ECT342 Embedded Systems

Dr. Sajesh Kumar U.

Thumb Instruction Set

- Thumb entry in CPSR -5th bit in CPSR is Thumb bit
- Thumb entry--- Using BX instructions, Using special instructions when returning from exceptions
- Thumb exit---Explicit switch to ARM instructions using BX thumb instruction. Implicit change to ARM execution occurs when returning from exceptions

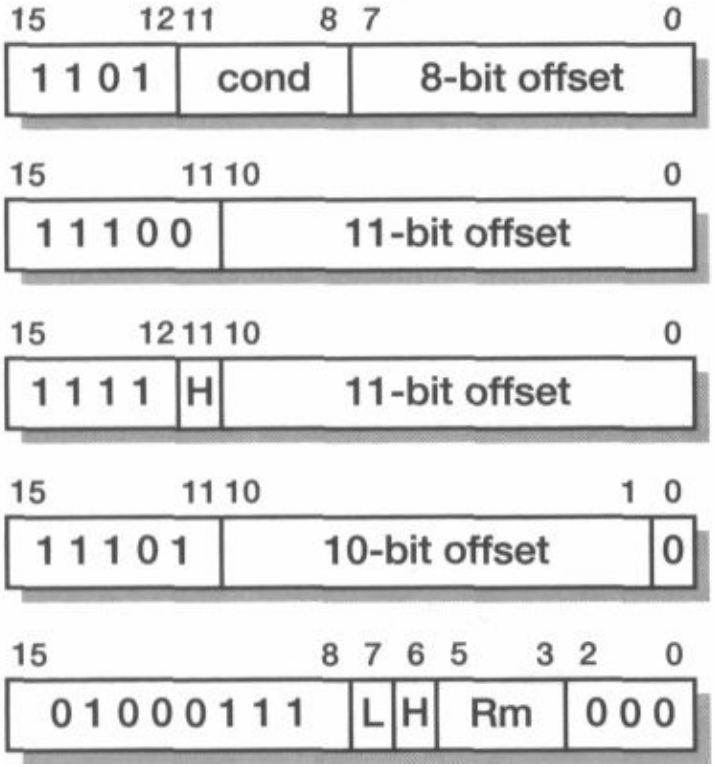
Thumb Programmers Model

- Registers R0 to R7 are used
- R13 (stack pointer), R14 (Link register), R15 (program counter) for special purposes
- R8 to R12 and CPSR have restricted access
- Instructions are 16 bit long
- CPSR flags are changed by arithmetic and logic instructions and control conditional branching instructions

Thumb Programmers Model

- Thumb instructions execute unconditionally
- Many thumb data processing instructions use 2 address format.
- Instruction format is less regular than ARM
- Return from exceptions involve return address adjustments to compensate the ARM pipeline behavior

Thumb Branch Instructions



(1) `B<cond> <label>`

(2) `B <label>`

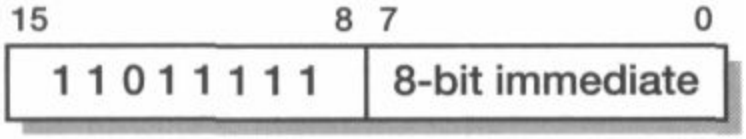
(3) `BL <label>`

(3a) `BLX <label>`

(4) `B{L}X Rm`

- BL offset can be extended to 22 bits by using 2 instructions.
- ARM supports word offsets whereas Thumb supports half word offsets

