ST-2900 CPU

Assembly Instructions
and
User's Manual

Sardis Technologies
2261 East 11th Avenue
Vancouver, B.C.
Canada V5N 1Z7

Although much effort has been made to ensure the accuracy of the hardware, software and documentation, David C. Wiens and Sardis Technologies disclaim any and all liability for consequential damages, economic loss, or any other injury arising from or on account of the use of, possession of, defect in, or failure of the supplied material.

This manual last revised September 7, 1984. (and May 14, 1986)

ST-2900 CPU Board Assembly Instructions

[] If you purchased a partially assembled board, skip this section and proceed to the section "Getting Started ...".

[] Read Appendix A and decide on which configuration you want.

[] Use a drill to enlarge the four mounting holes, if necessary. If you are also building the FDC board, drill both boards together to ensure the holes will line up.

[] Carefully check both sides of the board visually for any incomplete etching, foil "bridges" or "breaks".

[] Note -- whenever the component markings on the board are not entirely clear, refer to the component layout elsewhere in this manual.

[] Use an ohmmeter between pins 1 and 4 of connector P3 to verify that there are no shorts between ground and +5vdc traces.

[] Install and solder all IC sockets. If they have a notch or indentation at one end, orient them to match the "half circle" marking at the end of each IC outline on the board. Note, that if you are using a 24 pin socket for U17 to "lower justify" it in the 28 holes provided, ie. leave the holes in the PCB for pins 1, 2,27,28 open, and the "half circle" exposed.

[] Install and solder all resistors, including RP1 and RP2.

[] Install and solder all capacitors. Be especially careful with C16,C17,C18, C20,C21 -- the "+" on the PC board indicates the side where the positive lead should go. A tantalum capacitor installed backwards can explode. Note that C20 and C21 each have 3 holes provided instead of only 2 -- this gives you a choice of .100" or .200" lead spacing. One lead must be inserted into the hole with the "+" beside it; the other lead can be inserted into either of the two other holes. C17 is not marked on the board, but is between C16 and C18.

[] Install and solder connectors P1 and P2 and P4. If they are a tight fit, use a small screwdriver (or wirewrapping tool over the posts) to force them in. Be careful -- the posts of some connectors have very sharp points.

[] If you will be using male header strips for the 3 wiring blocks (see Appendix E), install and solder them now.

[] Install and solder the 3 pin header strips for jumpers J1 and J6.

[] Install and solder the 2 pin jumpers J2,J3,J5. If you don't anticipate changing these settings, just solder wire jumpers across.

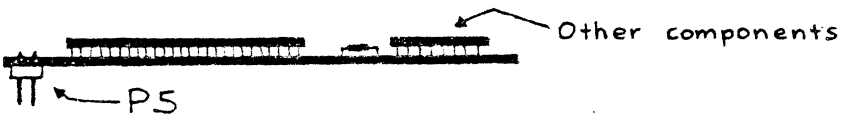
[] Install and solder test points TP1 and TP2. These can be made by bending a length of bare wire into a loop:

[] If you will not be using crystal Y1, see the instructions in Appendix A. Otherwise solder a jumper wire across J4, and another one at J7 from the center hole to the "I" hole.

[] Install and solder the power connector P3. If the connector you are using has square pins, you might have to round the inside corners of pins 1 and 4 so they fit into the holes.

[] Install and solder all diodes. Note that the banded ends should be oriented to match the arrow-head and bar symbol on the diode outlines on the PC board. In other words, the banded ends of the diodes should be closest to C18. Using a heatsink on the diode leads while soldering reduces the chance of damage.

[] Install and solder connector P5. NOTE -- this connector is inserted into the back side of the board and soldered on the top side of the board. With so many pins, it might take quite a bit of wiggling to force the connector into place.



[] Install and solder the crystals. To be on the safe side, use a piece of electrician's tape to insulate the crystal case from the PC board. Be careful not to overheat the crystals while soldering.

[] Double check your work, especially looking for missed or improperly soldered joints or accidental solder bridges.

[] You can now proceed to the section entitled "Getting Started -- Partially Assembled CPU Board".

Getting Started -- Partially Assembled CPU Board

[] NOTE -- static electricity can damage MOS integrated circuits. Even a static electricity charge too weak to feel can cause problems, and the damage, if it merely weakened the chip, might not cause the chip to fail until weeks or months later. While handling any MOS IC's you should ground your body and all tools that will touch the IC leads. Use a 1 Megohm resistor in series between you and ground to protect yourself against dangerous shocks. Also, handle the assembled board with care -- just because the IC's have been installed in their sockets doesn't make them immune to static electricity. Better safe than sorry.

[] Make up a cable and connector from your power supply to connector P3. Note that pin 1 of P3 is the one closest to connector P4. If the connectors you are using don't physically prevent you from plugging the socket into P3 backwards or one pin "off", make sure to mark pin 1 of both plug and socket.

[] At this point all IC's should be removed from the board. Connect the CPU board to the power supply and turn it on. Measure the supply voltages at several points.

Eg. U1 pin 1 should be between -10.8 vdc and -13.2 vdc
U1 pin 14 " " " +11.4 vdc and +12.6 vdc
U15 pin 14 " " " +4.75 vdc and +5.25 vdc

[] Turn the power supply off and disconnect from CPU board, then short pins 2, 3, and 4 to pin 1 at connector P3 to discharge the capacitors.

[] If you will be running the FLEX or STAR-DOS Level I operating systems, use a jumper wire to connect the "IP2" pin to the "15.625 KHz" pin at the main wiring block (see Appendix E).

[] Put shorting blocks on the jumpers as necessary for your particular configuration. See Appendix B for a description of all jumpers.

[] Install all IC's (but NOT the 8 RAM chips, U3-U10) into their sockets. Be sure you plug them in the right way. Double check against the component layout diagram. If you are inserting a 2716 or 2732 into a 28 pin socket at U17, the IC should be "lower justified", ie. leave pins 1,2,27,28 of the socket open.

[] Make up the cable to connect P1 to the terminal (see Appendix D), and hook it up.

[] Hook up the reset switch. The two pins of connector P4 sould go to a SPST momentary contact (normally open) switch.

[] Follow the instructions in Appendix C of the ST-MON 2.04 manual regarding configuring the serial port and terminal for data bits and parity.

[] Connect the CPU board to the power supply, but don't turn it on yet.

[] Turn on the CRT terminal and wait until the cursor is visible.

[] Now power up the CPU board. After about 2 seconds you should see "M2CW?" displayed on the terminal. Turn off the CPU power supply and the terminal. If you didn't see the "M2CW?" message, follow the instructions in the "Trouble Shooting Hints" section until you get that message. Don't go onto the next step until this step works.

[] Now install the 8 RAM chips (U3-U10) into their sockets, making sure to plug them in the right way.

[] Power up the terminal and then the CPU board as before. This time all you should see is a "CW?" message. Respond by typing the letter "C", after which you should get the ST-MON copyright message and command prompt.

[] Momentarily depress the reset switch. After approx. 3 seconds you should see the "CW?" message again. Respond with "C".

[] Before proceeding, thoroughly check out the operation of the CPU board. See the ST-MON User Manual for the various commands available.

Theory of Operation

Two integrated circuit chips are mostly responsible for the ST-2900's architecture -- the 6809E and 6883.

The 6809E microprocessor (U13) does the actual processing.

The 6883 Synchronous Address Multiplexor (SAM) (U14) takes care of all the timing/refreshing/multiplexing of the dynamic RAM as well as playing a major part (together with U18) in system address decoding. A few selected pages from Motorola's data sheet are included at the end of this manual.

Integrated circuit U17 is an EPROM of either 2K, 4K, or 8K bytes. The program stored in it gains control whenever the system is powered up or after the RESET switch is pressed.

Integrated circuits U3-U10 provide either 16K (16,384) or 64K (65,536) bytes of read/write memory. Since these chips are organized as 16Kx1 or 64Kx1, one 8 bit byte is stored with one bit in each of the 8 chips, with the high order bit in U3 and down to the low order bit in U10.

U2 is a buffer which, when the EXXX line (pin 13 of P5) is pulled low, isolates the RAM outputs from the data bus to allow a user provided device to occupy part of the RAM's normal address space.

Integrated circuit U12 implements two asynchronous serial I/O ports which are buffered to RS-232 levels by U1 and U11. The entire Signetics data sheet (except for pages 1 and 2) is included at the end of this manual.

Integrated circuit U16 divides down a 4 MHz clock signal to provide 9 various clock frequencies.

Two gates of U19, along with some resistors, capacitors, and diodes, generate three different reset signals whenever the system is powered up or the reset switch is pressed.

Trouble-shooting Hints

The first step is to double check the settings of jumpers J1-J7. Refer to the component layout diagram for the orientation of J2 and J3.

The second step is to double check and triple check that all components are in the right locations, and that they are oriented properly. Also make sure that the pins of all integrated circuits were correctly inserted into the sockets; sometimes a pin gets bent underneath the IC body.

If everything looks OK so far, carefully inspect all of the solder joints on the board, looking for missed connections, "cold" solder joints, and solder "bridges".

Use a voltmeter to measure the +5, +12, -12 volt supplies at each chip -- use the schematic as a guide. See the section "Getting Started ..." for the acceptable voltage ranges. Test points TP1 and TP2 provide a convenient connection to ground.

Double check your connections between connector P1 and your terminal. Note that the terms TxData/RxData/RTS/CTS are named from the 2681 DUART's point of view, not from the terminal's. See Appendix D. The CTSA line must be connected, and should result in a logic 0 level at U12 pin 7.

When the reset switch is held down (or while pin 1 is shorted to pin 2 at connector P4) the voltage at U19 pin 10 should be <= 0.5 vdc. When not held down, the voltage should read >= 2.7 vdc.

If you still can't get the CPU board working, you will need access to an oscilloscope.

Check clock signals "E" and "Q".

Check the signals at the crystal oscillator circuit of U12. If not OK, try different values for C11 and C13 (between 0 and 20 pF).

Check the signal at U12 pin 30 -- after you have pressed the reset button, then released it, you should see a continuous logic 1 level. However, approximately 3 seconds after releasing reset you should see a very short burst of logic zeroes and ones, representing the output of the "CW?" message.

Parts List - CPU Board

*	U1	1488 RS-232 driver
*	U2	74LS244 octal buffer
t*	U3-U10	4164 or 6665 64Kx1 dynamic RAM (128 cycle refresh) 200 nsec. (or 4517 16Kx1) (Texas Instr. TMS4164 will NOT work)
*	U11	1489 RS-232 receiver
*	U12	2681 DUART (Signetics SCN2681AC1N40) (also made by Motorola)
*	U13	6809E microprocessor
*	U14	6883 SAM (synchronous address multiplexor) (74LS783 or 74LS785)
*	U15	74ALS00 quad 2 input NAND (ALS preferred over LS)
*	U16	74LS393 dual 4 bit binary ripple counter
t*	U17	2716 single supply (or 2732 or 2764) EPROM (450 nsec)
*	U18	74LS155 dual 2 line to 4 line decoder/demultiplexor
*	U19	74LS14 hex inverter Schmitt trigger
t	Y1	3.6864 MHz crystal (HC-18 package)
t	Y2	16 MHz crystal " " "
*	D1,D3	1N4153 or 1N4148 diodes
*	D2	1N4153 diode (1N4148 usually works, too)
	R1-R5	4.7K ohm 1/4 watt resistors
	R6-R8	33 ohm " " "
	R9	4.7K ohm " " "
	R10,R11	100K ohm " " "
	R12	1.5K ohm " " "
	RP1,RP2	33 ohm resistor network, 8 pin SIP, 4 isolated res.
	C1,C2	.1uF or .15uF bypass capacitors
	C3-C6	470pF capacitors
	C7-C10	.1uF or .15uF bypass capacitors
t	C11	5pF capacitor (may be omitted)
	C12	22pF capacitor
t	C13	15pF capacitor
	C14	33 pF capacitor
	C15	.1uF or .15uF bypass capacitor
*	C16	10uF tantalum capacitor 6.3vdc
*	C17	2.2uF tantalum capacitor 16vdc
*	C18	10uF tantalum capacitor 16vdc
	C19	.1uF or .15uF bypass capacitor
*	C20	.1uF capacitor (any type OK)
*	C21	1.0uF capacitor (any type OK)
	C22,C23	.1uF or .15uF bypass capacitors
	C24-C27	.1uF or .15uF bypass capacitors
*		5 - 14 pin IC sockets
*		9 - 16 pin IC sockets
*		1 - 20 pin IC socket
t*		1 - 24 pin or 28 pin IC socket
*		3 - 40 pin IC sockets

	P1,P2	14 pin (2 row x 7) .100" center male header with .025" square posts (eg. can be cut from 72 pin strip such as AP Products 929836-01-36-R)
	P3	4 pin .156" center (.045" pin) connector (eg. Molex 09-75-1048 plug and 09-50-3061 socket plus two 15-04-0219 polarizing keys)
	P4	2 pin .100" center male header (or Molex 22-05-3021 plug with 22-01-2027 socket)
*	P5	60 pin (2 row x 30) .100" center male header
	J1,J6	3 pin .100" center male header (eg. can be cut from 36 pin strip such as AP Products 929834-01-36-R)
t	J2,J3	2 pin .100" center male header
t	J5	2 pin .100" center male header
t	--	20 pin (2 row x 10) .100" center male header (used as programming strip / wiring block)
t	--	4 pin (2 row x 2) .100" center male header
t	--	1 pin male header

* these components must be oriented as marked on PCB
t see appendix A for optional configurations

(Note - AP Products are also known as Aptronics. AMP, Molex, etc. also make similar connectors)

APPENDIX A - Optional Configurations

1) Eliminating crystal Y1

You can eliminate the 3.6864 MHz crystal if you can tolerate running the CPU at slightly less than 1 MHz. If you will be using the FDC board with double density operation, do NOT use this option.

a) substitute a 15 MHz crystal for Y2
b) omit Y1, C11, C13
c) wire J7 to the "E" position, not the "I" position
d) leave J4 open
e) add a jumper wire between the 4 MHz (has now become 3.75 MHz) wiring point and the lower pin (closest to C13) of J4. Refer to Appendix E.

Note that all the 9 clock frequencies from U16 are changed:

15.625	KHz	becomes	14.648 KHz
31.25	"	"	29.297 "
62.5	"	"	58.594 "
125.0	"	"	117.188 "
250.0	"	"	234.375 "
500.0	"	"	468.75 "
1	MHz	"	937.5 "
2	"	"	1.875 MHz
4	"	"	3.75 "

The baud rates generated by the 2681 DUART will be approx. 1.7% fast using this option, but that should cause no problems.

2) Socket for EPROM

If you will ever use a 2764 for U17, install a 28 pin socket. If only 2716's or 2732's will be used, a 24 pin socket will do.

3) 16K RAM / 64K RAM

Either 16K or 64K dynamic RAM chips can be used, but they must be 5 volt only supply types. Chips such as the 4116, which use 3 supply voltages, cannot be used. Also, 64K DRAM's that use a 256 cycle refresh (such as Texas Instruments) will not work. Note that ST-MON version 2.04 will only work with 64K.

4) J2/J3/J5/Wiring Block

You can either solder wires directly into these holes, or use header strips with jumper blocks or wire-wrap wires.

APPENDIX B -- JUMPERS

J1 - shorting center pin to "I" pin allows 2681 to generate IRQ interrupts. This is the setting used by current software.
   - shorting center pin to "F" allows 2681 to generate FIRQ
J2 - always shorted for currently supported configurations
J3 - " " " " " "
J4 - shorted when using Y1
   - open when not using Y1 (but one side connected to 3.75 MHz
J5 - always shorted for currently supported configurations
J6 - shorting center pin to "16" pin when using 2716/2516 EPROM
   - shorting center pin to "32/64" pin when using 2732 or 2764
J7 - shorting center pin to "I" pin when using Y1
   - shorting center pin to "E" pin when not using Y1

APPENDIX C -- MEMORY MAPS

After powerup or reset, but before the "CW?" prompt is answered:

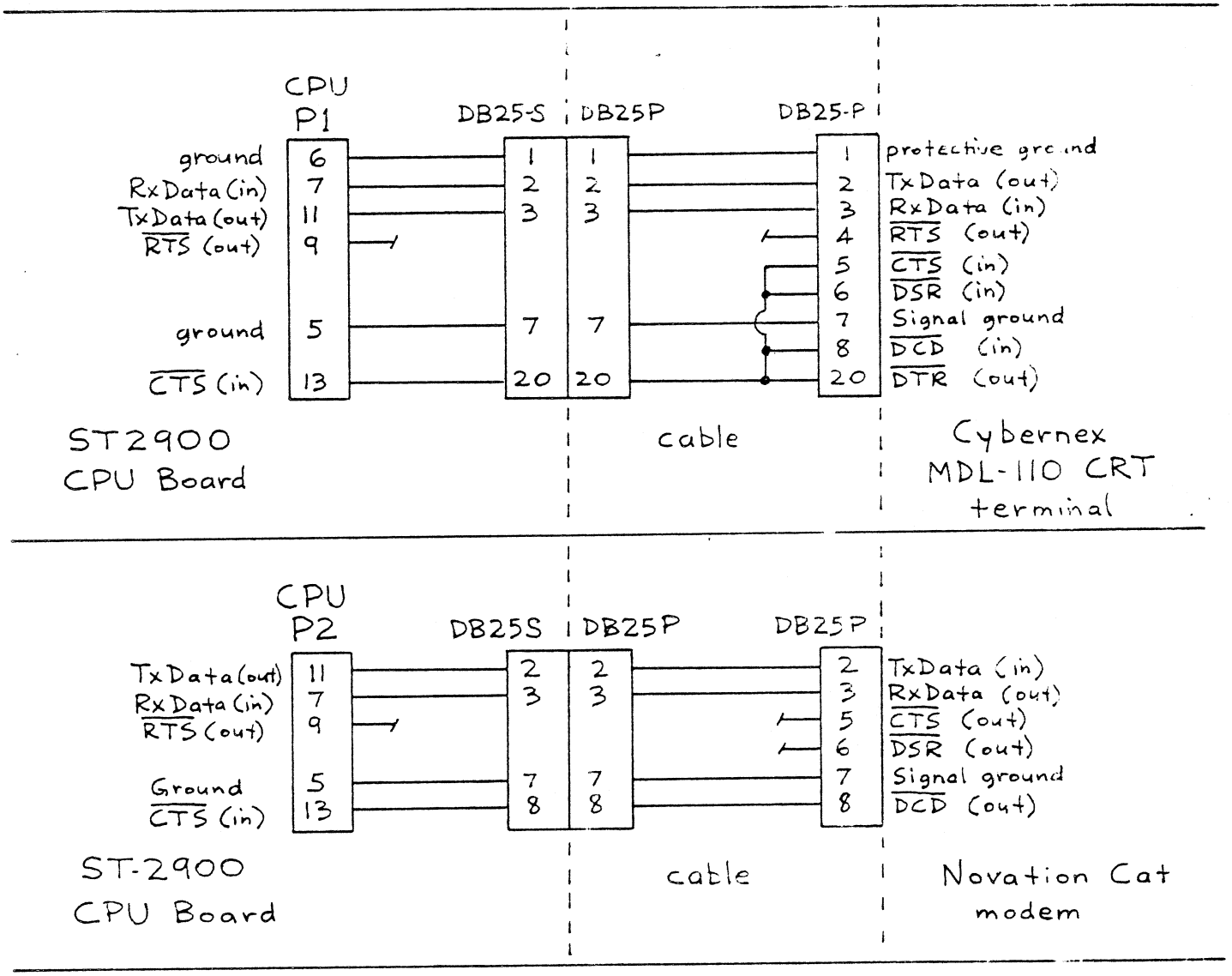
0000-7FFF	RAM
8000-9FFF	ROM 0 (off board)
A000-BFFF	on board EPROM (to A7FF if 2716, AFFF for 2732)
C000-FEFF	ROM 2 (off board)
FF00-FF1F	I/O 0 (off board)
FF20-FF3F	2681 DUART (FF30-FF3F mirrors FF20-FF2F)
FF40-FF5F	I/O 2 (off board)
FF60-FFF1	various SAM registers
FFF2-FFFF	interrupt vectors in EPROM

