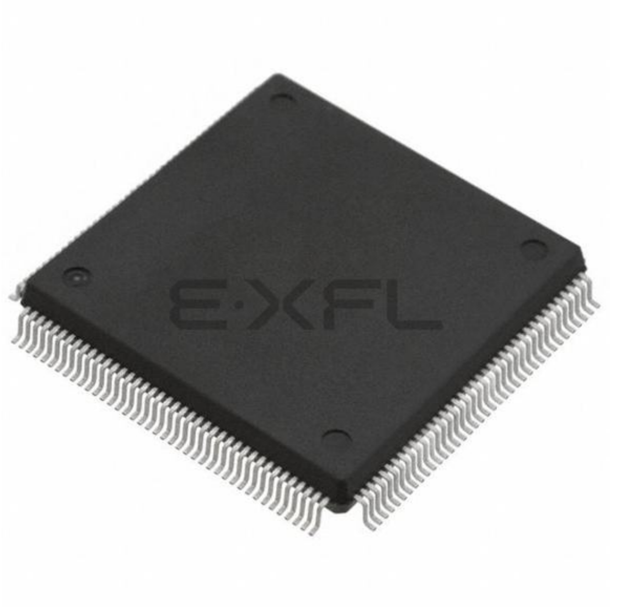

Details	
Product Status	Obsolete
Core Processor	CPU32
Core Size	32-Bit Single-Core
Speed	25MHz
Connectivity	CANbus, EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	18
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	7.5K x 8
Voltage - Supply (Vcc/Vdd)	4.75V ~ 5.25V
Data Converters	A/D 16x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	160-BQFP
Supplier Device Package	160-QFP (28x28)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/sc68376bavab25

TABLE OF CONTENTS
(Continued)

Paragraph **Title** **Page**

$\overline{\text{DSACK[1:0]}}$	—	Data and Size Acknowledge
DSCLK	—	Development Serial Clock
DSI	—	Development Serial Input
DSO	—	Development Serial Output
ECLK	—	MC6800 Devices and Peripherals Bus Clock
ETRIG[2:1]	—	QADC External Trigger
EXTAL	—	Crystal Oscillator Input
FC[2:0]	—	Function Codes
FREEZE	—	Freeze
$\overline{\text{HALT}}$	—	Halt
$\overline{\text{IFETCH}}$	—	Instruction Fetch
$\overline{\text{IPIPE}}$	—	Instruction Pipeline
$\overline{\text{IRQ[7:1]}}$	—	Interrupt Request
MA[2:0]	—	QADC Multiplexed Address
MISO	—	QSM Master In Slave Out
MODCLK	—	Clock Mode Select
MOSI	—	QSM Master Out Slave In
PCS[3:0]	—	QSM Peripheral Chip-Selects
PQA[7:0]	—	QADC Port A
PQB[7:0]	—	QADC Port B
PC[6:0]	—	SIM Port C
PE[7:0]	—	SIM Port E
PF[7:0]	—	SIM Port F
QUOT	—	Quotient Out
R/$\overline{\text{W}}$	—	Read/Write
$\overline{\text{RESET}}$	—	Reset
$\overline{\text{RMC}}$	—	Read-Modify-Write Cycle
RXD	—	SCI Receive Data
SCK	—	QSPI Serial Clock
SIZ[1:0]	—	Size
$\overline{\text{SS}}$	—	Slave Select
T2CLK	—	TPU Clock In
TPUCH[15:0]	—	TPU Channel Signals
TSC	—	Three-State Control
$\overline{\text{TSTME}}$	—	Test Mode Enable
V_{RH}	—	QADC High Reference Voltage
V_{RL}	—	QADC Low Reference Voltage
XFC	—	External Filter Capacitor
XTAL	—	Crystal Oscillator Output

2.5 Conventions

Logic level one is the voltage that corresponds to a Boolean true (1) state.

Logic level zero is the voltage that corresponds to a Boolean false (0) state.

Set refers specifically to establishing logic level one on a bit or bits.

Clear refers specifically to establishing logic level zero on a bit or bits.

Asserted means that a signal is in active logic state. An active low signal changes from logic level one to logic level zero when asserted. An active high signal changes from logic level zero to logic level one.

Negated means that an asserted signal changes logic state. An active low signal changes from logic level zero to logic level one when negated. An active high signal changes from logic level one to logic level zero.

A specific mnemonic within a range is referred to by mnemonic and number. A15 is bit 15 of accumulator A; ADDR7 is line 7 of the address bus; CSOR0 is chip-select option register 0. **A range of mnemonics** is referred to by mnemonic and the numbers that define the range. VBR[4:0] are bits four to zero of the vector base register; CSOR[0:5] are the first six option registers.

Parentheses are used to indicate the content of a register or memory location rather than the register or memory location itself. (A) is the content of accumulator A. (M : M + 1) is the content of the word at address M.

LSB means least significant bit. **MSB** means most significant bit. References to low and high bytes are spelled out.

LSW means least significant word. **MSW** means most significant word.

ADDR is the address bus. ADDR[7:0] are the eight LSBs of the address bus.

DATA is the data bus. DATA[15:8] are the eight MSBs of the data bus.

SECTION 4 CENTRAL PROCESSOR UNIT

The CPU32, the instruction processing module of the M68300 family, is based on the industry-standard MC68000 processor. It has many features of the MC68010 and MC68020, as well as unique features suited for high-performance controller applications. This section is an overview of the CPU32. For detailed information concerning CPU operation, refer to the *CPU32 Reference Manual* (CPU32RM/AD).

