ARM7TDMI Embedded Microprocessor ASIC

CMOS Embedded Systems

Preliminary Information

DS4872 - 1.0 March 1998

INTRODUCTION

The ARM7TDMI Embedded Microprocessor ASIC product combines the flexibility of Mitel Semiconductor's Embedded Array ASIC technologies with the widely adopted ARM7TDMI microprocessor from Advanced RISC Machines.

The Embedded Array ASIC technologies from Mitel Semiconductor allow the combination of RAM, ROM, microprocessor cores and other custom logic blocks within a sea-of-gates gate array. As well as being offered at standard array sizes, the Embedded Array ASICs can also be optimised to meet the I/O and gate count requirements of specific designs.

The ARM7TDMI microprocessor is a 32-bit RISC microprocessor specifically designed for deeply embedded applications, such as wireless communications, networking, media and mass-storage. It is a member of the SystemBuilder™ library of hard and synthesisable macrofunctions

FEATURES

- Industry standard ARM7TDMI microprocessor
  - Small size
  - Low Power Consumption
  - High Performance
  - High Code Density
- Available on both 0.6µm (MV90000) and 0.35µm (MV200) CMOS technologies
- Embedded functions and pin-out of device defined in advance, allowing base layers of the Embedded Array ASIC to be manufactured in parallel with the verification and fine-tuning of the design
- After design sign-off, only the metal layers remain to be fabricated
- Prototype cycle time equivalent to gate array (11 days for 0.6µm), instead of the 2-3 months typically required for standard cell
- Optimised silicon architecture for excellent silicon utilisation
- Direct sign-off on Industry Standard CAE tools
- SystemBuilder megacell libraries
- Worldwide design center support
- 'How-to' development guides
- Dedicated software and hardware applications engineering support

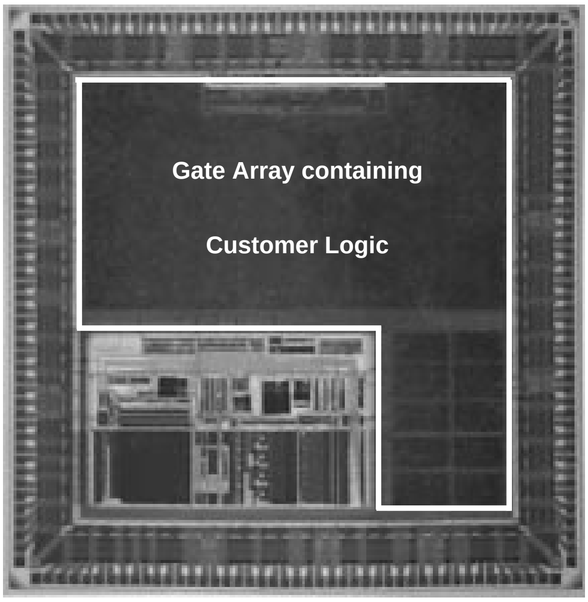

Fig.1 ARM7TDMI Embedded Microprocessor ASIC

BENEFITS

- Ideal for low-cost consumer applications where performance and power consumption are critical
- Fast time-to-market
- Ability to derive multiple solutions to different applications needs from a common base
- Cost-effective solutions
- Over five years experience of working with ARM to develop highly integrated silicon systems

The ARM7TDMI Embedded Microprocessor ASIC is one in a family of products from Mitel Semiconductor aimed at allowing customers to complete complex system integration designs in the shortest possible time.

Feature	0.35µm (3.3V)	0.6µm (5V)
ARM7TDMI Speed	54MHz	35MHz
ARM7TDMI Power Consumption	1.2mW/MHz	5.5mW/MHz
ARM7TDMI Core Size	2.26mm^2	5mm^2
Maximum available gates	3 million	1.15 million
Maximum available I/O	700	520

Embedded Microprocessor ASICs

The Embedded Microprocessor ASIC product from Mitel Semiconductor offers designers the capability to integrate designs of up to 3 million equivalent gates. Using automated Embedded Array base constructor software, an embedded array can be built to a customer's specification which gives designers the ability to specify the required embedded functions within a standard array. The use of a standard array size has benefits such as lower engineering costs for masks, probe cards, test engineering, etc.

The same software may be used to create optimised arrays for high volume applications, where the lower manufacturing costs of a custom array size will offset the increased engineering costs.

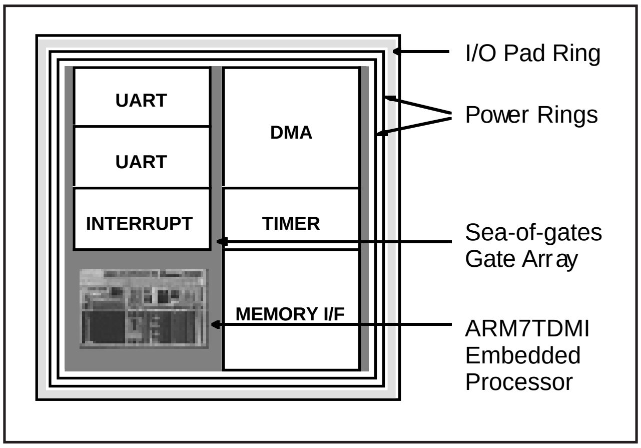

Fig.2 Example ARM7TDMI Embedded Microprocessor ASIC

Embedded Array Architecture

The core area consists of a dense array of core cells. Each core cell contains four transistors, two NMOS and two PMOS, whose sizes have been optimised for high density and low power. These are built on a structure consisting of a shared central source/drain region with independently available polysilicon gates. This core cell layout has been designed to allow very efficient metal interconnections including over-cell routing resulting in high utilisation. The core architecture also allows highly efficient register file RAM to be implemented.

