AccessionIndex: TCD-SCSS-T.20250919.009
Accession Date: 19-Sep-2025
Accession By: Dr.Brian Coghlan
Object name: Texas Instruments TMS1000 microcontroller and associated chips
Vintage: 1974
Synopsis: First high-volume general-purpose commercial microcontroller.

Description:
Following its success with its TMS1802NC single-chip calculator (later renamed TMS0102, the first of the TMS0100 family), Texas Instruments introduced the TMS1000 microcontroller in October 1974 [1][2][3][4][5][6][7][8]. It was the first high-volume general-purpose commercial single-chip microcontroller, containing CPU, ROM, RAM and I/O all on one chip, although it was patented as a calculator chip [9]. The difference was that the TMS0100 were *N*-word *register processors*, whereas the TMS1000 were single-word *digit processors*. Both were extremely successful products. Later the TMS1000 evolved into a family of at least a dozen 4-bit microcontrollers, some listed below.

	TMS 1000	**TMS 1200**	**TMS 1070**	**TMS 1270**	**TMS 1100**	**TMS 1300**
Pins	28	40	28	40	28	40
ROM bytes	1024	1034	1024	1024	2048	2048
RAM bytes	64	64	64	64	128	128
control outputs R	11	13	11	13	11	16
data outputs O	8	8	8	10	8	8

Like most microcontrollers, the TSM1000 had a *Harvard Architecture*, where code and data are stored and accessed separately. Code and data had different sizes, 8-bits for instructions and 4-bits for data. A 256-bit RAM was arranged as four banks (pages) of sixteen 4-bit registers. Since the space for RAM was limited, new circuits were developed for RAM, see the excellent explanation by Ken Shirriff [10].

There was a 4-bit ALU and accumulator **A**. Both RAM and ROM were paged. RAM was addressed by 4-bit **Y** (pointer) register and a 2 or 3-bit **X** (page) register, where the width of **X** depended on the size of RAM. Similarly, ROM was addressed by a 6-bit **PC** (program counter) and a 4-bit **PA** (page address) register, plus a **CA** (chapter address) flag when there was 2kB of ROM. Any ROM page changes had to be preloaded into a **PB** (page buffer) register. No stack was provided, but an **SR** (subroutine) register was provided to store the **PC** and Carry flag to allow for at least one level of subroutine. There was no such thing as an interrupt.

Data was read via a 4-bit **K** bus using specific input instructions. There were two kinds of outputs, **O** data outputs and **R** control outputs (the number of each varied across the family). A register determined which **R** outputs were set or reset. The **O** outputs were controlled by a mask-programmed PLA and driven by a 5-bit latch. An external reset (**INIT**) had to be activated for at least 6 clock cycles after power-up in order that the **PA** and **PB** registers be preloaded with binary ones, and the **PC**, **O** and **R** be zeroed.

From Ken Shirriff:

The outputs are also unusual: it has 8 "**O**" output lines, but these are not individually controllable. Instead, a 5-bit value is converted to the eight outputs by a customizable PLA decoder. The motivation behind this is to drive a 7-segment display. The microcontroller also has 11 "**R**" outputs, which are typically used to multiplex the LED display and to scan the keyboard. Another curious feature of the TMS1000 is that the instruction set was somewhat customizable.

The instruction set data movement, arithmetic, branching, looping, and I/O control. BCD arithmetic instructions were provided, but there were no instructions for AND or OR of registers. All instructions executed in 6 clock cycles at a maximum of 0.4MHz. The clock could be generated internally based on an external RC circuit, or fed from an external clock source. Again from Ken Shirriff:

The TMS1000 is implemented with complex logic circuitry, using a five-phase clock. The TMS1000 uses a mixture of depletion loads, gated loads, or precharge logic, for power savings. I'm not sure why the TMS1000 uses a five-phase clock. Four-phase logic was a logic design methodology at the time, but the TMS1000 circuitry doesn't appear to use four-phase principles. Among other things, the TMS1000 phases are irregular and $\Phi 4$ pulses twice per cycle.