4.1 General

Ease of programming is an important consideration in using a microcontroller. The CPU32 instruction format reflects a philosophy emphasizing register-memory interaction. There are eight multifunction data registers and seven general-purpose addressing registers.

All data resources are available to all operations requiring those resources. The data registers readily support 8-bit (byte), 16-bit (word), and 32-bit (long-word) operand lengths for all operations. Word and long-word operations support address manipulation. Although the program counter (PC) and stack pointers (SP) are special-purpose registers, they are also available for most data addressing activities. Ease of program checking and diagnosis is further enhanced by trace and trap capabilities at the instruction level.

A block diagram of the CPU32 is shown in **Figure 4-1**. The major blocks operate in a highly independent fashion that maximizes concurrency of operation while managing the essential synchronization of instruction execution and bus operation. The bus controller loads instructions from the data bus into the decode unit. The sequencer and control unit provide overall chip control, managing the internal buses, registers, and functions of the execution unit.

Bus cycles terminated by DSACK assertion normally require a minimum of three CLKOUT cycles. To support systems that use CLKOUT to generate DSACK and other inputs, asynchronous input setup time and asynchronous input hold times are specified. When these specifications are met, the MCU is guaranteed to recognize the appropriate signal on a specific edge of the CLKOUT signal.

For a read cycle, when assertion of DSACK is recognized on a particular falling edge of the clock, valid data is latched into the MCU on the next falling clock edge, provided that the data meets the data setup time. In this case, the parameter for asynchronous operation can be ignored.

When a system asserts DSACK for the required window around the falling edge of S2 and obeys the bus protocol by maintaining DSACK and BERR or HALT until and throughout the clock edge that negates AS (with the appropriate asynchronous input hold time), no wait states are inserted. The bus cycle runs at the maximum speed of three clocks per cycle.

To ensure proper operation in a system synchronized to CLKOUT, when either BERR or BERR and HALT is asserted after DSACK, BERR (or BERR and HALT) assertion must satisfy the appropriate data-in setup and hold times before the falling edge of the clock cycle after DSACK is recognized.

5.6.2 Regular Bus Cycles

The following paragraphs contain a discussion of cycles that use external bus control logic. Refer to **5.6.3 Fast Termination Cycles** for information about fast termination cycles.

To initiate a transfer, the MCU asserts an address and the SIZ[1:0] signals. The SIZ signals and ADDR0 are externally decoded to select the active portion of the data bus. Refer to **5.5.2 Dynamic Bus Sizing**. When AS, DS, and R/W are valid, a peripheral device either places data on the bus (read cycle) or latches data from the bus (write cycle), then asserts a DSACK[1:0] combination that indicates port size.

The DSACK[1:0] signals can be asserted before the data from a peripheral device is valid on a read cycle. To ensure valid data is latched into the MCU, a maximum period between DSACK assertion and DS assertion is specified.

There is no specified maximum for the period between the assertion of AS and DSACK. Although the MCU can transfer data in a minimum of three clock cycles when the cycle is terminated with DSACK, the MCU inserts wait cycles in clock period increments until either DSACK signal goes low.

Table 5-17 SIM Pin Reset States

Pin(s)	Pin State While $\overline{\text{RESET}}$ Asserted	Pin State After $\overline{\text{RESET}}$ Released			
		Default Function		Alternate Function	
		Pin Function	Pin State	Pin Function	Pin State
$\overline{\text{CS10}}$/ADDR23/ECLK	V_{DD}	$\overline{\text{CS10}}$	V_{DD}	ADDR23	Unknown
$\overline{\text{CS[9:6]}}$/ADDR[22:19]/PC[6:3]	V_{DD}	$\overline{\text{CS[9:6]}}$	V_{DD}	ADDR[22:19]	Unknown
ADDR[18:0]	High-Z	ADDR[18:0]	Unknown	ADDR[18:0]	Unknown
$\overline{\text{AS}}$/PE5	High-Z	$\overline{\text{AS}}$	Output	PE5	Input
$\overline{\text{AVEC}}$/PE2	High-Z	$\overline{\text{AVEC}}$	Input	PE2	Input
$\overline{\text{BERR}}$	High-Z	$\overline{\text{BERR}}$	Input	$\overline{\text{BERR}}$	Input
$\overline{\text{CS1}}$/$\overline{\text{BG}}$	V_{DD}	$\overline{\text{CS1}}$	V_{DD}	$\overline{\text{BG}}$	V_{DD}
$\overline{\text{CS2}}$/$\overline{\text{BGACK}}$	V_{DD}	$\overline{\text{CS2}}$	V_{DD}	$\overline{\text{BGACK}}$	Input
$\overline{\text{CS0}}$/$\overline{\text{BR}}$	V_{DD}	$\overline{\text{CS0}}$	V_{DD}	$\overline{\text{BR}}$	Input
CLKOUT	Output	CLKOUT	Output	CLKOUT	Output
$\overline{\text{CSBOOT}}$	V_{DD}	$\overline{\text{CSBOOT}}$	V_{SS}	$\overline{\text{CSBOOT}}$	V_{SS}
DATA[15:0]	Mode select	DATA[15:0]	Input	DATA[15:0]	Input
$\overline{\text{DS}}$/PE4	High-Z	$\overline{\text{DS}}$	Output	PE4	Input
$\overline{\text{DSACK0}}$/PE0	High-Z	$\overline{\text{DSACK0}}$	Input	PE0	Input
$\overline{\text{DSACK1}}$/PE1	High-Z	$\overline{\text{DSACK1}}$	Input	PE1	Input
$\overline{\text{CS[5:3]}}$/FC[2:0]/PC[2:0]	V_{DD}	$\overline{\text{CS[5:3]}}$	V_{DD}	FC[2:0]	Unknown
$\overline{\text{HALT}}$	High-Z	$\overline{\text{HALT}}$	Input	$\overline{\text{HALT}}$	Input
$\overline{\text{IRQ[7:1]}}$/PF[7:1]	High-Z	$\overline{\text{IRQ[7:1]}}$	Input	PF[7:1]	Input
MODCLK/PF0	Mode Select	MODCLK	Input	PF0	Input
R/$\overline{\text{W}}$	High-Z	R/$\overline{\text{W}}$	Output	R/$\overline{\text{W}}$	Output
$\overline{\text{RESET}}$	Asserted	$\overline{\text{RESET}}$	Input	$\overline{\text{RESET}}$	Input
$\overline{\text{RMC}}$/PE3	High-Z	$\overline{\text{RMC}}$	Output	PE3	Input
SIZ[1:0]/PE[7:6]	High-Z	SIZ[1:0]	Unknown	PE[7:6]	Input
$\overline{\text{TSTME}}$/TSC	Mode select	TSC	Input	TSC	Input

