Synertek.

SY6530

Memory, I/O, Timer Array

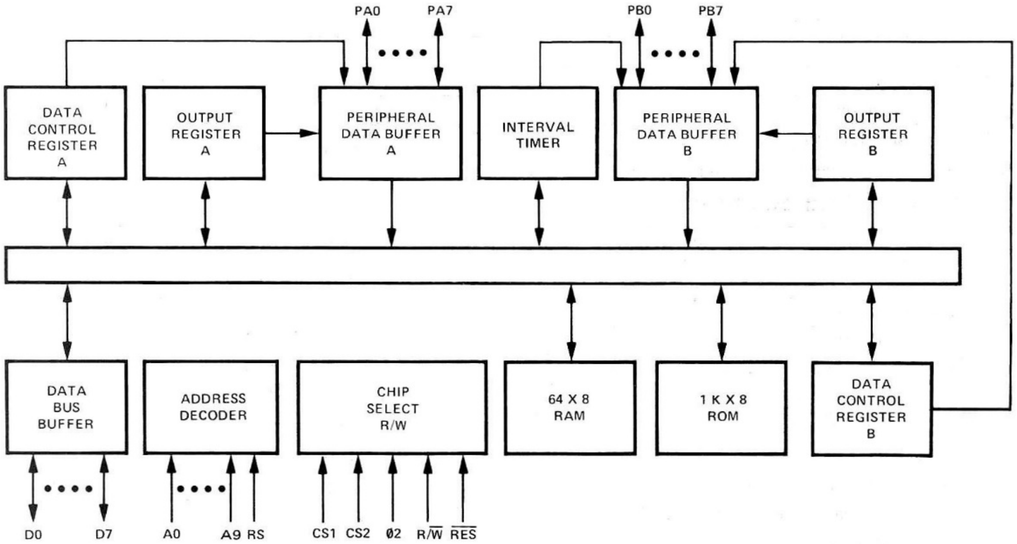
Features

- 8 bit bi-directional Data Bus for direct communication with the microprocessor
- 1024 x 8 ROM
- 64 x 8 static RAM
- Two 8 bit bi-directional data ports for interface to peripherals
- Two programmable I/O Peripheral Data Direction Registers
- Programmable Interval Timer
- Programmable Interval Timer Interrupt
- TTL & CMOS compatible peripheral lines
- Peripheral pins with Direct Transistor Drive Capability
- High Impedance Three-State Data Pins
- Allows up to 7K contiguous bytes of ROM with no external decoding

Description

The SY6530 is designed to operate in conjunction with the SY6500 microprocessor Family. It is comprised of a mask programmable 1024 x 8 ROM, a 64 x 8 static RAM, two software controlled 8 bit bi-directional data ports allowing direct interfacing between the microprocessor unit and peripheral devices, and a software programmable interval timer with interrupt, capable of timing in intervals from 1 to 262,144 clock periods.

Figure 1. SY6530 Block Diagram



Input/Output Voltage (V_{IN}) . . . . -.3 to +7.0V
Operating Temperature (T_{OP}) . . . . 0 to 70°C
Storage Temperature Range (T_{STG}). . . . . -55 to +150°C

Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

D.C. Characteristics

(V_{CC} = 5.0V ±5%, V_{SS} = 0V, T_A = 0°C to 70°C)

	Symbol	Min.	Typ.	Max.	Unit
Input High Voltage	V_{IH}	2.4		V_{CC}	V
Input Low Voltage	V_{IL}	-0.3		0.4	V
Input Leakage Current; V_{IN} = V_{SS} +5V AØ-A9, RS, R/$\overline{W}$, $\overline{RES}$, Ø2, PB6*, PB5*	I_{IN}		1.0	2.5	μA
Input Leakage Current for High Impedence State (Three State); V_{IN} = .4V to 2.4V; DØ-D7	I_{TSI}		±1.0	±10.0	μA
Input High Current; V_{IN} = 2.4V PAØ-PA7, PBØ-PB7	I_{IH}	-100.	-300.		μA
Low Input Current; V_{IN} = .4V PAØ-PA7, PBØ-PB7	I_{IL}		1.0	1.6	mA
Output High Voltage V_{CC} = MIN, I_{LOAD} ≤ -100μA (PAØ-PA7, PBØ-PB7, DØ-D7) I_{LOAD} ≤ -3mA (PAØ, PBØ)	V_{OH}	 2.4 1.5			V
Output Low Voltage V_{CC} = MIN, I_{LOAD} ≤ 1.6mA	V_{OL}			0.4	V
Output High Current (Sourcing); V_{OH} ≥ 2.4V (PAØ-PA7, PBØ-PB7, DØ-D7) ≥ 1.5V Available for other than TTL (Darlingtons) (PAØ, PBØ)	I_{OH}	-100 -3.0	-1000 -5.0		μA mA
Output Low Current (Sinking); V_{OL} ≤ .4V	I_{OL}	1.6			mA
Clock Input Capacitance	C_{CLK}			30	pF
Input Capacitance	C_{IN}			10	pF
Output Capacitance	C_{OUT}			10	pF
Power Dissipation (V_{CC}= 5.25V)	P_D			700	mW

