UMC

UM82C381
System Controller

T-52-33-15

I General Description

The UM82C381 is a system controller. It provides bus control signals to PC/AT and 32-bit memory expansion bus. It also provides synchronized reset for CPU and peripherals, refresh control, math-coprocessor (80287/80387/3167) interface, address decoding logic CLK2, CLK* and timer clock generation.

II Features

- CPU interface and bus control
- PC/AT expansion bus interface
- Clock generation
- Numerical processor 80287/80387 interface
- Peripheral chips interface
- Refresh and DMA logic
- Reset and shut down logic
- Advanced 1.2μM CMOS technology
- TTL compatible input
- 120 pin flat package

III Block Diagram

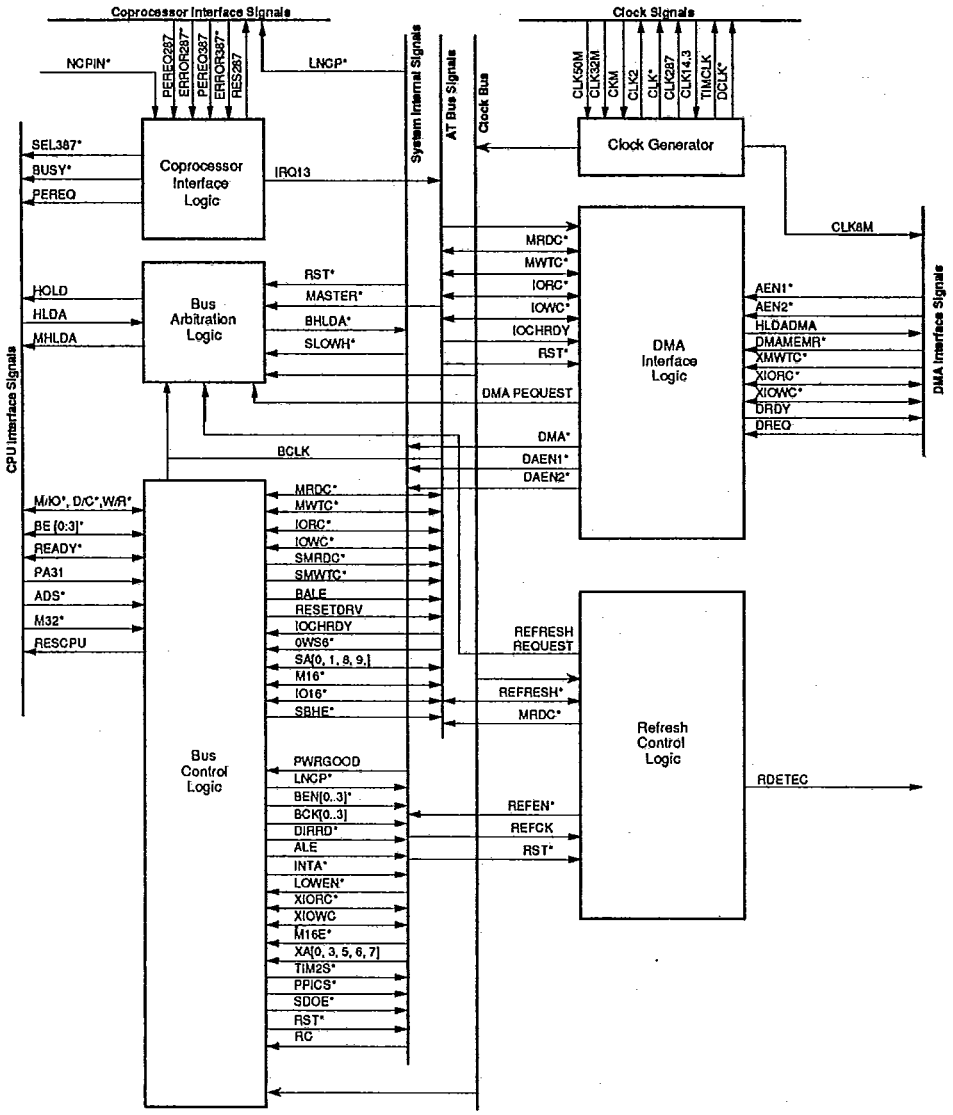

Figure 1. Block Diagram

PC Mainboard

IV Pin Configuration

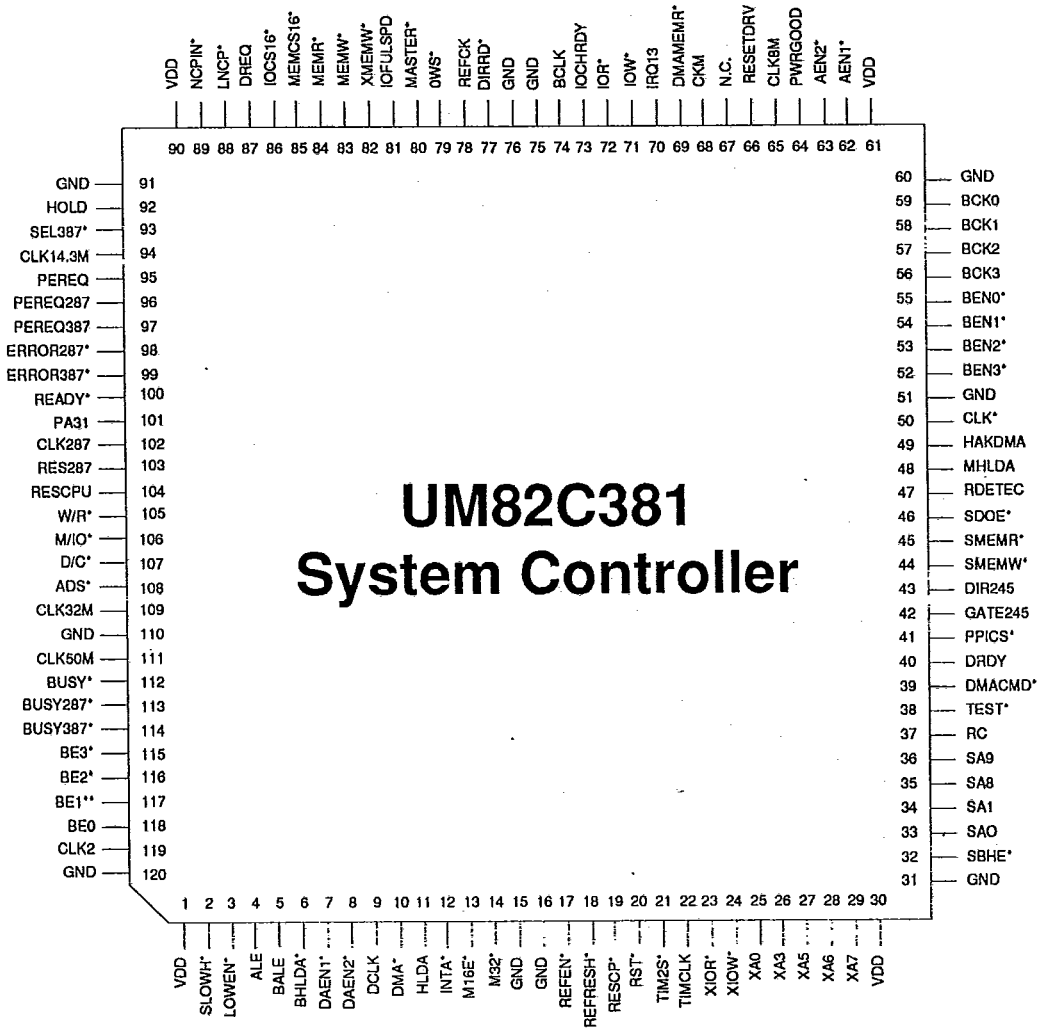

Figure 2. Pin Configuration

T-52-33-15

VI Functional Description

The function of UM82C381 can be divided into six sub-modules (see block diagram of Figure 1)

- Clock generator
- Bus control logic
- DMA interface logic
- Refresh logic
- Bus arbitration logic
- Numerical processor interface logic

6.1 Clock Generator

The clock generator may operate in both synchronous and asynchronous modes. It has two clock inputs - CK50M and CK32M, both of which are driven from a TTL crystal oscillator.

