Toshiba Launches 16-Bit Microcontroller Family

TLCS-900 Offers Signal-Processing Instructions

By Mark Thorson

Toshiba has announced US availability of the TLCS-900 family of 16-bit microcontrollers, which, when it was introduced in Japan in 1992, was marketed on a selective, private basis rather than being formally announced. Three versions of the chip are available in the US: ROM-less, 32K mask ROM, and 32K one-time-programmable ROM. The architecture's features and performance make it fairly typical of recent 16-bit microcontroller designs, such as NEC's K6 (see MPR 6/20/90 p.1) and Hitachi's H8/532. It also competes against such automotive-oriented designs as Siemen's 80C166 family (see MPR 3/7/90 p.8), Motorola's 68HC16 (see MPR 10/17/90 p. 6), and Intel's 8096 and 80196 families.

Figure 1 is a block diagram of the first generation of devices. As with the vast majority of microcontrollers, most of the pins on the package are available as parallel I/O lines—in this case 65 pins on an 80-pin PQFP. Alternate functions such as pulse-width modulated (PWM) outputs can be selected on a pin-by-pin basis by programming a port control register.

Instruction Set and Registers

One goal of the TLCS-900 architecture is to be an upward migration path for Toshiba's 8-bit TLCS-90 family, which in turn is a near-superset of the Zilog Z80. The TLCS-90 extends the Z80 instruction set with arithmetic and logical instructions combining an 8-bit immediate operand with any register or memory location, greatly relieving the accumulator bottleneck in the original Z80 architecture. The TLCS-90 also differs from the Z80 in that it lacks a separate I/O address space, so most binary applications can't be ported from the Z80 family to the TLCS-90 without modification.

The TLCS-900 is a completely new architecture, not binary compatible with the TLCS-90. The new extensions make the TLCS-900 a full 32-bit architecture with a 16-bit implementation. Only a few instructions, such as multiply and divide, are not supported in 32-bit forms. Most arithmetic and logical instructions have two-bit fields that specify the operand size.

The CPU has two main modes controlled by a bit in the processor's control and status register (SR):

- Minimum Mode—an architecture that is forward compatible at the assembly-language level from the TLCS-90. The program counter is 16 bits, and the general registers are organized as eight banks of four 16-bit registers, which can also be referenced as bytes.
- Maximum Mode—a full 32-bit architecture, including a 32-bit program counter. The general registers are organized as four banks of four 32-bit registers, which can also be referenced as bytes and 16-bit words.

Figure 2 shows the register set. The dashed lines indicate registers that become wider in maximum mode. Not shown are the four or eight alternate banks of data registers. Fastest execution results when operands are accessed in the current bank, but alternate register banks may be accessed with a one-cycle penalty.

When the 8-bit operand size is specified, a 3-bit field in the instruction selects one of eight 8-bit registers. When a 16- or 32-bit operand size is used, the field selects either one of four registers from the current bank or one of the four user-mode address registers. The 32-bit arithmetic and logical instructions are enabled in both modes, although they operate only on the 32-bit address registers in minimum mode. The index registers and stack pointers are always treated as 32 bits, but only the lower 24 bits are available on the external bus.

The user stack pointer is intended for use by applications programs. During interrupt handling, the CPU switches to a privileged mode that uses the system stack pointer instead. Some instructions, such as HALT and the instructions which access DMA controller registers, can be executed only in privileged mode, which is indicated by the SYSM bit in the SR register.

As shown in Table 1, the TLCS-900 has a rich instruction set, with both signed and unsigned multiplication and division, multiply-accumulate, sign extension, and stack management instructions for procedure entry and exit. It has one unusual instruction: bit-reverse. Called mirror-invert (MIRR), this instruction reverses the

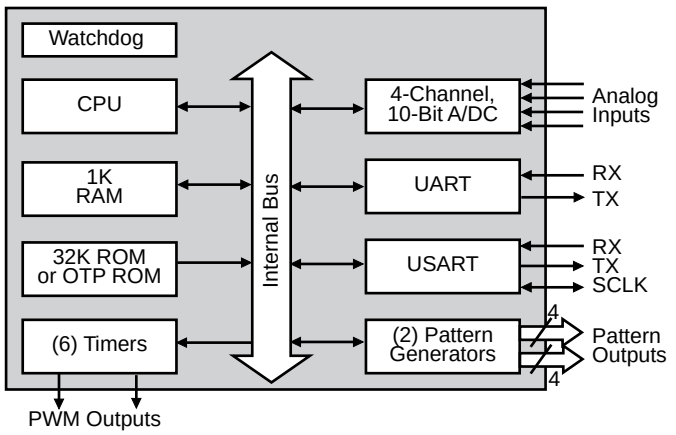

Figure 1. Block diagram of TLCS-900.