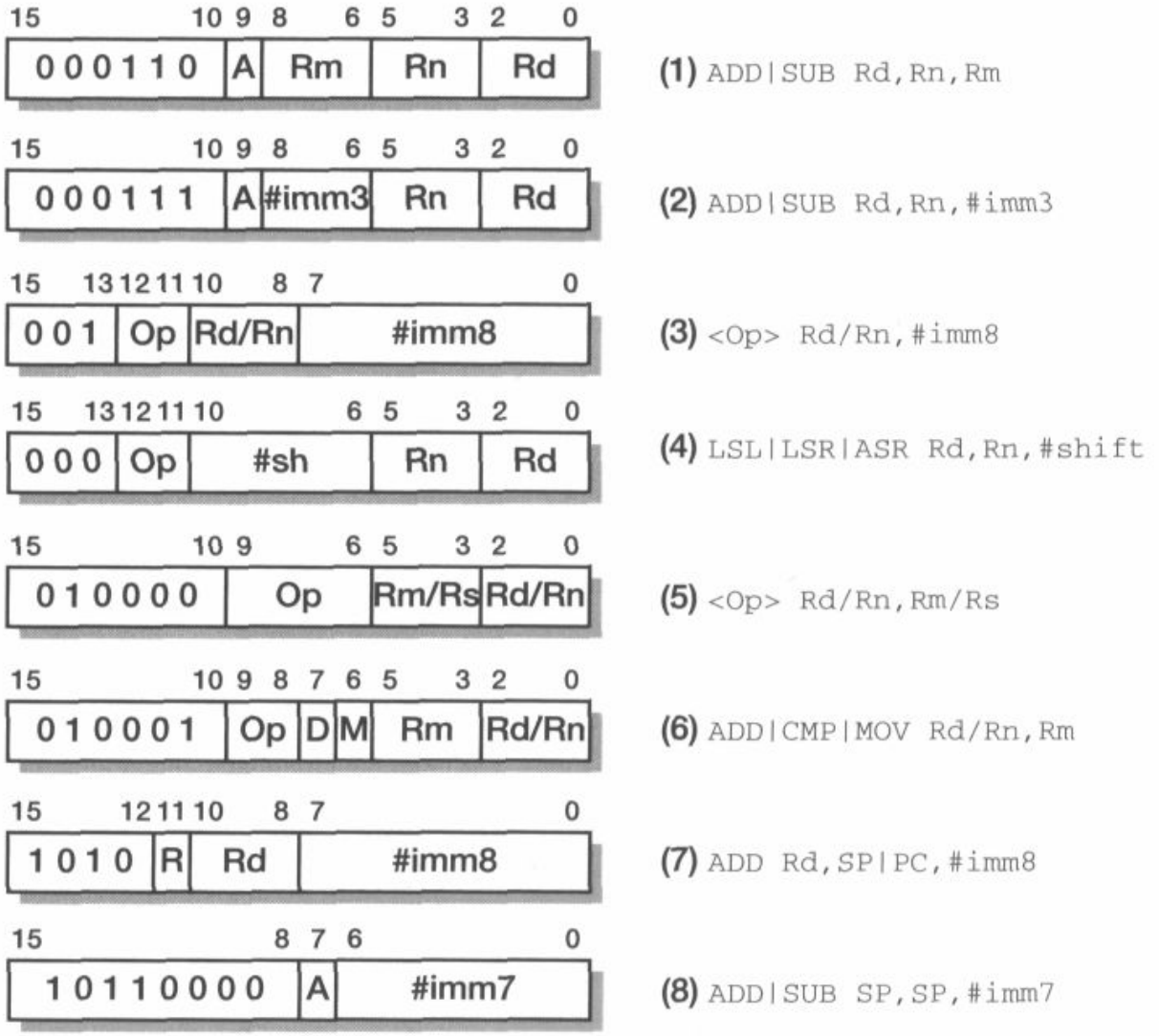
Thumb Software Interrupt Instruction



SWI 8bit immediate

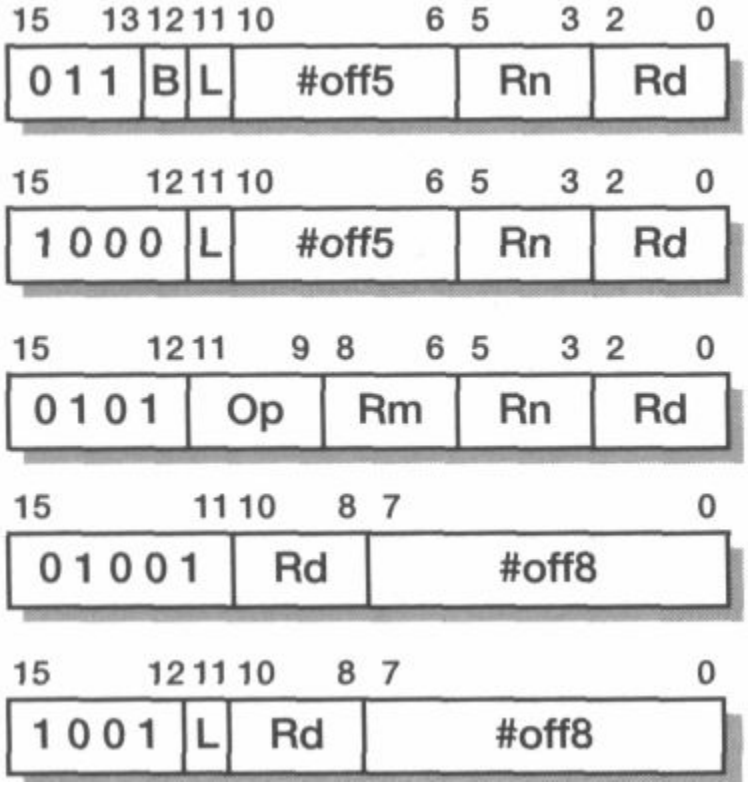
- Address of next thumb instruction is stored in R14_svc, CPSR in SPSR_svc
- IRQ is disabled
- PC is forced to 0x08 location
- ARM SWI handler is entered as exceptions clear thumb mode

Thumb Data Processing Instructions



- Shift and ALU operations in separate instructions
- All data processing operations set the flags if operated on R0 to R7.
- Only CMP sets flags if operated on Hi registers

Thumb Single register transfer Instruction



(1) `LDR|STR{B} Rd,[Rn,#off5]`

(2) `LDRH|STRH Rd,[Rn,#off5]`

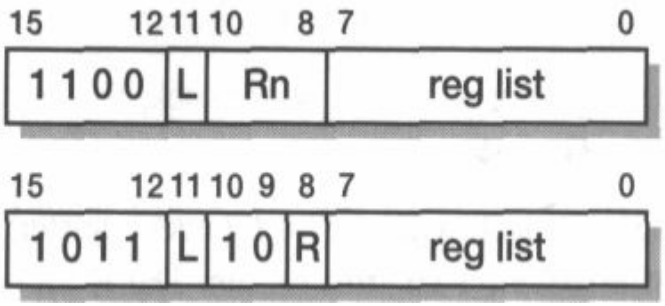
(3) `LDR|STR{S}{H|B} Rd,[Rn,Rm]`

(4) `LDR Rd,[PC,#off8]`

(5) `LDR|STR Rd,[SP,#off8]`

Same as ARM instructions

Thumb multiple register transfer Instruction



(1) LDMIA|STMIA Rn!, {<reg list>}

(2) POP|PUSH {<reg list>{,R}}

- PUSH, POP instructions
- Bottom bit of a loaded PC updates thumb bit

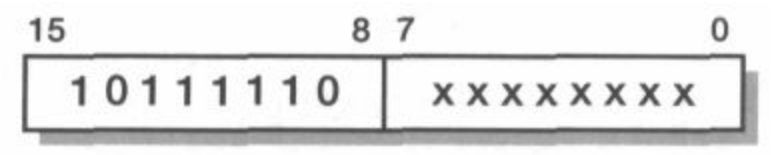
LDMIA Rn!, {<reg list>)