After the "CW?" prompt is answered:

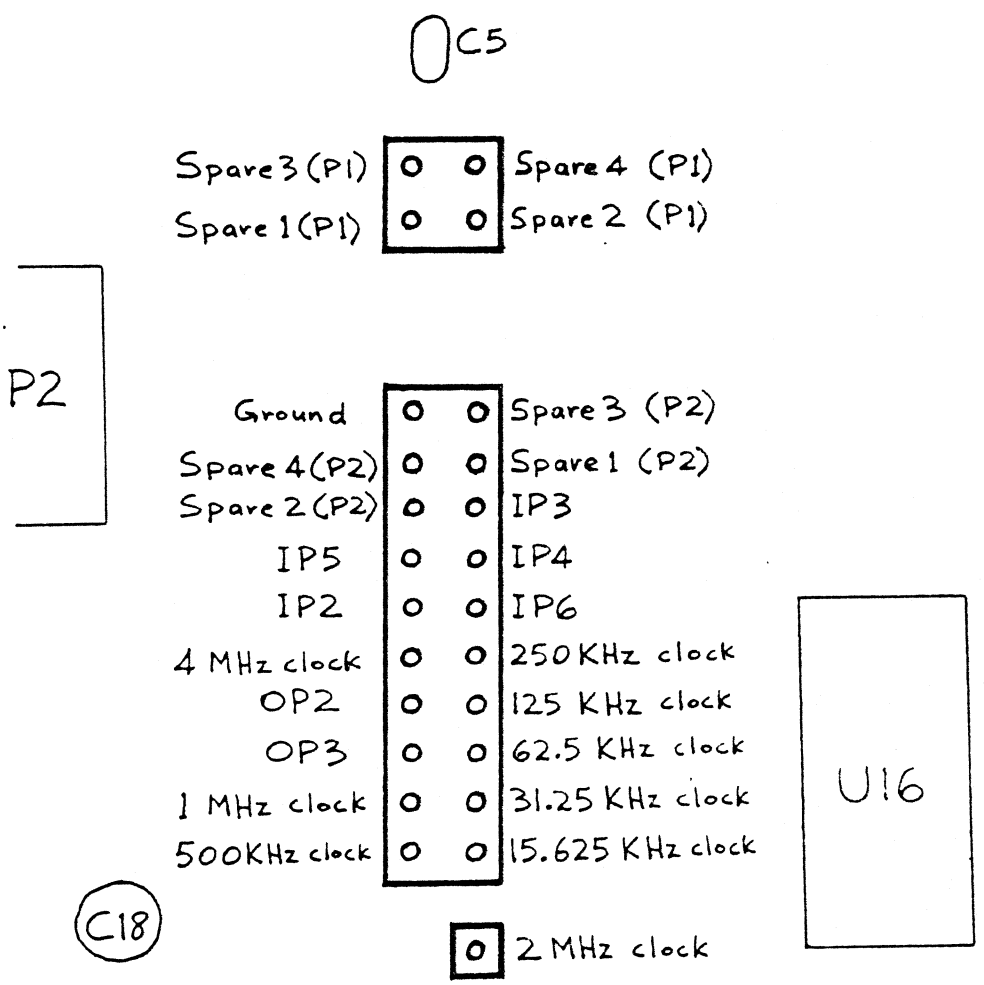
0000-FEFF	RAM (refer to ST-MON manual for details)
FF00-FF1F	I/O 0 (off board)
FF20-FF3F	2681 DUART (FF30-FF3F mirrors FF20-FF2F) (refer to the 2681 data sheet)
FF40-FF5F	I/O 2 (off board)
FF60-FFF1	various SAM registers (refer to 6883 data sheet)
FFF2-FFFF	interrupt vectors in EPROM

APPENDIX D -- RS-232 Cabling

Many CRT's only require the signals on pins 2,3,7 on the DB-25 connector. Others require handshaking. In many cases some of the signals required by the CRT are generated via jumpers on the connector at the CRT end of the cable. A few examples are provided as a general guide. Note that the current version of ST-MON and the FLEX Conversion Package configure both serial ports to require asserting CTS in order to enable that port's transmitter.



APPENDIX E - Programming Strips (Wiring Blocks)



APPENDIX F - Expansion Connector P5

Pin	In/ Out	Name	Description
1	0	Ground	
2	0	+5vdc	power supply
3	0	A14	address line from microprocessor
4	0	A12	" " " "
5	0	A10	" " " "
6	0	A8	" " " "
7	0	A6	" " " "
8	0	A4	" " " "
9	0	A2	" " " "
10	0	R/$\overline{W}$	Read/Write control signal from microprocessor
11	0	E	E clock signal from 6883
12	0	Q	Q " " " "
13	I	$\overline{EXXX}$	when pulled low, disables access to RAM/EPROM
14	I	$\overline{IRQ}$	pull line low to generate IRQ interrupt
15	I	$\overline{NMI}$	" " " " " NMI "
16	I	$\overline{FIRQ}$	" " " " " FIRQ "
17	0	$\overline{Reset}$	master reset signal
18	0	$\overline{WE}$	write enable
19	I/O	D6	data bus line
20	I/O	D4	" " "
21	I/O	D2	" " "
22	I/O	D0	" " "
23	I	IP5	from DUART
24	I	IP2	" "
25	0	OP4	" "
26	I	IP3	" "
27	0	OP5	" "
28	0	-12vdc	power supply
29	0	+5vdc	" "
30	0	Ground	
31	0	Ground	
32	0	+5vdc	power supply
33	0	A15	address line from microprocessor
34	0	A13	" " " "
35	0	A11	" " " "
36	0	A9	" " " "
37	0	A7	" " " "
38	0	A5	" " " "
39	0	A3	" " " "
40	0	A1	" " " "
41	0	A0	" " " "
42	I	$\overline{Halt}$	pull line low to halt the microprocessor
43	0	$\overline{RE}$	Read enable
44	0	$\overline{ROM\ 2}$	EPROM chip select ($C000-$FEFF)
45	0	$\overline{I/O\ 0}$	I/O chip select ($FF00-$FF1F)
46	0	$\overline{ROM\ 0}$	EPROM chip select ($8000-$9FFF)
47	0	I/O 2	I/O chip select ($FF40-$FF5F)

48	I/O	D7	data bus line
49	I/O	D5	" " "
50	I/O	D3	" " "
51	I/O	D1	" " "
52	I	IP4	from DUART
53	I	IP6	" "
54	0	OP2	" "
55	0	OP6	" "
56	0	OP3	" "
57	0	OP7	" "
58	0	+12vdc	power supply
59	0	+5vdc	" "
60	0	Ground	

APPENDIX G - Suggested Sources of Further Information

1) "6800 Software Gourmet Guide and Cookbook" by Robert Findley, published by Hayden, although not written for the 6809, has lots of useful subroutines that could be converted. Some of the chapter titles are "Conversion Routines" (incl. ASCII to/from Baudot), "Floating Point Routines", "Decimal Arithmetic Routines", and "Search and Sort Routines".

2) "The MC6809 Cookbook" by Carl. D. Warren, published by TAB, introduces the 6809 architecture and addressing modes, and includes descriptions of each machine instruction. A bonus is the complete assembly language source code of VTL-09 (Very Tiny Language for the 6809), a Tiny-BASIC-like interpreter that only occupies approx. 1K bytes!

3) "'68' Micro Journal", a monthly magazine for 6800/6809/68000 based systems, published by Computer Publishing Inc., 5900 Cassandra Smith, PO Box 849, Hixson, Tennessee, U.S.A. 37343. Current and back issues often contain source code listings of public domain programs.

4) "Microcomputing" magazine, formerly called "Kilobaud Microcomputing" or "Kilobaud", had many 6800 and 6809 articles from 1978 to 1982, especially during 1980/81.

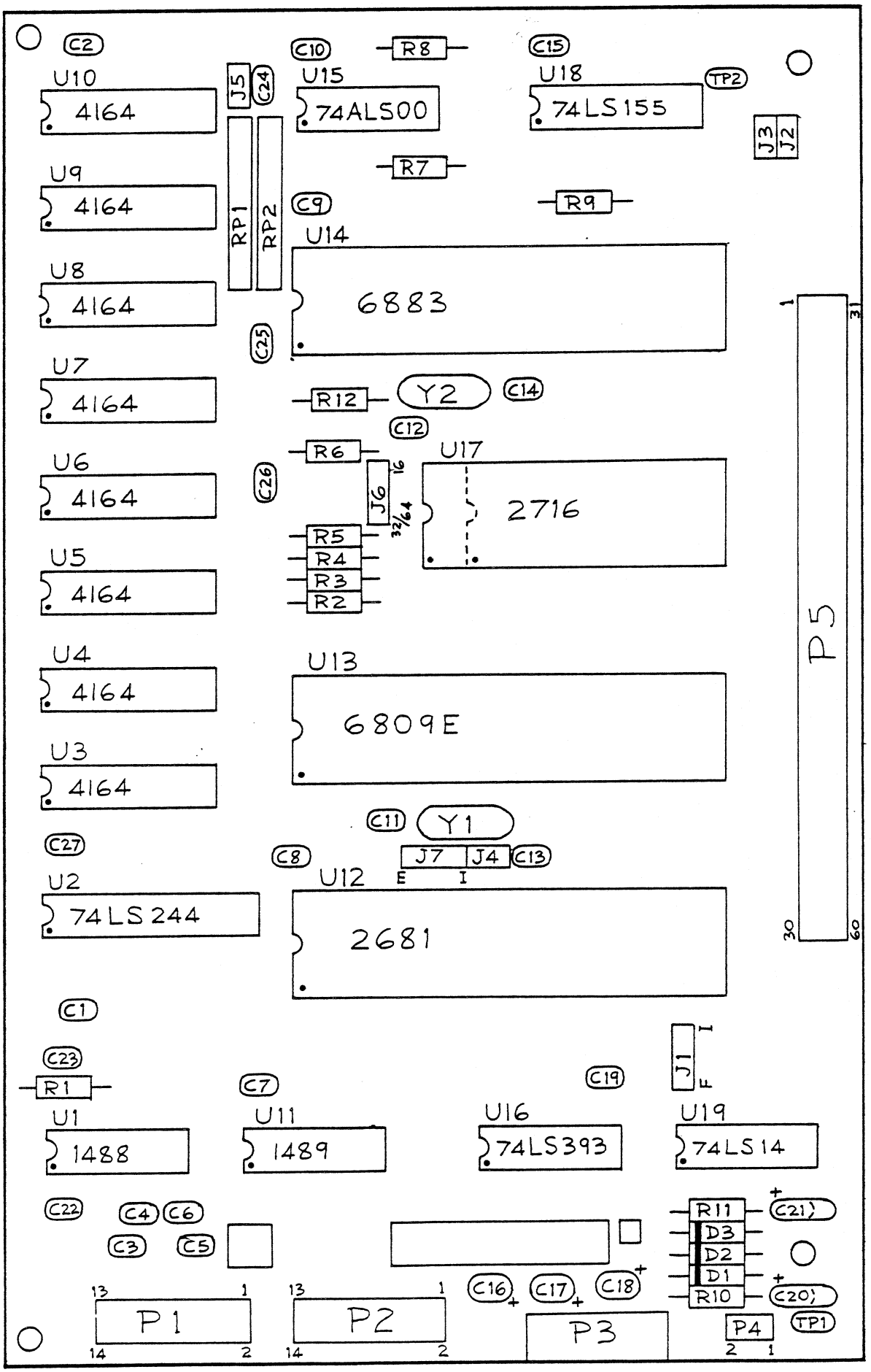
5) Mountain View Press, Inc., P.O. Box 4656, Mountain View, California, U.S.A. 94040 (see their ads in BYTE magazine) have source listings for 6809 fig-FORTH available for approximately (US) $15.

6) Several past issues of BYTE magazine have had articles dealing with the 6809. Some examples are March 1981 p.90, February 1979 p.32, and January 1979 p.14.

7) Assembler source code for a Fast Fourier Transform was published in the February '79 issue of BYTE, p.108. Although written for the 6800, it should be possible to convert to the 6809.

....

CPU - Component Layout



Revision B

CPU

Revision B
Apr. 5/84

P4 — to reset switch

R10 100K · D1 1N4148 · D3 1N4148 · D2 1N4153 · R11 100K · C20 0.1µF · C21 1.0µF · U19 74LS14 · 4.7K R5 R4 R3 R2

U13 6809E — TSC, BA, BS, LIC, A/VMA, Busy, Halt, FIRQ, IRQ, NMI, Reset, D7–D0, A15–A0, E, Q, R/W, Vcc, Vss

U17 2716 or 2732 or (2764) — Vpp/A11, A9, A8, A7–A0, D7–D0, Vcc, Vpp, Pgm, Vss, OE, A12

J6 — 32/64

U16 74LS393 — Gnd, MRa, MRb, Vcc, Clka, Clkb; 15.625KHz, 31.25KHz, 62.5KHz, 125KHz, 250KHz, 500KHz, 1MHz, 2MHz, 4MHz

P1 — Spare, +5, −12, +12, +12, Gnd, Gnd

C11 5pF · C13 15pF · Y1 3.6864 MHz · J7 · J4 · X2 · CLK/X1 · Reset · INTR

R1 4.7K

470pF C4 · C5 · C6 · C3

U1 1488 · U11 1489

P2 — TxDB, RTSB, CTSB, RxDB, Spare, +5, −12, +12, +12, Gnd, Gnd

U12 2681 — IP0 (CTSA), IP1 (CTSB), IP2–IP6, TXDA, RXDA, TXDB, RXDB, OP0 (RTSA), OP1 (RTSB), OP2–OP7, RD, WR, A3, A2, A1, A0, D0–D7, CE, Vcc, Gnd

U15 74ALS00 · U19 74LS14 · spare gate

C12 20pF · C14 33pF · Y2 16MHz · R12 1.5k

33Ω R7 · RP1, RP2 · R8 · RG

U14 6883 — VClk, Xin, Xout, Z7–Z0, WE, CAS, RAS, Vcc, Gnd, HS, DAO, S2, S1, S0, A15–A0, E, Q, R/W

U2 74LS244 — Vcc, Gnd, E1, E2

U18 74LS155 — Vcc, Gnd, Ea, Eb, A1, A0

J5 · U3 U4 U5 U6 U7 U8 U9 4517/4164/6665 — Vcc, Vss, WE, CAS, RAS

J3 · J2

R9 4.7K

P3 — −12, +12, +5, GND, Gnd

C17 2.2µF tantalum · C18 10µF tantalum · C16 10µF tantalum · TP1, TP2

P5 — Gnd, Gnd, Gnd, Gnd, +5, +5, +5, +12, −12; IP2, IP3, IP4, IP5, IP6, OP2, OP3, OP4, OP5, OP6, OP7; Reset, NMI, IRQ, Halt; D7, D6, D5, D4, D3, D2, D1, D0; ROM0, ROM2, I/O0, I/O2; A15, A14, A13, A12, A11, A10, A9, A8, A7, A6, A5, A4, A3, A2, A1, A0; E, Q, R/W, Exxx; RE, WE

Note: C1, 2, 7, 8, 9, 10, 15, 19, 22–27 are .1µF decoupling (bypass) capacitors.

MOTOROLA
SEMICONDUCTORS
P.O. BOX 20912 • PHOENIX, ARIZONA 85036

SN74LS783/ MC6883 SN74LS785

Advance Information

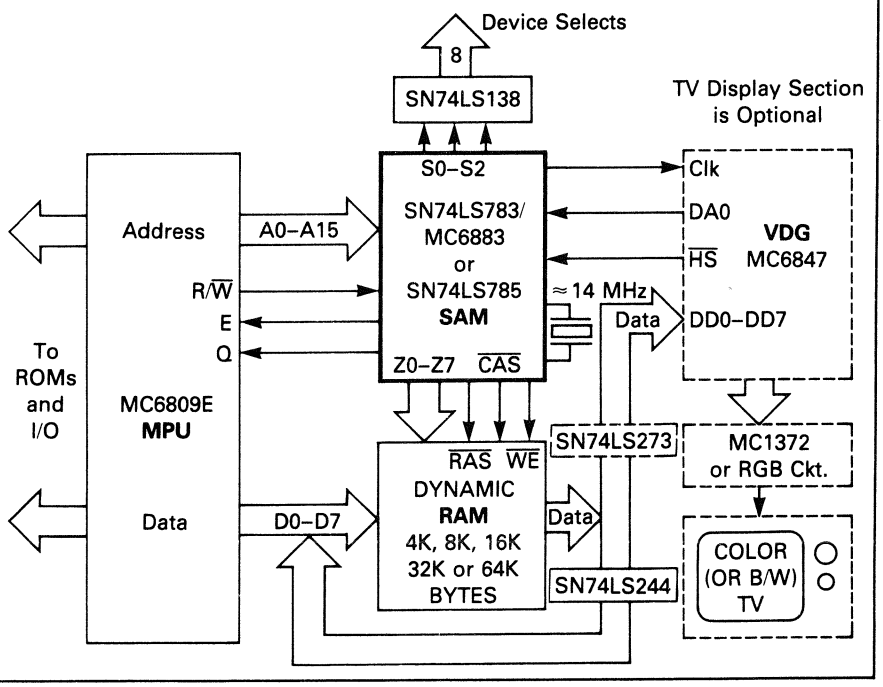
SYNCHRONOUS ADDRESS MULTIPLEXER

The SN74LS783/MC6883 and SN74LS785 bring together the MC6809E (MPU), the MC6847 (Color Video Display Generator) and dynamic RAM to form a highly effective, compact and cost effective computer and display system.

The SN74LS783/MC6883 is designed to support 4K x 1, 16K x 1 and 64K x 1 (128 column refresh) dynamic RAMs. The SN74LS785 has been modified to support the above listed products as well as 16K x 4 and 64K x 1 (256 column refresh) dynamic RAMs. A further enhancement allows the LS785 to support low power dynamic ROMs (such as MCM68364) without additional logic.