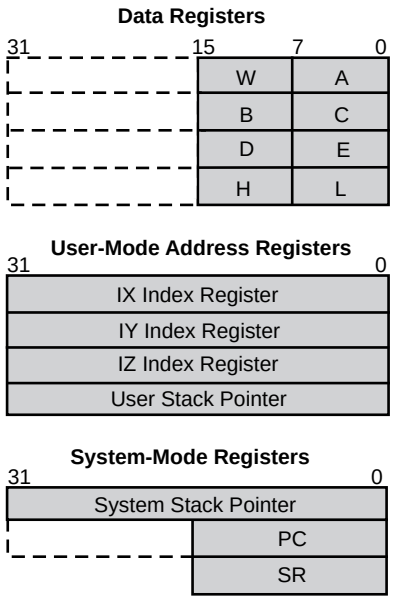

Figure 2. TLCS-900 register file.

bit order of the destination operand. It is typically used in address calculations for fast Fourier transforms (FFTs).

The double-operand arithmetic and logical instructions have five principal formats: register-to-register, register-to-memory, memory-to-register, immediate-to-register, and immediate-to-memory. There are no general memory-to-memory formats for arithmetic and logical instructions. Most single-operand instructions, such as NEG, can access only registers.

At 20 MHz, the minimum instruction time for simple instructions such as complementing the carry flag is 200 ns (4 cycles). Register-register add takes 400 ns at the same clock frequency. Signed or unsigned 16 × 16 multiply to a 32-bit product requires 2.6 µs. Division of a 32-bit dividend by a 16-bit divisor to produce a 16-bit quotient and remainder takes 3 µs (unsigned) or 3.2 µs (signed). Multiplication and division are available only for bytes and words, not double words.

As with the Z80 and TLCS-90, the Z flag can be operated with a single bit in memory. The TLCS-900 goes further by providing bit-scan-forward and bit-scan-backward instructions. A bit address consists of an address-mode operand and a 3- or 4-bit immediate value that specifies the position of the addressed bit in the byte- or word-length operand. There are no instructions that operate on variable (register-selectable) bits.

The instruction set includes modulo-arithmetic increment and decrement instructions for updating 16-bit pointers to circular buffers. The buffers must reside at 4K-aligned addresses. A 16-bit immediate operand specifies the buffer size, and a 16-bit register is the destination. Pointers can be incremented or decremented by one, two, or four. These circular buffers are often used in signal processing algorithms.

Addressing Modes

Memory addressing is little-endian (i.e., the byte address of a word or doubleword is the address of the least-significant byte). Stacks grow down, using pre-decrement/post-increment addressing.

A 5-bit address mode field is available in some of the double-operand instruction formats. In some cases, the encoding of these five bits is an escape code to get an additional opcode byte, which includes a 6-bit register specifier. In modes that don't use the extra opcode byte, only the four 16- or 32-bit data registers of the current bank or the four 32-bit address registers are available as the base for address arithmetic. With the extra byte, any 16- or 32-bit data register in any bank can be referenced, as can the DMA control registers.

The general addressing modes are: absolute, register indirect, push (auto-increment), pop (auto-decrement), register+immediate, and register+register. In the latter two modes, the index can be either 8 or 16 bits.

Peripherals

A four-channel DMA controller is implemented in microcode for high-bandwidth block transfers, similar to the DMA controller offered in SGS-Thomson's ST9 family of 8-bit microcontrollers *(see **061002.PDF**)*. Up to four interrupt vectors can be specified to invoke DMA transfers at the full bus bandwidth. A dedicated set of DMA registers provides control for up to 64K byte or word transfers with 32-bit address calculations (although only 24 bits are available on the external bus). Performance is 1.6 µs per transfer. Because the DMA is microcoded, normal instruction execution is suspended during a DMA transfer.

Interrupts are routed through a 64-byte vector table at address 0x8000. There isn't a trap for division by zero, but there are separate traps for attempting to execute a privileged instruction while the SYSM bit is clear, and for attempting to execute undefined opcodes. A software interrupt instruction is available, but it can call only the first eight interrupts in the table.

Price and Availability

The TLCS-900 family is in production, packaged in an 80-pin PQFP. In 10K quantities, the ROM-less TMP96C141 is priced at $7, the 32K mask ROM TMP96CM40 at $9, and the OTP ROM TMP96PM40 at $18. A development system, consisting of an ISA-based hardware emulator with C and assembly-language support, is available for $7,000.

Contact Toshiba America Electronic Components, 9775 Toledo Way, Irvine, Calif., 92718; 714.455.2000, fax 714.859.3963.

A multiplexed external bus with 24 bits of address and 16 bits of data is available. It supports dynamic bus sizing between 8- and 16-bit bus widths.

A four-channel, "10-bit" A/D converter (ADC) is provided. Its conversion time is 16 µs. The specifications for the ADC guarantee only four bits of accuracy across the full temperature and voltage range, or six bits at 25° C and 5.0 V. Typical specs are presumably better.