STMIA Rn!, {<reg list>}

POP {<reg list>{, pc}}

PUSH {<reg list>{, lr}}

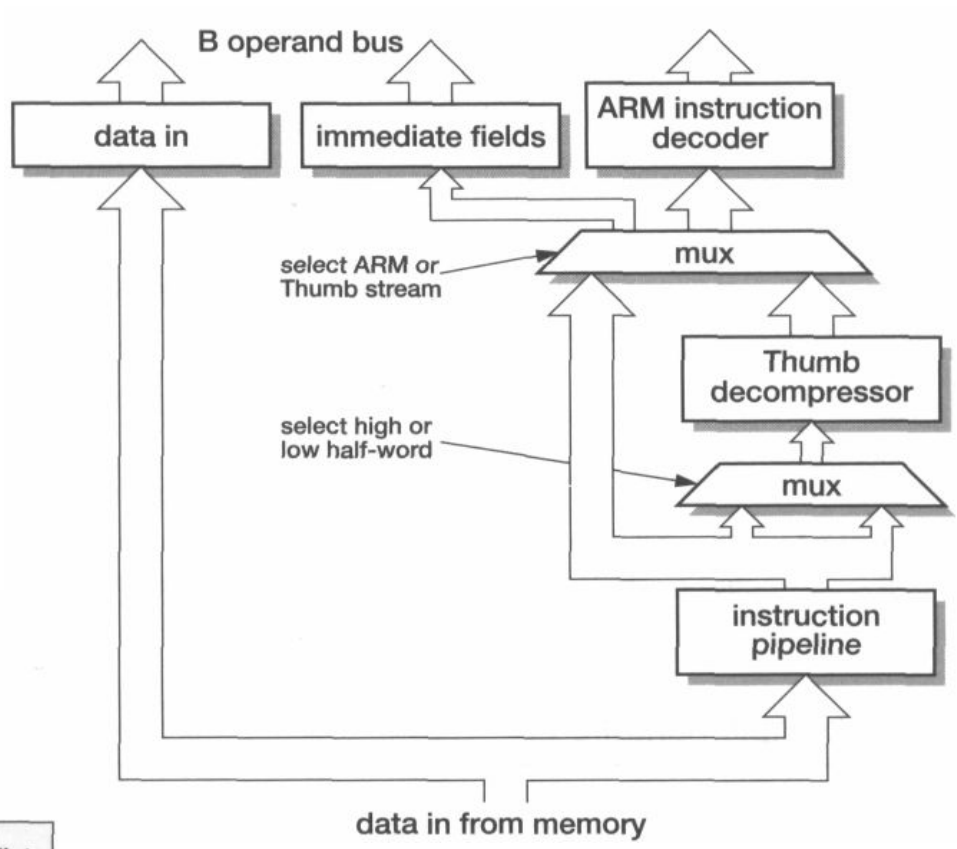
Thumb Break Point Instruction

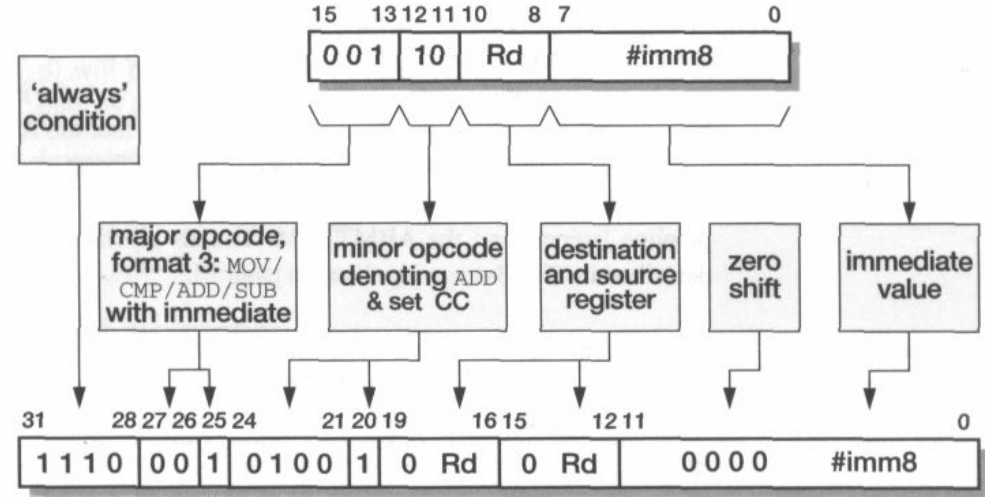


- BKPT
- Processor takes prefetch abort for debugging

Thumb Implementation

- Thumb de-compressor
- 16 bit thumb instruction to 32 bit ARM instruction





Thumb Implementation

- In thumb only conditional codes are branch instructions
- Data processing instructions modifying Conditional flags is implicit in Thumb
- Thumb 2 address format to ARM 3 address format by replicating a register specifier

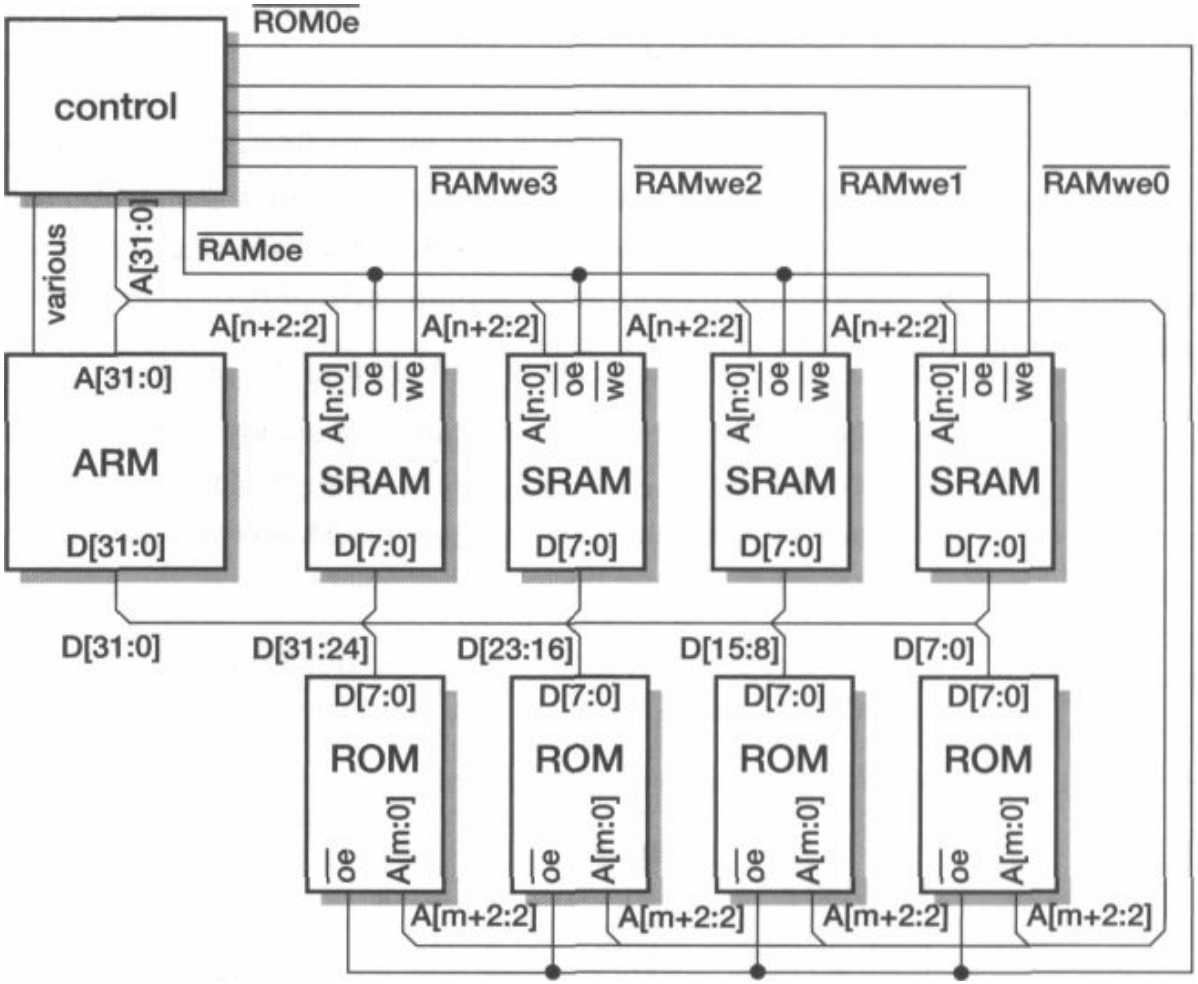
Thumb Applications

- Thumb uses 70% of the ARM code
- Thumb uses 40% more instructions
- With 32 bit memory ARM code is faster and with 16 bit Thumb is faster
- Thumb uses 30% less external memory power than ARM code

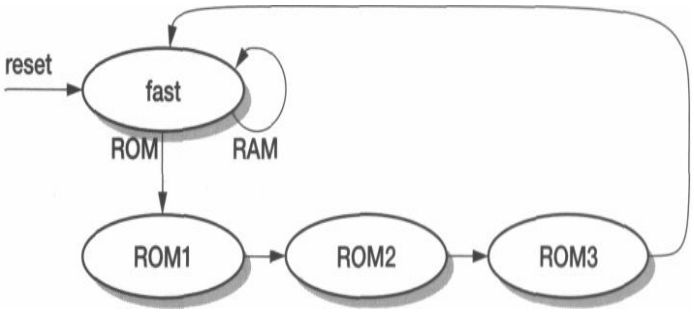
Architectural support for system development

- Memory interface
- 32 bit address bus and 32 bit bidirectional data bus
- Signal for memory needed (mreq), sequential address (seq) for memory control logic
- R/W direction and size of data for transfer
- Bus timing and control signals