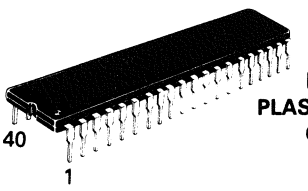
- MC6809E, MC6800, MC6801E, MC68000 and MC6847 (VDG) Compatible
- Transparent MPU/VDG/Refresh
- RAM size — 4K, 8K, 16K, 32K or 64K Bytes (Dynamic or Static)
- Addressing Range — 96K Bytes
- Single Crystal Provides All Timing
- Register Programmable:
  - VDG Addressing Modes
  - VDG Offset (0 to 64K)
  - RAM Size
  - Page Switch
  - MPU Rate (Crystal ÷ 16 or ÷ 8)
  - MPU Rate (Address Dependent or Independent)
- System "Device Selects" Decoded 'On Chip'
- Timing is Optimized for Standard Dynamic RAMs
- +5.0 V Only Operation
- Easy Synchronization of Multiple SAM Systems
- DMA Mode

TYPICAL SYSTEM BLOCK DIAGRAM

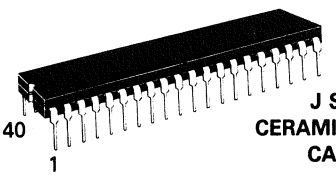


SYNCHRONOUS ADDRESS MULTIPLEXER

LOW POWER SCHOTTKY

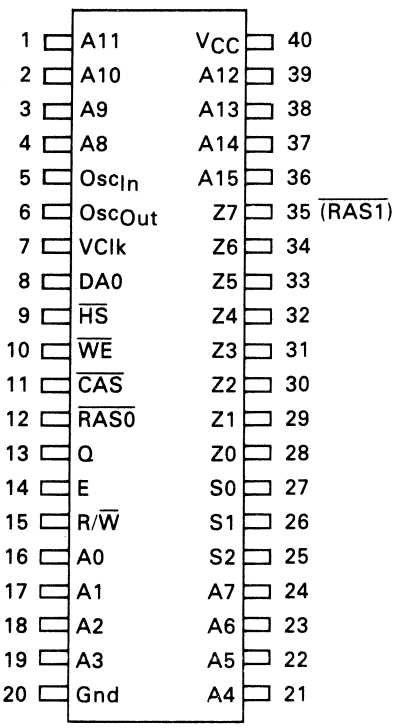


N SUFFIX
PLASTIC PACKAGE
CASE 711



J SUFFIX
CERAMIC PACKAGE
CASE 734

PIN ASSIGNMENT



This document contains information on a new product. Specifications and information herein

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	−0.5 to +7.0	Vdc
Input Voltage (Except Osc_{In})	V_I	−0.5 to 10	Vdc
Input Current (Except Osc_{In})	I_I	−30 to +5.0	mA
Output Voltage	V_O	−0.5 to +7.0	Vdc
Operating Ambient Temperature Range	T_A	0 to +70	°C
Storage Temperature Range	T_{stg}	−65 to +150	°C
Input Voltage Osc_{In}	$V_{IOsc_{In}}$	−0.5 to V_{CC}	Vdc
Input Current Osc_{In}	$I_{IOsc_{In}}$	−0.5 to +5.0	mA

GUARANTEED OPERATING RANGES

Parameter	Symbol	Min	Typ	Max	Unit
Supply Voltage	V_{CC}	4.75	5.0	5.25	V
Operating Ambient Temperature Range	T_A	0	25	75	°C
Output Current High	I_{OH}				mA
$\overline{RAS0}$, $\overline{RAS1}$, $\overline{CAS}$, $\overline{WE}$		—	—	−1.0	
All Other Outputs		—	—	−0.2	
Output Current Low	I_{OL}				mA
$\overline{RAS0}$, $\overline{RAS1}$, $\overline{CAS}$, $\overline{WE}$		—	—	8.0	
VClk		—	—	0.8	
All Other Outputs		—	—	4.0	

DC CHARACTERISTICS (Unless otherwise noted specifications apply over recommended power supply and temperature ranges.)

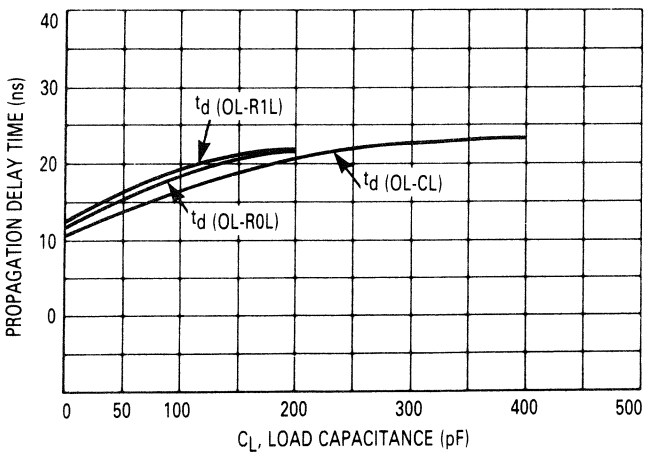
Characteristic	Symbol	Min	Typ	Max	Units
Input Voltage — High Logic State	V_{IH}	2.0	—	—	V
Input Voltage — Low Logic State	V_{IL}	—	—	0.8	V
Input Clamp Voltage (V_{CC} = Min, I_{in} = −18 mA) All Inputs Except Osc_{In}	V_{IK}	—	—	−1.5	V
Input Current — High Logic State at Max Input Voltage	I_I				µA
(V_{CC} = Max, V_{in} = 5.25 V) VClk Input		—	—	200	
(V_{CC} = Max, V_{in} = 5.25 V) DA0 Input		—	—	100	
(V_{CC} = Max, V_{in} = 5.25 V Osc_{In} = Gnd) Osc_{Out} Input		—	—	250	
(V_{CC} = Max, V_{in} = 7.0 V) All Other Inputs Except Osc_{In}		—	—	100	
Input Current High Logic State All Inputs Except VClk, (V_{CC} = Max, V_{in} = 2.7 V) DA0 Osc_{In}, Osc_{Out}	I_{IH}	—	—	20	µA
Input Current — Low Logic State	I_{IL}				mA
(V_{CC} = Max, V_{in} = 0.4 V) DA0 Input		—	—	−1.2	
(V_{CC} = Max, V_{in} = 0.4 V) VClk Input		—	−30	−60	
(V_{CC} = Max, V_{in} = 0.4 V, Osc_{In} = Gnd) Osc_{Out} Input		—	—	−8	
(V_{CC} = Max, V_{in} = 0.4 V) All Other Inputs Except Osc_{In}		—	—	−.4	
Output Voltage — High Logic State					V
(V_{CC} = Min, I_{OH} = −1.0 mA) $\overline{RAS0}$, $\overline{RAS1}$, $\overline{CAS}$, $\overline{WE}$	$V_{OH(C)}$	3.0	—	—	
(V_{CC} = Min, I_{OH} = −0.2 mA) E, Q	$V_{OH(E)}$	V_{CC} − 0.75	—	—	
(V_{CC} = Min, I_{OH} = −0.2 mA) All Other Outputs	V_{OH}	2.7	—	—	
Output Voltage — Low Logic State					V
(V_{CC} = Min, I_{OL} = 8.0 mA) $\overline{RAS0}$, $\overline{RAS1}$, $\overline{CAS}$, $\overline{WE}$	$V_{OL(C)}$	—	—	0.5	
(V_{CC} = Min, I_{OL} = 4.0 mA) E, Q Outputs	$V_{OL(E)}$	—	—	0.5	
(V_{CC} = Min, I_{OL} = 0.8 mA) VClk Output	$V_{OL(V)}$	—	—	0.6	
(V_{CC} = Min, I_{OL} = 4.0 mA) All Other Outputs	V_{OL}	—	—	0.5	
Power Supply Current	I_{CC}	—	180	230	mA
Output Short-Circuit Current	I_{OS}	30	—	225	mA

AC CHARACTERISTICS ($4.75\ V \leq V_{CC} \leq 5.25\ V$ and $0 \leq T_A \leq 70°C$, unless otherwise noted).

Characteristic	Symbol	Min	Typ	Max	Units
Propagation Delay Times					ns
(See Circuit in Figure 9) Oscillator-In ↘ to Oscillator-Out ↗	$t_{d(OL-OH)}$	—	3.0	—	
Oscillator-In ↗ to Oscillator-Out ↘	$t_{d(OH-OL)}$	—	20	—	
(C_L = 195 pF) A0 thru A15 to Z0, Z1, Z2 thru Z7	$t_{d(A-Z)}$	—	28	—	
(C_L = 30 pF) A0 thru A15, R/W to S0, S1, S3	$t_{d(A-S)}$	—	18	—	
(C_L = 95 pF) Oscillator-Out ↘ to $\overline{RAS0}$ ↗	$t_{d(OL-R0H)}$	—	20	—	
(C_L = 95 pF) Oscillator-Out ↘ to $\overline{RAS0}$ ↘	$t_{d(OL-R0L)}$	—	18	—	
(C_L = 95 pF) Oscillator-Out ↘ to $\overline{RAS1}$ ↗	$t_{d(OL-R1H)}$	—	22	—	
(C_L = 95 pF) Oscillator-Out ↘ to $\overline{RAS1}$ ↘	$t_{d(OL-R1L)}$	—	20	—	
(C_L = 195 pF) Oscillator-Out ↘ to $\overline{CAS}$ ↗	$t_{d(OL-CH)}$	—	20	—	
(C_L = 195 pF) Oscillator-Out ↘ to $\overline{CAS}$ ↘	$t_{d(OL-CL)}$	—	20	—	
(C_L = 195 pF) Oscillator-Out ↘ to $\overline{WE}$ ↗	$t_{d(OL-WH)}$	—	22	—	
(C_L = 195 pF) Oscillator-Out ↘ to $\overline{WE}$ ↘	$t_{d(OL-WL)}$	—	40	—	
(C_L = 100 pF) Oscillator-Out ↘ to E ↗	$t_{d(OL-EH)}$	—	55	—	
(C_L = 100 pF) Oscillator-Out ↘ to E ↘	$t_{d(OL-EL)}$	—	25	—	
(C_L = 100 pF) Oscillator-Out ↘ to Q ↗	$t_{d(OL-QH)}$	—	55	—	
(C_L = 100 pF) Oscillator-Out ↘ to Q ↘	$t_{d(OL-QL)}$	—	25	—	
(C_L = 30 pF) Oscillator-Out ↗ to VClk ↗	$t_{d(OH-VH)}$	—	50	—	
(C_L = 30 pF) Oscillator-Out ↗ to VClk ↘	$t_{d(OH-VL)}$	—	65	—	
(C_L = 195 pF) Oscillator-Out ↘ to Row Address	$t_{d(OL-AR)}$	—	36	—	
(C_L = 195 pF) Oscillator-Out ↘ to Column Address	$t_{d(OL-AC)}$	—	33	—	
(C_L = 15 pF) Oscillator-Out ↘ to DA0 ↗ Earliest(1)	$t_{d(OL-DH)}$	—	−15	—	
(C_L = 15 pF) Oscillator-Out ↘ to DA0 ↗ Latest(1)	$t_{d(OL-DH)}$	—	+15	—	
(C_L = 95 pF on $\overline{RAS}$, C_L = 195 pF on $\overline{CAS}$) $\overline{CAS}$ ↘ to $\overline{RAS}$ ↗	$t_{d(CL-RH)}$	—	208	—	
Setup Time for A0 thru A15, R/$\overline{W}$ Rate = ÷16	$t_{su(A)}$	—	28	—	ns
Rate = ÷ 8		—	28		
Hold Time for A0 thru A15, R/$\overline{W}$ Rate = ÷16	$t_{h(A)}$	—	30	—	ns
Rate = ÷ 8		—	30	—	
Width of $\overline{HS}$ Low 2	$t_{WL(HS)}$	2.0	5.0	6.0	µs

Notes: 1. When using the SAM with an MC6847, the rising edge of DA0 is confined within the range shown in the timing diagrams (unless the synchronizing process is incomplete.) The synchronization process requires a maximum of 32 cycles of Osc_{Out} for completion.
2. $t_{WL(HS)}$ wider than 6.0 µs may yield more than 8 sequential refresh addresses.

FIGURE 1 — PROPAGATION DELAY TIMES versus LOAD CAPACITANCE



PIN DESCRIPTION TABLE

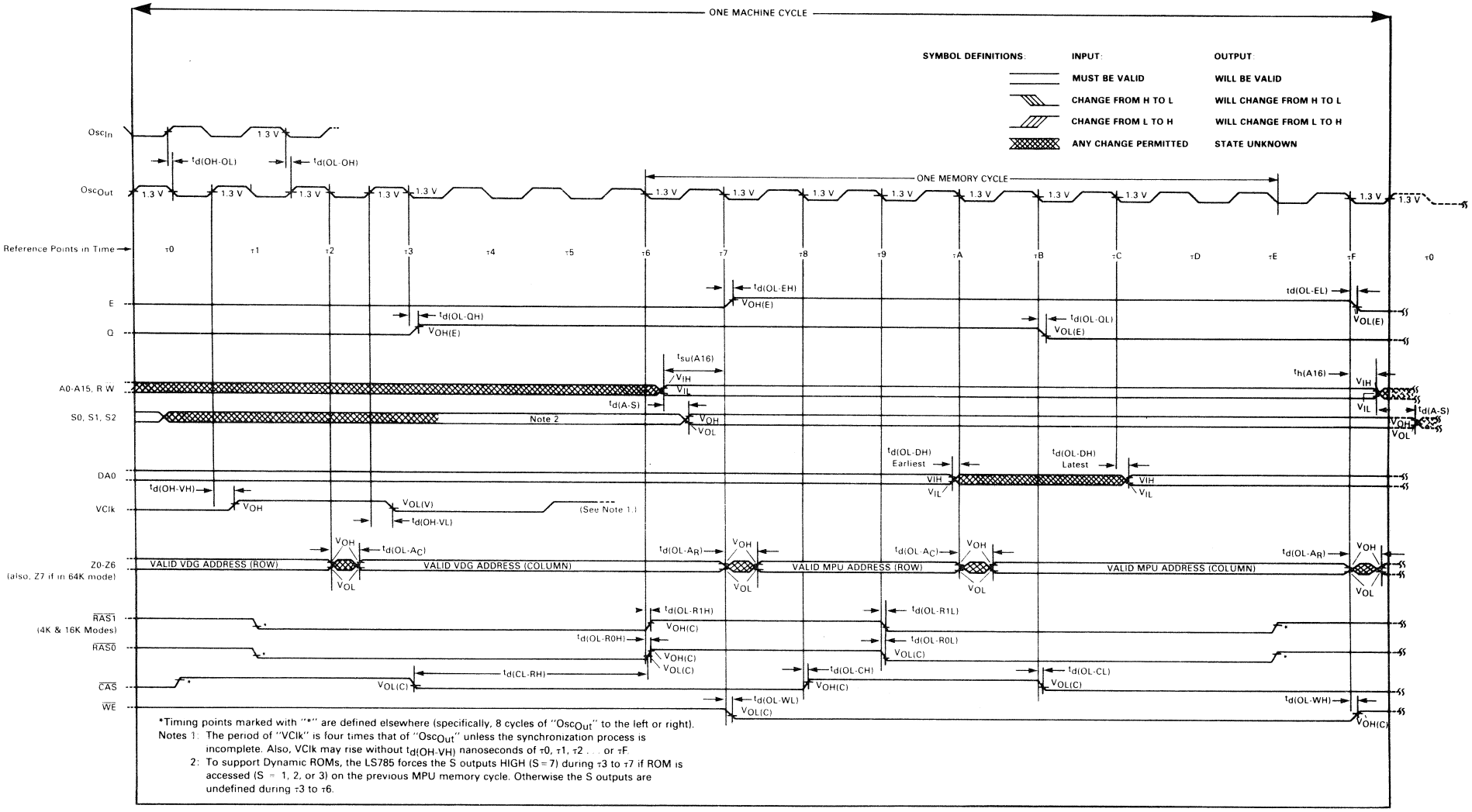
		Name	No.	Function
Input Pins	Power	V_{CC}	40	Apply + 5 volts ± 5%. SAM draws less than 230 mA.
		Gnd	20	Return Ground for +5 volts.
	MPU Address and Control	A15	36	Most Significant Bit.
		A14	37	
		A13	38	MPU address bits A0-A15. These 16 signals come directly from the MPU and are used to directly address up to 64K memory locations or to indirectly address up to 96K memory locations. (See pages 17 and 18 for memory maps). Each input is approximately equivalent to one low power Schottky load.
		A12	39	
		A11	1	
		A10	2	
		A9	3	
		A8	4	
		A7	24	
		A6	23	
		A5	22	
		A4	21	
		A3	19	
		A2	18	
		A1	17	
		A0	16	Least Significant Bit.
		R/$\overline{W}$	15	MPU READ or $\overline{WRITE}$. This signal comes directly from the MPU and is used to enable writing to the SAM control register, dynamic RAM (via $\overline{WE}$), and to enable device select #0.
	VDG Control	Osc_{In}	5	Apply 14.31818* MHz crystal and 2.5–30 pF trimmer to ground. See page 12.
		DA0	8	Display Address DA0. The primary function of this pin is to input the least significant bit of a 16-bit video display address. The more significant 15-bits are outputs from an internal 15-bit counter which is clocked by DA0. The secondary function of this pin is to indirectly input the logic level of the VDG "$\overline{FS}$" (field synchronization pulse) for vertical video address updating.
		$\overline{HS}$	9	Horizontal Synchronization. The primary function of this pin is to detect the falling edge of VDG "$\overline{HS}$" pulse in order to initiate eight dynamic RAM refresh cycles. The secondary function is to reset up to 4 least significant bits of the internal video address counter.
		VClk	7	VDG Clock. The primary function of this pin is to **output** a 3.579545 MHz square wave** to the VDG "Clk" pin. The secondary function resets the SAM when this VClk pin is pulled to logic "0" level, acting as an **input**.
Output Pins		Osc_{Out}	6	Apply 1.5 kΩ resistor to 14.31818* MHz crystal and 33 pF capacitor to ground. See page 12.
	Device Selects	S2	25	Most Significant Bit (Device Select Bits). The binary value of S2, S1, S0 selects one of eight "chunks" of MPU address space (numbers 0 through 7). Varying in length, these "chunks" provide efficient memory mapping for ROMs, RAMs, Input/Output devices, and MPU Vectors. (Requires 74LS 138-type demultiplexer).
		S1	26	
		S0	27	Least Significant Bit.
	MPU Clocks	E	14	E (Enable Clock) "E" and "Q" are 90° out of phase and are both used as MPU clocks for the MC6809E. For the MC6800 and MC6801E, only "E" is used. "E" is also used for many MC6800 peripheral chips.
		Q	13	Q (Quadrature Clock).
	RAM Address	Z7†	35	Most Significant Bit
		Z6†	34	First, the least significant address bits from the MPU or "VDG" are presented to Z0–Z5 (4K x 1 RAMs) or Z0–Z6 (16K x 1 RAMs) or Z0–Z7 (64K x 1 RAMs). Next, the most significant address bits from the MPU or "VDG" are presented to Z0 –Z5 (4K x 1 RAMs) or Z0 –Z6 (16K x 1 RAMs) or Z0 –Z7 (64K x 1 RAMs). Note that for 4K x 1 and 16K x 1 RAMs, Z7 (Pin 35) is not needed for address information. Therefore, Pin 35 is used for a second row address select which is labeled ($\overline{RAS1}$).
		Z5†	33	
		Z4†	32	
		Z3†	31	
		Z2†	30	
		Z1†	29	
		Z0†	28	Least Significant Bit.
	RAM Control	$\overline{RAS1}$†	35	Row Address Strobe One. This pulse strobes the least significant 6,7 or 8 address bits into dynamic RAMs in Bank #1.
		$\overline{RAS0}$†	12	Row Address Strobe Zero. This pulse strobes the least significant 6,7 or 8 address bits into dynamic RAMs in Bank #0.
		$\overline{CAS}$†	11	Column Address Strobe. This pulse strobes the most significant 6,7 or 8 address bits into dynamic RAMs.
		$\overline{WE}$†	10	Write Enable. When low, this pulse enables the MPU to write into dynamic RAM.

*14.31818 MHz is 4 times 3.579545 MHz television color subcarrier. Other frequencies may be used. (See page 12.)

**When VDG and SAM are not yet synchronized the "square wave" will stretch (see page 10.)

† Due to fast transitions, ferrite beads in series with these outputs may be necessary to avoid high frequency (≈ 60 MHz) resonances.

FIGURE 2 — TIMING WAVEFORMS for MPU RATE = SLOW



Timing points marked with "" are defined elsewhere (specifically, 8 cycles of "Osc_{Out}" to the left or right).

Notes 1: The period of "VClk" is four times that of "Osc_{Out}" unless the synchronization process is incomplete. Also, VClk may rise without $t_{d(OH-VH)}$ nanoseconds of τ0, τ1, τ2 . . . or τF.

2: To support Dynamic ROMs, the LS785 forces the S outputs HIGH (S = 7) during τ3 to τ7 if ROM is accessed (S = 1, 2, or 3) on the previous MPU memory cycle. Otherwise the S outputs are undefined during τ3 to τ6.