Gate array memory is offered for both dual and single port RAM, and for ROM. Single and dual port embedded RAM and embedded ROM are available for large memories and applications that require very fast memory access times.

Clock and Power Distribution

It is known that large, complex designs working at high speed are vulnerable to problems associated with poor clock and power distribution. The Embedded Array architecture tackles this problem in several ways.

Power Distribution

The Embedded Array utilises a grid methodology for power distribution. This grid, which is automatically constructed during layout, uses metal layers one and three for horizontal power rails and metal layer two for vertical connections. Metal layer four is also available on 0.35um for vertical connections, which may be useful on some larger arrays. Methods of implementation are available for use with flat layout, manual methods, or hierarchical layout.

Clock Distribution

The Embedded Array supports a number of Clock distribution methodologies which may be implemented depending on the particular design and the CAD tools being used by the designer. For small designs with a light clock load, a single large buffer may be sufficient. For large designs, with large clock loads, a clock grid or clock tree is recommended to avoid metal electromigration in the clock network. Clock trees can either be synthesized or manually specified as a clock hierarchy by the designer.The clock grid methodology uses up to three stages of buffers, where each stage drives a grid which feeds the next stage cells. The final stage grid is a starting point for routing to the actual clocked inputs. The figure below illustrates the layout of each buffer stage, which is done automatically at layout.

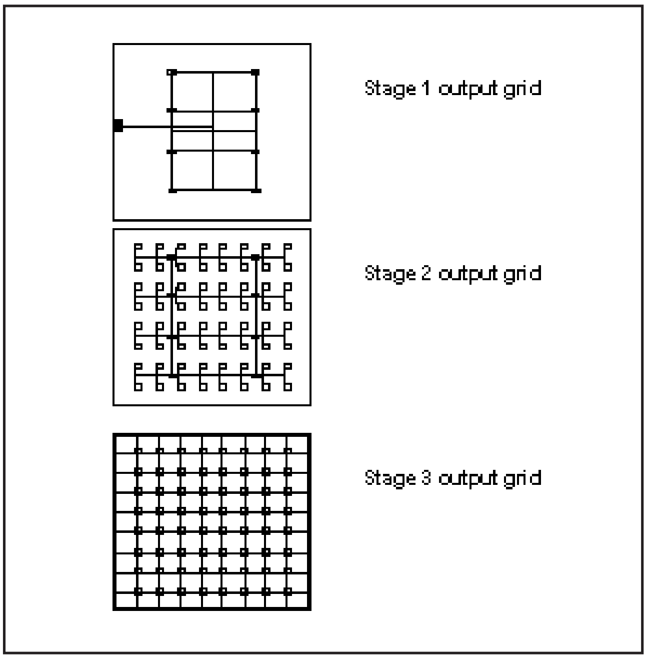

Fig.3 Embedded Microprocessor ASIC Clock Grid Methodology

ARM7TDMI Microprocessor

The ARM7TDMI is a member of the Advanced RISC Machines (ARM) family of general purpose 32-bit microprocessors, which offer high performance for very low power consumption and price.

The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles, and the instruction set and related decode mechanism are much simpler than those of microprogrammed Complex Instruction Set Computers. This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-effective core.

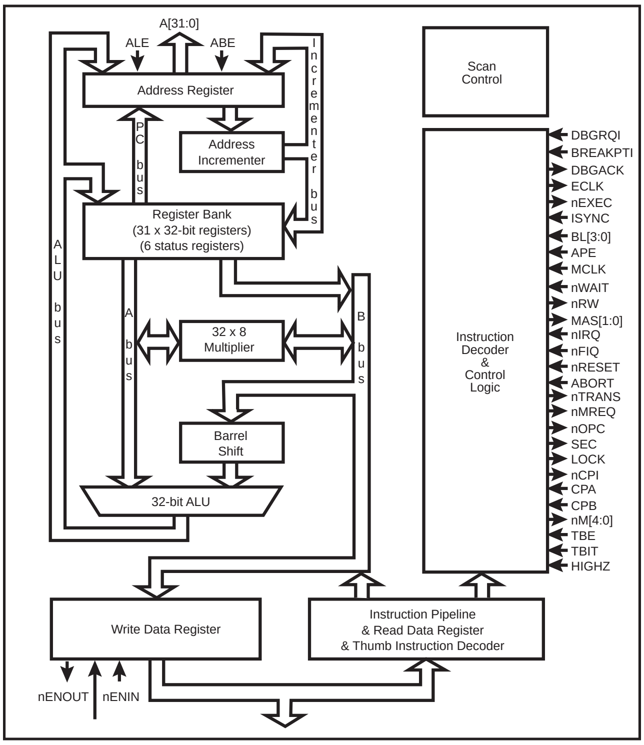

Fig.4 ARM7TDMI Architecture

Pipelining is employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory.

The ARM memory interface has been designed to allow the performance potential to be realised without incurring high costs in the memory system. Speed-critical control signals are pipelined to allow system control functions to be implemented in standard low-power logic, and these control signals facilitate the exploitation of the fast local access modes offered by industry standard dynamic RAMs.