*When Programmed as address pins   All values are D.C. readings

Write Timing Characteristics

Characteristic	Symbol	Min.	Typ.	Max.	Unit
Clock Period	T_{CYC}	1		10	μs
Rise & Fall Times	T_R, T_F			25	ns
Clock Pulse Width	T_C	470			ns
R/$\overline{W}$ valid before positive transition of clock	T_{WCW}	180			ns
Address valid before positive transition of clock	T_{ACW}	180			ns
Data bus valid before negative transition of clock	T_{DCW}	300			ns
Data Bus Hold Time	T_{HW}	10			ns
Peripheral data valid after negative transition of clock	T_{CPW}			1	μs
Peripheral data valid after negative transition of clock driving CMOS (Level = V_{CC} -30%)	T_{CMOS}			2	μs
R/$\overline{W}$ hold time after negative clock transition	T_{CWW}	0			ns
Address hold time	T_{CAH}	0			ns

Figure 2. Write Timing Characteristics

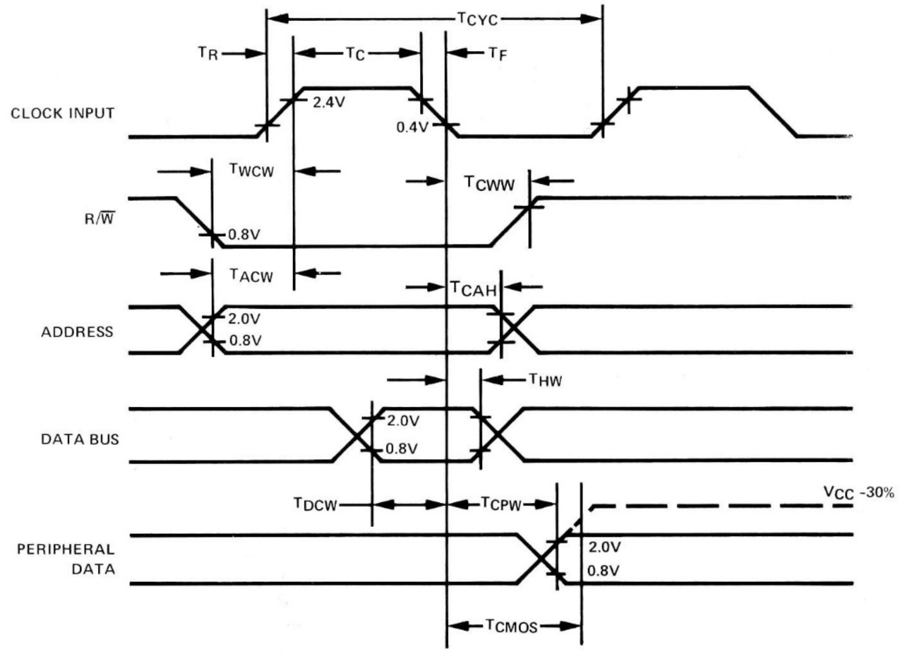
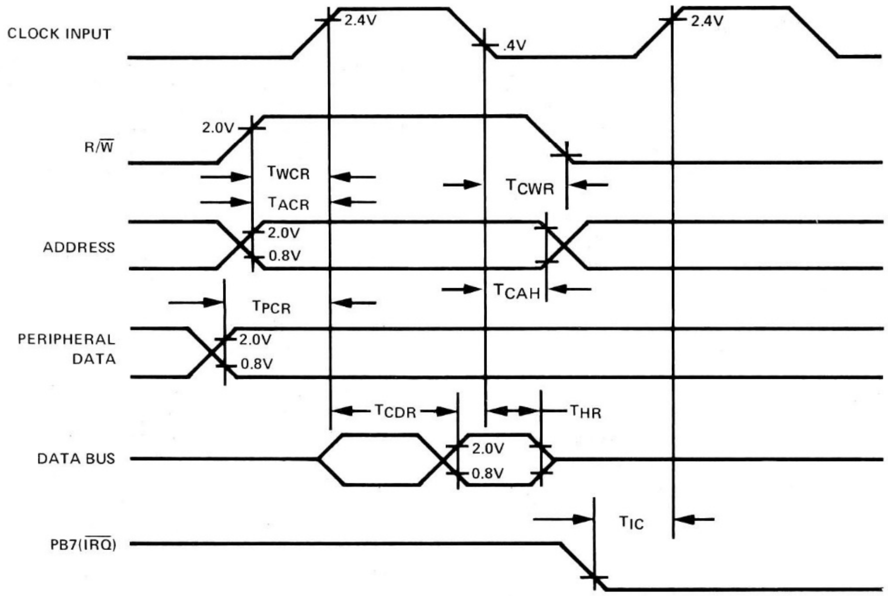
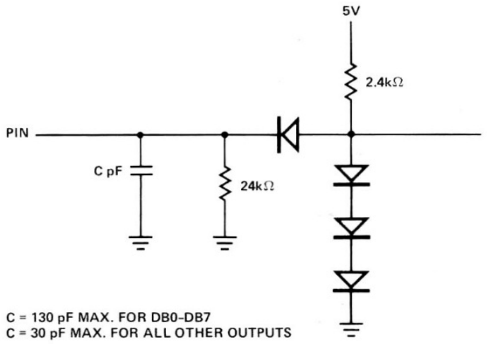