(1) When IOFULSPD is high, it operates in synchronous mode

CLK2 = CK50M
CLK* = CK50M/2
BLCK = CK50M/4

(2) When IOFULSPD is low, it operates in asynchronous mode

a. During CPU local memory cycle

CLK2 = CK50M
CLK* = CK50M/2
BCLK = CK50M/4

b. During CPU AT BUS cycle or DMA cycle

CLK2 = CK32M
CLK* = CK32M/2
BCLK = CK32M/4

If the target of system clock is 16MHz, IOFULSPD can be pulled high and CK50M is driven by 32MHz clock. In this case, CK32M does not matter for any operation. If the target of system clock is more than 16 MHz, IOFULSPD must be pulled low, CK50M is driven by double frequency of system clock and CK32M is driven by 32MHz clock in order to provide 8 MHz AT bus timing during non-local bus operation.

6.2 Bus Control Logic

a. Reset operation

The UM82C381 provides two reset inputs, PWRGOOD and RC. PWRGOOD is the power good signal from the power supply. When PWRGOOD is inactive (low), UM82C381 asserts RESCPU and RESETDRV for a system reset. RC is generated from the keyboard controller when a "warm reset" is required. The warm reset activates RESCPU to reset the 80386 CPU. RESCPU is also activated by the UM82C381 after a shutdown condition is detected, and held high for at least 16 CLK2 cycles and then deasserted. RESCPU is synchronous with CLK2, ensuring proper CPU operation. RESCP* is used to disable RTC during CPU reset period.

b. Bus status definitions

During CPU memory cycle (ADS* low and M/IO* high), if M32* is sampled low, it means current cycle is CPU local memory cycle, otherwise, it is an AT bus cycle.

The bus control logic gains control when M32* is detected high or M/IO* is low under ADS* active. It will perform the necessary synchronization of control and status signals between the AT bus and the processor. The UM82C381 supports 8 and 16 bit transfers between the processor and 8 or 16 bit memory or I/O devices located on the AT bus.

The data conversion definitions between CPU status signals and AT 8- and 16-bit bus are shown in Table 1.

T-52-33-15

IO16*/ M16*	BE3*	BE2*	BE1*	BE0*	SA1	SA0	BHE*	Cycle Type
X	1	1	1	0	0	0	1	Low byte on 8 or 16 bit bus
1	1	1	0	1	0	1	0	High byte on 8 bit bus
0	1	1	0	1	0	1	0	High byte on 16 bit bus
X	1	0	1	1	1	0	1	Low byte on 8 or 16 bit bus
1	0	1	1	1	1	1	0	High byte on 8 bit bus
0	0	1	1	1	1	1	0	High byte on 16 bit bus
0	1	1	0	0	0	0	0	Word on 16 bit bus
1	1	1	0	0	0	0	0	Word on 8 bit bus
					0	1	0	
0	1	0	0	1	1	0	1	2 bytes split on 16 bit bus
1	1	0	0	1	0	1	0	2 bytes split on 8 bit bus
					1	0	1	
0	0	0	1	1	1	0	0	Word on 16 bit bus
1	0	0	1	1	1	0	0	Word on 8 bit bus
					1	1	0	
0	1	0	0	0	0	0	0	3 bytes on16-bit bus
					1	0	1	
1	1	0	0	0	0	0	0	3 bytes on 8 bit bus
					0	1	0	
					1	0	1	
0	0	0	0	1	0	1	0	3 bytes on 16 bit bus
					1	0	0	
1	0	0	0	1	0	1	0	3 bytes on 8 bit bus
					1	0	0	
					1	1	0	
0	0	0	0	0	0	0	0	4 bytes on 16 bit bus
					1	0	0	
1	0	0	0	0	0	0	0	4 bytes on 8 bit bus
					0	1	0	
					1	0	0	
					1	1	0	

Notes: 1. I/O16* signal during I/O cycle
2. M16* signal during memory cycle
3. X means that the value of this signal doesn't matter for that cycle

Table 1. Data conversion definitions

c.Bus timing

The UM82C381 provides control signals to emulate the 8MHz AT bus signals. The number of AT bus cycles for CPU operation is shown in Table 2. For slow I/O devices, IOCHRDY is used to extend command cycle.