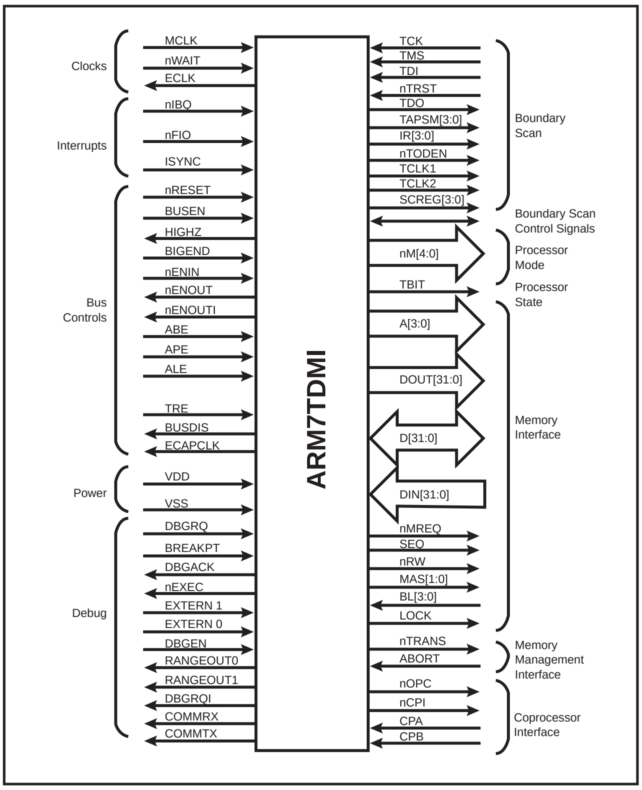

Fig.5 ARM7TDMI Interfaces

The ARM7TDMI microprocessor is surrounded by a scan chain. This allows it to be isolated from the embedded system for debug purposes. Register or memory values may be examined, and breakpoints and watchpoints may be set via the JTAG interface. As ARM instructions are conditionally executed, the microprocessor pipeline follows breakpoints to determine whether a trigger condition exists.

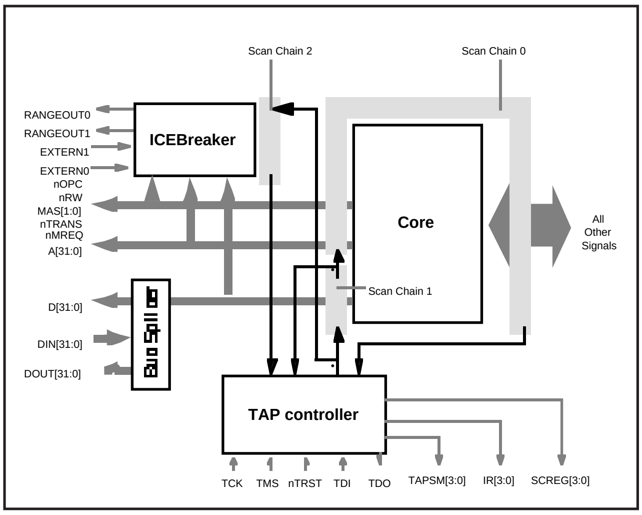

Fig.6 ARM7TDMI Debug Interface

ARM7TDMI Instruction Set Architecture

The ARM7TDMI microprocessor employs a unique architectural strategy known as THUMB, which makes it ideally suited to high-volume applications with memory restrictions, or applications where high code density is essential.

The THUMB Concept

The key idea behind THUMB is that of a super-reduced instruction set. Essentially, the ARM7TDMI microprocessor has two instruction sets.

The ARM 32-bit instruction set offers flexibility in instruction format and operand manipulation while producing maximum performance from 32-bit memory systems.

Mnemonic	Instruction	Action
ADC	Add with carry	Rd := Rn + Op2 + Carry
ADD	Add	Rd := Rn + Op2
AND	AND	Rd := Rn AND Op2
B	Branch	R15 := address
BIC	Bit Clear	Rd := Rn AND NOT Op2
BL	Branch with Link	R14 := R15, R15 := address
BX	Branch and Exchange	R15 := Rn, T bit := Rn[0]
CDP	Coprocesor Data Processing	(Coprocessor-specific)
CMN	Compare Negative	CPSR flags := Rn + Op2
CMP	Compare	CPSR flags := Rn - Op2
EOR	Exclusive OR	Rd := (Rn AND NOT Op2) OR (op2 AND NOT Rn)
LDC	Load coprocessor from memory	Coprocessor load
LDM	Load multiple registers	Stack manipulation (Pop)
LDR	Load register from memory	Rd := (address)
MCR	Move CPU register to coprocessor register	cRn := rRn {<op>cRm}
MLA	Multiply Accumulate	Rd := (Rm * Rs) + Rn
MOV	Move register or constant	Rd : = Op2
MRC	Move from coprocessor register to CPU register	Rn := cRn {<op>cRm}
MRS	Move PSR status/flags to register	Rn := PSR
MSR	Move register to PSR status/flags	PSR := Rm
MUL	Multiply	Rd := Rm * Rs
MVN	Move negative register	Rd := 0xFFFFFFFF EOR Op2
ORR	OR	Rd := Rn OR Op2
RSB	Reverse Subtract	Rd := Op2 - Rn
RSC	Reverse Subtract with Carry	Rd := Op2 - Rn - 1 + Carry
SBC	Subtract with Carry	Rd := Rn - Op2 - 1 + Carry
STC	Store coprocessor register to memory	address := CRn
STM	Store Multiple	Stack manipulation (Push)
STR	Store register to memory	<address> := Rd
SUB	Subtract	Rd := Rn - Op2
SWI	Software Interrupt	OS call
SWP	Swap register with memory	Rd := [Rn], [Rn] := Rm
TEQ	Test bitwise equality	CPSR flags := Rn EOR Op2
TST	Test bits	CPSR flags := Rn AND Op2

Table.1 Standard 32-bit ARM instruction set

The THUMB set's 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM's performance advantage over a traditional 16-bit microprocessor using 16-bit registers. This is possible because THUMB code operates on the same 32-bit register set as ARM code.