THEORY OF OPERATION

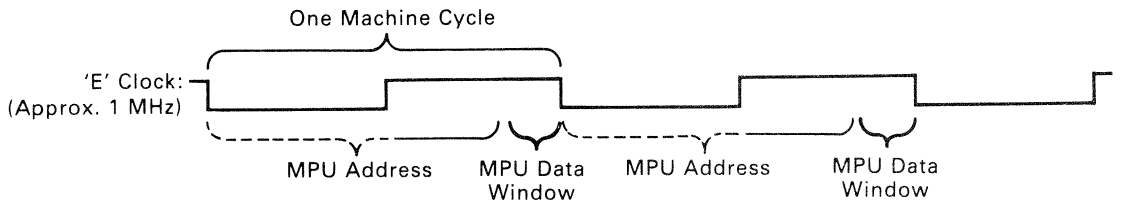
Video or No Video

Although the SAM may be used as a dynamic RAM controller **without** a video display*, most applications are likely to include a MC6847 video display generator (VDG). Therefore, this document emphasizes use of the SAM with MC6847 systems.

Shared RAM (with interleaved DMA)

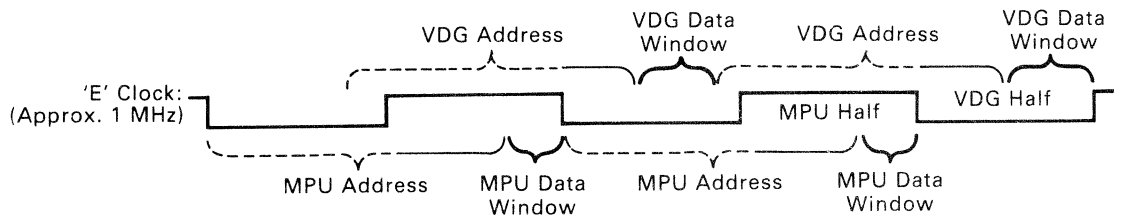
To minimize the number of RAM and interface chips, both the MPU and VDG share common dynamic RAM. Yet, the use of common RAM creates an apparent difficulty. That is, the MPU and VDG must both access the RAM without contention. This difficulty is overcome by taking advantage of the timing and architecture of Motorola MPU's (MC6800, MC6801E, MC6809E, MC68000). Specifically, **all** MPU accesses of external memory **always** occur in the **latter half** of the machine cycle, as shown below:

FIGURE 11 — MOTOROLA MPU TIMING



Similarly, the MC6847 (non-interlaced) VDG transfers a data byte in a half machine cycle (E or Φ2). Thus, when properly positioned, VDG and MPU RAM accesses interleave without contention as shown below:

FIGURE 12 — MOTOROLA MPU WITH VDG TIMING



This Interleaved Direct Memory Access (IDMA) is synchronized via the MC6883 by centering the VDG data window half-way between MPU data windows.**

The result is a shared RAM system without MPU/VDG RAM access contention, with both MPU and VDG running uninterrupted at normal operating speed, each transparent to the other.

RAM Refresh

Dynamic RAM refresh is accomplished by accessing eight*** sequential row addresses every 64*** microseconds until all addresses have been accessed. To avoid RAM access contention between REFRESH and MPU, each of the refresh accesses occupies the "VDG half" of the interleaved DMA (IDMA). Furthermore, refresh accesses occur only during the television retrace period (at which time the VDG doesn't need to access RAM).

In summary, the VDG, MPU and SAM's Refresh Counter all transparently access the common dynamic RAM without contention or interruption.

Why IDMA?

Use of the interleaved direct memory access results in fast modification to variable portions of display RAM, by the MPU, without any distracting flashes on the screen (due to RAM access contention). In addition, the MPU is not slowed down nor stopped by the SAM; thereby, assuring accurate software timing loops without costly additional hardware timers. Furthermore, additional hardware and software to give "access permission" to the MPU is eliminated since the MPU may access RAM at **any** time.

*Only 1 pin, (DA0) out of 40 pins is dedicated to the video display.

**See VDG synchronization (page 10) for more detail.

***When not using a MC6847, $\overline{HS}$ may be wired low for continuous transparent refresh.

MPU Rate

Two bits (R1, R0) control the clock rate to the MC6809E MPU. The options are:

RATE (FREQUENCY OF "E" CLOCK)	R1	R0
0.9 MHz (Crystal Frequency ÷ 16) Slow	0	0
0.9/1.8 MHz (Address Dependent Rate)	0	1
1.8 MHz (Crystal Frequency ÷ 8) Fast	1	X
(Typical Crystal Frequency = 14.31818 MHz)		

In the "address dependent rate" mode, accesses to $0000-$7FFF and $FF00-$FF1F are slowed to 0.9 MHz (crystal frequency ÷ 16) and all other addresses are accessed at 1.8 MHz (crystal frequency ÷ 8).

Note: "Slow" (0.9 MHz) operation can be accomplished using 1.0 MHz MC6809E and MC6821 devices. For "Fast" (1.8 MHz) operation, 2.0 MHz MC68B09E and MC68B21 devices must be used.

Memory Size

Two bits (M1 and M0) determine RAM memory size. the options are:

SIZE	M1	M0
One or two banks of 4K x 1 dynamic RAMs	0	0
One or two banks of 16K x 1 dynamic RAMs	0	1
One bank of 16K x 4 dynamic RAMs①	0	1
One bank of 64K x 1 dynamic RAMs	1	0
Up to 64K static RAM*	1	1

① This option is only available when using the LS785.

* Requires a latch for demultiplexing the RAM address.

IMPORTANT!

Note: Be sure to program the SAM for the correct memory size **before** using RAM (i.e., for a subroutine stack).

Map Type

One bit (TY) is used to select between two memory map configurations.

Refer to Figures 14–16 for details. Early versions of the LS783 did not allow the "Fast" MPU rate to be used in conjunction with Map Type "TY = 1." Devices manufactured after January 1, 1983 allow both "Fast" and "Slow" MPU rates to be used with Map Type "TY = 1." (Date of manufacture is marked on devices as YYWW where YY is the year and WW is the week of manufacture).

Writing To The SAM Control Register

Any bit in the control register (CR) may be set by writing to a specific unique address. Each bit has two unique addresses . . . writing to the **even** # address **clears** the bit and writing to the **odd** # address **sets** the bit. (Data on the data bus is irrelevant in this procedure.) The specific addresses are tabulated in Figures 14–16.

If desired, a short routine may be written to program the SAM CR "a word at a time." For example, the following routine copies "B" bits from "A" register to SAM CR addresses beginning with address "X."

```
SAM1    46          ROR   A
        24  06      BCC   SAM2
        30  01      INX   (LEAX1,X)
        A7  80      STA   O,X+
        20  02      BRA   SAM3
SAM2    A7  81      STA   O,X++
SAM3    5A          DEC   B
        26  F2      BNE   SAM1
        39          RTS
```

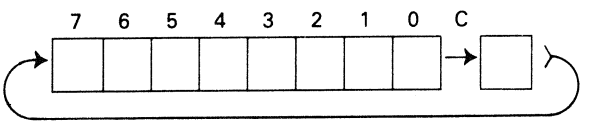
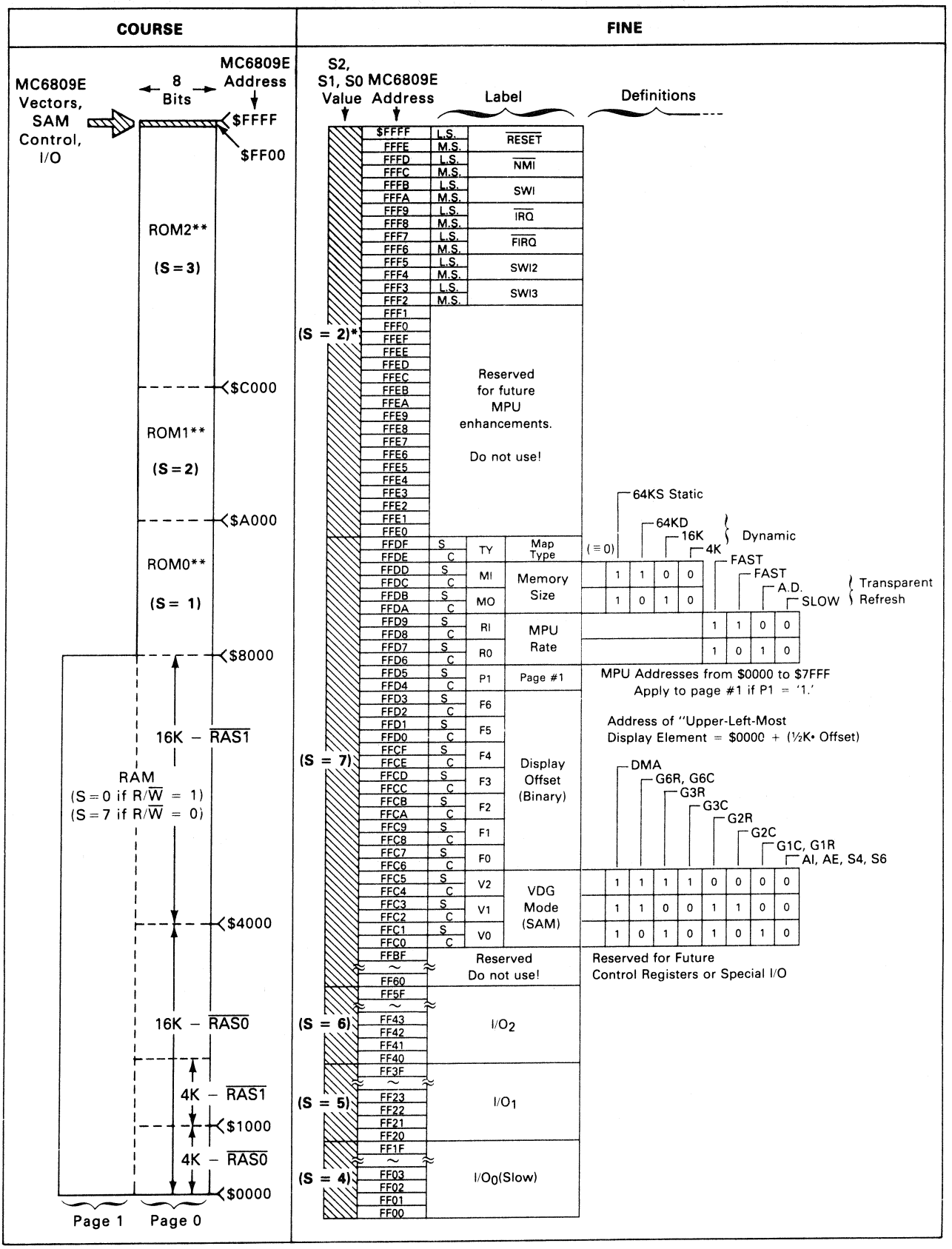


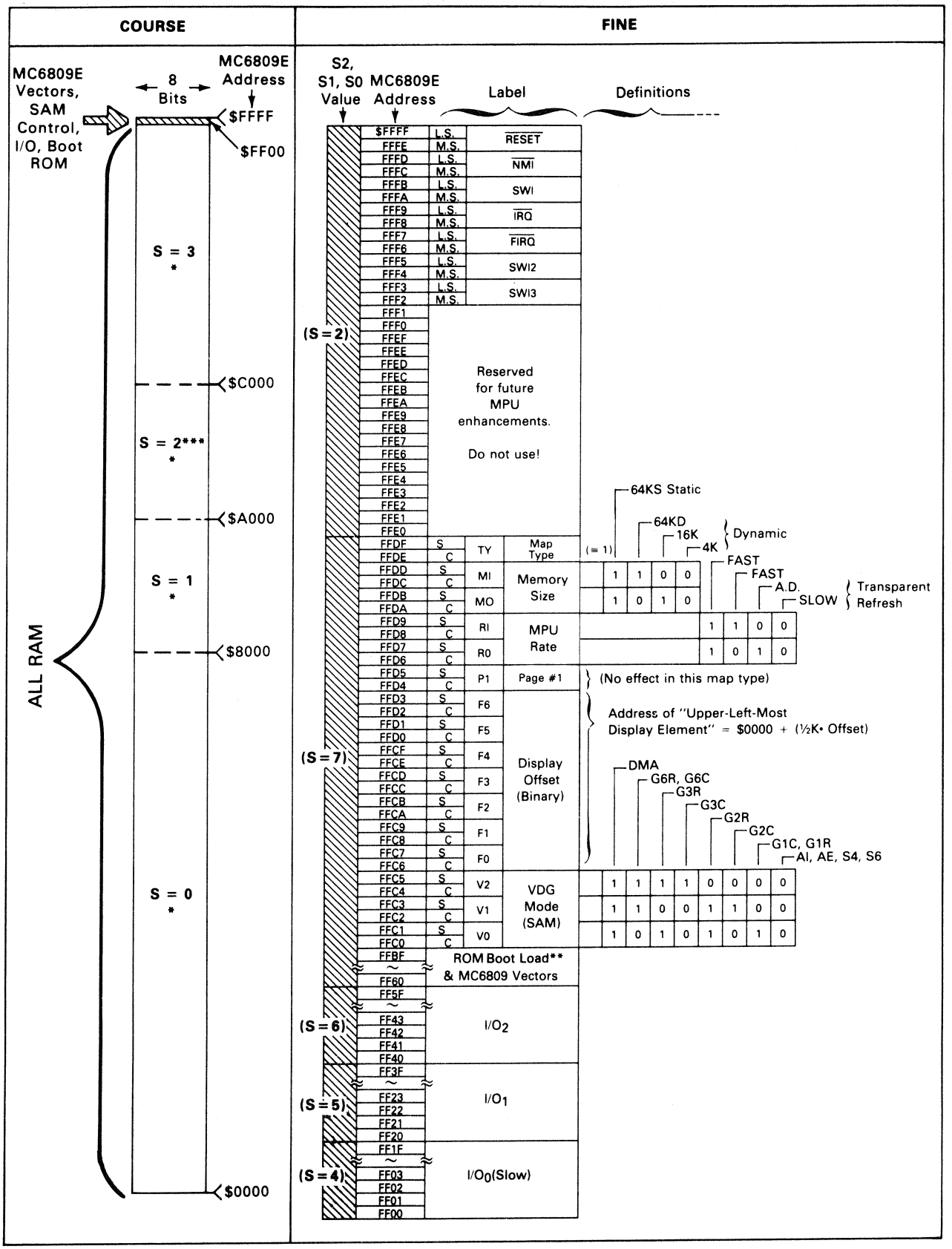
FIGURE 14 — MEMORY MAP (TYPE #0)



Abbreviations:
- M.S. ≡ Most Significant
- L.S. ≡ Least Significant
- S ≡ Set Bit
- C ≡ Clear Bit (All bits are cleared when SAM is reset.)
- **S** ≡ Device Select value = 4 x S2 + 2 x S1 + 1 x S0

NOTES:
- *On LS785, S = 7 if R/$\overline{W}$ = 0
- **This memory area may also be RAM. However, locations $FFE0-$FFFF must be ROM when using LS785.

FIGURE 15 — LS783 MEMORY MAP (TYPE #1 64K RAM)



Abbreviations:
M.S. ≡ Most Significant
L.S. ≡ Least Significant
S ≡ Set Bit
C ≡ Clear Bit (All bits are cleared when SAM is reset.)
S ≡ Device Select value = 4 x S2 + 2 x S1 + 1 x S0

NOTES:
*S = 0 if R/$\overline{W}$ = 1
**Decode S2, S1, and S0 with an open collector SN74LS156 and 'wire-or' state 7 with state 2. (See Appendix B for suggested decode circuit.)
***To avoid ROM enable during R/$\overline{W}$ = LOW, the ROM at S = 2 must be gated with R/$\overline{W}$. (See Appendix B for suggested decode circuit.)

FIGURE 17 — MEMORY ALLOCATION TABLE
(Also, see the memory MAPs on pages 17 and 18.)

Type # 0: (Primarily for ROM based systems)

Address Range	Intended Use
$FFF2 to FFFF	MC6809E Vectors: $\overline{\text{Reset}}$, $\overline{\text{NMI}}$, SWI, $\overline{\text{IRQ}}$, $\overline{\text{FIRQ}}$, SWI2, SWI3.
FFE0 to FFF1	Reserved for future MPU enhancements.
FFC0 to FFDF	SAM Control Register: V0, – V2, F0 – F6, P, R0, R1, M0, M1, TY.
FF60 to FFBF	Reserved for future control register enhancements.
FF40 to FF5F	I/O_2: Input/Output (PIAs, ACIAs, etc.) To subdivide, use A0 – A4.
FF20 to FF3F	I/O_1: Input/Output (PIAs, ACIAs, etc.) To subdivide, use A0 – A4.
FF00 to FF1F	I/O_0: Input/Output (PIAs, ACIAs, etc.) To subdivide, use A0 – A4.
C000 to FEFF	ROM2: 16K addresses. External cartridge ROM*.
A000 to BFFF	ROM1: 8K addresses. Internal ROM*. Note that MC6809E vector addresses select this ROM*.
8000 to 9FFF	ROM0: 8K addresses. Internal ROM*.
0000 to 7FFF	RAM: 32K addresses. RAM shared by MPU and VDG.

*Not restricted to ROM. For example, RAM or I/O may be used here.

Type # 1: (Primarily for RAM based systems)

Address Range	Intended Use
$FFF2 to FFFF	MC6809E Vectors: $\overline{\text{Reset}}$, $\overline{\text{NMI}}$, SWI, $\overline{\text{IRQ}}$, $\overline{\text{FIRQ}}$, SWI2, SWI3.
FFE0 to FFF1	Reserved for future MPU enhancements.
FFC0 to FFDF	SAM Control Register: V0 – V2, F0 – F6, P, R0, R1, M0, M1, TY.
FF60 to FFBF	Small ROM: Boot load program and initial MC6809 vectors.
FF40 to FF5F	I/O_2: Input/Output (PIAs, ACIAs, etc.) To subdivide, use A0 – A4.
FF20 to FF3F	I/O_1: Input/Output (PIAs, ACIAs, etc.) To subdivide, use A0 – A4.
FF00 to FF1F	I/O_1: Input/Output (PIAs, ACIAs, etc.) To subdivide, use A0 – A4.
0000 to FEFF	RAM: 64K(–256) addresses, shared by MPU and VDG. (If $R/\overline{W}$ = 0 then S = 3 for $C000-$FEFF; S = 2 for $A000-$BFFF; S = 1 for $8000-$9FFF and S = 7 for $0000-$7FFF.)

DUAL ASYNCHRONOUS RECEIVER/TRANSMITTER (DUART) SCN2681 SERIES

Preliminary

DESCRIPTION

The Signetics SCN2681 Dual Universal Asynchronous Receiver/Transmitter (DUART) is a single chip MOS-LSI communications device that provides two independent full-duplex asynchronous receiver/transmitter channels in a single package. It interfaces directly with microprocessors and may be used in a polled or interrupt driven system.

The operating mode and data format of each channel can be programmed independently. Additionally, each receiver and transmitter can select its operating speed as one of eighteen fixed baud rates, a 16x clock derived from a programmable counter/timer, or an external 1x or 16x clock. The baud rate generator and counter/timer can operate directly from a crystal or from external clock inputs. The ability to independently program the operating speed of the receiver and transmitter make the DUART particularly attractive for dual-speed channel applications such as clustered terminal systems.

Each receiver is quadruply buffered to minimize the potential of receiver overrun or to reduce interrupt overhead in interrupt driven systems. In addition, a flow control capability is provided to disable a remote DUART transmitter when the buffer of the receiving device is full.