Operation	Bus Size	Number of Cycles Required	
		0 W.S.	Normal W.S.
memory read	16 bit	4	6
memory read	8 bit	6	12
memory write	16 bit	4	6
memory write	8 bit	6	12
I/O read	16 bit	6	6
I/O read	8 bit	6	12
I/O write	16 bit	6	6
I/O write	8 bit	6	12

Note: A normal wait state occurs if the device being accessed does not specify the number of wait states that should be generated by the system.

Table 2. Number of AT Bus Cycles

6.3 DMA Interface Logic

Essentially, DMA operation is controlled by UM82C206. During DMA cycle, UM82C381 provides interface among UM82C206, local bus and AT bus. It provides command signals to AT bus and generates bus cycle status signal to local bus for memory operation. The DMA operation is memory to I/O only and its bus size can be 8 or 16 bits.

The bus cycle status definitions for the non-CPU cycle are shown in Table 3:

BHDLA	M/IO	D/C	W/R	Cycle Time
1	0	0	0	Invalid cycle time
1	0	0	1	Non-CPU refresh read
1	0	1	0	Non-CPU memory write
1	0	1	1	Non-CPU memory read(*)
1	1	0	0	Invalid cycle type
1	1	0	1	Non-CPU refresh cycle
1	1	1	0	Non-CPU cycle (*)
1	1	1	1	Non-CPU, no cycle in progress

* This cycle can occur during the transition to and from memory write cycles.

Table 3. Non-CPU cycle status definitions

6.4 Refresh Logic

The refresh control logic is triggered by REFCK, witch generates the refresh control signals to the AT bus cycle. MEMR* is asserted low during a refresh cycle and the refresh address A0-A9 is provided by UM82C382 when REFEN* is active. It also provides the bus cycle status to the local bus for memory refresh operation. Its definitions during the refresh cycle are shown in Table 3.

6.5 Bus Arbitration Logic

The UM82C381 controls all bus activity and provides arbitration between the CPU, DMA, DRAM refresh and Master device and SLOWH*. SLOWH* comes from UM82C382 and is used to control the system speed. The UM82C381 generates HOLD request to the CPU and arbitrates those requests in a non-preemptive manner. The CPU relinquishes the bus by issuing HLDA. During a refresh cycle, the refresh logic has control of the bus until REFRESH* goes inactive. During a DMA cycle, the DMA controller has control of the bus until DREQ (HRQ of UM82C206) goes inactive.

6.6 Numerical Processor Interface Logic

The UM82C381 provides signals to interface between CPU and coprocessor. The coprocessor can be either UM80287 or UM80387. If NCPIN* is low, it means that UM80287 or UM80387 is installed; otherwise, no coprocessor is present.

Since 80386 samples its ERROR* input during initialization to determine the type of coprocessor present, SEL387* is low after RESCPU goes low and before execution of the first instruction, this pin can be used to seperate 80387 or 80287 present.

If 80387 is present, 82C381 handles BUSY387*, PEREQ387* and ERROR387* signals from the 80387 to the CPU and generates IRQ13 for error handling. During executing a task, 80387 issues a BUSY387* signal to the 82C381. Under normal operation, it is passed out to the CPU as BUSY*. If during this busy period, a coprocessor error occurs, ERROR387* input to the 82C381 becomes active resulting in latching of the BUSY* output and assertion of IRQ13. BUSY* stays active until cleared by I/O write cycle to address 0F0(H).

If 80287 is present, 82C381 provides the decoding required for selecting and resetting the 80287, handling BUSY287*, PEREQ287 and ERROR287* signals from the 80287 to the CPU and generates IRQ13 for error handling. The 80287 chip select LNCP* is active for I/O address OF8(H)-OFF(H) and PA31 is high. During execution of a task, 80287 issues a BUSY287* signal to the 82C381. Under normal operation , it is passed out to the CPU as BUSY*. If during this busy period, a coprocessor error occurs, ERROR287* input to the 82C381 becomes active, resulting in latching of the BUSY* output and assertion of IRQ13. BUSY* stays active until cleared by I/O write cycle to address 0F0(H) or 0F1(H).

The clock signal to 80287 is also provided by 82C381. If CKM is low, CLK287 is a 16 MHz clock signal with 50% duty cycle and 80287 is running at 5.4MHz. If CKM is high, CLK287 is a 10.67MHz clock signal with 33% duty cycle and 80287 is running at 10.67 MHz.