THUMB code is able to provide up to 65% of the code size of ARM, and 160% of the performance of an equivalent ARM microprocessor connected to a 16-bit memory system.

Mnemonic	Instruction	Action	Lo/Hi register operands	Condition codes set
ADC	Add with Carry	Rd := Rd + Rs + C	Lo	Yes
ADD	Add	Rd := Rn + Rs	Lo/Hi	Yes*
AND	AND	Rd := Rd AND Rs	Lo	Yes
ASR	Arithmetic Shift Right	Rd := Rd ASR Rs	Lo	Yes
B	Unconditional branch	PC := PC +/- Offset11	Lo	
Bxx	Conditional branch	PC := PC +/- Offset8	Lo	
BIC	Bit Clear	Rd := Rd AND NOT Rs	Lo	Yes
BL	Branch and Link	PC := PC +/- Offset	LR := PC + 2	
BX	Branch and Exchange	PC := Rs	Lo / Hi	
CMN	Compare Negative	Rd + Rs	Lo	Yes
CMP	Compare	CPSR flags :=Rd - Rs	Lo / Hi	Yes
EOR	EOR	Rd := Rd EOR Rs	Lo	Yes
LDMIA	Load multiple	Stack manipulation (Pop)	Lo	
LDR	Load word	Rd32 := [Rb + Immediate5]	Lo	
LDRB	Load byte	Rd8 := [Rb + Immediate5]	Lo	
LDRH	Load halfword	Rd16 := [Rb + Immediate5]	Lo	
LSL	Logical Shift Left	Rd := Rd << Rs	Lo	Yes
LDSB	Load sign-extended byte	Rd8 := [Rb + Immediate5]	Lo	
LDSH	Load sign-extended halfword	Rd16 := [Rb + Immediate5]	Lo	
LSR	Logical Shift Right	Rd := Rd >> Rs	Lo	Yes
MOV	Move register	Rd := Immediate8	Lo / Hi	Yes*
MUL	Multiply	Rd := Rs * Rd	Lo	Yes
MVN	Move Negative register	Rd := NOT Rs	Lo	Yes
NEG	Negate	Rd := -Rs	Lo	Yes
ORR	OR	Rd := Rd OR Rs	Lo	Yes
POP	Pop registers	[SP] ++ := Rlist (LR)	Lo	
PUSH	Push registers	Rlist (LR):= [SP] --	Lo	
ROR	Rotate Right	Rd := Rd ROR Rs	Lo	Yes
SBC	Subtract with Carry	Rd := Rd -Rs - NOT C	Lo	Yes
STMIA	Store Multiple	[Rb]++ := Rlist	Lo	
STR	Store word	[Rb + Immediate5] := Rd32	Lo	
STRB	Store byte	[Rb + Immediate5] := Rd8	Lo	
STRH	Store halfword	[Rb + Immediate5] := Rd16	Lo	
SWI	Software Interrupt	OS call		
SUB	Subtract	Rd := Rd - Immediate8	Lo	Yes
TST	Test bits	CPSR flags :=Rd AND Rs	Lo	Yes

Table.2 16-bit THUMB instruction set

THUMB's Advantages

THUMB instructions operate with the standard ARM register configuration, allowing excellent interoperability between ARM and THUMB states. Each 16-bit THUMB instruction has a corresponding 32-bit ARM instruction with the same effect on the microprocessor model.

The major advantage of a 32-bit (ARM) architecture over a 16-bit architecture is its ability to manipulate 32-bit integers with single instructions, and to address a large address space efficiently. When processing 32-bit data, a 16-bit architecture will take at least two instructions to perform the same task as a single ARM instruction.

However, not all the code in a program will process 32-bit data (for example, code that performs character string handling), and some instructions, like Branches, do not process any data at all.

If a 16-bit architecture only has 16-bit instructions, and a 32-bit architecture only has 32-bit instructions, then overall the 16-bit architecture will have better code density, and better than one half the performance of the 32-bit architecture. Clearly 32-bit performance comes at the cost of code density.

THUMB breaks this constraint by implementing a 16-bit instruction length on a 32-bit architecture, making the processing of 32-bit data efficient with a compact instruction coding. This provides far better performance than a 16-bit architecture, with better code density than a 32-bit architecture.

THUMB also has a major advantage over other 32-bit architectures with 16-bit instructions. This is the ability to switch back to full ARM code and execute at full speed. Thus critical loops for applications such as **fast interrupts, DSP algorithms** can be coded using the full ARM instruction set, and linked with THUMB code. The overhead of switching from THUMB code to ARM code is folded into sub-routine entry time. Various portions of a system can be optimised for speed or for code density by switching between THUMB and ARM execution as appropriate.