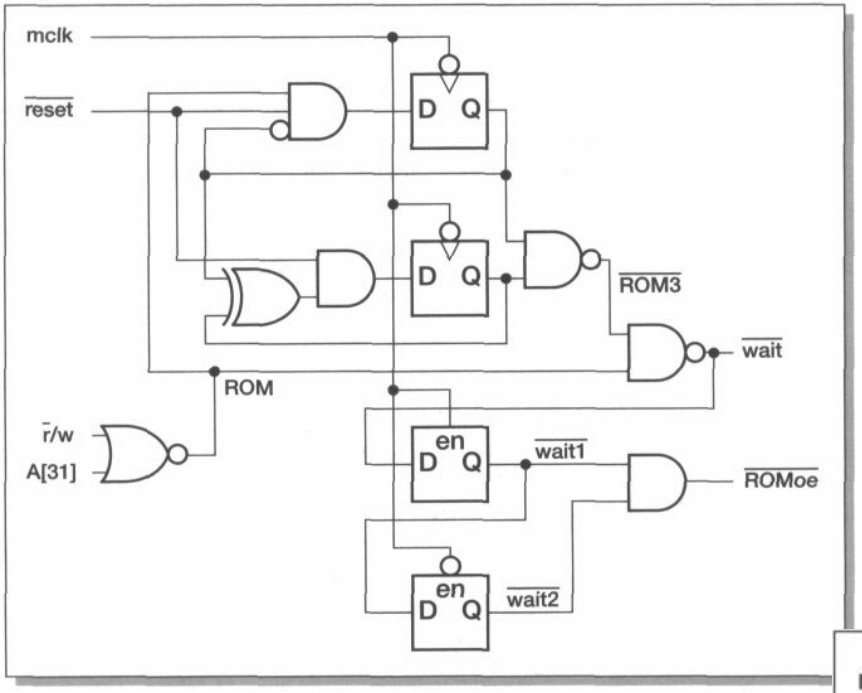
Memory Interface

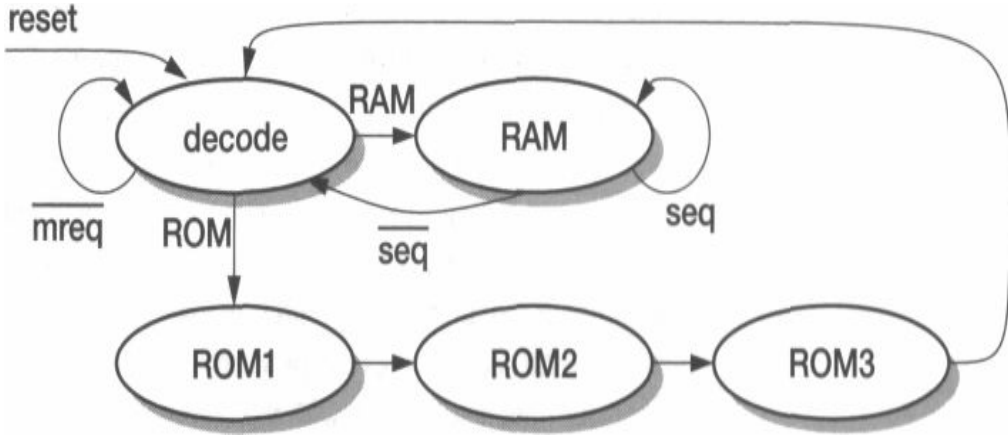


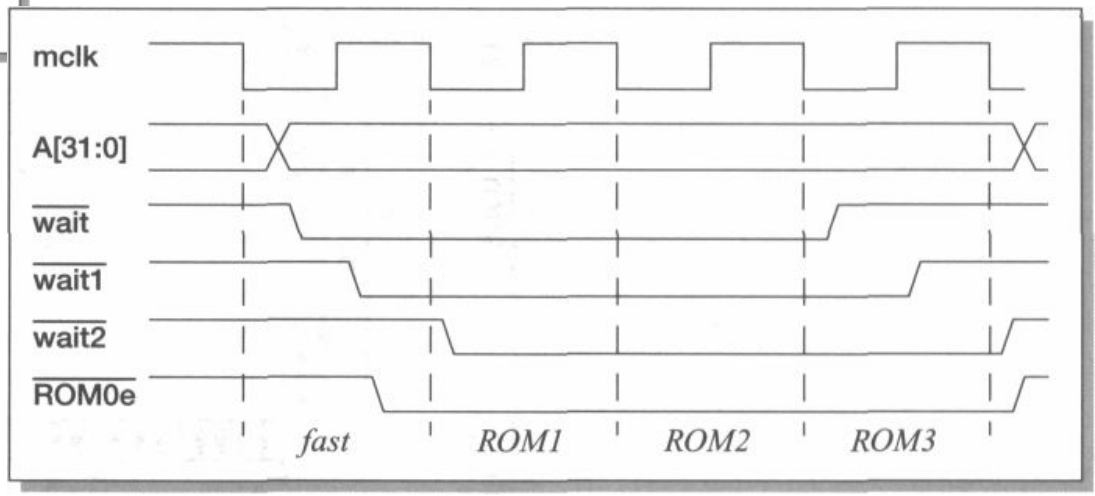
- **ROM wait states**



Memory Wait States and Address Decoding

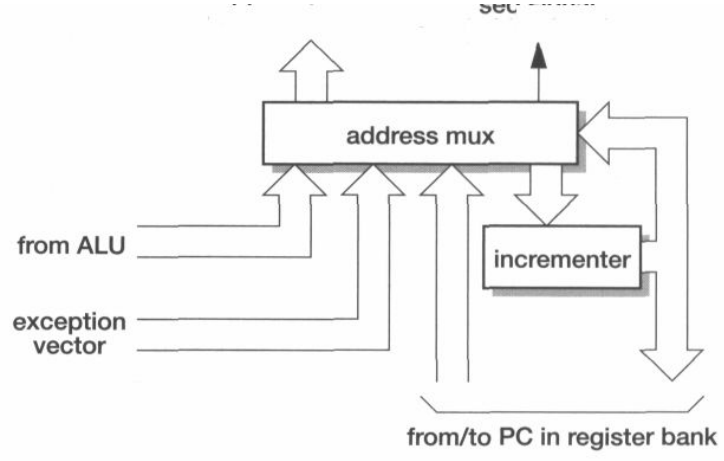