Figure 3. Read Timing Characteristics

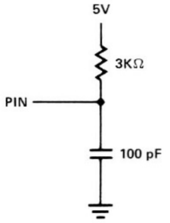


Test Load



C = 130 pF MAX. FOR DB0-DB7
C = 30 pF MAX. FOR ALL OTHER OUTPUTS

OPEN COLLECTOR
OUTPUT TEST LOAD



Read Timing Characteristics

Characteristic	Symbol	Min.	Typ.	Max.	Unit
R/$\overline{W}$ valid before positive transition of clock	T_{WCR}	180			ns
Address valid before positive transition of clock	T_{ACR}	180			ns
Peripheral data valid before positive transition of clock	T_{PCR}	300			ns
Data bus valid after positive transition of clock	T_{CDR}			395	ns
Data Bus Hold Time	T_{HR}	10			ns
$\overline{IRQ}$ (Interval Timer Interrupt) valid before positive transition of clock	T_{IC}	200			ns
R/$\overline{W}$ hold time after negative clock transition	T_{CWR}	0			ns
Address hold time	T_{CAH}	0			ns

Loading = 30 pF + 1 TTL load for PAØ-PA7, PBØ-PB7
= 130 pF + 1 TTL load for DØ-D7

Interface Signal Description

Reset ($\overline{RES}$)

During system initialization a low (≤0.4V) on the $\overline{RES}$ input will cause a zeroing of all four I/O registers. This in turn will cause all I/O buses to act as inputs, protecting external components from possible damage and erroneous data while the system is being configured under software control. The Data Bus Buffers are put into an OFF-STATE during reset. Interrupt capability is disabled with the $\overline{RES}$ signal. The $\overline{RES}$ signal must be held low for at least one clock period when reset is required.

Input Clock (ϕ 2)

The input clock is a system Phase Two clock.

Read/Write (R/$\overline{W}$)

R/$\overline{W}$ is supplied by the microprocessor and is used to control the transfer of data to and from the SY6530. A high on the R/$\overline{W}$ pin allows the processor to read (with proper addressing) the SY6530. A low on the R/$\overline{W}$ pin allows a write (with proper addressing) to the SY6530.

Interrupt Request ($\overline{IRQ}$)

The $\overline{IRQ}$ output is derived from the interval timer. The same line, if not used as an interrupt, can be used as a peripheral I/O (PB7). When used as an interrupt, the pin should be set to an input in the data direction register. As $\overline{IRQ}$ the output will be normally high with a low indicating an interrupt from the SY6530. An external pull-up device is not required; however, if collector-OR'd with other devices, the internal pull-up may be omitted with a mask option.