5.7.5.2 Reset States of Pins Assigned to Other MCU Modules

As a rule, module pins that are assigned to general-purpose I/O ports go into a high-impedance state following reset. Other pin states are determined by individual module control register settings. Refer to sections concerning modules for details. However, during power-on reset, module port pins may be in an indeterminate state for a short period. Refer to **5.7.7 Power-On Reset** for more information.

5.7.6 Reset Timing

The $\overline{\text{RESET}}$ input must be asserted for a specified minimum period for reset to occur. External $\overline{\text{RESET}}$ assertion can be delayed internally for a period equal to the longest bus cycle time (or the bus monitor time-out period) in order to protect write cycles from being aborted by reset. While $\overline{\text{RESET}}$ is asserted, SIM pins are either in an inactive, high-impedance state or are driven to their inactive states.

6.6 Reset

Reset places the SRAM in low-power stop mode, enables program space access, and clears the base address registers and the register lock bit. These actions make it possible to write a new base address into the registers.

When a synchronous reset occurs while a byte or word SRAM access is in progress, the access is completed. If reset occurs during the first word access of a long-word operation, only the first word access is completed. If reset occurs during the second word access of a long-word operation, the entire access is completed. Data being read from or written to the RAM may be corrupted by asynchronous reset. Refer to **5.7 Reset** for more information about resets.

2. Set up SCCR1
   a. Select serial mode (M)
   b. Enable use (PE) and type (PT) of parity check.
   c. Select use (RWU) and type (WAKE) of receiver wake-up.
   d. Enable idle-line detection (ILT) and interrupt (ILIE).
   e. Enable or disable wired-OR operation (WOMS).
   f. Enable or disable break transmission (SBK).
3. To receive:
   a. Set the receiver (RE) and receiver interrupt (RIE) bits in SCCR1.
4. To transmit:
   a. Set transmitter (TE) and transmitter interrupt (TIE) bits in SCCR1.
   b. Clear the TDRE and TC flags by reading SCSR and writing data to SCDR.

In the low-power stop mode, QADCMCR, the interrupt register (QADCINT), and the test register (QADCTEST) are not reset and fully accessible. The data direction register (DDRQA) and port data registers (PORTQA and PORTQB) are not reset and are read-only accessible. Control register 0 (QACR0), control register 1 (QACR1), control register 2 (QACR2), and status register (QASR) are reset and are read-only accessible. The CCW table and result table are not reset and not accessible. In addition, the QADC clock (QCLK) and the periodic/interval timer are held in reset during low-power stop mode.

If the STOP bit is clear, low-power stop mode is disabled. Refer to **D.5.1 QADC Module Configuration Register** for more information.

8.6.2 Freeze Mode

The QADC enters freeze mode when background debug mode is enabled and a breakpoint is processed. This is indicated by assertion of the FREEZE line on the IMB. The FRZ bit in QADCMCR determines whether or not the QADC responds to an IMB FREEZE assertion. Freeze mode is useful when debugging an application.

When the IMB FREEZE line is asserted and the FRZ bit is set, the QADC finishes any conversion in progress and then freezes. Depending on when the FREEZE is asserted, there are three possible queue freeze scenarios:

- When a queue is not executing, the QADC freezes immediately.
- When a queue is executing, the QADC completes the current conversion and then freezes.
- If during the execution of the current conversion, the queue operating mode for the active queue is changed, or a queue 2 abort occurs, the QADC freezes immediately.

When the QADC enters the freeze mode while a queue is active, the current CCW location of the queue pointer is saved.

In freeze mode, the analog logic is held in reset and is not clocked. Although QCLK is unaffected, the periodic/interval timer is held in reset. External trigger events that occur during freeze mode are not recorded. The CPU32 may continue to access all QADC registers, the CCW table, and the result table. Although the QADC saves a pointer to the next CCW in the current queue, software can force the QADC to execute a different CCW by writing new queue operating modes before normal operation resumes. The QADC looks at the queue operating modes, the current queue pointer, and any pending trigger events to decide which CCW to execute.