The chips were fabricated using pMOS and needed a single 9V or 15V power supply, consuming about 6mA. Logic levels were not TTL-compatible. Its outputs were open drain, with customisable pull-downs. This allowed the device to interface with higher voltage loads and multiplexed displays without complex drivers. Some variants could drive high-voltage displays directly.

The TMS1100 was introduced by Texas Instruments in 1975 as an expanded memory version of the TMS1000.

Many thanks to Brian Coghlan for donating these items.

The homepage for this catalog is at: https://www.scss.tcd.ie/SCSSTreasuresCatalog/
Click *'Accession Index'* (1st column listed) for related folder, or *'About'* for further guidance.
Some of the items below may be more properly part of other categories of this catalog,
but are listed here for convenience.

Accession Index	Object with Identification
TCD-SCSS-T.20250919.009	Texas Instruments TMS1000 and associated chips. First high-volume general-purpose commercial microcontroller. 1974.
TCD-SCSS-T.20250919.009.01	Texas Instruments TMS1000 microcontroller. 1974.
TCD-SCSS-T.20250919.009.02	Texas Instruments TMS1100 microcontroller. 1975.
TCD-SCSS-X.20250916.001	Dr.Brian Coghlan's Collection of Early Microprocessors. An extensive and nearly complete set of unused 1970s microprocessor chips, most accompanied with documentation, some with demonstration boards. 1971.

References:

1. Wikipedia, *Texas Instruments TMS1000*, see:
   https://en.wikipedia.org/wiki/Texas_Instruments_TMS1000
   Last browsed to on 19-Sep-2025.

2. Wikipedia, *Texas Instruments TMS1100*, see:
   https://en.wikipedia.org/wiki/Texas_Instruments_TMS1100
   Last browsed to on 19-Sep-2025.

3. Texas Instruments, *TMS1000 Series MOS LSI One-Chip Microcomputers*, 1975, see:
   http://bitsavers.informatik.uni-stuttgart.de/components/ti/TMS1000/TMS_1000_Series_MOS_LSI_One-Chip_Microcomputers_1975.pdf
   Also: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250919.009/TMS1000-Series-MOS-LSI-One-Chip-Microcomputers-1975.pdf
   Last browsed to on 19-Sep-2025.

4. Texas Instruments, *TMS1000 Programmer's Reference Manual*, 1975, see:
   https://bitsavers.trailing-edge.com/components/ti/TMS1000/TMS1000pgmRef_1975.pdf
   Also: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250919.009/TMS1000-ProgrammersReferenceManual-1975.pdf
   Last browsed to on 19-Sep-2025.

5. Datamath, *Texas Instruments TMS1000 Product Family*, see:
   http://www.datamath.org/Chips/TMS1000.htm
   Last browsed to on 19-Sep-2025.

6. NAMIP Computer Museum, *Curated resources about TMS 1000 microcontroller and handheld games/calculators*, see:
   https://github.com/NAMIP-Computer-Museum/tms1000
   Last browsed to on 19-Sep-2025.

7. Jenny List, *The TMS1000: the first commercially available microcontroller*, see: https://hackaday.com/2020/02/18/the-tms1000-the-first-commercially-available-microcontroller/
Also: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250919.009/TMS1000-The-First-Commercially-Available-Microcontroller-Hackaday-JennyList.pdf
Last browsed to on 19-Sep-2025.

8. Steven Leibson, *A History of Early Microcontrollers, Part 2: The Texas instruments TMS1000*, 21st November, 2022, see: https://www.eejournal.com/article/a-history-of-early-microcontrollers-part-2-the-texas-instruments-tms1000/
Also: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250919.009/History-of-EarlyMicrocontrollers-TMS1000-EEJournal-StevenLeibson-21Nov2022.pdf
Last browsed to on 19-Sep-2025.

9. Gary Boone and Michael J. Cochran, *Variable function programmed calculator*, 1977 (continuation of Ser.No.163,565, filed 19th July, 1971 [abandoned]), see: https://patents.google.com/patent/US4074351
Last browsed to on 19-Sep-2025.