Operating Modes

ARM7TDMI supports seven modes of operation:

User (usr)	The normal ARM program execution state
FIQ (fiq)	Designed to support a data transfer or channel process
IRQ (irq)	Used for general-purpose interrupt handling
Supervisor (svc)	Protected mode for the operating system
Abort mode (abt)	Entered after a data or instruction prefetch abort
System (sys)	A privileged user mode for the operating system
Undefined (und)	Entered when an undefined instruction is executed

Mode changes may be made under software control, or may be brought about by external interrupts or exception processing. Most application programs will execute in User mode. The non-user modes - known as privileged modes - are entered in order to service interrupts or exceptions, or to access protected resources.

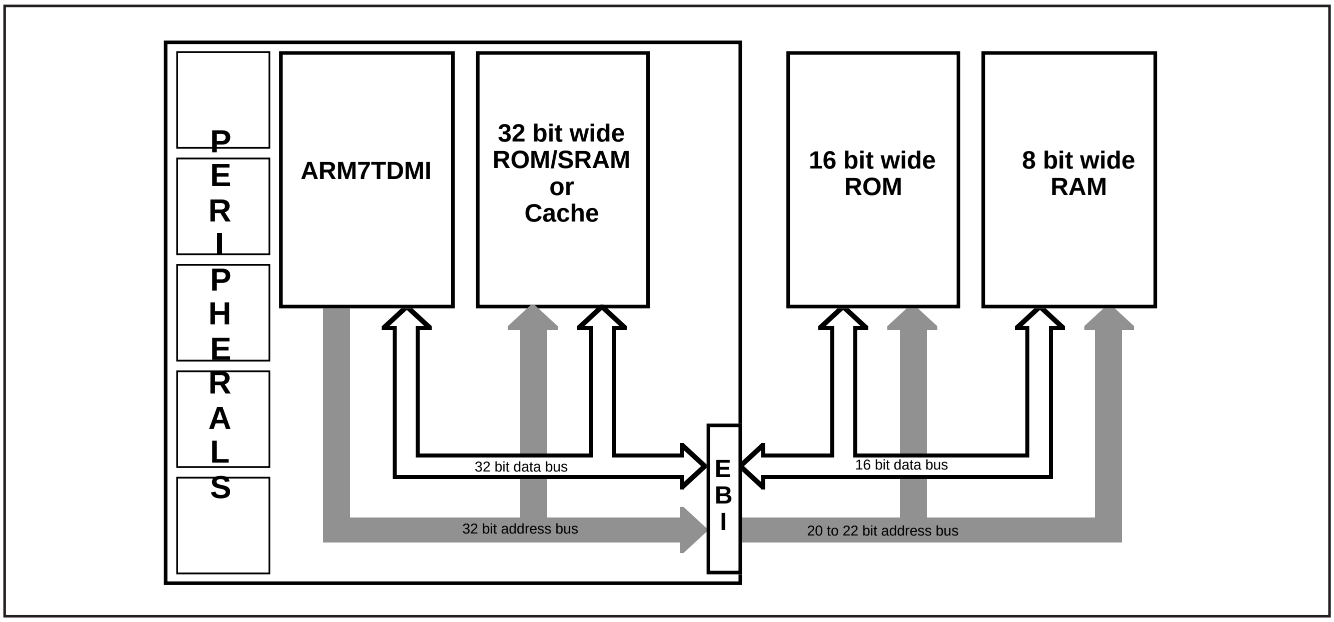

Fig.7 Mixed memory system

Register Sets

In ARM state, 16 general registers and one or two status registers are visible at any one time. In privileged (non-User) modes, mode-specific banked registers are switched in. The table below shows which registers are available in each mode: the banked registers are indicated by a triangle:

ARM State General Registers and Program Counter

System & User	FIQ	Supervisor	Abort	IRQ	Undefined
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7
R8	◣R8_fiq	R8	R8	R8	R8
R9	◣R9_fiq	R9	R9	R9	R9
R10	◣R10_fiq	R10	R10	R10	R10
R11	◣R11_fiq	R11	R11	R11	R11
R12	◣R12_fiq	R12	R12	R12	R12
R13	◣R13_fiq	◣R13_svc	◣R13_abt	◣R13_irq	◣R13_und
R14	◣R14_fiq	◣R14_svc	◣R14_abt	◣R14_irq	◣R14_und
R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)

ARM State Program Status Registers

CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
	◣SPSR_fiq	◣SPSR_svc	◣SPSR_abt	◣SPSR_irq	◣SPSR_und

◣= *banked register*

The THUMB state register set is a subset of the ARM state set. The programmer has direct access to eight general registers, R0-R7, as well as the Program Counter (PC), a stack pointer register (SP), a link register (LR), and the CPSR. There are banked Stack Pointers, Link Registers and Saved Process Status Registers (SPSRs) for each privileged mode. This is shown below:

THUMB State General Registers and Program Counter

System & User	FIQ	Supervisor	Abort	IRQ	Undefined
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7
SP	◣SP_fiq	◣SP_svc	◣SP_abt	◣SP_irq	◣SP_und
LR	◣LR_fiq	◣LR_svc	◣LR_abt	◣LR_irq	◣LR_und
PC	PC	PC	PC	PC	PC

THUMB State Program Status Registers

CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
	◣SPSR_fiq	◣SPSR_svc	◣SPSR_abt	◣SPSR_irq	◣SPSR_und

◣= *banked register*

Design Support

Mitel Semiconductor offers fully flexible design support allowing the customer a wide choice of design interfaces. Each customer design is supported full-time by an design center engineer as it proceeds through the design flow. Hardware and software integration expertise is provided by dedicated teams of hardware design engineers and system applications engineers.

