TEKMOS

TK68020 Microprocessor

November 8, 2019 **Product Overview**

Features

- Addressing Mode Extensions for Enhanced Support of High-Level Languages
- Object-Code Compatible with Earlier M68000 Microprocessors
- Addressing Mode Extensions for Enhanced Support of High-Level Languages
- New Bit Field Data Type Accelerates Bit-Oriented Applications e.g., Video Graphics
- An On-Chip Instruction Cache for Faster Instruction Execution
- Coprocessor Interface to Companion 32-Bit Peripherals-the MC68881 and MC68882 Floating-Point Coprocessors and the MC68851 Paged Memory Management Unit
- Pipelined Architecture with High Degree of Internal Parallelism Allowing Multiple
- Instructions To Be Executed Concurrently
- High-Performance Asynchronous Bus Is Non-multiplexed and Full 32 Bits
- Dynamic Bus Sizing Efficiently Supports 8-/16-/32-Bit Memories and Peripherals
- Full Support of Virtual Memory and Virtual Machine
- Sixteen 32-Bit General-Purpose Data and Address Registers
- Two 32-Bit Supervisor Stack Pointers and Five Special-Purpose Control Registers
- Eighteen Addressing Modes and Seven Data Types
- 4-Gbyte Direct Addressing Range
- 16-Mbyte Direct Addressing Rang
- Selection of Processor Speeds 16.67. 20, 25, and 33.33 MHz
- Available in modified 132 pin BQFP (Using adaptor) – see mechanical data section.

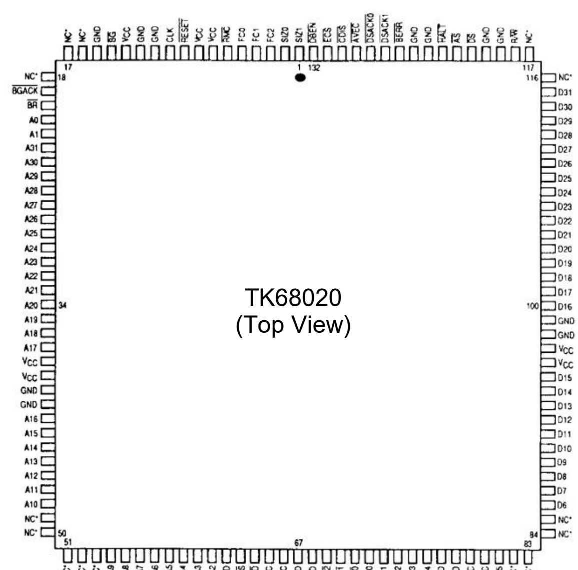
General Description

The TK68020 is a functional equivalent of the Motorola M68020 and they are pin-for-pin compatible. It is a full 32-bit implementation of the M68000 family of microprocessors from Motorola.

The TK68020 is a full 32-bit implementation of the 68000 family of microprocessors from Motorola. The TK68020 is implemented with 32-bit registers and data paths, 32-bit addresses, a rich instruction set, and versatile addressing modes.

The TK68020 is object-code compatible with earlier members of the 68000 family and has the added features of new addressing modes in support of high-level languages, an on-chip instruction cache, and a flexible coprocessor interface with full IEEE floating-point support available. The internal operations of this microprocessor operate in parallel, allowing multiple instructions to be executed concurrently.

The asynchronous bus structure of the TK68020 uses a non-multiplexed bus with 32 bits of address and 32 bits of data. The processor supports a dynamic bus sizing mechanism that allows the processor to transfer operands to or from external devices while automatically determining device port size on a cycle-by-cycle basis. The dynamic bus interface allows access to devices of differing data bus widths, in addition to eliminating all data alignment restrictions.



*NC—Do not connect to this pin.