A watchdog timer helps correct faults caused by software errors or hardware glitches such as noise. It can be programmed to issue a non-maskable interrupt (NMI) or reset to the CPU if it isn't cleared periodically. There are six general-purpose timers: four 8-bit timers, and two 16-bit timers. Two of the 8-bit timers are set up for generating PWM outputs, but they also can be used as general-purpose timers. The 16-bit timers each have two sets of compare and capture registers, used for implementing complex timing mechanisms such as digital phase-locked loops (for example, in automotive applications for high-resolution tracking of crankshaft rotation).

Two four-output pattern generators support synthesis of stepper-motor control signals. They can be programmed to issue quadrature signals with a 50% duty cycle or loaded with a user-defined pattern that is recirculated in a shift register. The general-purpose timers supply timing for the pattern generators.

There are two serial channels with baud rate generators. Both can handle asynchronous I/O; one can also handle synchronous serial I/O. Maximum asynchronous communication speed is 625 kbps; synchronous is 2.5 Mbps. The latter is commonly used to interface to serial peripheral chips such as ADCs and LCD display drivers.

Conclusion

The TLCS-900 has the features needed to be competitive in the 16-bit arena, but its speed is somewhat low compared to the fastest 16-bit microcontrollers. Although 200 ns is quoted as the minimum instruction cycle, only a few instructions, such as NOP, achieve this speed. NEC's K6 and Siemens' 80C166 achieve minimum instruction cycles of 125 ns and 100 ns, respectively, even though they use the same number of clock cycles for their fastest instructions. Toshiba says that, in 1Q94, it will introduce new family members that use fewer cycles per instruction.

At $9 in quantities of 10K for the 32K mask ROM version, the TLCS-900 is very competitive with the 80C166, which is $16.50 with 32K ROM, also in 10K quantity. NEC's K6 series is not marketed in the US, but average Japanese pricing is ¥1,500 (about $15) for unquantified "typical" volumes. The instruction set is unique among general-purpose microcontrollers in its support for bit-reversal and circular buffers. These features could attract low-end DSP applications, allowing the chip to serve as a DSP and a controller. ♦

Operand Size			Instruction	Description
8	16	32		
•	•	•	LD	Load (also store)
•	•	•	PUSH, POP	Push, pop
	•	•	LDA, LDAR	Load address (relative)
•	•		EX	Exchange
	•		MIRR	Mirror (bit reverse)
•	•		LDI, LDIR	Load increment (repeat)
•	•		LDD, LDR	Load decrement (repeat)
•	•		CPI, CPIR	Compare increment (repeat)
•	•		CPD, CPDR	Compare decrement (repeat)
•	•	•	ADD, ADC	Add (with carry)
•	•	•	SUB, SBC	Subtract (with carry)
•	•	•	CP	Compare
•	•	•	AND, OR	Logical AND, OR
•	•	•	XOR	Logical Exclusive OR
•	•	•	INC, DEC	Increment ,Decrement
•	•		MUL, DIV	Unsigned Multiply/Divide
•	•		MULS, DIVS	Signed Multiply/Divide
	•		MULA	Multiply-accumulate
	•		MINC1, 2, 4	Modulo-addressing, add 1, 2, or 4
	•		MDEC1, 2, 4	Modulo-addressing, subtract 1, 2, or 4
•	•		NEG, CPL	Negate, Complement
	•	•	EXTZ	Zero extend
	•	•	EXTS	Sign extend
•			DAA	Decimal adjust accumulator
	•	•	PAA	Pointer adjust accumulator
			LDCF, STCF	Load/Store carry flag
			ANDCF,ORCF	And/Or carry flag
			XORCF	Exclusive OR carry flag
			RCF, SCF	Reset/Set carry flag
			CCF	Complement carry flag
			ZCF	Copy zero flag to carry flag
			BIT	Bit test
			RES, SET	Bit reset/set
			CHG	Bit invert
			TSET	Bit test and set
			BS1F, BS1B	Bit scan forward/backward
			NOP	No operation
			NORMAL	Normal mode (i.e., minimum mode)
			MAX	Maximum mode
			EI,DI	Enable/Disable interrupt
	•		PUSH SR	Push SR register
	•		POP SR	Pop SR register
			SWI	Software interrupt
			HALT	Halt
•	•	•	LDC	Load control register
			LDX	Load extract
		•	LINK	Allocate local variables for subroutine
		•	UNLK	Deallocate local variables
			LDF	Load register file pointer (bank select)
			INCF	Increment register file pointer
			DECF	Decrement register file pointer
			SCC	Set condition code
•	•	•	RLC, RRC	Rotate left/right without carry
•	•	•	RL, RR	Rotate left/right with carry
•	•	•	SLA, SRA	Shift left/right arithmetic
•	•	•	SLL,SRL	Shift left/right logical
•			RLD,RRD	Rotate left/right digit (i.e., nibble)
			JR, JRL	Jump relative (long)
			JP	Jump
			CALL, CALR	Call subroutine (relative)
			DJNZ	Decrement and jump if not zero
			RET	Return
			RETD	Return and deallocate stack frame
			RETI	Return from interrupt

Table 1. TLCS-900 instruction set.