If the FRZ bit is clear, assertion of the IMB FREEZE line is ignored. Refer to **D.5.1 QADC Module Configuration Register** for more information.

8.6.3 Supervisor/Unrestricted Address Space

The QADC memory map is divided into two segments: supervisor-only data space and assignable data space. Access to supervisor-only data space is permitted only when the CPU32 is operating in supervisor mode. Assignable data space can have either restricted to supervisor-only data space access or unrestricted supervisor and user

SECTION 10 CONFIGURABLE TIMER MODULE 4

This section is an overview of CTM4 function. Refer to the *CTM Reference Manual* (CTMRM/AD) for a comprehensive discussion of CTM capabilities.

10.1 General

The configurable timer module 4 (CTM4) consists of several submodules which are located on either side of the CTM4 internal submodule bus (SMB). All data and control signals within the CTM4 are passed over this bus. The SMB is connected to the outside world via the bus interface unit submodule (BIUSM), which is connected to the intermodule bus (IMB), and subsequently the CPU32. This configuration allows the CPU32 to access the data and control registers in each CTM4 submodule on the SMB. Three time base buses (TBB1, TBB2 and TBB4), each 16-bits wide, are used to transfer timing information from counters to action submodules. **Figure 10-1** shows a block diagram of the CTM4.

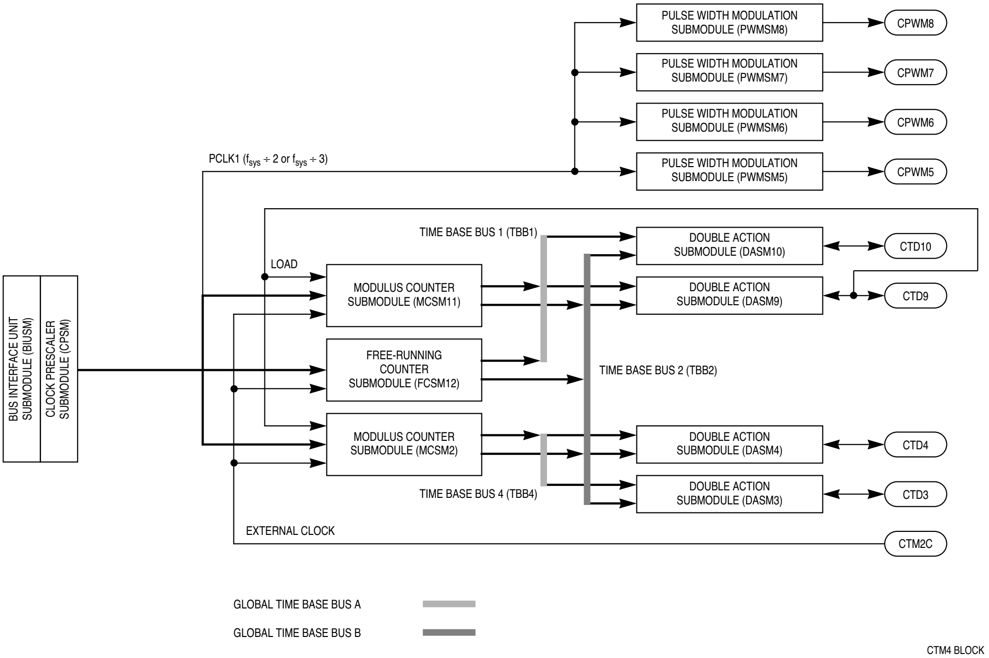
PULSE WIDTH MODULATION SUBMODULE (PWMSM8)
CPWM8
PULSE WIDTH MODULATION SUBMODULE (PWMSM7)
CPWM7
PULSE WIDTH MODULATION SUBMODULE (PWMSM6)
CPWM6
PCLK1 (f_{sys} ÷ 2 or f_{sys} ÷ 3)
PULSE WIDTH MODULATION SUBMODULE (PWMSM5)
CPWM5
TIME BASE BUS 1 (TBB1)
DOUBLE ACTION SUBMODULE (DASM10)
CTD10
LOAD
MODULUS COUNTER SUBMODULE (MCSM11)
DOUBLE ACTION SUBMODULE (DASM9)
CTD9
BUS INTERFACE UNIT SUBMODULE (BIUSM)
CLOCK PRESCALER SUBMODULE (CPSM)
FREE-RUNNING COUNTER SUBMODULE (FCSM12)
TIME BASE BUS 2 (TBB2)
MODULUS COUNTER SUBMODULE (MCSM2)
DOUBLE ACTION SUBMODULE (DASM4)
CTD4
TIME BASE BUS 4 (TBB4)
DOUBLE ACTION SUBMODULE (DASM3)
CTD3
EXTERNAL CLOCK
CTM2C
GLOBAL TIME BASE BUS A
GLOBAL TIME BASE BUS B
CTM4 BLOCK