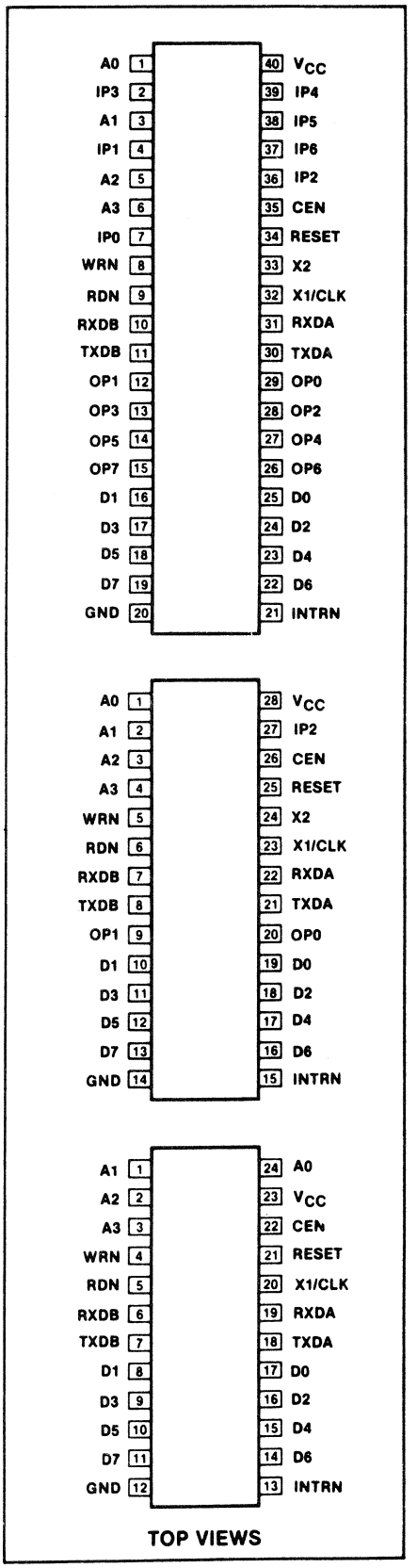
Also provided on the SCN2681 are a multipurpose 7-bit input port and a multipurpose 8-bit output port. These can be used as general purpose I/O ports or can be assigned specific functions (such as clock inputs or status/interrupt outputs) under program control.

The SCN2681 is available in three package versions to satisfy various system requirements: 40-pin and 28-pin, both 0.6″ wide DIPs, and a compact 24-pin, 0.4″ wide, DIP.

FEATURES

- **Dual full-duplex asynchronous receiver/transmiter**
- **Quadruple buffered receiver data registers**
- **Programmable data format**
  - **—5 to 8 data bits plus parity**
  - **—Odd, even, no parity or force parity**
  - **—1, 1.5 or 2 stop bits programmable in 1/16 bit increments**
- **Programmable baud rate for each receiver and transmiter selectable from:**
  - **—18 fixed rates: 50 to 38.4K baud**
  - **—One user defined rate derived from programmable timer/counter**
  - **—External 1x or 16x clock**
- **Parity, framing, and overrun error detection**
- **False start bit detection**
- **Line break detection and generation**
- **Programmable channel mode**
  - **—Normal (full duplex)**
  - **—Automatic echo**
  - **—Local loopback**
  - **—Remote loopback**
- **Multi-function programmable 16-bit counter/timer**
- **Multi-function 7-bit input port**
  - **—Can serve as clock or control inputs**
  - **—Change of state detection on four inputs**
- **Multi-function 8-bit output port**
  - **—Individual bit set/reset capability**
  - **—Outputs can be programmed to be status/interrupt signals**
- **Versatile interrupt system**
  - **—Single interrupt output with eight maskable interrupting conditions**
  - **—Output port can be configured to provide a total of up to six separate wire-OR'able interrupt outputs**
- **Maximum data transfer: 1X — 1MB/sec, 16X — 125KB/sec**
- **Automatic wake-up mode for multidrop applications**
- **Start-end break interrupt/status**
- **Detects break which originates in the middle of a character**
- **On-chip crystal oscillator**
- **TTL compatible**
- **Single +5V power supply**

PIN CONFIGURATION



TOP VIEWS

ORDERING CODE

PACKAGES	V_{CC} = 5V ± 5%, T_A = 0°C to 70°C		
	24 Pin[1]	28 Pin[2]	40 Pin[2]
Ceramic DIP	Not available	SCN2681AC1I28	SCN2681AC1I40
Plastic DIP	SCN2681AC1N24	SCN2681AC1N28	SCN2681AC1N40

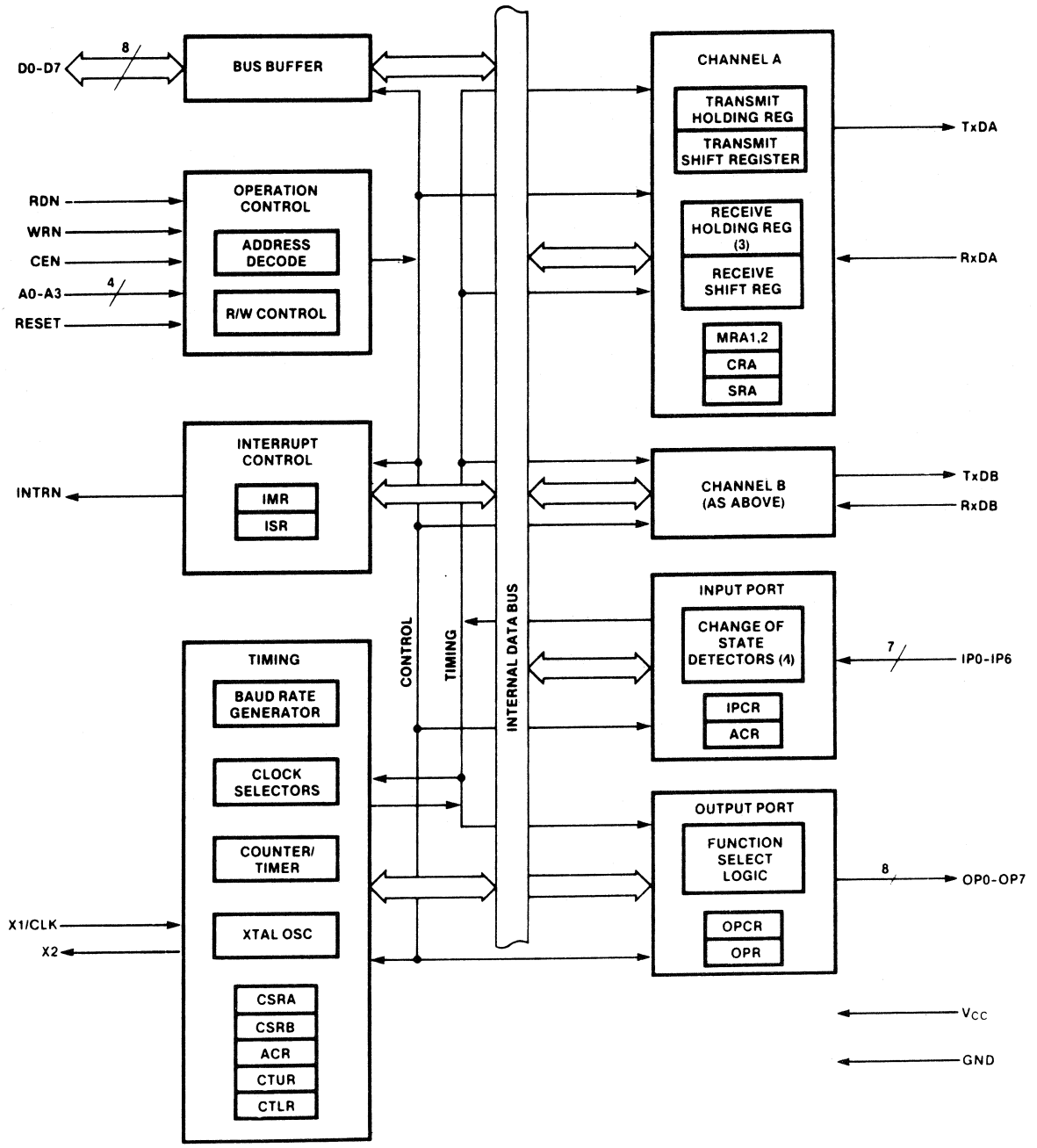
[1] 400 mil wide DIP

[2] 600 mil wide DIP

DUAL ASYNCHRONOUS RECEIVER/TRANSMITTER (DUART) SCN2681 SERIES

Preliminary

BLOCK DIAGRAM



Preliminary

PIN DESIGNATION

MNEMONIC	APPLICABLE			TYPE	NAME AND FUNCTION
	40	28	24		
D0-D7	X	X	X	I/O	**Data Bus:** Bidirectional 3-state data bus used to transfer commands, data and status between the DUART and the CPU. D0 is the least significant bit.
CEN	X	X	X	I	**Chip Enable:** Active low input signal. When low, data transfers between the CPU and the DUART are enabled on D0-D7 as controlled by the WRN, RDN and A0-A3 inputs. When high, places the D0-D7 lines in the 3-state condition.
WRN	X	X	X	I	**Write Strobe:** When low and CEN is also low, the contents of the data bus is loaded into the addressed register. The transfer occurs on the rising edge of the signal.
RDN	X	X	X	I	**Read Strobe:** When low and CEN is also low, causes the contents of the addressed register to be presented on the data bus. The read cycle begins on the falling edge of RDN.
A0-A3	X	X	X	I	**Address Inputs:** Select the DUART internal registers and ports for read/write operations.
RESET	X	X	X	I	**Reset:** A high level clears internal registers (SRA, SRB, IMR, ISR, OPR, OPCR), puts OP0-OP7 in the high state, stops the counter/timer, and puts channels A and B in the inactive state, with the TxDA and TxDB outputs in the mark (high) state.
INTRN	X	X	X	O	**Interrupt Request:** Active low, open drain, output which signals the CPU that one or more of the eight maskable interrupting conditions are true.
X1/CLK	X	X	X	I	**Crystal 1:** Crystal or external clock input. A crystal or clock of the specified limits must be supplied at all times. When a crystal is used, a capacitor must be connected from this pin to ground (see figure 5).
X2	X	X		O	**Crystal 2:** Connection for other side of the crystal. Should be connected to ground if a crystal is not used. When a crystal is used, a capacitor must be connected from this pin to ground (see figure 5).
RxDA	X	X	X	I	**Channel A Receiver Serial Data Input:** The least significant bit is received first. 'Mark' is high, 'space' is low.
RxDB	X	X	X	I	**Channel B Receiver Serial Data Input:** The least significant bit is received first. 'Mark' is high, 'space' is low.
TxDA	X	X	X	O	**Channel A Transmitter Serial Data Output:** The least significant bit is transmitted first. This output is held in the 'mark' condition when the transmitter is disabled, idle, or when operating in local loopback mode. 'Mark' is high, 'space' is low.
TxDB	X	X	X	O	**Channel B Transmitter Serial Data Output:** The least significant bit is transmitted first. This output is held in the 'mark' condition when the transmitter is disabled, idle, or when operating in local loopback mode. 'Mark' is high, 'space' is low.
OP0	X	X		O	**Output 0:** General purpose output, or channel A request to send (RTSAN, active low). Can be deactivated on receive or transmit.
OP1	X	X		O	**Output 1:** General purpose output, or channel B request to send (RTSBN, active low). Can be deactivated on receive or transmit.
OP2	X			O	**Output 2:** General purpose output, or channel A transmitter 1X or 16X clock output, or channel A receiver 1X clock output.
OP3	X			O	**Output 3:** General purpose output, or open drain, active low counter/timer output, or channel B transmitter 1X clock output, or channel B receiver 1X clock output.
OP4	X			O	**Output 4:** General purpose output, or channel A open drain, active low, RxRDYA/FFULLA output.
OP5	X			O	**Output 5:** General purpose output, or channel B open drain, active low, RxRDYB/FFULLB output.
OP6	X			O	**Output 6:** General purpose output, or channel A open drain, active low, TxRDYA output.
OP7	X			O	**Output 7:** General purpose output, or channel B open drain, active low, TxRDYB output.
IP0	X			I	**Input 0:** General purpose input, or channel A clear to send active low input (CTSAN).
IP1	X			I	**Input 1:** General purpose input, or channel B clear to send active low input (CTSBN).
IP2	X	X		I	**Input 2:** General purpose input, or counter/timer external clock input.
IP3	X			I	**Input 3:** General purpose input, or channel A transmitter external clock input (TxCA). When the external clock is used by the transmitter, the transmitted data is clocked on the falling edge of the clock.

Preliminary

PIN DESIGNATION (Continued)

MNEMONIC	APPLICABLE			TYPE	NAME AND FUNCTION
	40	28	24		
IP4	X			I	**Input 4:** General purpose input, or channel A receiver external clock input (RxCA). When the external clock is used by the receiver, the received data is sampled on the rising edge of the clock.
IP5	X			I	**Input 5:** General purpose input, or channel B transmitter external clock input (TxCB). When the external clock is used by the transmitter, the transmitted data is clocked on the falling edge of the clock.
IP6	X			I	**Input 6:** General purpose input or channel B receiver external clock input (RxCB). When the external clock is used by the receiver, the received data is sampled on the rising edge of the clock.
V_{CC}	X	X	X	I	**Power Supply:** +5V supply input
GND	X	X	X	I	**Ground**

BLOCK DIAGRAM

The 2681 DUART consists of the following eight major sections: data bus buffer, operation control, interrupt control, timing, communications channels A and B, input port and output port. Refer to the block diagram.

Data Bus Buffer

The data bus buffer provides the interface between the external and internal data busses. It is controlled by the operation control block to allow read and write operations to take place between the controlling CPU and the DUART.

Operation Control

The operation control logic receives operation commands from the CPU and generates appropriate signals to internal sections to control device operation. It contains address decoding and read and write circuits to permit communications with the microprocessor via the data bus buffer.

Interrupt Control

A single active low interrupt output (INTRN) is provided which is activated upon the occurence of any of eight internal events. Associated with the interrupt system are the interrupt mask register (IMR) and the interrupt status register (ISR). The IMR may be programmed to select only certain conditions to cause INTRN to be asserted. The ISR can be read by the CPU to determine all currently active interrupting conditions.

Outputs OP3-OP7 can be programmed to provide discrete interrupt outputs for the transmitters, receivers, and counter/timer.

Timing Circuits

The timing block consists of a crystal oscillator, a baud rate generator, a programmable 16-bit counter/timer, and four clock selectors. The crystal oscillator operates directly from a 3.6864MHz crystal connected across the X1/CLK and X2 inputs. If an external clock of the appropriate frequency is available, it may be connected to X1/CLK. The clock serves as the basic timing reference for the baud rate generator (BRG), the counter/timer, and other internal circuits. A clock signal within the limits specified in the specifications section of this data sheet must always be supplied to the DUART.

The baud rate generator operates from the oscillator or external clock input and is capable of generating 18 commonly used data communications baud rates ranging from 50 to 38.4K baud. The clock outputs from the BRG are at 16X the actual baud rate. The counter/timer can be used as a timer to produce a 16X clock for any other baud rate by counting down the crystal clock or an external clock. The four clock selectors allow the independent selection, for each receiver and transmitter, of any of these baud rates or an external timing signal.

The counter/timer (C/T) can be programmed to use one of several timing sources as its input. The output of the C/T is available to the clock selectors and can also be programmed to be output at OP3. In the counter mode, the contents of the C/T can be read by the CPU and it can be stopped and started under program control. In the timer mode, the C/T acts as a programmable divider.

Communications Channels A and B

Each communications channel of the 2681 comprises a full duplex asynchronous receiver/transmitter (UART). The operating frequency for each receiver and transmitter can be selected independently from the baud rate generator, the counter timer, or from an external input.

The transmitter accepts parallel data from the CPU, converts it to a serial bit stream, inserts the appropriate start, stop, and optional parity bits and outputs a composite serial stream of data on the TxD output pin. The receiver accepts serial data on the RxD pin, converts this serial input to parallel format, checks for start bit, stop bit, parity bit (if any), or break condition and sends an assembled character to the CPU.

Input Port

The inputs to this unlatched 7-bit port can be read by the CPU by performing a read operation at address D_{16}. A high input results in a logic 1 while a low input results in a logic 0. D_7 will always be read as a logic 1. The pins of this port can also serve as auxiliary inputs to certain portions of the DUART logic.

Four change-of-state detectors are provided which are associated with inputs IP3, IP2, IP1, and IP0. A high-to-low or low-to-high transition of these inputs lasting longer than 25-50μs will set the corresponding bit in the input port will change register. The bits are cleared when the register is read by the CPU. Any change of state can also be programmed to generate an interrupt to the CPU.

Preliminary

Output Port

The 8-bit multi-purpose output port can be used as a general purpose output port, in which case the outputs are the complements of the output port register (OPR). OPR[n] = 1 results in OP[n] = low and vice-versa. Bits of the OPR can be individually set and reset. A bit is set by performing a write operation at address E_{16} with the accompanying data specifying the bits to be set (1 = set, 0 = no change). Likewise, a bit is reset by a write at address F_{16} with the accompanying data specifying the bits to be reset (1 = reset, 0 = no change).

Outputs can be also individually assigned specific functions by appropriate programming of the channel A mode registers (MR1A, MR2A), the channel B mode registers (MR1B, MR2B), and the output port configuration register (OPCR).

OPERATION

Transmitter

The 2681 is conditioned to transmit data when the transmitter is enabled through the command register. The 2681 indicates to the CPU that it is ready to accept a character by setting the TxRDY bit in the status register. This condition can be programmed to generate an interrupt request at OP6 or OP7 and INTRN. When a character is loaded into the transmit holding register (THR), the above conditions are negated. Data is transferred from the holding register to the transmit shift register when it is idle or has completed transmission of the previous character. The TxRDY conditions are then asserted again which means one full character time of buffering is provided. Characters cannot be loaded into the THR while the transmitter is disabled.

The transmitter converts the parallel data from the CPU to a serial bit stream on the TxD output pin. It automatically sends a start bit followed by the programmed number of data bits, an optional parity bit, and the programmed number of stop bits. The least significant bit is sent first. Following the transmission of the stop bits, if a new character is not available in the THR, the TxD output remains high and the TxEMT bit in the status register (SR) will be set to 1. Transmission resumes and the TxEMT bit is cleared when the CPU loads a new character into the THR. If the transmitter is disabled, it continues operating until the character currently being transmitted is completely sent out. The transmitter can be forced to send a continuous low condition by issuing a send break command.

The transmitter can be reset through a software command. If it is reset, operation ceases immediately and the transmitter must be enabled through the command register before resuming operation. If CTS operation is enabled, the CTSN input must be low in order for the character to be transmitted. if it goes high in the middle of a transmission, the character in the shift register is transmitted and TxDA then remains in the marking state until CTSN goes low. The transmitter can also control the deactivation of the RTSN output. If programmed, the RTSN output will be reset one bit time after the character in the transmit shift register and transmit holding register (if any) are completely transmitted, if the transmitter has been disabled.

Receiver

The 2681 is conditioned to receive data when enabled through the command register. The receiver looks for a high to low (mark to space) transition of the start bit on the RxD input pin. If a transition is detected, the state of the RxD pin is sampled each 16X clock for 7-1/2 clocks (16X clock mode) or at the next rising edge of the bit time clock (1X clock mode). If RxD is sampled high, the start bit is invalid and the search for a valid start bit begins again. If RxD is still low, a valid start bit is assumed and the receiver continues to sample the input at one bit time intervals at the theoretical center of the bit, until the proper number of data bits and the parity bit (if any) have been assembled, and one stop bit has been detected. The least sigificant bit is received first. The data is then transferred to the receive holding register (RHR) and the RxRDY bit in the SR is set to a 1. This condition can be programmed to generate an interrupt at OP4 or OP5 and INTRN. If the character length is less than eight bits, the most significant unused bits in the RHR are set to zero.