10. Ken Shirriff, *Reverse engineering RAM storage in early Texas Instruments calculator chip*, see: https://www.righto.com/2020/11/reverse-engineering-ram-storage-in.html
Also: https://www.scss.tcd.ie/SCSSTreasuresCatalog/hardware/TCD-SCSS-T.20250919.009/Reverse-engineering-RAM-in-early-TexasInstruments-chips-KenShirriff.pdf
Last browsed to on 19-Sep-2025.

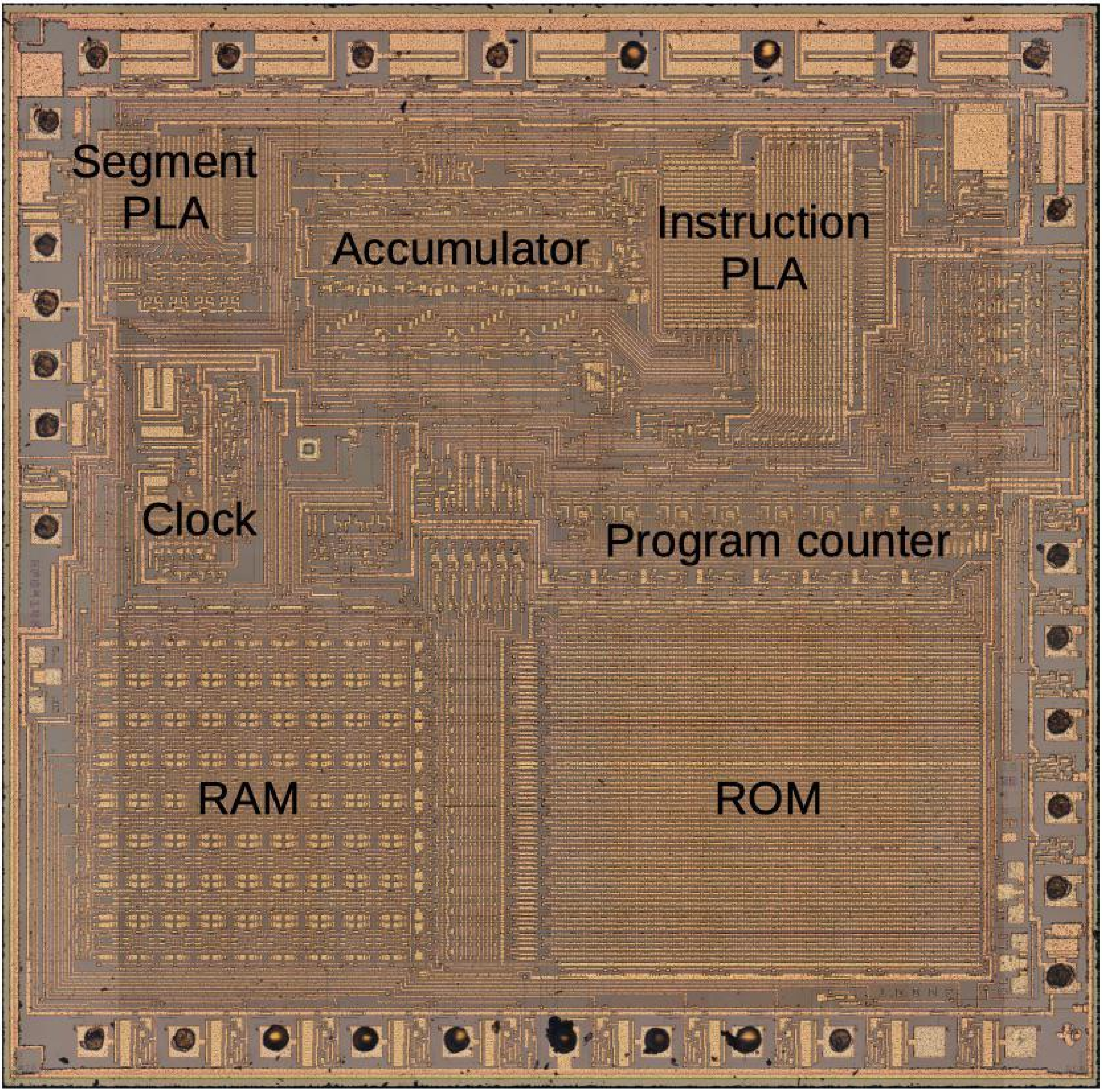

Figure 1: Texas Instruments TMS1000 chip die micrograph (from Ken Shirriff)..