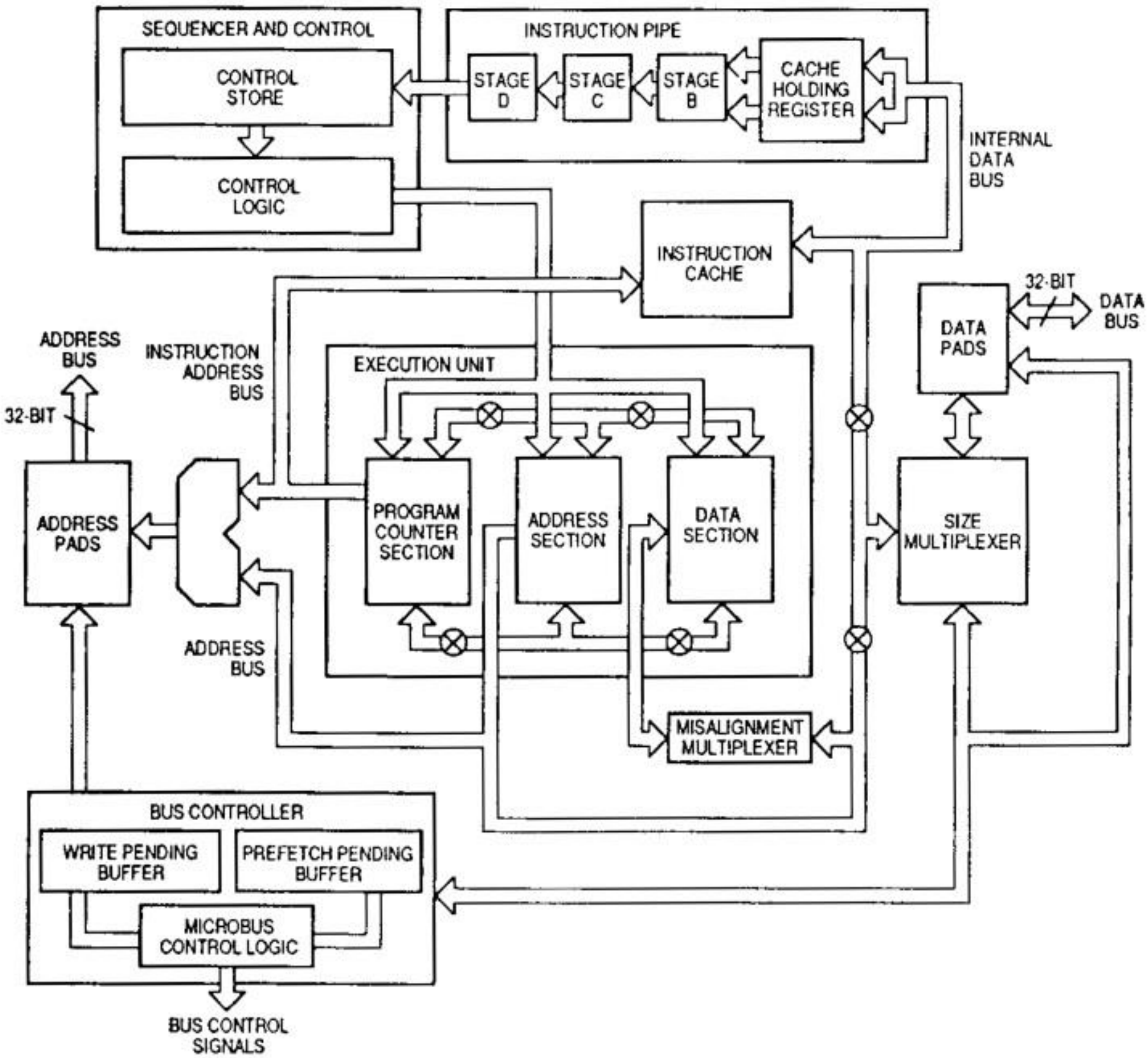


Figure 1 TK68020 Block Diagram

Programming Model

The programming model of the TK68020 consists of two groups of registers, the user model and the supervisor model, that correspond to the user and supervisor privilege levels, respectively. User programs executing at the user privilege level use the registers of the user model. System software executing at the supervisor level uses the control registers of the supervisor level to perform supervisor functions.

As shown in the programming models (see Figure 2 and Figure 3), the TK68020 has 16 32-bit general-purpose registers, a 32-bit PC, two 32-bit SSPs, a 16-bit SR, a 32-bit VBR, two 3-bit alternate function code registers, and two 32-bit cache handling (address and control) registers.

The user programming model remains unchanged from earlier 68000 family microprocessors. The supervisor programming model supplements the user programming model and is used exclusively by TK68020 system programmers who utilize the supervisor privilege level to implement sensitive operating system functions. The supervisor programming model contains all the controls to access and enable the special features of the TK68020. All application software, written to run at the non-privileged user level, migrates to the MC68020 from any 68000 platform without modification.

Registers D7-D0 are data registers used for bit and bit field (1 to 32 bits}, byte (8 bit), word (16 bit), long-word (32 bit). and quad-word (64 bit) operations. Registers A6-AO and the USP, ISP, and MSP are address registers that may be used as software stack pointers or base address registers. Register A7 (shown as A7 in Figure 1-2 and as A7' and A7'' in Figure 1-3) is a register designation that applies to the USP in the user privilege level and to either the ISP or MSP in the supervisor privilege level. In the supervisor privilege level, the active stack pointer (interrupt or master) is called the SSP. In addition, the address registers may be used for word and long-word operations. All of the 16 general-purpose registers (D7-D0, A7-AO) may be used as index registers.

The PC contains the address of the next instruction to be executed by the TK68020. During instruction execution and exception processing, the processor automatically increments the contents of the PC or places a new value in the PC, as appropriate.