In addition, a series of 'How To' guides is available to ease the process of learning how to integrate the microprocessor core into the design.

The design process incorporates a design audit procedure to verify compliance with the customer's specification and to ensure manufacturability. The procedure includes three design reviews held at key stages of the design process to ensure device performance and timescales.

Design Review 1: Held at the beginning of the design cycle to check and agree on performance, packaging, specification and design time scales

Design Review 2: Held after logic simulation but prior to layout to ensure satisfactory functionality, timing performance and adequate fault coverage

Design Review 3: Held after layout and post layout simulation verification of satisfactory design performance after insertion of actual track loads. This is the final check of all device specifications prior to prototype manufacture

Design Flow

The Embedded Microprocessor ASIC product is offered within the standard ASIC design flows supported by Mitel Semiconductor.

Features of design kits include:

- 'How-To' design guides
- Full top-down design flow support
- VHDL/Verilog simulation models for the ARM7TDMI microprocessor core
- Synopsys Design Compiler timing model for the ARM7TDMI microprocessor core
- Sign-off simulation by the customer on Cadence Verilog, Mentor Quicksim, and Synopsys VSS
- Support for floorplanning
- Full test patterns (95% coverage) for both serial scan and parallel access for the ARM7TDMI microprocessor core
- Direct routes to layout and test

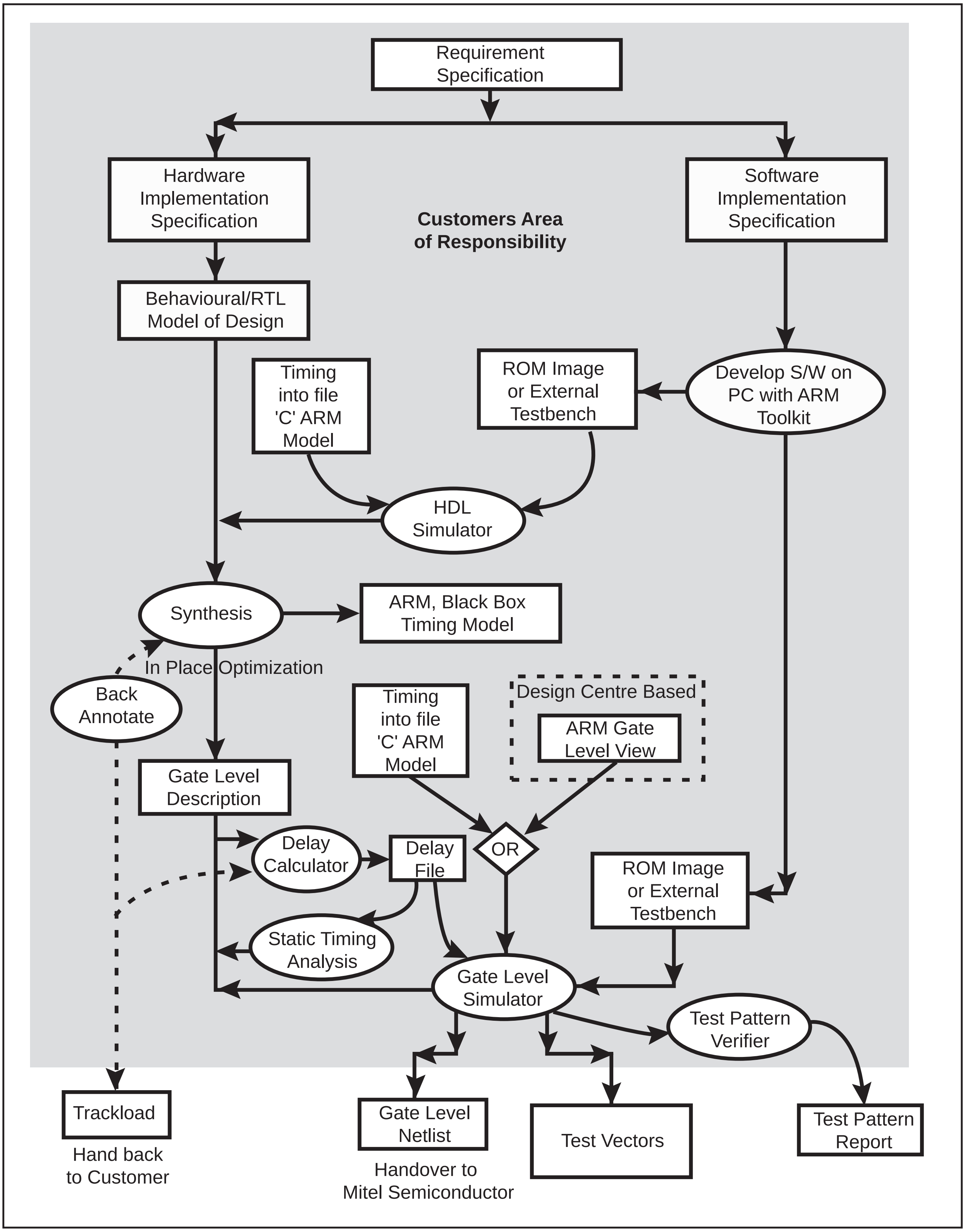
Requirement Specification
Hardware Implementation Specification
Customers Area of Responsibility
Software Implementation Specification
Behavioural/RTL Model of Design
Timing into file 'C' ARM Model
ROM Image or External Testbench
Develop S/W on PC with ARM Toolkit
HDL Simulator
Synthesis
ARM, Black Box Timing Model
In Place Optimization
Back Annotate
Timing into file 'C' ARM Model
Design Centre Based
ARM Gate Level View
Gate Level Description
Delay Calculator
Delay File
OR
ROM Image or External Testbench
Static Timing Analysis
Gate Level Simulator
Test Pattern Verifier
Trackload
Hand back to Customer
Gate Level Netlist
Handover to Mitel Semiconductor
Test Vectors
Test Pattern Report