After the stop bit is detected, the receiver will immediately look for the next start bit. However, if a non-zero character was received without a stop bit (framing error) and RxD remains low for one half of the bit period after the stop bit was sampled, then the receiver operates as if a new start bit transition had been detected at that point (one-half bit time after the stop bit was sampled).

The parity error, framing error, overrun error and received break state (if any) are strobed into the SR at the received character boundary, before the RxRDY status bit is set. If a break condition is detected (RxD is low for the entire character including the stop bit), a character consisting of all zeros will be loaded into the RHR and the received break bit in the SR is set to 1. The RxD input must return to a high condition for at least one-half bit time before a search for the next start bit begins.

The RHR consists of a first-in-first-out (FIFO) stack with a capacity of three characters. Data is loaded from the receive shift register into the topmost empty position of the FIFO. The RxRDY bit in the status register is set whenever one or more characters are available to be read, and a FFULL status bit is set if all three stack positions are filled with data. Either of these bits can be selected to cause an interrupt. A read of the RHR outputs the data at the top of the FIFO. After the read cycle, the data FIFO and its associated status bits (see below) are 'popped' thus emptying a FIFO position for new data.

In addition to the data word, three status bits (parity error, framing error, and received break) are also appended to each data character in the FIFO (overrun is not). Status can be provided in two ways, as programmed by the error mode control bit in the mode register. In the 'character' mode, status is provided on a character-by-character basis: the status applies only to the character at the top of the FIFO. In the 'block' mode, the status provided in the SR for these three bits is the logical OR of the status for all characters coming to the top of the FIFO since the last 'reset error' command was issued. In either mode reading the SR does not affect the FIFO. The FIFO is 'popped' only when the RHR is read. Therefore the status register should be read prior to reading the FIFO.

If the FIFO is full when a new character is received, that character is held in the receive shift register until a FIFO position is available. If an additional character is received while this state exits, the contents of the FIFO are not affected: the character previously in the shift register is lost and the overrun error status bit (SR[4]) will be set upon receipt of the start bit of the new (overruning) character.

The receiver can control the deactivation of RTS. If programmed to operate in this mode, the RTSN output will be negated when a valid start bit was received and the FIFO is full. When a FIFO position becomes available, the RTSN output will be re-asserted automatically. This feature can be used to prevent an overrun, in the

receiver, by connecting the RTSN output to the CTSN input of the transmitting device.

If the receiver is disabled, the FIFO characters can be read. However, no additional characters can be received until the receiver is enabled again. If the receiver is reset, the FIFO and all of the receiver status, and the corresponding output ports and interrupt are reset. No additional characters can be received until the receiver is enabled again.

Multidrop Mode

The DUART is equipped with a wake up mode used for multidrop applications. This mode is selected by programming bits MR1A[4:3] or MR1B[4:3] to '11' for channels A and B respectively. In this mode of operation, a 'master' station transmits an address character followed by data characters for the addressed 'slave' station. The slave stations, with receivers that are normally disabled, examine the received data stream and 'wake-up' the CPU (by setting RxRDY) only upon receipt of an address character. The CPU compares the received address to its station address and enables the receiver if it wishes to receive the subsequent data characters. Upon receipt of another address character, the CPU may disable the receiver to initiate the process again.

A transmitted character consists of a start bit, the programmed number of data bits, an address/data (A/D) bit, and the programmed number of stop bits. The polarity of the transmitted A/D bit is selected by the CPU by programming bit MR1A[2]/MR1B[2]. MR1A[2]/MR1B[2] = 0 transmits a zero in the A/D bit position, which identifies the corresponding data bits as data, while MR1A[2]/MR1B[2] = 1 transmits a one in the A/D bit position, which identifies the corresponding data bits as an address. The CPU should program the mode register prior to loading the corresponding data bits into the THR.

In this mode, the receiver continuously looks at the received data stream, whether it is enabled or disabled. If disabled, it sets the RxRDY status bit and loads the character into the RHR FIFO if the received A/D bit is a one (address tag), but discards the received character if the received A/D bit is a zero (data tag). If enabled, all received characters are transferred to the CPU via the RHR. In either case, the data bits are loaded into the data FIFO while the A/D bit is loaded into the status FIFO position normally used for parity error (SRA[5] or SRB[5]). Framing error, overrun error, and break detect operate normally whether or not the receiver is enabled.

PROGRAMMING

The operation of the DUART is programmed by writing control words into the appropriate registers. Operational feedback is provided via status registers which can be read by the CPU. The addressing of the registers is described in table 1.

The contents of certain control registers are initialized to zero on RESET. Care should be exercised if the contents of a register are changed during operation, since certain changes may cause operational problems. For example, changing the number of bits per character while the transmitter is active may cause the transmission of an incorrect character. In general, the contents of the MR, the CSR, and the OPCR should only be changed while the receiver(s) and transmitter(s) are not enabled, and certain changes to the ACR should only be made while the C/T is stopped.

Mode registers 1 and 2 of each channel are accessed via independent auxiliary pointers. The pointer is set to MR1x by RESET or by issuing a 'reset pointer' command via the corresponding command register. Any read or write of the mode register while the pointer is at MR1x switches the pointer to MR2x. The pointer then remains at MR2x, so that subsequent accesses are always to MR2x unless the pointer is reset to MR1x as described above.

Mode, command, clock select, and status registers are duplicated for each channel to provide total independent operation and control. Refer to table 2 for register bit descriptions.

MR1A — Channel A Mode Register 1

MR1A is accessed when the channel A MR pointer points to MR1. The pointer is set to MR1 by RESET or by a 'set pointer' command applied via CRA. After reading or writing MR1A, the pointer will point to MR2A.

MR1A[7] — Channel A Receiver Request-to-Send Control — This bit controls the deactivation of the RTSAN output (OP0) by the receiver. This output is normally asserted by setting OPR[0] and negated by resetting OPR[0]. MR1A[7] = 1 causes RTSAN to be negated upon receipt of a valid start bit if the channel A FIFO is full. However, OPR[0] is not reset and RTSAN will be asserted again when an empty FIFO position is available. This feature can be used for flow control to prevent overrun in the receiver by using the RTSAN output signal to control the CTSN input of the transmitting device.

MR1A[6] — Channel A Receiver Interrupt Select — This bit selects either the channel A receiver ready status (RXRDY) or the channel A FIFO full status (FFULL) to be used for CPU interrupts. It also causes the selected bit to be output on OP4 if it is programmed as an interrupt output via the OPCR.

MR1A[5] — Channel A Error Mode Select — This bit selects the operating mode of the three FIFOed status bits (FE, PE, received break) for channel A. In the 'character' mode, status is provided on a character-by-character basis: the status applies only to the character at the top of the FIFO. In the 'block' mode, the status provided in the SR for these bits is the ac-

Table 1 2681 REGISTER ADDRESSING

A3	A2	A1	A0	READ (RDN = 0)	WRITE (WRN = 0)
0	0	0	0	Mode Register A (MR1A, MR2A)	Mode Register A (MR1A, MR2A)
0	0	0	1	Status Register A (SRA)	Clock Select Reg. A (CSRA)
0	0	1	0	*Reserved*	Command Register A (CRA)
0	0	1	1	RX Holding Register A (RHRA)	TX Holding Register A (THRA)
0	1	0	0	Input Port Change Reg. (IPCR)	Aux. Control Register (ACR)
0	1	0	1	Interrupt Status Reg. (ISR)	Interrupt Mask Reg. (IMR)
0	1	1	0	Counter/Timer Upper (CTU)	C/T Upper Register (CTUR)
0	1	1	1	Counter/Timer Lower (CTL)	C/T Lower Register (CTLR)
1	0	0	0	Mode Register B (MR1B, MR2B)	Mode Register B (MR1B, MR2B)
1	0	0	1	Status Register B (SRB)	Clock Select Reg. B (CSRB)
1	0	1	0	*Reserved*	Command Register B (CRB)
1	0	1	1	RX Holding Register B (RHRB)	TX Holding Register B (THRB)
1	1	0	0	*Reserved*	*Reserved*
1	1	0	1	Input Port	Output Port Conf. Reg. (OPCR)
1	1	1	0	Start Counter Command	Set Output Port Bits Command
1	1	1	1	Stop Counter Command	Reset Output Port Bits Command

Preliminary

Table 2 REGISTER BIT FORMATS

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
MR1A MR1B	RX RTS CONTROL	RX INT SELECT	ERROR MODE	PARITY MODE		PARITY TYPE	BITS PER CHAR.	
	0 = no 1 = yes	0 = RXRDY 1 = FFULL	0 = char 1 = block	00 = with parity 01 = force parity 10 = no parity 11 = multi-drop mode		0 = even 1 = odd	00 = 5 01 = 6 10 = 7 11 = 8	

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
MR2A MR2B	CHANNEL MODE		Tx RTS CONTROL	CTS ENABLE Tx	STOP BIT LENGTH*			
	00 = Normal 01 = Auto echo 10 = Local loop 11 = Remote loop		0 = no 1 = yes	0 = no 1 = yes	0 = 0.563 1 = 0.625 2 = 0.688 3 = 0.750	4 = 0.813 5 = 0.875 6 = 0.938 7 = 1.000	8 = 1.563 9 = 1.625 A = 1.688 B = 1.750	C = 1.813 D = 1.875 E = 1.938 F = 2.000

*Add 0.5 to values shown for 0-7 if channel is programmed for 5 bits/char.

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
CSRA CSRB	RECEIVER CLOCK SELECT				TRANSMITTER CLOCK SELECT			
	See text				See text			

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
CRA CRB	not used— must be 0	MISCELLANEOUS COMMANDS			DISABLE Tx	ENABLE Tx	DISABLE Rx	ENABLE Rx
		See text			0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
SRA SRB	RECEIVED BREAK	FRAMING ERROR	PARITY ERROR	OVERRUN ERROR	TxEMT	TxRDY	FFULL	RxRDY
	0 = no 1 = yes *	0 = no 1 = yes *	0 = no 1 = yes *	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes

*These status bits are appended to the corresponding data character in the receive FIFO. A read of the status register provides these bits (7:5) from the top of the FIFO together with bits 4:0. These bits are cleared by a 'reset error status' command. In character mode they are discarded when the corresponding data character is read from the FIFO.

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
OPCR	OP7	OP6	OP5	OP4	OP3		OP2	
	0 = OPR[7] 1 = TxRDYB	0 = OPR[6] 1 = TxRDYA	0 = OPR[5] 1 = RxRDY/ FFULLB	0 = OPR[4] 1 = RxRDY/ FFULLA	00 = OPR[3] 01 = C/T OUTPUT 10 = TxCB (1X) 11 = RxCB (1X)		00 = OPR[2] 01 = TxCA (16X) 10 = TxCA (1X) 11 = RxCA (1X)	

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ACR	BRG SET SELECT	COUNTER/TIMER MODE AND SOURCE			DELTA IP3 INT	DELTA IP2 INT	DELTA IP1 INT	DELTA IP0 INT
	0 = set1 1 = set2	See table 4			0 = off 1 = on	0 = off 1 = on	0 = off 1 = on	0 = off 1 = on

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
IPCR	DELTA IP3	DELTA IP2	DELTA IP1	DELTA IP0	IP3	IP2	IP1	IP0
	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = low 1 = high	0 = low 1 = high	0 = low 1 = high	0 = low 1 = high

Table 2 REGISTER BIT FORMATS (Continued)

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ISR	INPUT PORT CHANGE	DELTA BREAK B	RxRDY/ FFULLB	TxRDYB	COUNTER READY	DELTA BREAK A	RxRDY/ FFULLA	TxRDYA
	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes	0 = no 1 = yes

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
IMR	IN. PORT CHANGE INT	DELTA BREAK B INT	RxRDY/ FFULLB INT	TxRDYB INT	COUNTER READY INT	DELTA BREAK A INT	RxRDY/ FFULLA INT	TxRDYA INT
	0 = off 1 = on	0 = off 1 = on	0 = off 1 = on	0 = off 1 = on	0 = off 1 = on	0 = off 1 = on	0 = off 1 = on	0 = off 1 = on

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
CTUR	C/T[15]	C/T[14]	C/T[13]	C/T[12]	C/T[11]	C/T[10]	C/T[9]	C/T[8]

	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
CTLR	C/T[7]	C/T[6]	C/T[5]	C/T[4]	C/T[3]	C/T[2]	C/T[1]	C/T[0]

cumulation (logical OR) of the status for all characters coming to the top of the FIFO since the last 'reset error' command for channel A was issued.

MR1A[4:3] — Channel A Parity Mode Select — If 'with parity' or 'force parity' is selected, a parity bit is added to the transmitted character and the receiver performs a parity check on incoming data. MR1A[4:3] = 11 selects channel A to operate in the special multidrop mode described in the Operation section.

MR1A[2] — Channel A Parity Type Select — This bit selects the parity type (odd or even) if the 'with parity' mode is programmed by MR1A[4:3], and the polarity of the forced parity bit if the 'force parity' mode is programmed. It has no effect if the 'no parity' mode is programmed. In the special multidrop mode it selects the polarity of the A/D bit.

MR1A[1:0] — Channel A Bits per Character Select — This field selects the number of data bits per character to be transmitted and received. The character length does not include the start, parity, and stop bits.

MR2A — Channel A Mode Register 2

MR2A is accessed when the channel A MR pointer points to MR2, which occurs after any access to MR1A. Accesses to MR2A do not change the pointer.

MR2A[7:6] — Channel A Mode Select — Each channel of the DUART can operate in one of four modes. MR2A[7:6] = 00 is the normal mode, with the transmitter and receiver operating independently. MR2A[7:6] = 01 places the channel in the automatic echo mode, which automatically retransmits the received data. The following conditions are true while in automatic echo mode:

1. Received data is reclocked and retransmitted on the TxDA output.
2. The receive clock is used for the transmitter.
3. The receiver must be enabled, but the transmitter need not be enabled.
4. The channel A TxRDY and TxEMT status bits are inactive.
5. The received parity is checked, but is not regenerated for transmission, i.e., transmitted parity bit is as received.
6. Character framing is checked, but the stop bits are retransmitted as received.
7. A received break is echoed as received until the next valid start bit is detected.
8. CPU to receiver communication continues normally, but the CPU to transmitter link is disabled.

Two diagnostic modes can also be configured. MR2A[7:6] = 10 selects local loopback mode. In this mode:

1. The transmitter output is internally connected to the receiver input.
2. The transmit clock is used for the receiver.
3. The TxDA output is held high.
4. The RxDA input is ignored.
5. The transmitter must be enabled, but the receiver need not be enabled.
6. CPU to transmitter and receiver communications continue normally.

The second diagnostic mode is the remote loopback mode, selected by MR2A[7:6] = 11. In this mode:

1. Received data is relocked and retransmitted on the TxDA output.
2. The receive clock is used for the transmitter.

3. Received data is not sent to the local CPU, and the error status conditions are inactive.
4. The received parity is not checked and is not regenerated for transmission, i.e., transmitted parity bit is as received.
5. The receiver must be enabled.
6. Character framing is not checked, and the stop bits are retransmitted as received.
7. A received break is echoed as received until the next valid start bit is detected.

The user must exercise care when switching into and out of the various modes. The selected mode will be activated immediately upon mode selection, even if this occurs in the middle of a received or transmitted character. Likewise, if a mode is deselected, the device will switch out of the mode immediately. An exception to this is switching out of autoecho or remote loopback modes: if the de-selection occurs just after the receiver has sampled the stop bit (indicated in autoecho by assertion of RxRDY), and the transmitter is enabled, the transmitter will remain in autoecho mode until the entire stop bit has been retransmitted.

MR2A[5] — Channel A Transmitter Request-to-Send Control — This bit controls the deactivation of the RTSAN output (OP0) by the transmitter. This output is normally asserted by setting OPR[0] and negated by resetting OPR[0]. MR2A[5] = 1 causes OPR[0] to be reset automatically one bit time after the characters in the channel A transmit shift register and in the THR, if any, are completely transmitted, including the programmed number of stop bits, if the transmitter is not enabled. This feature can be used to automatically terminate the transmission of a message as follows:

1. Program auto-reset mode: MR2A[5] = 1.
2. Enable transmitter.
3. Assert RTSAN: OPR[0] = 1.
4. Send message.
5. Disable transmitter after the last character is loaded into the channel A THR.
6. The last character will be transmitted and OPR[0] will be reset one bit time after the last stop bit, causing RTSAN to be negated.

MR2A[4] — Channel A Clear-to-Send Control — If this bit is 0, CTSAN has no effect on the transmitter. If this bit is a 1, the transmitter checks the state of CTSAN (IP0) each time it is ready to send a character. If IP0 is asserted (low), the character is transmitted. If it is negated (high), the TxDA output remains in the marking state and the transmission is delayed until CTSAN goes low. Changes in CTSAN while a character is being transmitted do not affect the transmission of that character.

MR2A[3:0] — Channel A Stop Bit Length Select — This field programs the length of the stop bit appended to the transmitted character. Stop bit lengths of 9/16 to 1 and 1-9/16 to 2 bits, in increments of 1/16 bit, can be programmed for character lengths of 6, 7, and 8 bits. For a character length of 5 bits, 1-1/16 to 2 stop bits can be programmed in increments of 1/16 bit. The receiver only checks for a 'mark' condition at the center of the first stop bit position (one bit time after the last data bit, or after the parity bit if parity is enabled) in all cases.

If an external 1X clock is used for the transmitter, MR2A[3] = 0 selects one stop bit and MR2A[3] = 1 selects two stop bits to be transmitted.

MR1B — Channel B Mode Register 1

MR1B is accessed when the channel B MR pointer points to MR1. The pointer is set to MR1 by RESET or by a 'set pointer' command applied via CRB. After reading or writing MR1B, the pointer will point to MR2B.

The bit definitions for this register are identical to the bit definitions for MR1A, except that all control actions apply to the channel B receiver and transmitter and the corresponding inputs and outputs.

MR2B — Channel B Mode Register 2

MR2B is accessed when the channel B MR pointer points to MR2, which occurs after any access to MR1B. Accesses to MR2B do not change the pointer.

The bit definitions for this register are identical to the bit definitions for MR2A, except that all control actions apply to the channel B receiver and transmitter and the corresponding inputs and outputs.

CSRA — Channel A Clock Select Register

CSRA[7:4] — Channel A Receiver Clock Select — This field selects the baud rate clock for the channel A receiver as follows:

				Baud Rate CLOCK = 3.6864MHz	
CSRA[7:4]				ACR[7] = 0	ACR[7] = 1
0	0	0	0	50	75
0	0	0	1	110	110
0	0	1	0	134.5	134.5
0	0	1	1	200	150
0	1	0	0	300	300
0	1	0	1	600	600
0	1	1	0	1,200	1,200
0	1	1	1	1,050	2,000
1	0	0	0	2,400	2,400
1	0	0	1	4,800	4,800
1	0	1	0	7,200	1,800
1	0	1	1	9,600	9,600
1	1	0	0	38.4K	19.2K
1	1	0	1	Timer	Timer
1	1	1	0	IP4—16X	IP4—16X
1	1	1	1	IP4—1X	IP4—1X

The receiver clock is always a 16X clock except for CSRA[7:4] = 1111.

CSRA[3:0] — Channel A Transmitter Clock Select — This field selects the baud rate clock for the channel A transmitter. The field definition is as per CSRA[7:4] except as follows:

				Baud Rate	
CSRA[3:0]				ACR[7] = 0	ACR[7] = 1
1	1	1	0	IP3—16X	IP3—16X
1	1	1	1	IP3—1X	IP3—1X

The transmitter clock is always a 16X clock except for CSRA[3:0] = 1111.