The SR (see Figure 4) stores the processor status. It contains the condition codes that reflect the results of a previous operation and can be used for conditional instruction execution in a program. The condition codes are extend (X), negative (N), zero (Z), overflow (V), and carry (C). The user byte, which contains the condition codes, is the only portion of the SR information available in the user privilege level, and it is referenced as the CCR in user programs. In the supervisor privilege level, software can access the entire SR, including the interrupt priority mask (three bit s) and control bits that indicate whether the processor is in:

1. One of two trace modes (T1, TO)
2. Supervisor or user privilege level (S)
3. Master or interrupt mode (M)

The VBA contains the base address of the exception vector table in memory. The displacement of an exception vector is added to the value in this register to access the vector table.

The alternate function code registers, SFC and DFC, contain 3-bit function codes. For the TK68020, function codes can be considered extensions of the 32-bit linear address that optionally provide as many as eight 4-Gbyte address spaces. Function codes are automatically generated by the processor to select address spaces for data and program at the user and supervisor privilege levels and to select a CPU address space for processor functions (e.g., coprocessor communications). Registers SFC and DFC are used by certain instructions to explicitly specify the function codes for operations.

The CACA controls the on-chip instruction cache of the TK68020. The CAAR stores an address for cache control functions.

Data Types And Addressing Modes Overview

For detailed information on the data types and addressing modes supported by the TK68020, refer to 68000 Family Programmer's Reference Manuals.

The TK68020 supports seven basic data types:

1. Bits
2. Bit Fields (Fields of consecutive bits, (32 bits long)
3. BCD Digits (Packed: 2 digits/byte, Unpacked: 1 digit/byte)
4. Byte Integers (8 bits)
5. Word Integers (16 bits)
6. Long-Word Integers (32 bits)
7. Quad-Word Integers (64 bits)

In addition, the TK68020 instruction set supports operations on other data types such as memory addresses. The coprocessor mechanism allows direct support of floating-point operations with the MC68881 and MC68882 floating-point coprocessors as well as specialized user-defined data types and functions.