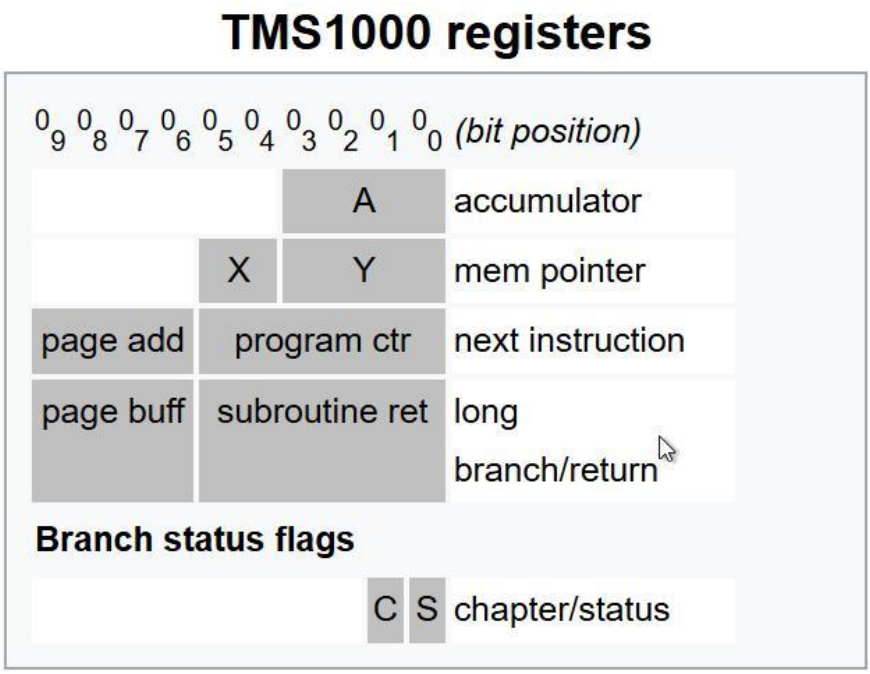

Figure 2: Texas Instruments TMS1000 architecture (from Wikipedia).

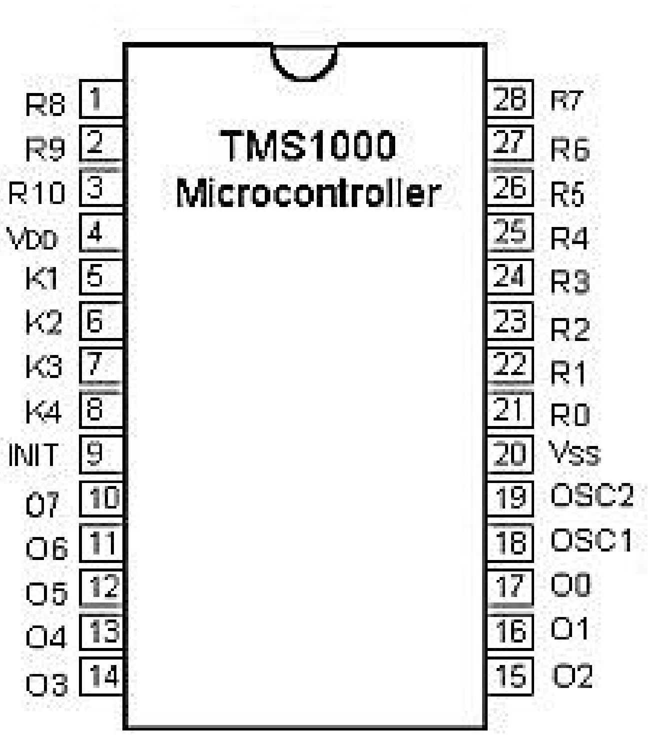

Figure 3: Texas Instruments TMS1000 pinout (from Wikipedia).

Figure 4: Texas Instruments TMS1000 front view.

Figure 5: Texas Instruments TMS1100 front view.