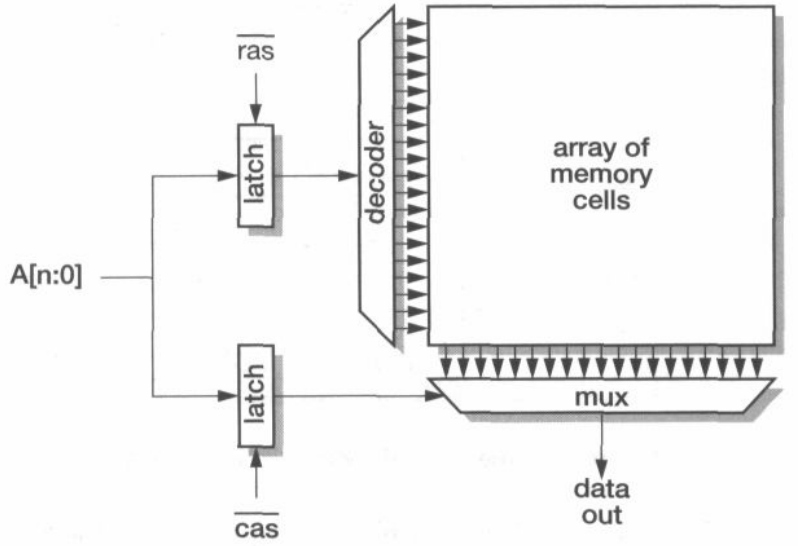


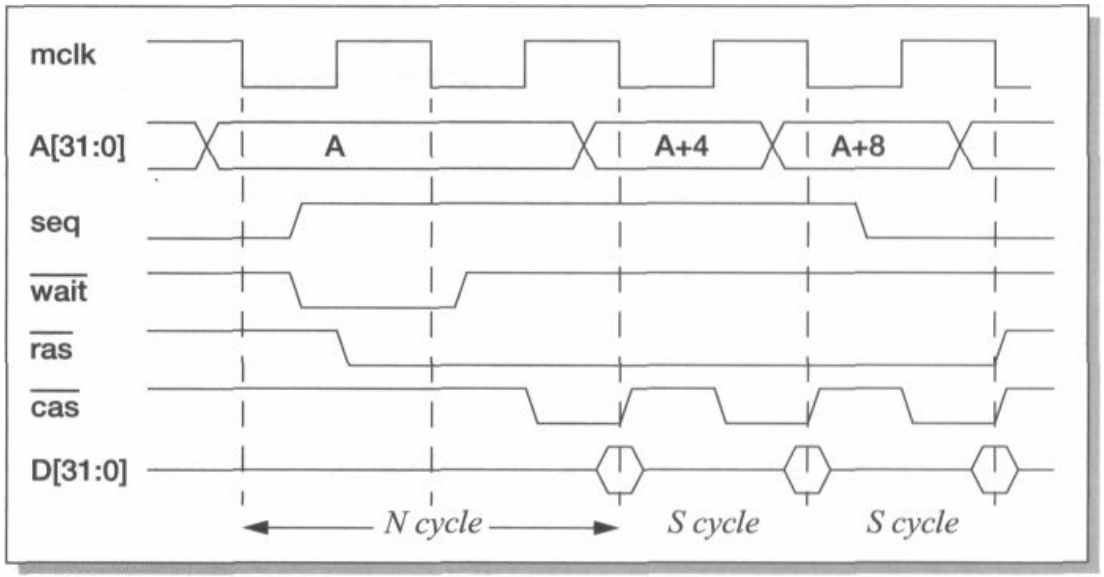


DRAM and Address Incrementer

- **DRAM is low cost**

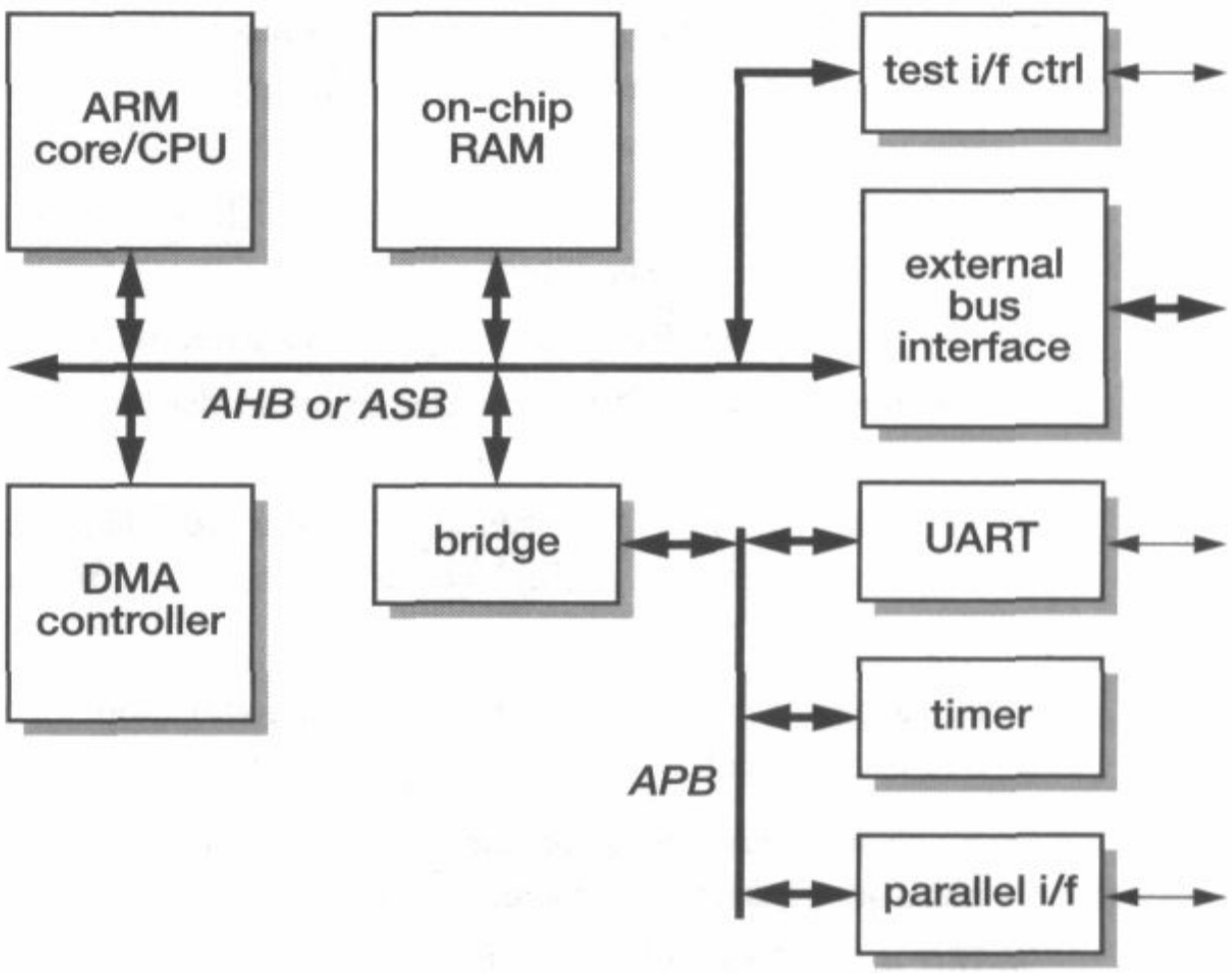






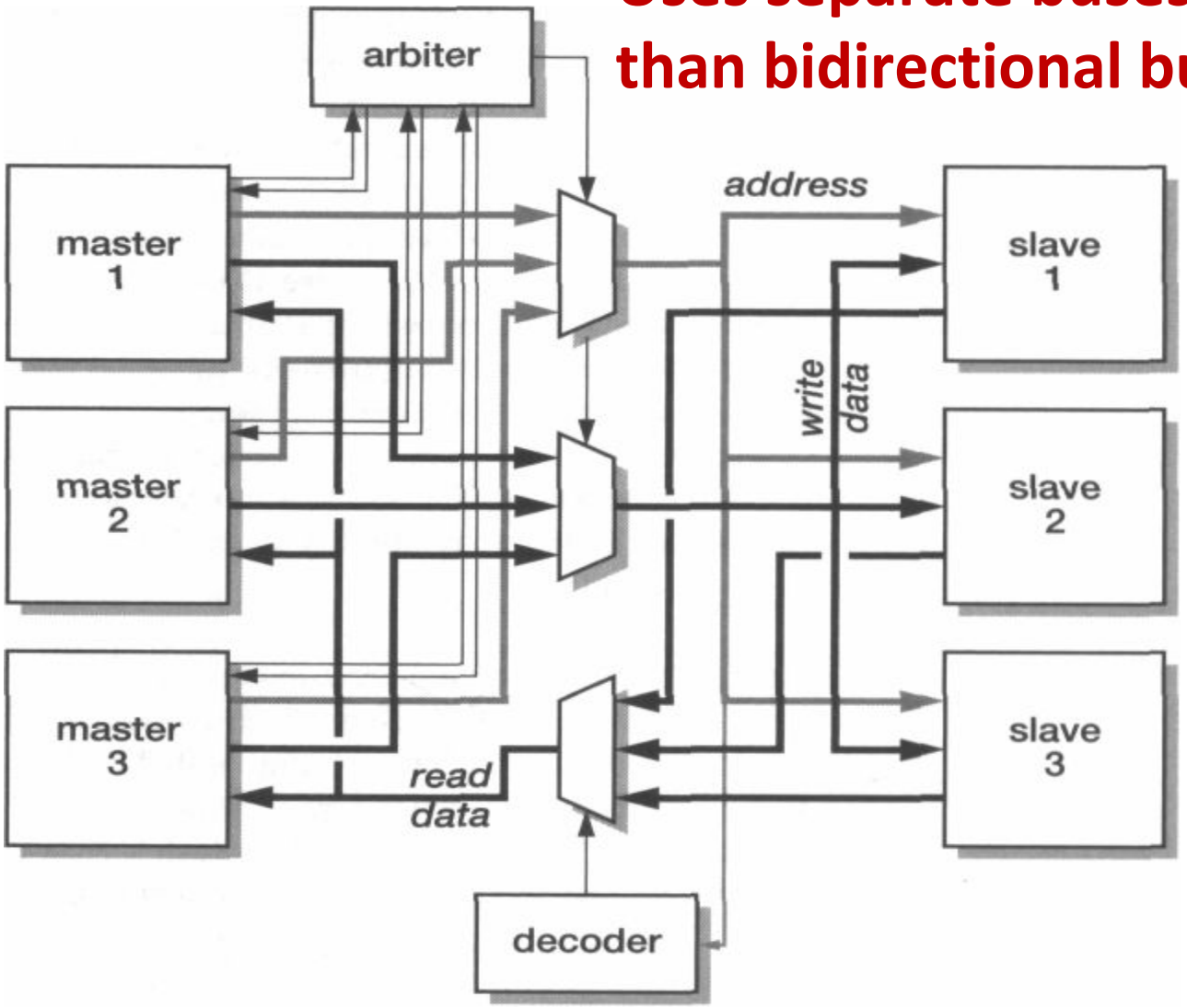
Advanced Microcontroller bus Architecture (AMBA) Bus