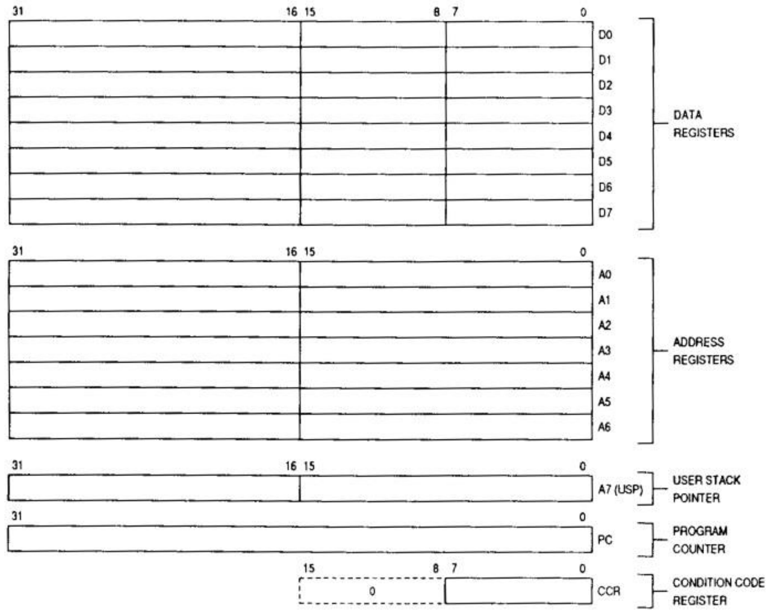


Figure 2

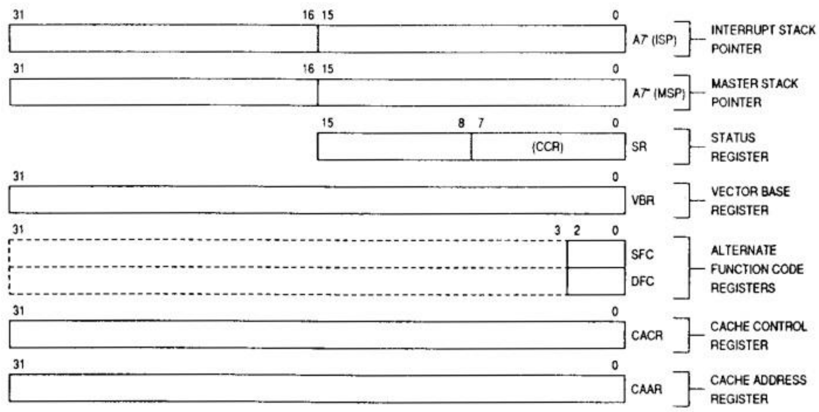


Figure 3

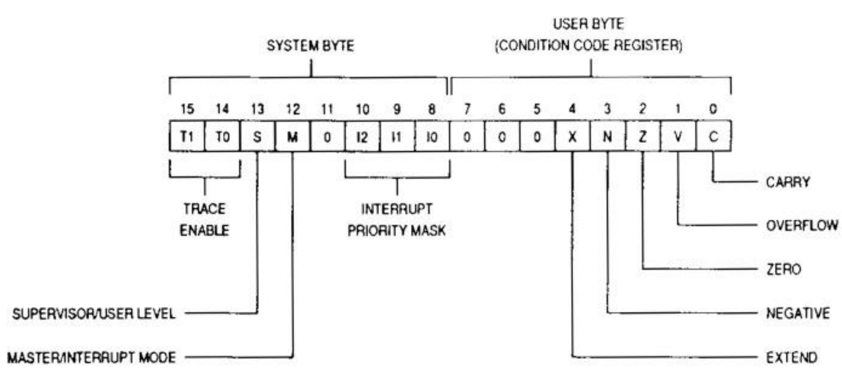


Figure 4

The 18 addressing modes listed in Table 1 include nine basic types:

1. Register Direct
2. Register Indirect
3. Register Indirect with Address
4. Memory Indirect
5. PC Indirect with Displacement
6. PC Indirect with Index
7. PC Memory Indirect
8. Absolute
9. Immediate

The register indirect addressing modes have post-increment, pre-increment, displacement and index capabilities. The PC modes have index and offset capabilities. Both modes are extended to provide indirect reference through memory. In addition to these addressing modes, many instructions specify the use of CCR, stack pointer, and/or PC.

Table 1. Addressing Modes

Addressing Modes	Syntax
Register Direct	
Data	Dn
Address	An
Register Indirect	
Address	(An)
Address with Postincrement	(An)+
Address with Predecrement	-(An)
Address with Displacement	(d_{16}, An)
Address Register Indirect with Index	
8-Bit Displacement	(d_8, An, Xn)
Base Displacement	(bd, An, Xn)
Memory Indirect	
Postindexed	([bd, An], Xn, od)
Preindexed	([bd, An, Xn], od)
PC Indirect with Displacement	(d_{16}, PC)
PC Indirect with Index	
8-Bit Displacement	(d_8, PC, Xn)
Base Displacement	(bd, PC, Xn)
PC Indirect	
Postindexed	([bd, PC], Xn, od)
Preindexed	([bd, PC, Xn], od)
Absolute Data Addressing	
Short	(xxx).W
Long	(xxx).L
Immediate	#<data>

NOTE

- Dn = Data Register, D7–D0
- An = Address Register, A7–A0
- d_8, d_{16} = A twos complement or sign-extended displacement added as part of the effective address calculation; size is 8 (d_8) or 16 (d_{16}) bits; when omitted, assemblers use a value of zero.
- Xn = Address or data register used as an index register; form is Xn.SIZE*SCALE, where SIZE is .W or .L (indicates index register size) and SCALE is 1, 2, 4, or 8 (index register is multiplied by SCALE); use of SIZE and/or SCALE is optional.
- bd = A twos-complement base displacement; when present, size can be 16 or 32 bits.
- od = Outer displacement added as part of effective address calculation after any memory indirection; use is optional with a size of 16 or 32 bits.
- PC = Program Counter
- <data> = Immediate value of 8, 16, or 32 bits
- ( ) = Effective Address
- [ ] = Use as indirect access to long-word address.

Instruction Set Overview

For detailed information on the TK68020 instruction set, refer to M6BOOOPM/AD.

The instructions in the TK68020 instruction set are listed in Table 2. The instruction set has been tailored to support structured high-level languages and sophisticated operating systems. Many instructions operate on bytes, words, or long word, and most instructions can use any of the 18 addressing modes.

Virtual Memory And Virtual Machine Concepts

The full addressing range of the TK68020 is 4 Gbytes (4,294,967,296 bytes) in each of eight address spaces. Even though most systems implement a smaller physical memory, the system can be made to appear to have a full 4 Gbytes of memory available to each user program by using virtual memory techniques.

In a virtual memory system, a user program can be written as if it has a large amount of memory available, although the physical memory actually present is much smaller. Similarly, a system can be designed to allow user programs to access devices that are not physically present in the system, such as tape drives, disk drives, printers, terminals, and so forth. With proper software emulation, a physical system can appear to be any other M68000 computer system to a user program, and the program can be given full access to all of the resources of that emulated system. Such an emulated system is called a virtual machine.

Virtual Memory

A system that supports virtual memory has a limited amount of high-speed physical memory that can be accessed directly by the processor and maintains an image of a much larger virtual memory on a secondary storage device such as a large-capacity disk drive. When the processor attempts to access a location in the virtual memory map that is not resident in physical memory, a page fault occurs. The access to that location is temporarily suspended while the necessary data is fetched from secondary storage and placed in physical memory. The suspended access is then either restarted or continued.