Figure 10-1 CTM4 Block Diagram

- If both STOP and SELFWAKE are set and a recessive to dominant edge immediately occurs on the CAN bus, the TouCAN may never set the STOPACK bit, and the STOP bit will be cleared.
- To prevent old frames from being sent when the TouCAN awakes from low-power stop mode via the self-wake mechanism, disable all transmit sources, including transmit buffers configured for remote request responses, before placing the TouCAN in low-power stop mode.
- If the TouCAN is in debug mode when the STOP bit is set, the TouCAN will assume that debug mode should be exited. As a result, it will try to synchronize with the CAN bus, and only then will it await the conditions required for entry into low-power stop mode.
- Unlike other modules, the TouCAN does not come out of reset in low-power stop mode. The basic TouCAN initialization procedure (see **13.5.2 TouCAN Initialization**) should be executed before placing the module in low-power stop mode.
- If the TouCAN is in low-power stop mode with the self-wake mechanism engaged and is operating with a single system clock per time quantum, there can be extreme cases in which TouCAN wake-up on recessive to dominant edge may not conform to the CAN protocol. TouCAN synchronization will be shifted one time quantum from the wake-up event. This shift lasts until the next recessive to dominant edge, which resynchronizes the TouCAN to be in conformance with the CAN protocol. The same holds true when the TouCAN is in auto power save mode and awakens on a recessive to dominant edge.

13.6.3 Auto Power Save Mode

Auto power save mode enables normal operation with optimized power savings. Once the auto power save (APS) bit in CANMCR is set, the TouCAN looks for a set of conditions in which there is no need for the clocks to be running. If these conditions are met, the TouCAN stops its clocks, thus saving power. The following conditions will activate auto power save mode.

- No RX/TX frame in progress.
- No transfer of RX/TX frames to and from a serial message buffer, and no TX frame awaiting transmission in any message buffer.
- No CPU32 access to the TouCAN module.
- The TouCAN is not in debug mode, low-power stop mode, or the bus off state.

While its clocks are stopped, if the TouCAN senses that any one of the aforementioned conditions is no longer true, it restarts its clocks. The TouCAN then continues to monitor these conditions and stops/restarts its clocks accordingly.

Table A-5 DC Characteristics

(V_{DD} and V_{DDSYN} = 5.0 Vdc ±5%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

Num	Characteristic	Symbol	Min	Max	Unit
1	Input High Voltage	V_{IH}	0.7 (V_{DD})	V_{DD} + 0.3	V
2	Input Low Voltage	V_{IL}	V_{SS} – 0.3	0.2 (V_{DD})	V
3	Input Hysteresis[1]	V_{HYS}	0.5	—	V
4	Input Leakage Current[2] $V_{in} = V_{DD}$ or V_{SS} Input-only pins	I_{in}	–2.5	2.5	µA
5	High Impedance (Off-State) Leakage Current[2] $V_{in} = V_{DD}$ or V_{SS} All input/output and output pins	I_{OZ}	–2.5	2.5	µA
6	CMOS Output High Voltage[2, 3] I_{OH} = –10.0 µA Group 1, 2, 4 input/output and all output pins	V_{OH}	V_{DD} – 0.2	—	V
7	CMOS Output Low Voltage[2] I_{OL} = 10.0 µA Group 1, 2, 4 input/output and all output pins	V_{OL}	—	0.2	V
8	Output High Voltage[2, 3] I_{OH} = –0.8 mA Group 1, 2, 4 input/output and all output pins	V_{OH}	V_{DD} – 0.8	—	V
9	Output Low Voltage[2] I_{OL} = 1.6 mA Group 1 I/O Pins, CLKOUT, FREEZE/QUOT, $\overline{\text{IPIPE}}$ I_{OL} = 5.3 mA Group 2 and Group 4 I/O Pins, $\overline{\text{CSBOOT}}$, $\overline{\text{BG/CS}}$ I_{OL} = 12 mA Group 3	V_{OL}	— — —	0.4 0.4 0.4	V
10	Three State Control Input High Voltage	V_{IHTSC}	1.6 (V_{DD})	9.1	V
11	Data Bus Mode Select Pull-up Current[4] $V_{in} = V_{IL}$ DATA[15:0] $V_{in} = V_{IH}$ DATA[15:0]	I_{MSP}	— –15	–120 —	µA
12A	MC68336 V_{DD} Supply Current[5] RUN [6] RUN, TPU emulation mode LPSTOP, 4.194 MHz crystal, VCO Off (STSIM = 0) LPSTOP (External clock input frequency = maximum f_{sys})	I_{DD} I_{DD} S_{IDD} S_{IDD}	— — — —	140 150 3 7	mA mA mA mA
12B	MC68376 V_{DD} Supply Current[5] RUN[6] RUN, TPU emulation mode LPSTOP, 4.194 MHz crystal, VCO Off (STSIM = 0) LPSTOP (External clock input frequency = maximum f_{sys})	I_{DD} I_{DD} S_{IDD} S_{IDD}	— — — —	150 160 3 7	mA mA mA mA
13	Clock Synthesizer Operating Voltage	V_{DDSYN}	4.75	5.25	V
14	V_{DDSYN} Supply Current[5] 4.194 MHz crystal, VCO on, maximum f_{sys} External Clock, maximum f_{sys} LPSTOP, 4.194 MHz crystal, VCO off (STSIM = 0) 4.194 MHz crystal, V_{DD} powered down	I_{DDSYN} I_{DDSYN} S_{IDDSYN} I_{DDSYN}	— — — —	3 5 3 3	mA mA mA mA

Table A-6 AC Timing

(V_{DD} and V_{DDSYN} = 5.0 Vdc ±5%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)[1]