- AHB
- ASB
- APB



Advanced High Performance Bus (AHB)

Uses separate buses for data rather than bidirectional buses with tri states

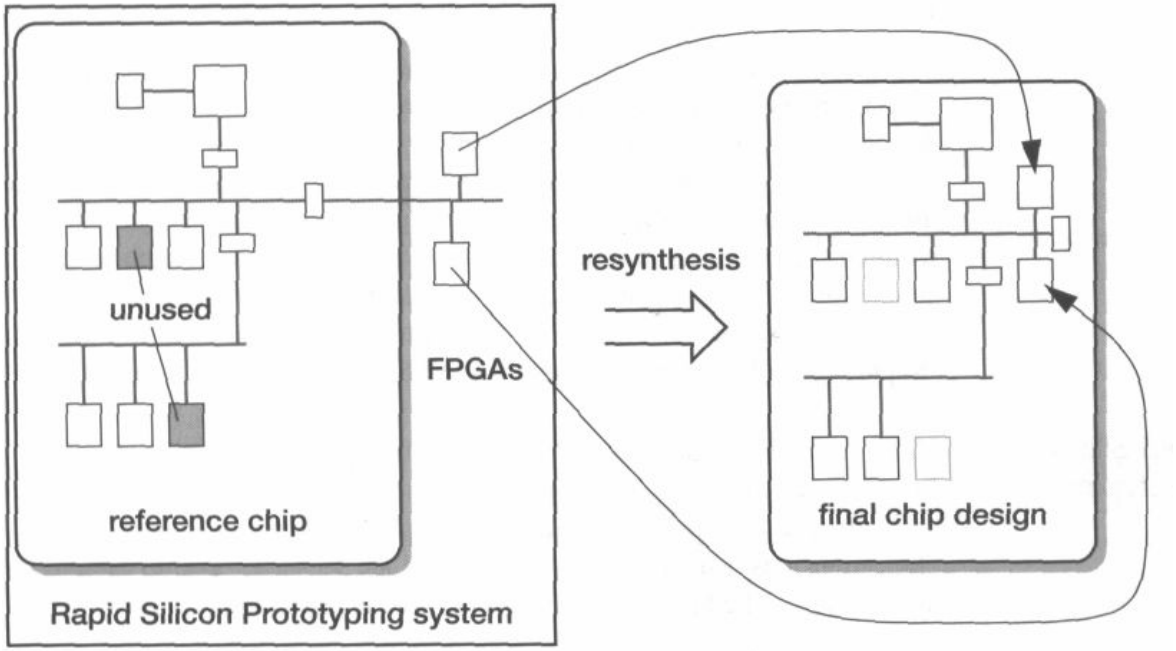


ARM Reference Peripheral Specification

- For operating system, number of components are required
- Memory map- Define base addresses of interrupt controller, counter time controller, reset controller and pause controller
- Interrupt controller- disabling, enabling, examining 32 IRQ sources, FIQ
- Counter timer—frequency division, prescaling
- Reset and pause controller---reset, pause status

Hardware System Prototyping Tools

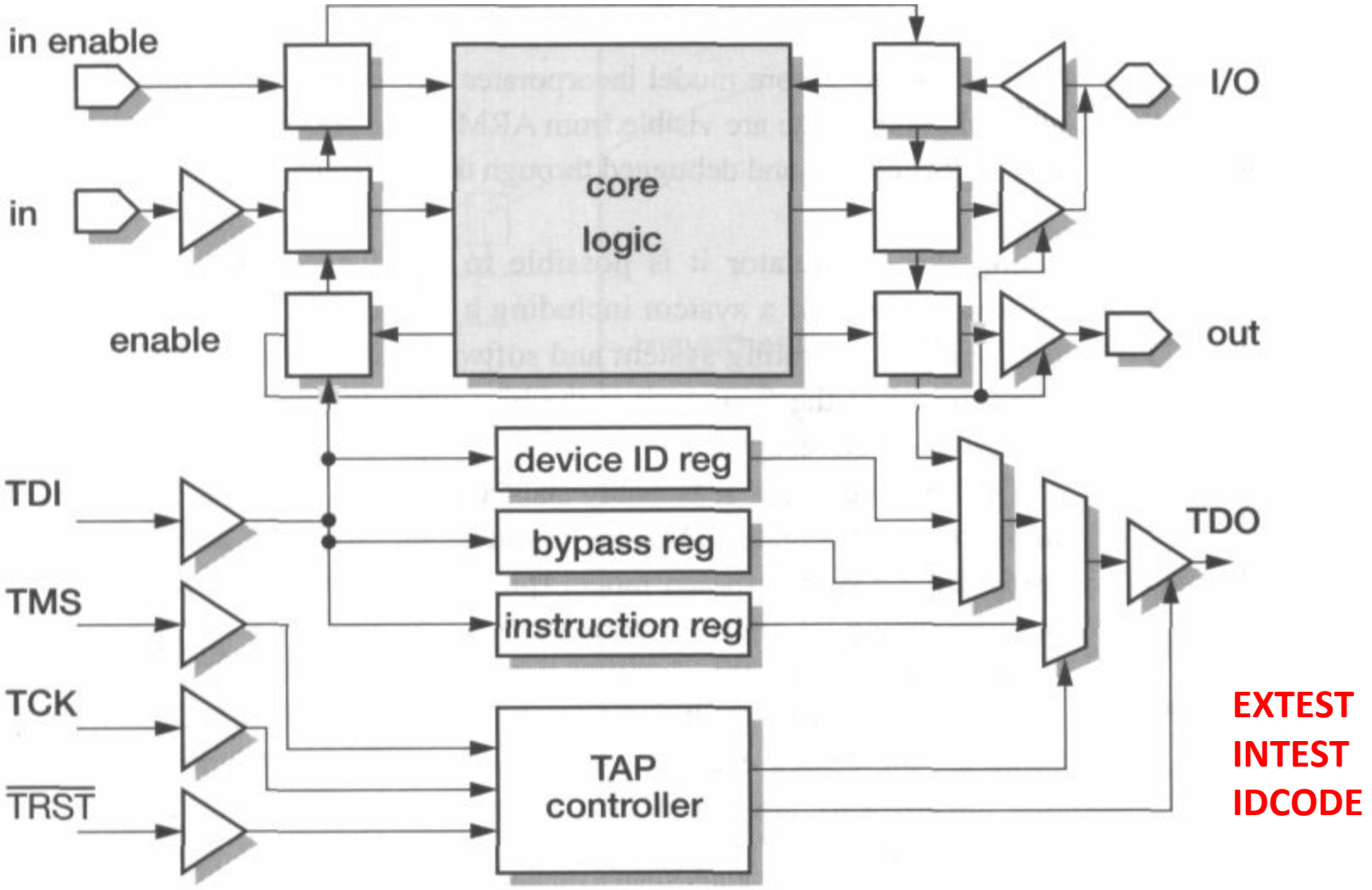
- Testing the components
- Rapid Silicon Prototyping
- Required blocks are implemented as off chip components
- On chip components not required are de configured



