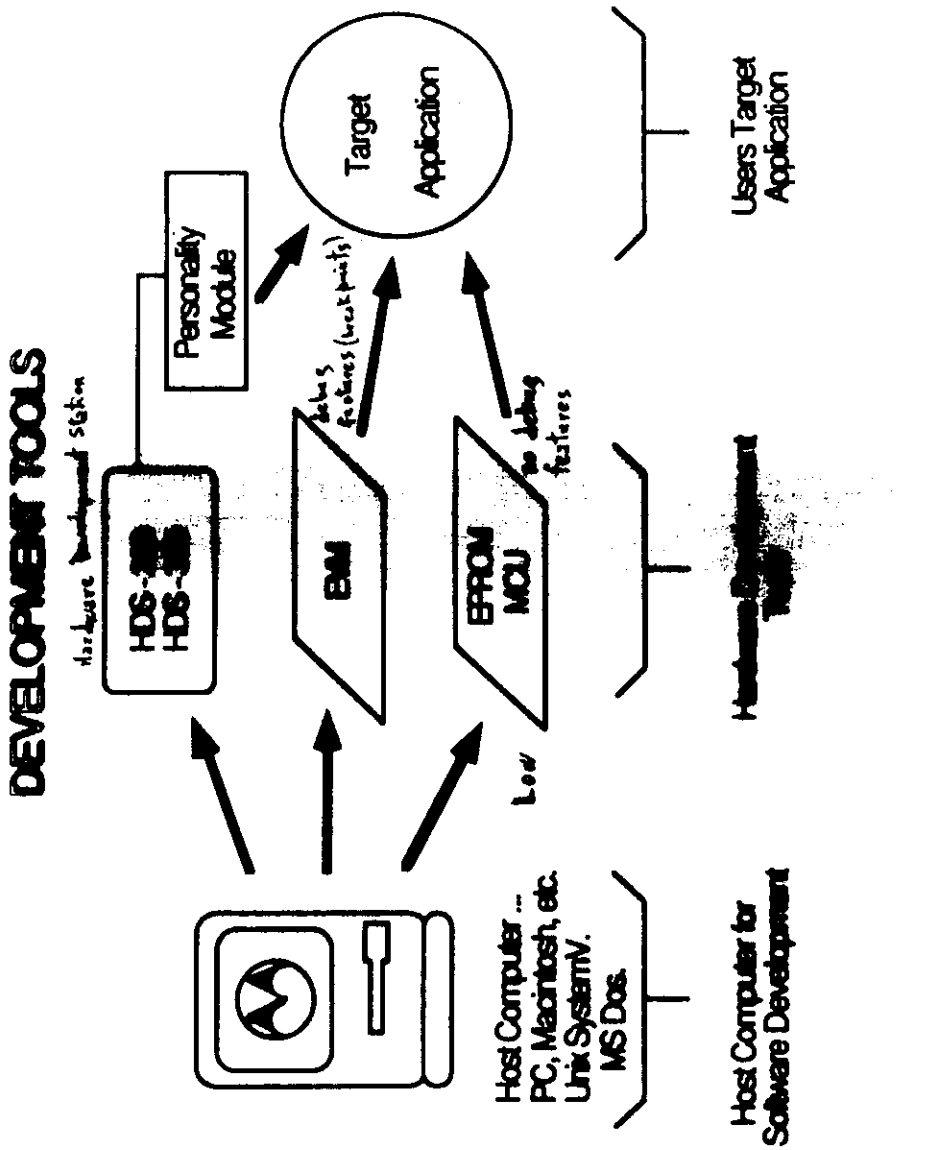
- 68HC11A8 DATA BOOK
- 68HC11A8 PROGRAMMERS' REFERENCE MANUAL
- 68HC11A8 PROGRAMMERS' REFERENCE GUIDE
- 68HC11 COOL AND POWERFUL BROCHURE
- 68HC24 DATA SHEET
- 68HC11 EVB BROCHURE
- 68HC11 EVM BROCHURE
- HDS300 BROCHURE
- THIRD PARTY SOFTWARE SUPPORT INFO SHEET
- THIRD PARTY HARDWARE SUPPORT INFO SHEET

## JSTOMER SPECIFIC VERSIONS

ne above functions or their combination does not suit your requirements it is possibl
o make a specific design based on Motorola ASIC standard cell or gate arra
oducts. The 68HC05 CPU and a number of the peripheral functions are available i
he standard cell library. This makes it possible to design a 68HC05 based MCU
xactly to your rquirements.

For larger systems it may be more effective to use a standard MCU with an expansior
us such as the MC68HC11A1, and an ASIC product for all the special functions tha
re not available on the MCU. This option makes it also easy to use inexpensive
xternal memory for large programs.



DEVELOPMENT TOOLS




DEVELOPMENT TOOLS




DEVELOPMENT TOOLS

The following diagram shows a possible MCU based application to ilustrate some c
the additional components that may be used with the MCU.