Num	Characteristic	Symbol	Min	Max	Unit
F1	Frequency of Operation[2]	f_{sys}	—	20.97	MHz
1	Clock Period	t_{cyc}	47.7	—	ns
1A	ECLK Period	t_{Ecyc}	381	—	ns
1B	External Clock Input Period[3]	t_{Xcyc}	47.7	—	ns
2, 3	Clock Pulse Width	t_{CW}	18.8	—	ns
2A, 3A	ECLK Pulse Width	t_{ECW}	183	—	ns
2B, 3B	External Clock Input High/Low Time[3]	t_{XCHL}	23.8	—	ns
3, 4	Clock Rise and Fall Time	t_{Crf}	—	5	ns
4A, 5A	Rise and Fall Time — All Outputs except CLKOUT	t_{rf}	—	8	ns
4B, 5B	External Clock Rise and Fall Time[4]	t_{XCrf}	—	5	ns
4	Clock High to Address, FC, SIZE, $\overline{\text{RMC}}$ Valid	t_{CHAV}	0	23	ns
5	Clock High to Address, Data, FC, SIZE, $\overline{\text{RMC}}$ High Impedance	t_{CHAZx}	0	47	ns
6	Clock High to Address, FC, SIZE, $\overline{\text{RMC}}$ Invalid[5]	t_{CHAZn}	0	—	ns
7	Clock Low to $\overline{\text{AS}}$, $\overline{\text{DS}}$, $\overline{\text{CS}}$ Asserted	t_{CLSA}	0	23	ns
8A	$\overline{\text{AS}}$ to $\overline{\text{DS}}$ or $\overline{\text{CS}}$ Asserted (Read)[6]	t_{STSA}	–10	10	ns
8C	Clock Low to $\overline{\text{IFETCH}}$, $\overline{\text{IPIPE}}$ Asserted	t_{CLIA}	2	22	ns
11	Address, FC, SIZE, $\overline{\text{RMC}}$ Valid to $\overline{\text{AS}}$, $\overline{\text{CS}}$ Asserted	t_{AVSA}	10	—	ns
12	Clock Low to $\overline{\text{AS}}$, $\overline{\text{DS}}$, $\overline{\text{CS}}$ Negated	t_{CLSN}	2	23	ns
12A	Clock Low to $\overline{\text{IFETCH}}$, $\overline{\text{IPIPE}}$ Negated	t_{CLIN}	2	22	ns
13	$\overline{\text{AS}}$, $\overline{\text{DS}}$, $\overline{\text{CS}}$ Negated to Address, FC, SIZE Invalid (Address Hold)	t_{SNAI}	10	—	ns
14	$\overline{\text{AS}}$, $\overline{\text{CS}}$ Width Asserted	t_{SWA}	80	—	ns
14A	$\overline{\text{DS}}$, $\overline{\text{CS}}$ Width Asserted (Write)	t_{SWAW}	36	—	ns
14B	$\overline{\text{AS}}$, $\overline{\text{CS}}$ Width Asserted (Fast Write Cycle)	t_{SWDW}	32	—	ns
15	$\overline{\text{AS}}$, $\overline{\text{DS}}$, $\overline{\text{CS}}$ Width Negated[7]	t_{SN}	32	—	ns
16	Clock High to $\overline{\text{AS}}$, $\overline{\text{DS}}$, R/$\overline{\text{W}}$ High Impedance	t_{CHSZ}	—	47	ns
17	$\overline{\text{AS}}$, $\overline{\text{DS}}$, $\overline{\text{CS}}$ Negated to R/$\overline{\text{W}}$ Negated	t_{SNRN}	10	—	ns
18	Clock High to R/$\overline{\text{W}}$ High	t_{CHRH}	0	23	ns
20	Clock High to R/$\overline{\text{W}}$ Low	t_{CHRL}	0	23	ns
21	R/$\overline{\text{W}}$ Asserted to $\overline{\text{AS}}$, $\overline{\text{CS}}$ Asserted	t_{RAAA}	10	—	ns
22	R/$\overline{\text{W}}$ Low to $\overline{\text{DS}}$, $\overline{\text{CS}}$ Asserted (Write)	t_{RASA}	54	—	ns
23	Clock High to Data Out Valid	t_{CHDO}	—	23	ns
24	Data Out Valid to Negating Edge of $\overline{\text{AS}}$, $\overline{\text{CS}}$	t_{DVASN}	10	—	ns
25	$\overline{\text{DS}}$, $\overline{\text{CS}}$ Negated to Data Out Invalid (Data Out Hold)	t_{SNDOI}	10	—	ns

Table A-8 ECLK Bus Timing

(V_{DD} = 5.0 Vdc ±5%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)[1]

Num	Characteristic	Symbol	Min	Max	Unit
E1	ECLK Low to Address Valid[2]	t_{EAD}	—	48	ns
E2	ECLK Low to Address Hold	t_{EAH}	10	—	ns
E3	ECLK Low to $\overline{CS}$ Valid ($\overline{CS}$ delay)	t_{ECSD}	—	120	ns
E4	ECLK Low to $\overline{CS}$ Hold	t_{ECSH}	10	—	ns
E5	$\overline{CS}$ Negated Width	t_{ECSN}	25	—	ns
E6	Read Data Setup Time	t_{EDSR}	25	—	ns
E7	Read Data Hold Time	t_{EDHR}	5	—	ns
E8	ECLK Low to Data High Impedance	t_{EDHZ}	—	48	ns
E9	$\overline{CS}$ Negated to Data Hold (Read)	t_{ECDH}	0	—	ns
E10	$\overline{CS}$ Negated to Data High Impedance	t_{ECDZ}	—	1	t_{cyc}
E11	ECLK Low to Data Valid (Write)	t_{EDDW}	—	2	t_{cyc}
E12	ECLK Low to Data Hold (Write)	t_{EDHW}	10	—	ns
E13	Address Access Time (Read)[3]	t_{EACC}	308	—	ns
E14	Chip Select Access Time (Read)[4]	t_{EACS}	236	—	ns
E15	Address Setup Time	t_{EAS}	1/2	—	t_{cyc}