ARMulator

- Software simulator which contains four components
- The processor core model
- Memory interface
- Coprocessor interface
- Operating system interface

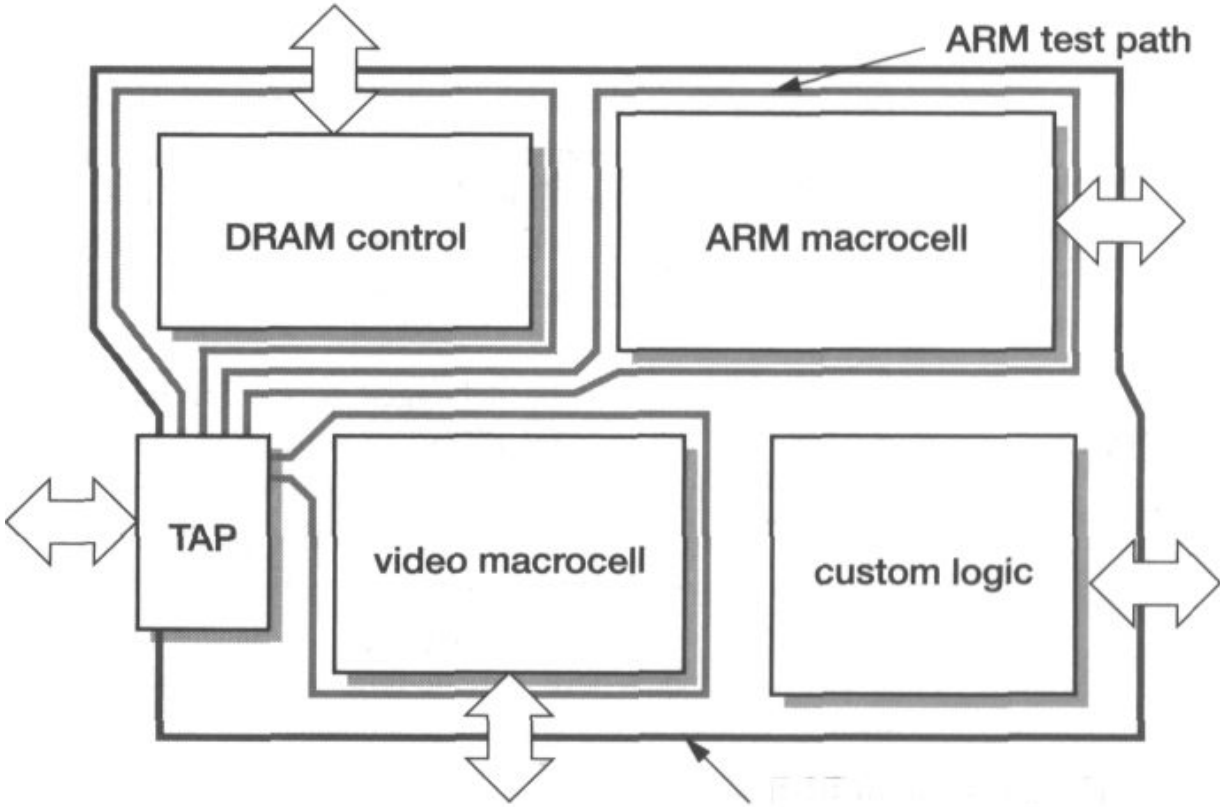
JTAG Boundary Scan Architecture



Embedded ICE

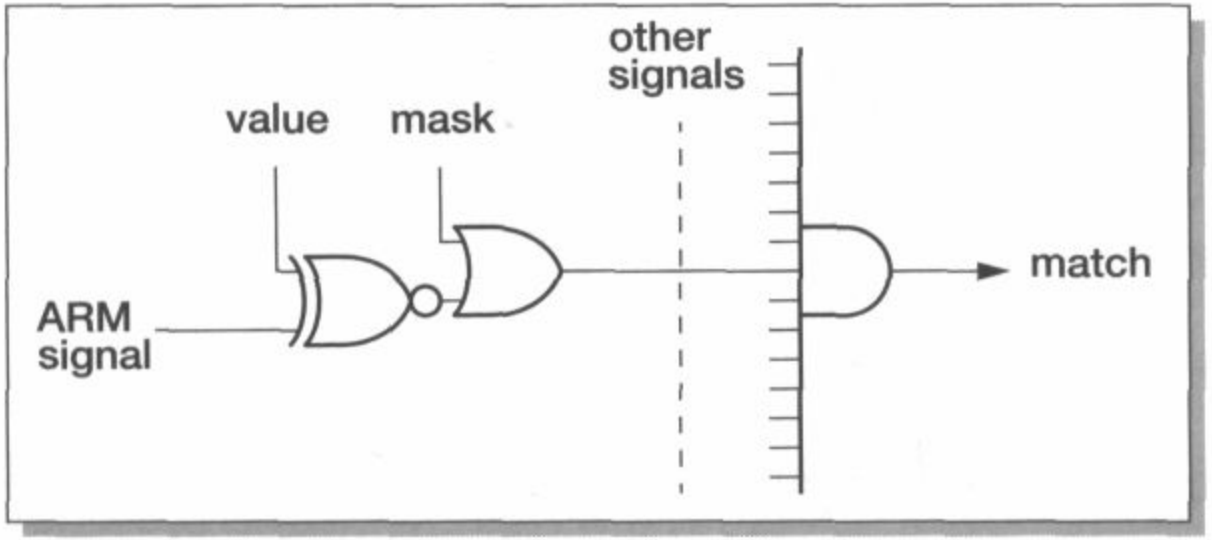
- Introduces breakpoints and watch point registers
- Accessed by JTAG instructions