The TK68020 uses instruction continuation to support virtual memory. When a bus cycle is terminated with a bus error, the microprocessor suspends the current instruction and executes the virtual memory bus error handler. When the bus error handler has completed execution, it returns control to the program that was executing when the error was detected, reruns the faulted bus cycle (when required), and continues the suspended instruction.

Virtual Machine

A typical use for a virtual machine system is the development of software, such as an operating system, for a new machine also under development and not yet available for programming use. In a virtual machine system, a governing operating system emulates the hardware of the new machine and allows the new software to be executed and debugged as though it were running on the new hardware. Since the new software is controlled by the governing operating system, it is executed at a lower privilege level than the governing operating system. Thus, any attempts by the new software to use virtual resources that are not physically present (and should be emulated) are trapped to the governing operating system and performed by its software.

In the TK6B020 implementation of a virtual machine, the virtual application runs at the user privilege level. The governing operating system executes at the supervisor privilege level and any attempt by the new operating system to access supervisor resources or execute privileged instructions causes a trap to the governing operation system.

Instruction continuation is used to support virtual 1/0 devices in memory mapped input/output systems. Control and data registers for the virtual device are simulated in the memory map. An access to a virtual register causes a fault, and the function of the register is emulated by software.

Pipelined Architecture

The TK68020 contains a three-word instruction pipe where instruction opcodes are decoded. As shown in Figure 5, instruction words (instruction operation words and all extension words) enter the pipe at stage 8 and proceed to stages C and D. An instruction word is completely decoded when it reaches stage D of the pipe. Each stage has a status bit that reflects whether the word in the stage was loaded with data from a bus cycle that was terminated abnormally. Stages of the pipe are only filled in response to specific prefetch requests issued by the sequencer.

Words are loaded into the instruction pipe from the cache holding register. Although the individual stages of the pipe are only 16 bits wide, the cache holding register is 32 bits wide and contains the entire long word. This long word is obtained from the instruction cache or the external bus in response to a prefetch request from the sequencer. When the sequencer requests an even-word (long-word-aligned) prefetch, the entire long word is accessed from the instruction cache or the external bus and loaded into the cache holding register, and the high-order word is also loaded into stage B of the pipe. The instruction word for the next sequential prefetch can then be accessed directly from the cacheholding register, and no external bus cycle or instruction cache access is required. The cache holding register provides instruction words to the pipe regardless of whether the instruction cache is enabled or disabled.

Table 2

Mnemonic	Description
ABCD	Add Decimal with Extend
ADD	Add
ADDA	Add Address
ADDI	Add Immediate
ADDQ	Add Quick
ADDX	Add with Extend
AND	Logical AND
ANDI	Logical AND Immediate
ASL, ASR	Arithmetic Shift Left and Right
Bcc	Branch Conditionally
BCHG	Test Bit and Change
BCLR	Test Bit and Clear
BFCHG	Test Bit Field and Change
BFCLR	Test Bit Field and Clear
BFEXTS	Signed Bit Field Extract
BFEXTU	Unsigned Bit Field Extract
BFFFO	Bit Field Find First One
BFINS	Bit Field Insert
BFSET	Test Bit Field and Set
BFTST	Test Bit Field
BKPT	Breakpoint
BRA	Branch Always
BSET	Test Bit and Set
BSR	Branch to Subroutine
BTST	Test Bit
CALLM	Call Module
CAS	Compare and Swap Operands
CAS2	Compare and Swap Dual Operands
CHK	Check Register Against Bound
CHK2	Check Register Against Upper and Lower Bound
CLR	Clear
CMP	Compare
CMPA	Compare Address
CMPI	Compare Immediate
CMPM	Compare Memory to Memory
CMP2	Compare Register Against Upper and Lower Bounds
DBcc	Test Condition, Decrement and Branch
DIVS, DIVSL	Signed Divide
DIVU, DIVUL	Unsigned Divide
EOR	Logical Exclusive OR
EORI	Logical Exclusive Or Immediate
EXG	Exchange Registers
EXT, EXTB	Sign Extend
ILLEGAL	Take Illegal Instruction Trap
JMP	Jump
JSR	Jump to Subroutine
LEA	Load Effective Address
LINK	Link and Allocate
LSL, LSR	Logical Shift Left and Right
MOVE	Move
MOVEA	Move Address
MOVE CCR	Move Condition Code Register
MOVE SR	Move Status Register