NOTES:

1. All AC timing is shown with respect to 20% V_{DD} and 70% V_{DD} levels unless otherwise noted.
2. When the previous bus cycle is not an ECLK cycle, the address may be valid before ECLK goes low.
3. Address access time = $t_{Ecyc} - t_{EAD} - t_{EDSR}$.
4. Chip select access time = $t_{Ecyc} - t_{ECSD} - t_{EDSR}$.

APPENDIX C DEVELOPMENT SUPPORT

This section serves as a brief reference to Motorola development tools for MC68336 and MC68376 microcontrollers.

Information provided is complete as of the time of publication, but new systems and software are continually being developed. In addition, there is a growing number of third-party tools available. The Motorola *Microcontroller Development Tools Directory* (MCUDEVTLDIR/D Revision. 3) provides an up-to-date list of development tools. Contact your Motorola representative for further information.

C.1 M68MMDS1632 Modular Development System

The M68MMDS1632 Motorola Modular Development System (MMDS) is a development tool for evaluating M68HC16 and M68300 MCU-based systems. The MMDS1632 is an emulator, bus state analyzer, and control station for debugging hardware and software. A separately purchased MPB completes MMDS functionality with regard to a particular MCU or MCU family. The many MPBs available let your MMDS emulate a variety of different MCUs. Contact your Motorola sales representative, who will assist you in selecting and configuring the modular system that fits your needs. A full-featured development system, the MMDS provides both in-circuit emulation and bus analysis capabilities, including:

- Real-time in-circuit emulation at maximum speed of 20 MHz
- Built-in emulation memory
  - 1-Mbyte main emulation memory (three-clock bus cycle)
  - 256-Kbyte fast termination (two-clock bus cycle)
  - 4-Kbyte dual-port emulation memory (three-clock bus cycle)
- Real-time bus analysis
  - Instruction disassembly
  - State-machine-controlled triggering
- Four hardware breakpoints, bitwise masking
- Analog/digital emulation
- Synchronized signal output
- Built-in AC power supply, 90–264 V, 50–60 Hz, FCC and EC EMI compliant
- RS-232 connection to host capable of communicating at 1200, 2400, 4800, 9600, 19200, 38400, or 57600 baud

C.2 M68MEVB1632 Modular Evaluation Board

The M68MEVB1632 Modular Evaluation Board (MEVB) is a development tool for evaluating M68HC16 and M68300 MCU-based systems. The MEVB consists of the M68MPFB1632 modular platform board, an MCU personality board (MPB), an in-circuit debugger (ICD16 or ICD32), and development software. MEVB features include:

D.2.6 Port E Data Register

PORTE0 — Port E0 Data Register **$YFFA11**
PORTE1 — Port E1 Data Register **$YFFA13**

15	8	7	6	5	4	3	2	1	0
NOT USED		PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
RESET:									
		U	U	U	U	U	U	U	U

PORTE is an internal data latch that can be accessed at two locations. It can be read or written at any time. If a port E I/O pin is configured as an output, the corresponding bit value is driven out on the pin. When a pin is configured as an output, a read of PORTE returns the latched bit value; when a pin is configured as an input, a read returns the pin logic level.

D.2.7 Port E Data Direction Register

DDRE — Port E Data Direction Register **$YFFA15**

15	8	7	6	5	4	3	2	1	0
NOT USED		DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0
RESET:									
		0	0	0	0	0	0	0	0

Bits in this register control the direction of the port E pin drivers when pins are configured for I/O. Setting a bit configures the corresponding pin as an output; clearing a bit configures the corresponding pin as an input. This register can be read or written at any time.

D.2.8 Port E Pin Assignment Register

PEPAR — Port E Pin Assignment **$YFFA17**

15	8	7	6	5	4	3	2	1	0
NOT USED		PEPA7	PEPA6	PEPA5	PEPA4	PEPA3	PEPA2	PEPA1	PEPA0
RESET:									
		DATA8	DATA8	DATA8	DATA8	DATA8	DATA8	DATA8	DATA8

Bits in this register determine the function of port E pins. Setting a bit assigns the corresponding pin to a bus control signal; clearing a bit assigns the pin to I/O port E. Refer to **Table D-5**.

To reset the software watchdog:

1. Write $55 to SWSR.
2. Write $AA to SWSR.

Both writes must occur in the order specified before the software watchdog times out, but any number of instructions can occur between the two writes.

D.2.16 Port C Data Register

PORTC — Port C Data Register **$YFFA41**

15	8	7	6	5	4	3	2	1	0
NOT USED		0	PC6	PC5	PC4	PC3	PC2	PC1	PC0
RESET:									
		0	1	1	1	1	1	1	1

PORTC latches data for chip-select pins configured as discrete outputs.