System on chip has many macro cells to be tested

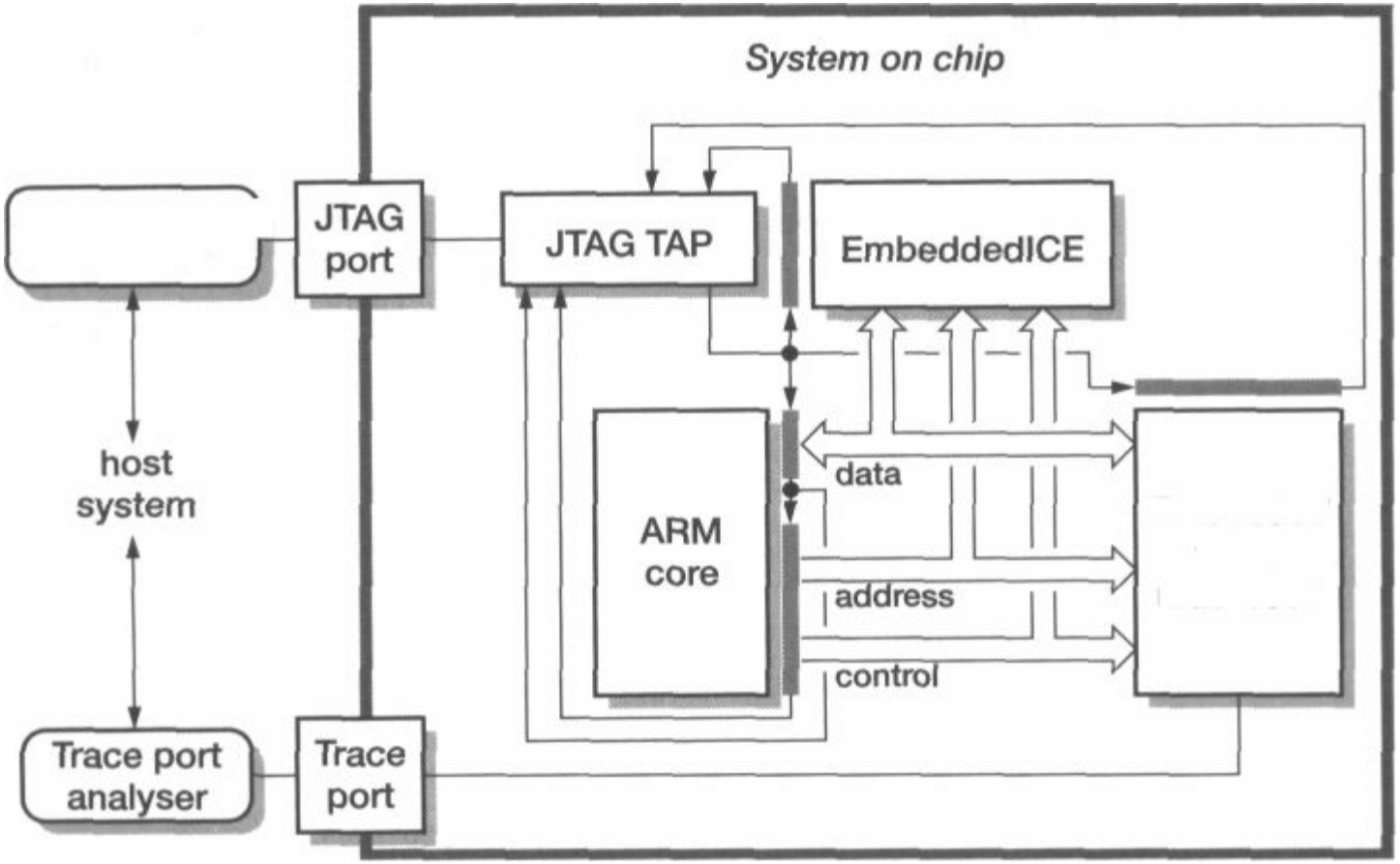


ARM Debug Architecture

- Use AMBA bus and access the test data parallel rather than serial in case boundary scan
- Embedded ICE contains two watch point registers, control and status registers

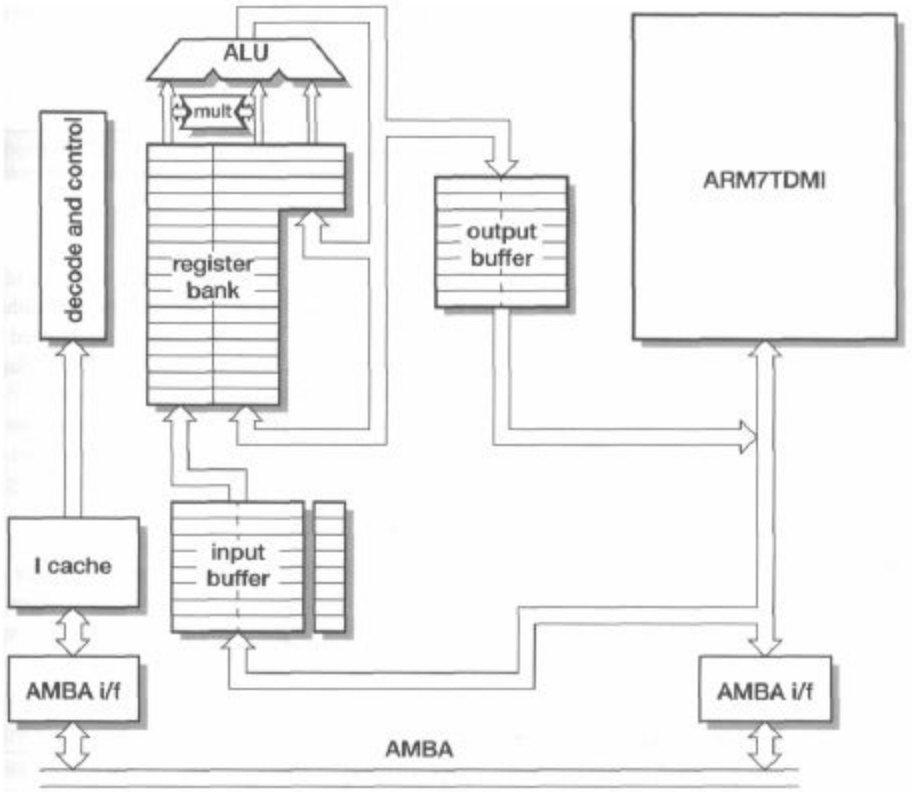


Embedded Trace-Real Time Debug System Organization

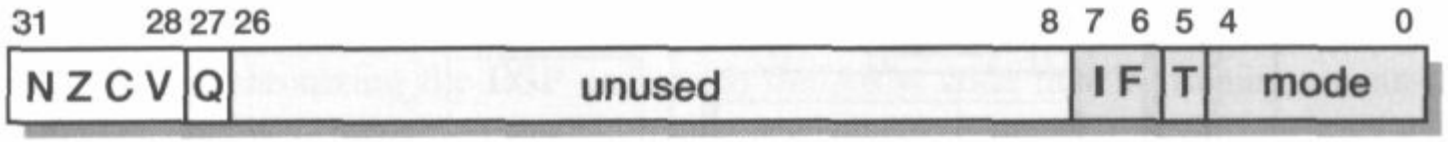


DSP Core

- Piccolo co processor, 16 bit
- 4, 48 bit registers



ARM v5TE



- Q flag for saturated arithmetic
- Multiply 16X16

QADD{cond} Rd,Rm,Rn

SMLAxy{cond} Rd,Rm,Rs,Rn

SMLAWy{cond} Rd,Rm,Rs,Rn

QDADD{cond} Rd,Rm,Rn

SMULWy{cond} Rd,Rm,Rs

SMLALxy{cond} RdLo,RdHi,Rm,Rs

SMULxy{cond} Rd,Rm,Rs