Mnemonic	Description
MOVE USP	Move User Stack Pointer
MOVEC	Move Control Register
MOVEM	Move Multiple Registers
MOVEP	Move Peripheral
MOVEQ	Move Quick
MOVES	Move Alternate Address Space
MULS	Signed Multiply
MULU	Unsigned Multiply
NBCD	Negate Decimal with Extend
NEG	Negate
NEGX	Negate with Extend
NOP	No Operation
NOT	Logical Complement
OR	Logical Inclusive OR
ORI	Logical Inclusive OR Immediate
ORI CCR	Logical Inclusive Or Immediate to Condition Codes
ORI SR	Logical Inclusive OR Immediate to Status Register
PACK	Pack BCD
PEA	Push Effective Address
RESET	Reset External Devices
ROL, ROR	Rotate Left and Right
ROXL,ROXR	Rotate with Extend Left and Right
RTD	Return and Deallocate
RTE	Return from Exception
RTM	Return from Module
RTR	Return and Restore Codes
RTS	Return from Subroutine
SBCD	Subtract Decimal with Extend
Scc	Set Conditionally
STOP	Stop
SUB	Subtract
SUBA	Subtract Address
SUBI	Subtract Immediate
SUBQ	Subtract Quick
SUBX	Subtract with Extend
SWAP	Swap Register Words
TAS	Test and Set an Operand
TRAP	Trap
TRAPcc	Trap Conditionally
TRAPV	Trap on Overflow
TST	Test Operand
UNLK	Unlink
UNPK	Unpack BCD

COPROCESSOR INSTRUCTIONS

Mnemonic	Description
cpBcc	Branch Conditionally
cpDBcc	Test Coprocessor Condition, Decrement and Branch
cpGEN	Coprocessor General Instruction
cpRESTORE	Restore Internal State of Coprocessor
cpSAVE	Save Internal State of Coprocessor
cpScc	Set Conditionally
cpTRAPcc	Trap Conditionally

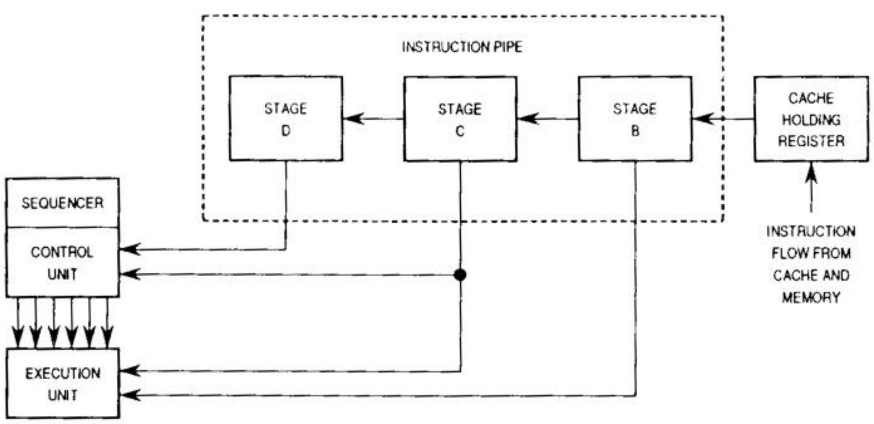


Figure 5

The sequencer is either executing microinstructions or awaiting completion of accesses that are necessary to continue executing microcode. The bus controller is responsible for all bus activity. The sequencer controls the bus controller, instruction execution, and internal processor operations such as the calculation of effective addresses and the setting of condition codes. The sequencer initiates instruction word prefetches and controls the validation of instruction words in the instruction pipe.

Prefetch requests are simultaneously submitted to the cache holding register, the instruction cache, and the bus controller. Thus, even if the instruction cache is disabled, an instruction prefetch may hit in the cache holding register and cause an external bus cycle to be aborted.

Cache Memory

Due to locality of reference, instructions that are used in a program have a high probability of being reused within a short time. Additionally, instructions that reside in proximity to the instructions currently in use also have a high probability of being utilized within a short period. To exploit these locality characteristics, the TK68020 contains an on-chip instruction cache.

The cache improves the overall performance of the system by reducing the number of bus cycles required by the processor to fetch information from memory and by increasing the bus bandwidth available for other bus masters in the system.

Electrical Characteristics

Rating	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	V
Input Voltage	V_{in}	−0.5 to +7.0	V
Operating Temperature Range Minimum Ambient Temperature Maximum Ambient Temperature PGA, PPGA, PQFP Maximum Junction Temperature CQFP	 T_A T_A T_J	 0 70 110	 °C °C °C
Storage Temperature Range	T_{stg}	−55 to 150	°C

The device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, normal precautions should be taken to avoid application of voltages higher than maximum-rated voltages to these high-impedance circuits. Tying unused inputs to the appropriate logic voltage level (e.g., either GND or V_{CC}) enhances reliability of operation.