D.2.17 Chip-Select Pin Assignment Registers

CSPAR0 — Chip-Select Pin Assignment Register 0 **$YFFA44**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	CS5PA[1:0]		CS4PA[1:0]		CS3PA[1:0]		CS2PA[1:0]		CS1PA[1:0]		CS0PA[1:0]		CSBTPA[1:0]	
RESET:															
0	0	DATA2	1	DATA2	1	DATA2	1	DATA1	1	DATA1	1	DATA1	1	1	DATA0

The chip-select pin assignment registers configure the chip-select pins for use as discrete I/O, an alternate function, or as an 8-bit or 16-bit chip-select. Each 2-bit field in CSPAR[0:1] (except for CSBTPA[1:0]) has the possible encoding shown in **Table D-9**.

Table D-9 Pin Assignment Field Encoding

CSxPA[1:0]	Description
00	Discrete output[1]
01	Alternate function[1]
10	Chip-select (8-bit port)
11	Chip-select (16-bit port)

NOTES:
1. Does not apply to the $\overline{\text{CSBOOT}}$ field.

CSPAR0 contains seven 2-bit fields that determine the function of corresponding chip-select pins. Bits [15:14] are not used. These bits always read zero; writes have no effect. CSPAR0 bit 1 always reads one; writes to CSPAR0 bit 1 have no effect. The alternate functions can be enabled by data bus mode selection during reset.

Table D-10 shows CSPAR0 pin assignments.

This field only affects the response of chip-selects and does not affect interrupt recognition by the CPU32.

$\overline{\text{AVEC}}$ — Autovector Enable

This field selects one of two methods of acquiring an interrupt vector during an interrupt acknowledge cycle. It is not usually used with a chip-select pin.

0 = External interrupt vector enabled
1 = Autovector enabled

If the chip select is configured to trigger on an interrupt acknowledge cycle (SPACE[1:0] = %00) and the $\overline{\text{AVEC}}$ field is set to one, the chip-select automatically generates $\overline{\text{AVEC}}$ in response to the interrupt acknowledge cycle. Otherwise, the vector must be supplied by the requesting device.

D.2.22 Master Shift Registers

TSTMSRA — Master Shift Register A **$YFFA30**

Used for factory test only.

TSTMSRB — Master Shift Register B **$YFFA32**

Used for factory test only.

D.2.23 Test Module Shift Count Register

TSTSC — Test Module Shift Count **$YFFA34**

Used for factory test only.

D.2.24 Test Module Repetition Count Register

TSTRC — Test Module Repetition Count **$YFFA36**

Used for factory test only.

D.2.25 Test Submodule Control Register

CREG — Test Submodule Control Register **$YFFA38**

Used for factory test only.

D.2.26 Distributed Register

DREG — Distributed Register **$YFFA3A**

Used for factory test only.

D.5 QADC Module

Table D-24 shows the QADC address map. The column labeled “Access” indicates the privilege level at which the CPU32 must be operating to access the register. A designation of “S” indicates that supervisor mode is required. A designation of “S/U” indicates that the register can be programmed for either supervisor mode access or unrestricted access.

Table D-24 QADC Address Map

Access	**Address[1]**	**15 8**	**7 0**
S	$YFF200	Module Configuration Register (QADCMCR)	
S	$YFF202	Test Register (QADCTEST)	
S	$YFF204	Interrupt Register (QADCINT)	
S/U	$YFF206	Port A Data (PORTQA)	Port B Data (PORTQB)
S/U	$YFF208	Port Data Direction Register (DDRQA)	
S/U	$YFF20A	Control Register 0 (QACR0)	
S/U	$YFF20C	Control Register 1 (QACR1)	
S/U	$YFF20E	Control Register 2 (QACR2)	
S/U	$YFF210	Status Register (QASR)	
—	$YFF212 – $YFF22E	Reserved	
S/U	$YFF230 – $YFF27E	Conversion Command Word (CCW) Table	
—	$YFF280 – $YFF2AE	Reserved	
S/U	$YFF2B0 – $YFF2FE	Result Word Table Right Justified, Unsigned Result Register (RJURR)	
—	$YFF300 – $YFF32E	Reserved	
S/U	$YFF330 – $YFF37E	Result Word Table Left Justified, Signed Result Register (LJSRR)	
—	$YFF380 – $YFF3AE	Reserved	
S/U	$YFF3B0 – $YFF3FE	Result Word Table Left Justified, Unsigned Result Register (LJURR)	

NOTES:

[1] Y = M111, where M is the logic state of the module mapping (MM) bit in SIMCR.

D.5.1 QADC Module Configuration Register

QADCMCR — Module Configuration Register **$YFF200**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STOP	FRZ	NOT USED						SUPV	NOT USED			IARB[3:0]			
RESET:															
0	0							1				0	0	0	0

STOP — Low-Power Stop Mode Enable

When the STOP bit is set, the clock signal to the QADC is disabled, effectively turning off the analog circuitry.

0 = Enable QADC clock.
1 = Disable QADC clock.

To clear an interrupt flag, first read the flag as a one, and then write it as a zero. Should a new flag setting event occur between the time that the CPU32 reads the flag as a one and writes the flag as a zero, the flag will not be cleared. This register can be written to zeros only.

D.10.15 Error Counters

RXECTR — Receive Error Counter **$YFF0A6**
TXECTR — Transmit Error Counter **$YFF0A7**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXECTR								TXECTR							
RESET:															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Both counters are read only, except when the TouCAN is in test or debug mode.