Fig.8 ARM7TDMI Embedded Microprocessor ASIC Design Flow

Software Development and Debug

Both software and hardware development tools are available to aid the designer in the development and debugging of the target application. The Software Development Toolkit contains an industry-standard optimising ‘C’ compiler and Assembler for both ARM and Thumb code, a linker, and a Windows-based debugger.

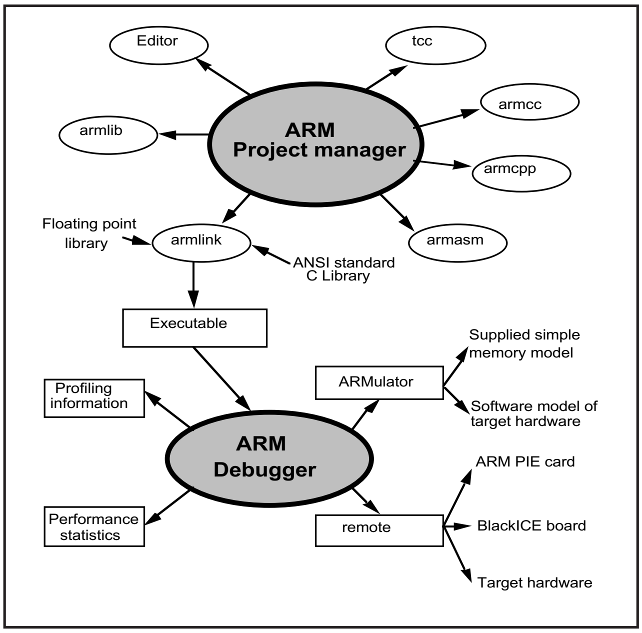

Fig.9 Software development toolkit - Tool Chain

The task of creating the project environment within which the user can write, maintain and build program code is significantly eased by the inclusion of the ARM Project Manager, an integrated development environment.

The Project Manager provides the mechanisms required to configure and build complex Embedded Applications. The interdependencies between source files in a project are automatically detected at build time, removing the need to write complex make files.

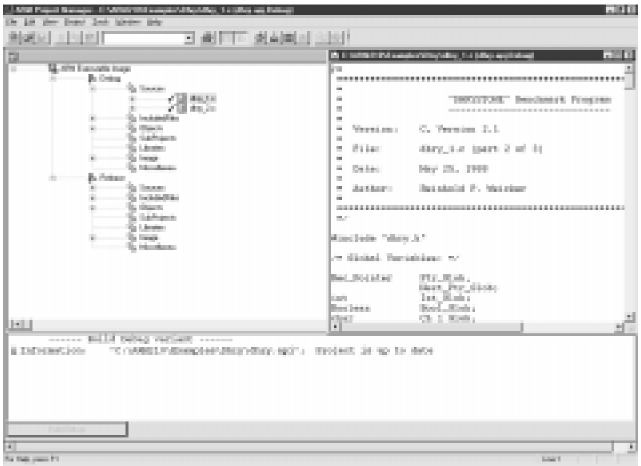

Fig.10 Software development toolkit - Project Manager Interface

The ARM Windows Debugger is a source-level debugger, with optional Source/Assembler interleaving. C and Assembler may be single-stepped and breakpointed. Breakpoints may be simple or compound, occurring only on a specific set of events. Variables and memory may be watched during execution. Modifying or setting memory and variables to specific values can trigger breakpoints.

The debugger can interface to a software simulation environment, such as the ARMulator software emulator, and also to a target-resident debug monitor via a serial port or other communications channel.

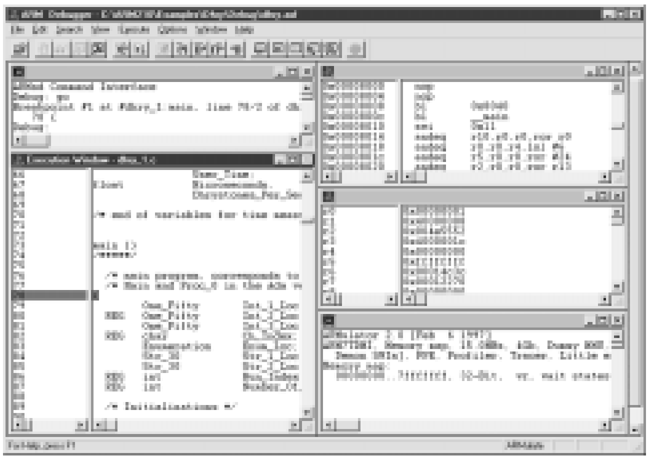

Fig.11 Software development toolkit - Debugger Interface

Alternatively, the debugger may connect directly to the microprocessor via the EmbeddedICE JTAG Debugger interface unit in the target hardware environment. For Embedded Microprocessor ASICs where the board-level hardware and interface software are still being developed, this represents a particularly attractive debug option. The correct operation of the ARM7TDMI microprocessor and other on-chip resources may be verified prior to the availability of debugged boards and software