CSRB — Channel B Clock Select Register

CSRB[7:4] — Channel B Receiver Clock Select — This field selects the baud rate clock for the channel B receiver. The field definition is as per CSRA[7:4] except as follows:

				Baud Rate	
CSRB[7:4]				ACR[7] = 0	ACR[7] = 1
1	1	1	0	IP6—16X	IP6—16X
1	1	1	1	IP6—1X	IP6—1X

The receiver clock is always a 16X clock except for CSRB[7:4] = 1111.

CSRB[3:0] — Channel B Transmitter Clock Select — This field selects the baud rate clock for the channel B transmitter. The field definition is as per CSRA[7:4] except as follows:

				Baud Rate	
CSRB[3:0]				ACR[7] = 0	ACR[7] = 1
1	1	1	0	IP5—16X	IP5—16X
1	1	1	1	IP5—1X	IP5—1X

The transmitter clock is always a 16X clock except for CSRB[3:0] = 1111.

Preliminary

CRA — Channel A Command Register

CRA is a register used to supply commands to channel A. Multiple commands can be specified in a single write to CRA as long as the commands are non-conflicting, e.g., the 'enable transmitter' and 'reset transmitter' commands cannot be specified in a single command word.

CRA[6:4] — Channel A Miscellaneous Commands — The encoded value of this field may be used to specify a single command as follows:

CRA[6:4]	COMMAND
0 0 0	No command.
0 0 1	Reset MR pointer. Causes the channel A MR pointer to point to MR1.
0 1 0	Reset receiver. Resets the channel A receiver as if a hardware reset had been applied. The receiver is disabled and the FIFO is flushed.
0 1 1	Reset transmitter. Resets the channel A transmitter as if a hardware reset had been applied.
1 0 0	Reset error status. Clears the channel A Received Break, Parity Error, Framing Error, and Overrun Error bits in the status register (SRA[7:4]). Used in character mode to clear OE status (although RB, PE, and FE bits will also be cleared) and in block mode to clear all error status after a block of data has been received.
1 0 1	Reset channel A break change interrupt. Causes the channel A break detect change bit in the interrupt status register (ISR[2]) to be cleared to zero.
1 1 0	Start break. Forces the TXDA output low (spacing). If the transmitter is empty the start of the break condition will be delayed up to two bit times. If the transmitter is active the break begins when transmission of the character is completed. If a character is in the THR, the start of the break will be delayed until that character, or any others loaded subsequently are transmitted. The transmitter must be enabled for this command to be accepted.
1 1 1	Stop Break. The TXDA line will go high (marking) within two bit times. TXDA will remain high for one bit time before the next character, if any, is transmitted.

CRA[3] — Disable Channel A Transmitter — This command terminates transmitter operation and resets the TxRDY and TxEMT status bits. However, if a character is being transmitted or if a character is in the THR when the transmitter is disabled, the transmission of the character(s) is completed before assuming the inactive state.

CRA[2] — Enable Channel A Transmitter — Enables operation of the channel A transmitter. The TxRDY status bit will be asserted.

CRA[1] — Disable Channel A Receiver — This command terminates operation of the receiver immediately — a character being received will be lost. The command has no effect on the receiver status bits or any other control registers. If the special multidrop mode is programmed, the receiver operates even if it is disabled. See Operation section.

CRA[0] — Enable Channel A Receiver — Enables operation of the channel A receiver. If not in the special wakeup mode, this also forces the receiver into the search for start-bit state.

CRB — Channel B Command Register

CRB is a register used to supply commands to channel B. Multiple commands can be specified in a single write to CRB as long as the commands are non-conflicting, e.g., the 'enable transmitter' and 'reset transmitter' commands cannot be specified in a single command word.

The bit definitions for this register are identical to the bit definitions for CRA, except that all control actions apply to the channel B receiver and transmitter and the corresponding inputs and outputs.

SRA — Channel A Status Register

SRA[7] — Channel A Received Break — This bit indicates that an all zero character of the programmed length has been received without a stop bit. Only a single FIFO position is occupied when a break is received: further entries to the FIFO are inhibited until the RxDA line returns to the marking state for at least one-half a bit time (two successive edges of the internal or external 1x clock). When this bit is set, the channel A 'change in break' bit in the ISR (ISR[2]) is set. ISR[2] is also set when the end of the break condition, as defined above, is detected.

The break detect circuitry can detect breaks that originate in the middle of a received character. However, if a break begins in the middle of a character, it must persist until at least the end of the next character time in order for it to be detected.

SRA[6] — Channel A Framing Error — This bit, when set, indicates that a stop bit was not detected when the corresponding data character in the FIFO was received. The stop bit check is made in the middle of the first stop bit position.

SRA[5] — Channel A Parity Error — This bit is set when the 'with parity' or 'force parity' mode is programmed and the corresponding character in the FIFO was received with incorrect parity.

In the special multidrop mode the parity error bit stores the received A/D bit.

SRA[4] — Channel A Overrun Error — This bit, when set, indicates that one or more characters in the received data stream have been lost. It is set upon receipt of a new character when the FIFO is full and a character is already in the receive shift register waiting for an empty FIFO position. When this occurs, the character in the receive shift register (and its break detect, parity error and framing error status, if any) is lost.

This bit is cleared by a 'reset error status' command.

SRA[3] — Channel A Transmitter Empty (TxEMTA) — This bit will be set when the channel A transmitter underruns, i.e., both the transmit holding register (THR) and the transmit shift register are empty. It is set after transmission of the last stop bit of a character if no character is in the THR awaiting transmission. It is reset when the THR is loaded by the CPU or when the transmitter is disabled.

SRA[2] — Channel A Transmitter Ready (TxRDYA) — This bit, when set, indicates that the THR is empty and ready to be loaded with a character. This bit is cleared when the THR is loaded by the CPU and is set when the character is transferred to the transmit shift register. TxRDY is reset when the transmitter is disabled and is set when the transmitter is first enabled, viz., characters loaded into the THR while the transmitter is disabled will not be transmitted.

Preliminary

SRA[1] — Channel A FIFO Full (FFULLA) — This bit is set when a character is transferred from the receive shift register to the receive FIFO and the transfer causes the FIFO to become full, i.e., all three FIFO positions are occupied. It is reset when the CPU reads the RHR. If a character is waiting in the receive shift register because the FIFO is full, FFULL will not be reset when the CPU reads the RHR.

SRA[0] — Channel A Receiver Ready (RxRDYA) — This bit indicates that a character has been received and is waiting in the FIFO to be read by the CPU. It is set when the character is transferred from the receive shift register to the FIFO and reset when the CPU reads the RHR, if after this read there are no more characters still in the FIFO.

SRB — Channel B Status Register

The bit definitions for this register are identical to the bit definitions for SRA, except that all status applies to the channel B receiver and transmitter and the corresponding inputs and outputs.

OPCR — Output Port Configuration Register

OPCR[7] — OP7 Output Select — This bit programs the OP7 output to provide one of the following:

- The complement of OPR[7]
- The channel B transmitter interrupt output, which is the complement of TxRDYB. When in this mode OP7 acts as an open collector output. Note that this output is not masked by the contents of the IMR.

OPCR[6] — OP6 Output Select — This bit programs the OP6 output to provide one of the followng:

- The complement of OPR[6]
- The channel A transmitter interrupt output, which is the complement of TxRDYA. When in this mode OP6 acts as an open collector output. Note that this output is not masked by the contents of the IMR.

OPCR[5] — OP5 Output Select — This bit programs the OP5 output to provide one of the following:

- The complement of OPR[5]
- The channel B receiver interrupt output, which is the complement of ISR[5]. When in this mode OP5 acts as an open collector output. Note that this output is not masked by the contents of the IMR.

OPCR[4] — OP4 Output Select — This bit programs the OP4 output to provide one of the following:

- The complement of OPR[4]
- The channel A receiver interrupt output, which is the complement of ISR[1]. When in this mode OP4 acts as an open collector output. Note that this output is not masked by the contents of the IMR.

OPCR[3:2] — OP3 Output Select — This field programs the OP3 output to provide one of the following:

- The complement of OPR[3]
- The counter/timer output, in which case OP3 acts as an open collector output. In the timer mode, this output is a square wave at the programmed frequency. In the counter mode, the output remains high until terminal count is reached, at which time it goes low. The output returns to the high state when the counter is stopped by a stop counter command. Note that this output is not masked by the contents of the IMR.
- The 1X clock for the channel B transmitter, which is the clock that shifts the transmitted data. If data is not being transmitted, a free running 1X clock is output.
- The 1X clock for the channel B receiver, which is the clock that samples the received data. If data is not being received, a free running 1X clock is output.

OPCR[1:0] — OP2 Output Select — This field programs the OP2 output to provide one of the following:

- The complement of OPR[2]
- The 16X clock for the channel A transmitter. This is the clock selected by CSRA[3:0], and will be a 1X clock if CSRA[3:0] = 1111.
- The 1X clock for the channel A transmitter, which is the clock that shifts the transmitted data. If data is not being transmitted, a free running 1X clock is output.
- The 1X clock for the channel A receiver, which is the clock that samples the received data. If data is not being received, a free running 1X clock is output.

ACR — Auxiliary Control Register

ACR[7] — Baud Rate Generator Set Select — This bit selects one of two sets of baud rates to be generated by the BRG:

Set 1: 50, 110, 134.5, 200, 300, 600, 1.05K, 1.2K, 2.4K, 4.8K, 7.2K, 9.6K, and 38.4K baud.

Set 2: 75, 110, 134.5, 150, 300, 600, 1.2K, 1.8K, 2.0K, 2.4K, 4.8K, 9.6K, and 19.2K baud.

The selected set of rates is available for use by the channel A and B receivers and transmitters as described in CSRA and CSRB. Baud rate generator characteristics are given in table 3.

Table 3 BAUD RATE GENERATOR CHARACTERISTICS CRYSTAL OR CLOCK = 3.6864MHz

NOMINAL RATE (BAUD)	ACTUAL 16X CLOCK (KHz)	ERROR (PERCENT)
50	0.8	0
75	1.2	0
110	1.759	-0.069
134.5	2.153	0.059
150	2.4	0
200	3.2	0
300	4.8	0
600	9.6	0
1050	16.756	-0.260
1200	19.2	0
1800	28.8	0
2000	32.056	0.175
2400	38.4	0
4800	76.8	0
7200	115.2	0
9600	153.6	0
19.2K	307.2	0
38.4K	614.4	0

NOTE
Duty cycle of 16X clock is 50% ± 1%.

Preliminary

ACR[6:4]—Counter/Timer Mode and Clock Source Select — This field selects the operating mode of the counter/timer and its clock source as shown in table 4.

ACR[3:0] — IP3, IP2, IP1, IPO Change of State Interrupt Enable — This field selects which bits of the Input Port Change register (IPCR) cause the input change bit in the interrupt status register (ISR[7]) to be set. If a bit is in the 'on' state, the setting of the corresponding bit in the IPCR will also result in the setting of ISR[7], which results in the generation of an interrupt output if IMR[7] = 1. If a bit is in the 'off' state, the setting of that bit in the IPCR has no effect on ISR[7].

IPCR — Input Port Change Register

IPCR[7:4] — IP3, IP2, IP1, IP0 Change of State — These bits are set when a change of state, as defined in the Input Port section of this data sheet, occurs at the respective input pins. They are cleared when the IPCR is read by the CPU. A read of the IPCR also clears ISR[7], the input change bit in the interrupt status register.

The setting of these bits can be programmed to generate an interrupt to the CPU.

IPCR[3:0] — IP3, IP2, IP1, IP0 Current State — These bits provide the current state of the respective inputs. The information is unlatched and reflects the state of the input pins at the time the IPCR is read.

ISR — Interrupt Status Register

This register provides the status of all potential interrupt sources. The contents of this register are masked by the interrupt mask register (IMR). If a bit in the ISR is a '1' and the corresponding bit in the IMR is also a '1', the INTRN output will be asserted. If the corresponding bit in the IMR is a zero, the state of the bit in the ISR has no effect on the INTRN output. Note that the IMR does not mask the reading of the ISR — the true status will be provided regardless of the contents of the IMR. The contents of this register are initialized to 00_{16} when the DUART is reset.

ISR[7] — Input Port Change Status — This bit is a '1' when a change of state has occurred at the IP0, IP1, IP2, or IP3 inputs and that event has been selected to cause an interrupt by the programming of ACR[3:0]. The bit is cleared when the CPU reads the IPCR.

Table 4 ACR [6:4] FIELD DEFINITION

ACR[6:4]	MODE	CLOCK SOURCE
0 0 0	Counter	External (IP2)
0 0 1	Counter	TXCA — 1X clock of channel A transmitter
0 1 0	Counter	TXCB — 1X clock of channel B transmitter
0 1 1	Counter	Crystal or external clock (X1/CLK) divided by 16
1 0 0	Timer	External (IP2)
1 0 1	Timer	External (IP2) divided by 16
1 1 0	Timer	Crystal or external clock (X1/CLK)
1 1 1	Timer	Crystal or external clock (X1/CLK) divided by 16

ISR[6] — Channel B Change in Break — This bit, when set, indicates that the channel B receiver has detected the beginning or the end of a received break. It is reset when the CPU issues a channel B 'reset break change interrupt' command.

ISR[5] — Channel B Receiver Ready or FIFO Full — The function of this bit is programmed by MR1B[6]. If programmed as receiver ready, it indicates that a character has been received in channel B and is waiting in the FIFO to be read by the CPU. It is set when the character is transferred from the receive shift register to the FIFO and reset when the CPU reads the RHR. If after this read there are more characters still in the FIFO the bit will be set again after the FIFO is 'popped'. If programmed as FIFO full, it is set when a character is transferred from the receive holding register to the receive FIFO and the transfer causes the channel B FIFO to become full, i.e., all three FIFO positions are occupied. It is reset when the CPU reads the RHR. If a character is waiting in the receive shift register because the FIFO is full, the bit will be set again when the waiting character is loaded into the FIFO.

ISR[4] — Channel B Transmitter Ready — This bit is a duplicate of TxRDYB (SRB[2]).

ISR[3] — Counter Ready — In the counter mode, this bit is set when the counter reaches terminal count and is reset when the counter is stopped by a stop counter command.

In the timer mode, this bit is set once each cycle of the generated square wave (every other time that the counter/timer reaches zero count). The bit is reset by a stop counter command. The command, however, does not stop the counter/timer.

ISR[2] — Channel A Change in Break — This bit, when set, indicates that the channel A receiver has detected the beginning or the end of a received break. It is reset when the CPU issues a channel A 'reset break change interrupt' command.

ISR[1] — Channel A Receiver Ready or FIFO Full — The function of this bit is programmed by MR1A[6]. If programmed as receiver ready, it indicates that a character has been received in channel A and is waiting in the FIFO to be read by the CPU. It is set when the character is transferred from the receive shift register to the FIFO and reset when the CPU reads the RHR. If after this read there are more characters still in the FIFO the bit will be set again after the FIFO is 'popped'. If programmed as FIFO full, it is set when a character is transferred from the receive holding register to the receive FIFO and the transfer causes the channel A FIFO to become full, i.e., all three FIFO positions are occupied. It is reset when the CPU reads the RHR. If a character is waiting in the receive shift register because the FIFO is full, the bit will be set again when the waiting character is loaded into the FIFO.

ISR[0] — Channel A Transmitter Ready — This bit is a duplicate of TxRDYA (SRA[2]).

IMR — Interrupt Mask Register

The programming of this register selects which bits in the ISR cause an interrupt output. If a bit in the ISR is a '1' and the corresponding bit in the IMR is also a '1', the INTRN output will be asserted. If the corresponding bit in the IMR is a zero, the state of the bit in the ISR has no effect on the INTRN output. Note that the IMR does not mask the programmable interrupt outputs OP3-OP7 or the reading of the ISR.

Preliminary

CTUR and CTLR — Counter/Timer Registers

The CTUR and CTLR hold the eight MSBs and eight LSBs respectively of the value to be used by the counter/timer in either the counter or timer modes of operation. The minimum value which may be loaded into the CTUR/CTLR registers is 0002_{16}. Note that these registers are write-only and cannot be read by the CPU.

In the timer (programmable divider) mode, the C/T generates a square wave with a period of twice the value (in clock periods) of the CTUR and CTLR. If the value in CTUR or CTLR is changed, the current half-period will not be affected, but subsequent half periods will be. In this mode the C/T runs continuously. Receipt of a start counter command (read with A3-A0 = 1110) causes the counter to terminate the current timing cycle and to begin a new cycle using the values in CTUR and CTLR.

The counter ready status bit (ISR[3]) is set once each cycle of the square wave. The bit is reset by a stop counter command (read with A3-A0 = 1111). The command, however, does not stop the C/T. The generated square wave is output on OP3 if it is programmed to be the C/T output.

In the counter mode, the C/T counts down the number of pulses loaded into CTUR and CTLR by the CPU. Counting begins upon receipt of a start counter command. Upon reaching terminal count (0000_{16}), the counter ready interrupt bit (ISR[3]) is set. The counter continues counting past the terminal count until stopped by the CPU. If OP3 is programmed to be the output of the C/T, the output remains high until terminal count is reached, at which time it goes low. The output returns to the high state and ISR[3] is cleared when the counter is stopped by a stop counter command. The CPU may change the values of CTUR and CTLR at any time, but the new count becomes effective only on the next start counter command. If new values have not been loaded, the previous count values are preserved and used for the next count cycle.

In the counter mode, the current value of the upper and lower 8 bits of the counter (CTU, CTL) may be read by the CPU. It is recommended that the counter be stopped when reading to prevent potential problems which may occur if a carry from the lower 8-bits to the upper 8-bits occurs between the times that both halves of the counter are read. However, note that a subsequent start counter command will cause the counter to begin a new count cycle using the values in CTUR and CTLR.

ABSOLUTE MAXIMUM RATINGS[1]

PARAMETER	RATING	UNIT
Operating ambient temperature[2]	0 to +70	°C
Storage temperature	−65 to +150	°C
All voltages with respect to ground[3]	−0.5 to +6.0	V

NOTES:
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or at any other condition above those indicated in the operation section of this specification is not implied.
2. For operating at elevated temperatures, the device must be derated based on +150°C maximum junction temperature.
3. This product includes circuitry specifically designed for the protection of its internal devices from damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying any voltages larger than the rated maxima.

DC ELECTRICAL CHARACTERISTICS $T_A = 0°C$ to $+70°C$, $V_{CC} = 5.0V \pm 5\%$[4,5,6]

PARAMETER		TEST CONDITIONS	LIMITS			UNIT
			Min	Typ	Max	
V_{IL}	Input low voltage				0.8	V
V_{IH}	Input high voltage (except X1/CLK)		2.0			V
V_{IH}	Input high voltage (X1/CLK)		4.0			V
V_{OL}	Output low voltage	$I_{OL} = 2.4mA$			0.4	V
V_{OH}	Output high voltage (except o.c. outputs)	$I_{OH} = -400\mu A$	2.4			V
I_{IL}	Input leakage current	$V_{IN} = 0$ to V_{CC}	−10		10	μA
I_{LL}	Data bus 3-state leakage current	$V_O = 0$ to V_{CC}	−10		10	μA
I_{OC}	Open collector output leakage current	$V_O = 0$ to V_{CC}	−10		10	μA
I_{CC}	Power supply current				150	mA

NOTES:
4. Parameters are valid over specified temperature range.
5. All voltage measurements are referenced to ground (GND). For testing, all input signals swing between 0.4V and 2.4V with a transition time of 20ns maximum. All time measurements are referenced at input voltages of 0.8V and 2.0V and output voltages of 0.8V and 2.0V as appropriate.
6. Typical values are at +25°C, typical supply voltages, and typical processing parameters.