Table 3

(V_{CC} = 5.0 V_{dc} ± 5%, GND = 0 V_{dc}; Temperature within defined ranges)

Characteristics		Symbol	Min	Max	Unit
Input High Voltage		V_{IH}	2.0	V_{CC}	V
Input Low Voltage		V_{IL}	GND −0.5	0.8	V
Input Leakage Current GND ≤ V_{in} ≤ V_{CC}	$\overline{BERR}$, $\overline{BR}$, $\overline{BGACK}$, CLK, $\overline{IPL2}$–$\overline{IPL0}$, $\overline{AVEC}$,$\overline{DSACK1}$, $\overline{DSACK0}$, $\overline{CDIS}$ $\overline{HALT}$, $\overline{RESET}$	I_{in}	 −1.0 −20	 1.0 20	μA
Hi-Z (Off-State) Leakage Current @ 2.4 V/0.5 V	A31–A0, $\overline{AS}$, $\overline{DBEN}$, $\overline{DS}$, D31–D0, FC2– FC0, R/$\overline{W}$, $\overline{RMC}$, SIZ1–SIZ0	I_{TSI}	 −20	 20	μA
Output High Voltage I_{OH} = 400 μA	A31–A0, $\overline{AS}$, $\overline{BG}$, D31–D0, $\overline{DBEN}$, $\overline{DS}$, R/$\overline{W}$, $\overline{ECS}$, $\overline{IPEND}$, $\overline{RMC}$, SIZ1–SIZ0, FC2–FC0	V_{OH}	 2.4	 —	V
Output Low Voltage I_{OL} = 3.2 mA I_{OL} = 5.3 mA I_{OL} = 2.0 mA I_{OL} = 10.7 mA	 A31–A0, FC2–FC0, SIZ1–SIZ0, $\overline{BG}$, D31–0 $\overline{AS}$, $\overline{DS}$, R/$\overline{W}$, $\overline{RMC}$, $\overline{DBEN}$, $\overline{IPEND}$, $\overline{ECS}$, $\overline{OCS}$ $\overline{HALT}$, $\overline{RESET}$	V_{OL}	 — — — —	 0.5 0.5 0.5 0.5	V
Power Dissipation (T_A = 0°C)		P_D	—	2.0	W
Capacitance (see Note) V_{in} = 0 V, T_A = 25°C, f = 1 MHz		C_{in}	—	20	pF
Load Capacitance	$\overline{ECS}$, $\overline{OCS}$ All Other	C_L	— —	50 130	pF

NOTE: Capacitance is periodically sampled rather than 100% tested.

Table 4

Mechanical Data

The Tekmos TK68020 features a 144 pin TQFP attached to an adapter card to allow the adapter to fit the same footprint as the BQFP it replaces. (Plastic BQFP are not available.) As shown in Figure 7, there are no corner bumpers on this adapter card version.