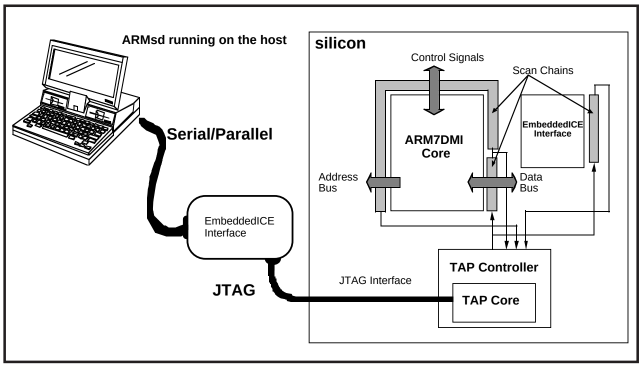

Fig.9 Software development toolkit - Tool Chain

Experience

Mitel Semiconductor has been an ARM semiconductor partner since 1992. As such, the company has developed significant expertise in the integration and manufacture of ARM-based designs. The company has worked closely with Advanced RISC Machines to develop reuse methodologies and internal bus standards for ARM-based designs. The company also manufactures ARM-based standard product microcontrollers, in addition to ARM Microcontroller ASICs developed for key customers in markets such as cellular communications, networking and mass-storage.

This experience in both the hardware and the software aspects of ARM-based design and integration is available through the regional design centers of Mitel Semiconductor, with additional support provided from a dedicated applications engineering team.

Manufacturing

The Embedded Microprocessor ASIC product is manufactured in Mitel Semiconductor’s state of the art facility near Plymouth, England. This facility is a purpose-built, vibration-free wafer fab equipped with the latest automated technology for 8-inch wafer processing. This equipment utilises mini-environments together with the use of SMIF boxes to achieve ultra clean processing conditions. Computer Aided Manufacture ensures production efficiency. In addition to the world class wafer fabrication facility, the probe and final test areas are equipped with the latest analog and digital testers. Mitel Semiconductor is committed to continuous investment to provide state-of-the-art CMOS ASICs.

A qualified second source for this silicon process is also available.

Availability

The ARM7TDMI Embedded Microprocessor ASIC product is available for design today on both 0.6µm and 0.35µm technologies.

PRIMARY ASIC SYSTEMS DESIGN CENTRES
UNITED KINGDOM: Swindon, Tel: (01793) 518000 Fax: (01793) 518411. **USA:** San Jose, CA, Tel: (408) 451-4700 Fax: (408) 451-4710. Boston, MA, Tel: (617) 251-0100. Fax: (617) 251-0104. Irvine, CA, Tel: (714) 852-3900. Fax: (714) 852-3910. **FRANCE:** Les Ulis Cedex, Tel: (01) 69 18 90 00 Fax: (01) 64 46 06 07. **GERMANY:** Munich, Tel: (089) 419508-0 Fax: (089) 419508-55. **JAPAN:** Tokyo, Tel: (3) 5276-5501. Fax: (3) 5276-5510.

HEADQUARTERS OPERATIONS
MITEL SEMICONDUCTOR
Cheney Manor, Swindon,
Wiltshire SN2 2QW, United Kingdom.
Tel: (01793) 518000
Fax: (01793) 518411

MITEL SEMICONDUCTOR
1500 Green Hills Road,
Scotts Valley, California 95066-4922
United States of America.
Tel (408) 438 2900
Fax: (408) 438 5576/6231

Internet: http://www.mitelsemi.com
CUSTOMER SERVICE CENTRES
- **FRANCE & BENELUX** Les Ulis Cedex Tel: (1) 69 18 90 00 Fax : (1) 64 46 06 07
- **GERMANY** Munich Tel: (089) 419508-20 Fax : (089) 419508-55
- **ITALY** Milan Tel: (02) 6607151 Fax: (02) 66040993
- **JAPAN** Tokyo Tel: (03) 5276-5501 Fax: (03) 5276-5510
- **KOREA** Seoul Tel: (2) 5668141 Fax: (2) 5697933
- **NORTH AMERICA** Scotts Valley, USA Tel: (408) 438 2900 Fax: (408) 438 5576/6231
- **SOUTH EAST ASIA** Singapore Tel:(65) 3827708 Fax: (65) 3828872
- **SWEDEN** Stockholm Tel: 46 8 702 97 70 Fax: 46 8 640 47 36
- **TAIWAN, ROC** Taipei Tel: 886 2 25461260 Fax: 886 2 27190260
- **UK, EIRE, DENMARK, FINLAND & NORWAY**
  Swindon Tel: (01793) 726666 Fax : (01793) 518582

These are supported by Agents and Distributors in major countries world-wide.

This publication is issued to provide information only which (unless agreed by the Company in writing) may not be used, applied or reproduced for any purpose nor form part of any order or contract nor to be regarded as a representation relating to the products or services concerned. No warranty or guarantee express or implied is made regarding the capability, performance or suitability of any product or service. The Company reserves the right to alter without prior notice the specification, design or price of any product or service. Information concerning possible methods of use is provided as a guide only and does not constitute any guarantee that such methods of use will be satisfactory in a specific piece of equipment. It is the user's responsibility to fully determine the performance and suitability of any equipment using such information and to ensure that any publication or data used is up to date and has not been superseded. These products are not suitable for use in any medical products whose failure to perform may result in significant injury or death to the user. All products and materials are sold and services provided subject to the Company's conditions of sale, which are available on request.