Preliminary

AC ELECTRICAL CHARACTERISTICS $T_A = 0°C$ to $+70°C$, $V_{CC} = 5.0V \pm 5\%$[4,5,6,7]

PARAMETER	TENTATIVE LIMITS			UNIT
	Min	Typ	Max	
Reset Timing (figure 1)				
t_{RES} RESET pulse width	1.0			μs
Bus Timing (figure 2)[8]				
t_{AS} A0-A3 setup time to RDN, WRN low	10			ns
t_{AH} A0-A3 hold time from RDN, WRN high	0			ns
t_{CS} CEN setup time to RDN, WRN low	0			ns
t_{CH} CEN hold time from RDN, WRN high	0			ns
t_{RW} WRN, RDN pulse width	225			ns
t_{DD} Data valid after RDN low			175	ns
t_{DF} Data bus floating after RDN high			100	ns
t_{DS} Data setup time before WRN high	100			ns
t_{DH} Data hold time after WRN high	20			ns
t_{RWD} High time between READs and/or WRITEs[9,10]	200			ns
Port Timing (figure 3)[8]				
t_{PS} Port input setup time before RDN low	0			ns
t_{PH} Port input hold time after RDN high	0			ns
t_{PD} Port output valid after WRN high			400	ns
Interrupt Timing (figure 4)				
t_{IR} INTRN (or OP3-OP7 when used as interrupts) high from:				
Read RHR (RXRDY/FFULL interrupt)			300	ns
Write THR (TXRDY interrupt)			300	ns
Reset command (delta break interrupt)			300	ns
Stop C/T command (counter interrupt)			300	ns
Read IPCR (input port change interrupt)			300	ns
Write IMR (clear of interrupt mask bit)			300	ns
Clock Timing (figure 5)				
t_{CLK} X1/CLK high or low time	100			ns
f_{CLK} X1/CLK frequency	2.0	3.6864	4.0	MHz
t_{CTC} CTCLK (IP2) high or low time	100			ns
f_{CTC} CTCLK (IP2) frequency	0		4.0	MHz
t_{RX} RxC high or low time	220			ns
f_{RX} RxC frequency (16X)	0		2.0	MHz
(1X)	0		1.0	MHz
t_{TX} TxC high or low time	220			ns
f_{TX} TxC frequency (16X)	0		2.0	MHz
(1X)	0		1.0	MHz
Transmitter Timing (figure 6)				
t_{TXD} TxD output delay from TxC low			350	ns
t_{TCS} TxC output skew from TxD output data	0		150	ns
Receiver Timing (figure 7)				
t_{RXS} RxD data setup time to RXC high	240			ns
t_{RXH} RxD data hold time from RXC high	200			ns

NOTES:

4. Parameters are valid over specified temperature range.
5. All voltage measurements are referenced to ground (GND). For testing, all input signals swing between 0.4V and 2.4V with a transition time of 20ns maximum. All time measurements are referenced at input voltages of 0.8V and 2.0V and output voltages of 0.8V and 2.0V as appropriate.
6. Typical values are at +25°C, typical supply voltages, and typical processing parameters.
7. Test condition for outputs: $C_L = 150pF$, except interrupt outputs. Test condition for interrupt outputs: $C_L = 50pF$, $R_L = 2.7K$ ohm to V_{CC}.
8. Timing is illustrated and referenced to the WRN and RDN inputs. The device may also be operated with CEN as the 'strobing' input. In this case, all timing specifications apply referenced to the falling and rising edges of CEN.
9. If CEN is used as the 'strobing' input, this parameter defines the minimum high time between one CEN and the next.
10. Consecutive write operations to the same command register require at least three edges of the X1 clock between writes.

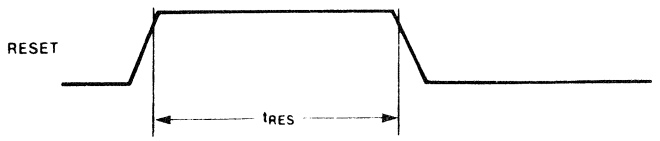


Figure 1. Reset Timing

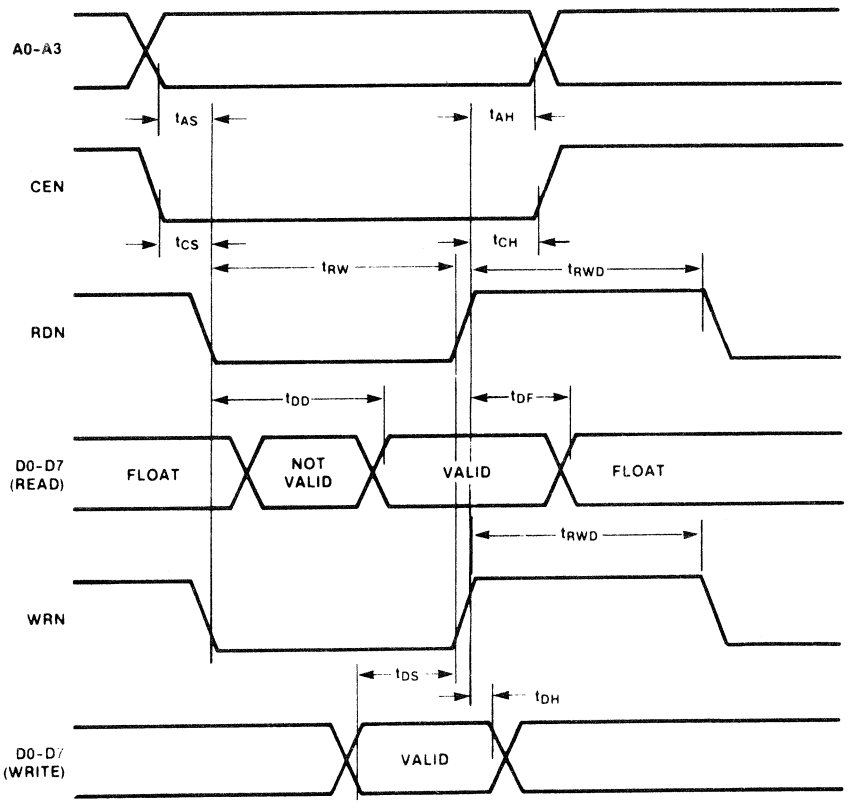


Figure 2. Bus Timing

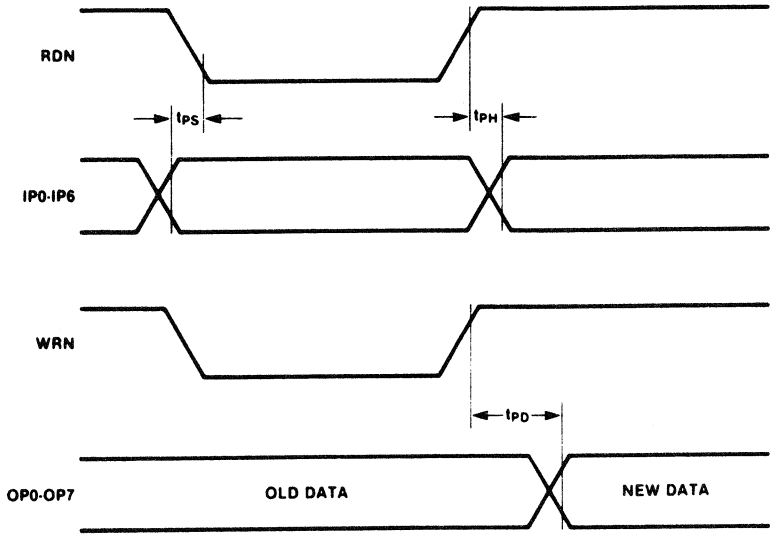


Figure 3. Port Timing

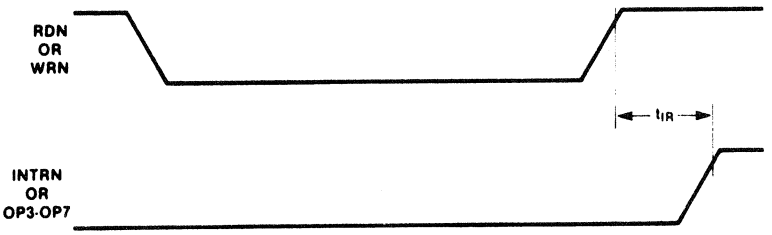


Figure 4. Interrupt Timing

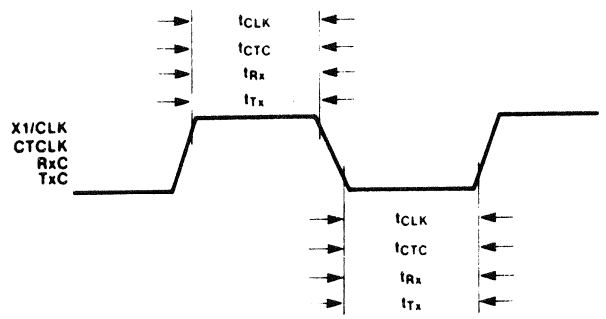
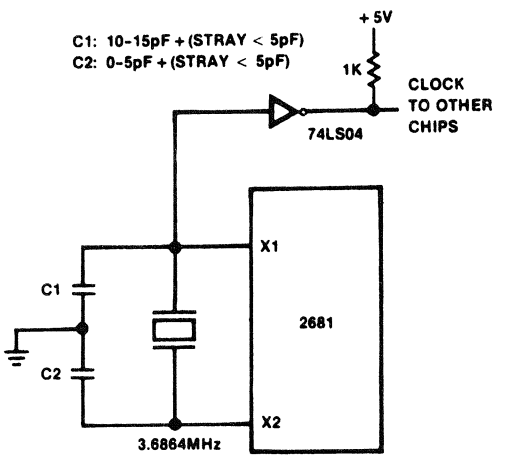


Figure 5. Clock Timing



CRYSTAL SERIES RESISTANCE SHOULD BE LESS THAN 180 OHMS.

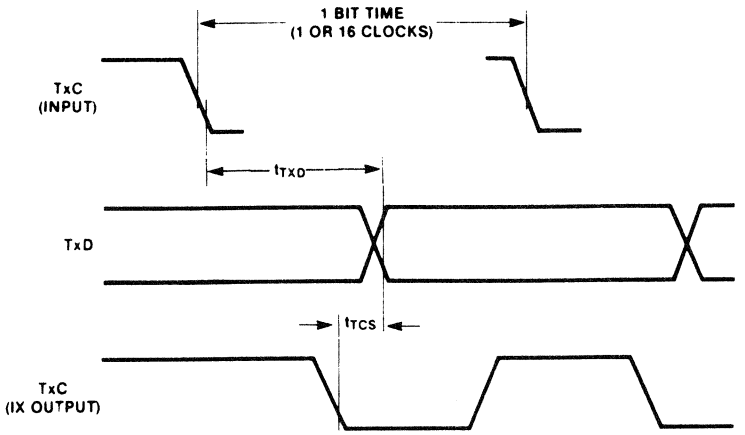


Figure 6. Transmit

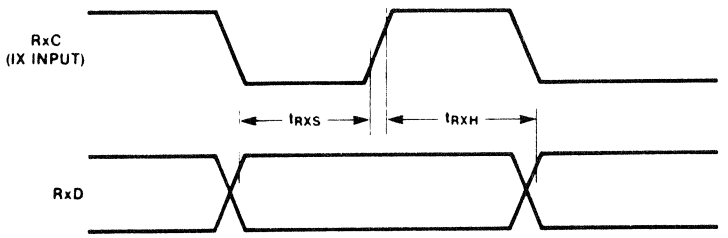
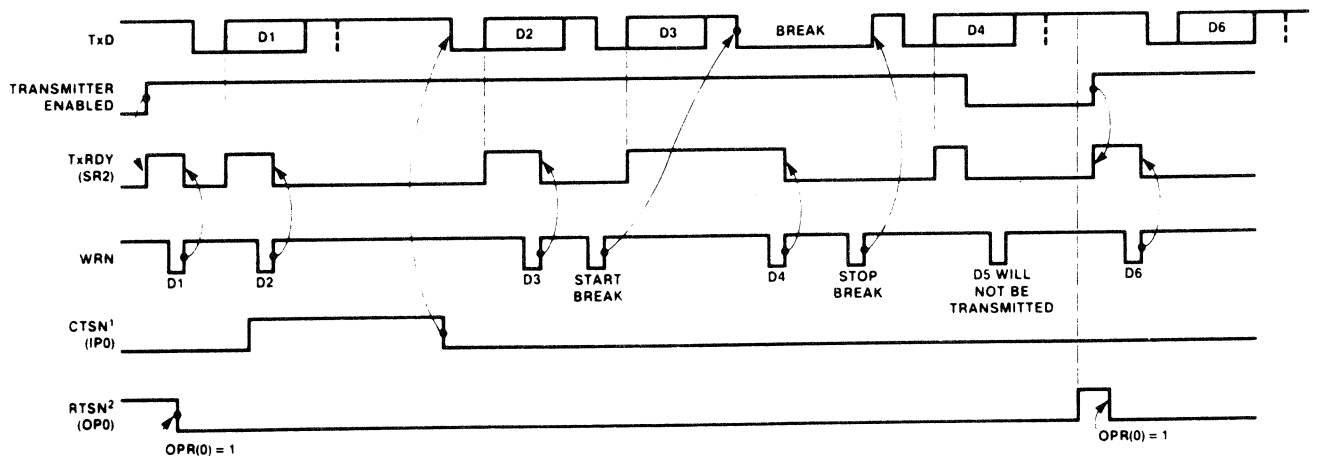
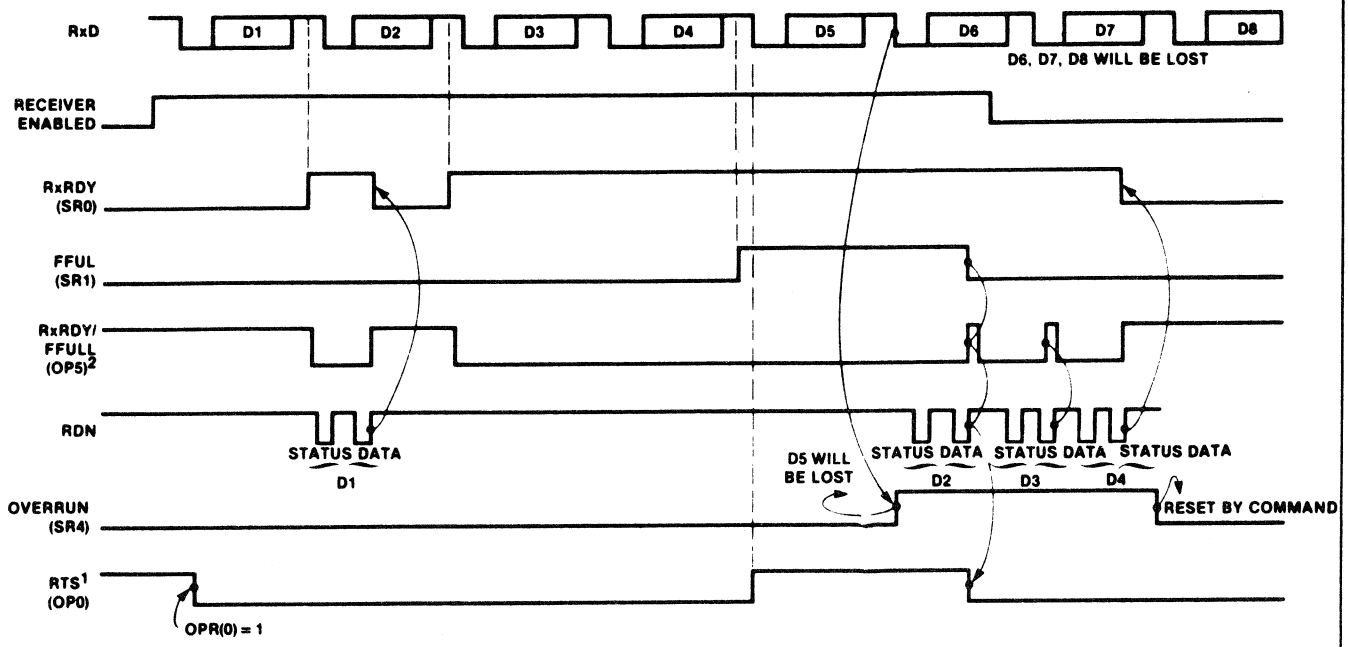


Figure 7. Receive



NOTES
1 TIMING SHOWN FOR MR2(4) = 1
2 TIMING SHOWN FOR MR2(5) = 1

Figure 8. Transmitter Timing



NOTES
1. TIMING SHOWN FOR MR1(7) = 1.
2. SHOWN FOR OPCR(4) = 1 AND MR(6) = 0.

Figure 9. Receiver Timing

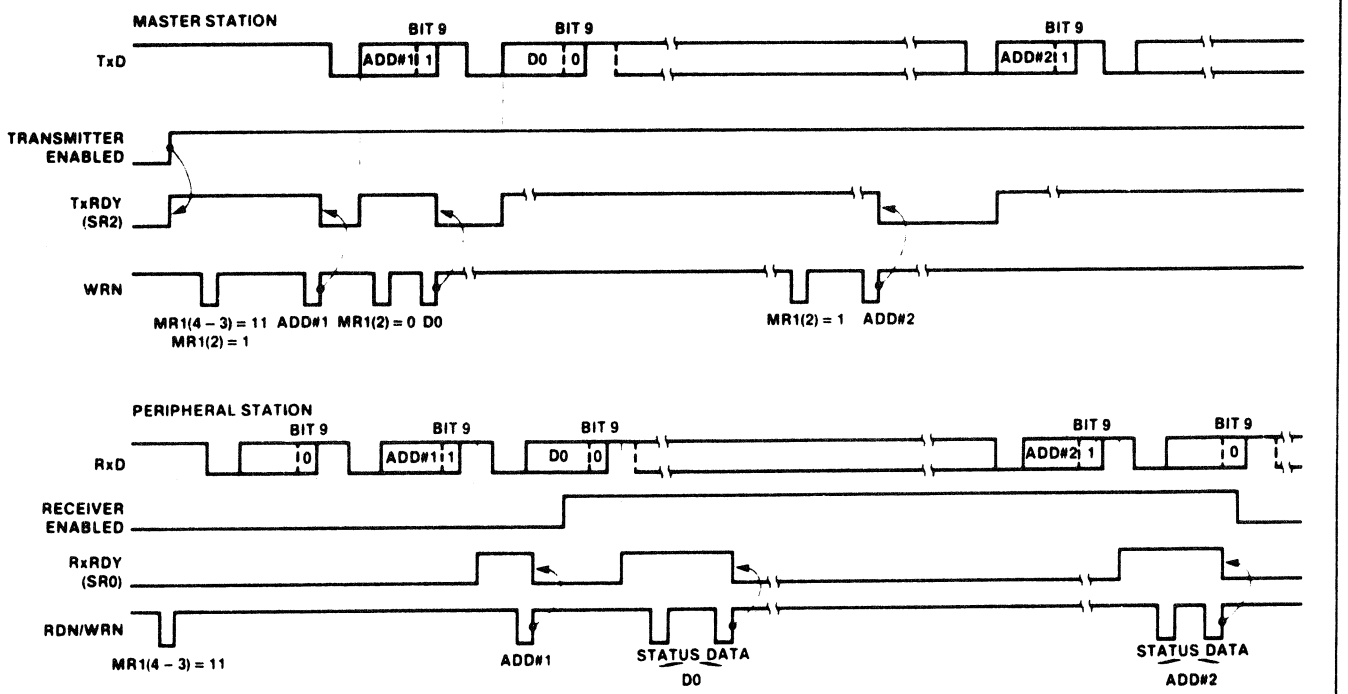


Figure 10. Wake Up Mode

Signetics Corporation
811 East Arques Avenue
P.O. Box 409
Sunnyvale, California 94086
Telephone 408/739-7700