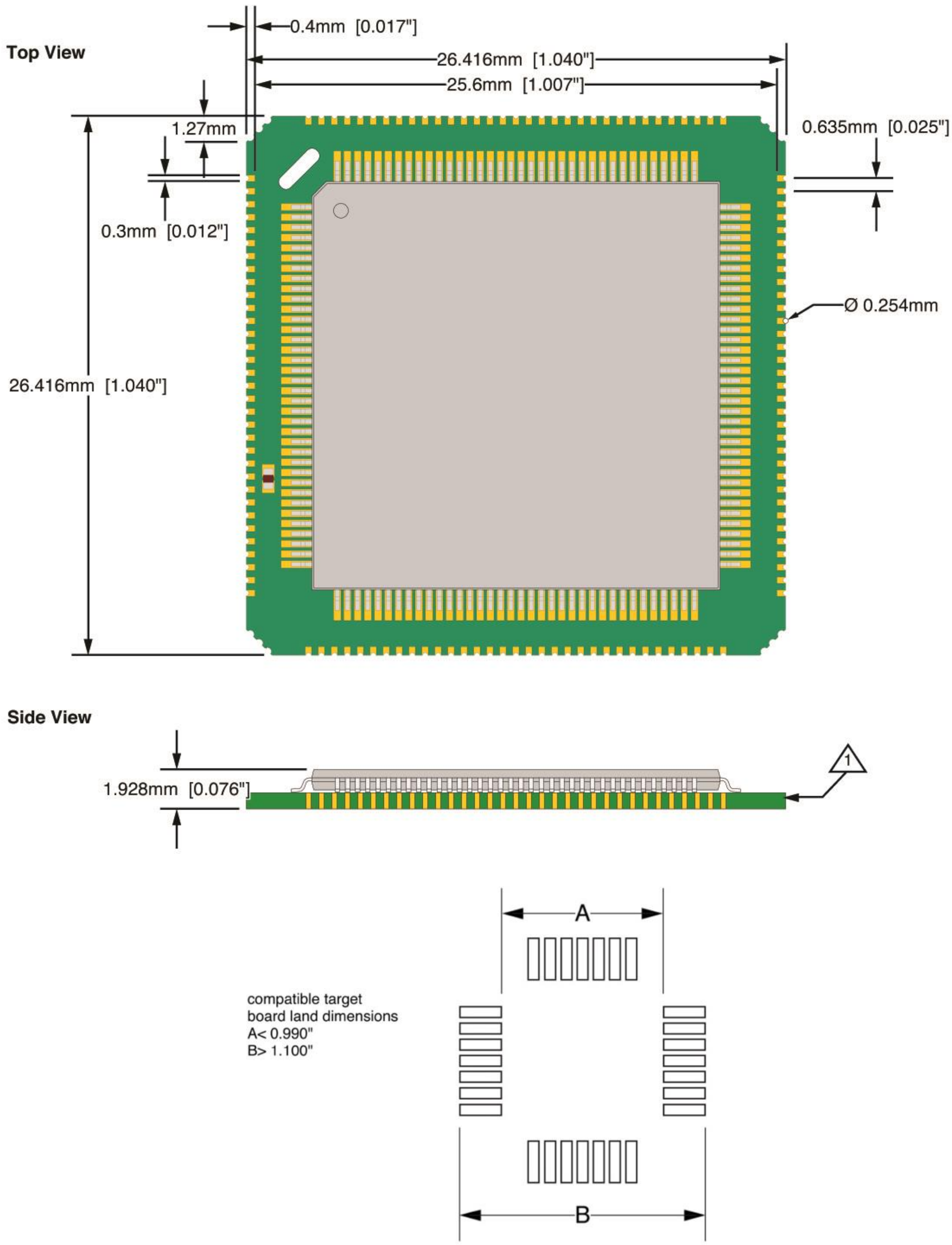
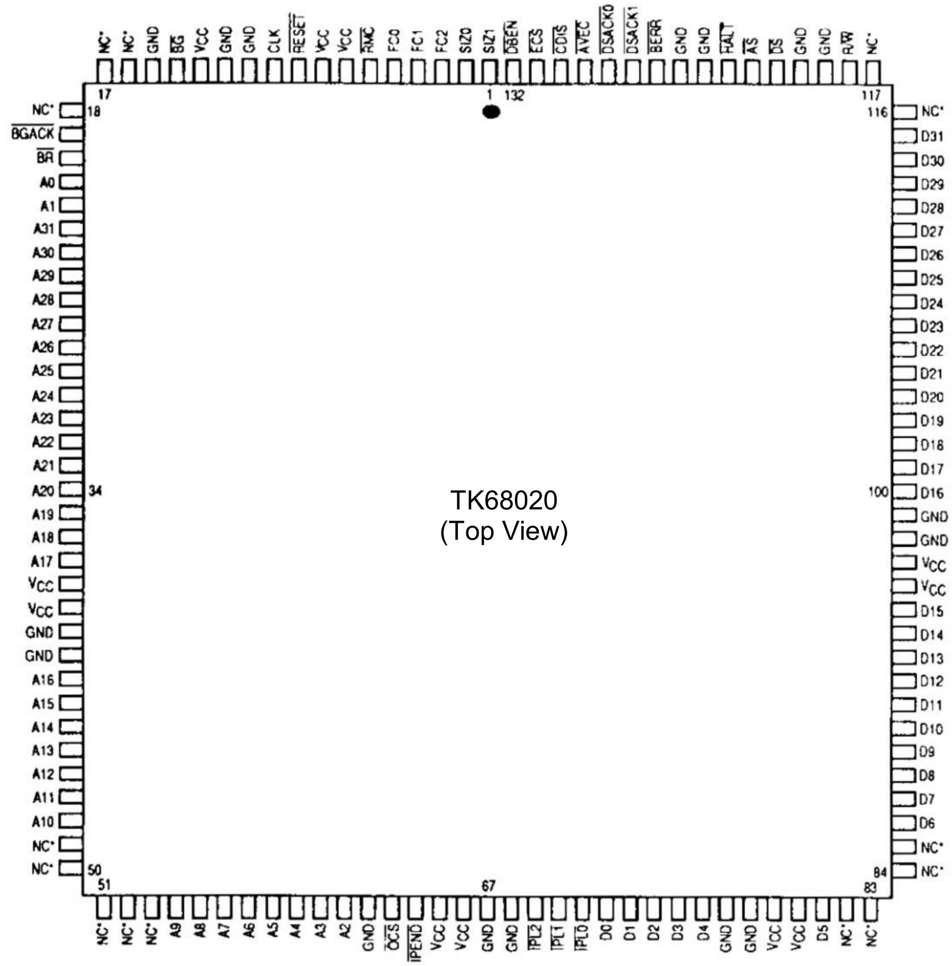


Figure 6



*NC—Do not connect to this pin.

Figure 7

Ordering Information

Code	Temperature	Package	Frequency
TK68020FC16E	0 to +70C	Modified 132 pin BQFP (Using adaptor) - RoHS	16 MHz

Contact Information

The TK68020 may be ordered directly from Tekmos:

512-342-9871 phone
Sales@Tekmos.com
www.Tekmos.com

Revision History

Date	Revision	Description
1/26/14	1.0	Initial Release
9/03/14	1.1	Ceramic package reference removed