Data Bus (D0-D7)

The SY6530 has eight bi-directional data lines (D0-D7). These lines connect to the system's data bus and allow transfer of data to and from the microprocessor. The output buffers remain in the off state except when a Read operation occurs.

Peripheral Data Ports (PA0-PA7, PB0-PB7)

The SY6530 has two 8-bit peripheral I/O ports, Port A (lines PA0-PA7) and Port B (lines PB0-PB7). Each line is individually software programmable as either an input or an output. By writing a "0" to any bit position of the Data Direction Register (DDRA or DDRB) the corresponding line will be programmed as an input. Likewise, by writing "1" to any bit position in the DDR will cause the corresponding line to act as an output.

When the Ports are programmed as inputs and their output registers (ORA and ORB) are read by the MPU, the level on the port lines will be transferred to the Data Bus. When the ports are programmed as outputs the lines will reflect the data written by the MPU into the output registers.

PA0 and PB0 are capable of direct transistor drive (source 3mA at 1.5V).

Address and Select Lines (A0-A9, RS, PB5 and PB6)
A0-A9 and ROM SELECT (RS) are always used as addressing lines. There are 2 additional lines which are mask programmable and can be used either individually or together as CHIP SELECTS. They are PB5 and PB6. When used as peripheral data lines they cannot be used as chip selects.

Internal Organization

A block diagram of the internal architecture is shown in Figure 1. The SY6530 is divided into four basic sections, RAM, ROM, I/O and TIMER. The RAM and ROM interface directly with the microprocessor through the system data bus and address lines. The I/O section consists of two 8-bit halves. Each half contains a Data Direction Register (DDR) and an I/O Register.

ROM 1 K Byte (8 K Bits)
The 8K ROM is in a 1024 x 8 configuration. Address lines A0-A9, as well as RS are needed to address the entire ROM. With the addition of CS1 and CS2, seven SY6530's may be addressed, giving 7168 x 8 bits of contiguous ROM.

RAM-64 Bytes (512 Bits)
A 64 x 8 static RAM is contained on the SY6530. It is addressed by A0-A5 (Byte Select), RS, A6, A7, A8, A9, and, depending on the number of chips in the system, CS1 and CS2.

Internal Peripheral Registers
There are four 8-bit internal registers, two data direction registers (DDRA and DDRB) and two peripheral I/O data registers (ORA and ORB). The two data direction registers control the direction of the data into and out of the peripheral line. A "1" written into the Data Direction Register sets up the corresponding peripheral buffer line as an output. Therefore, anything then written into the I/O Register will appear on that corresponding peripheral pin. A "0" written into the DDR inhibits the output buffer from transmitting data to or from the I/O Register. For example, a "1" loaded into data direction A, position 3, sets up peripheral line PA3 as an output. If a "0" had been loaded, PA3 would be configured as an input and remain in the high state. The two data I/O registers are used to latch data from the Data Bus during a Write operation until the peripheral device can read the data supplied by the microprocessor.

During a read operation by the microprocessor the SY6530 transfers the TTL level on the peripheral data lines to the data bus. For the peripheral data lines which are programmed as outputs the microprocessor will read the corresponding data bits of the I/O Register. The only way the I/O Register data can be changed is by a microprocessor Write operation. The I/O Register is not affected by a Read of the data on the peripheral lines.

Interval Timer
The Timer section of the SY6530 contains three basic parts: preliminary divide down register, programmable 8-bit register and interrupt logic. These are illustrated in Figure 4.

The Interval Timer can be programmed to count up to 256 time intervals. Each time interval can be either 1T, 8T, 64T or 1024T increments, where T is the system clock period. When a full count is reached, the interrupt flag is set to a logic "1". After the interrupt flag is set the internal clock continues counting down at a 1T rate to a maximum of -255T. This allows the user to read the counter and then determine how long the interrupt has been set.

The 8-bit system Data Bus is used to transfer data to and from the Interval Timer. If a count of 52 time intervals were to be counted, the pattern 0 0 1 1 0 1 0 0 would be put on the Data Bus and written into the Interval Time register.

At the same time that data is being written into the Interval Timer, the counting intervals of 1, 8, 64, 1024T are decoded from address lines A0 and A1. During a Read or Write operation address line A3 controls the interrupt capability of PB7, i.e., A3 = 1 enables $\overline{IRQ}$ on PB7, A3= 0 disables $\overline{IRQ}$ on PB7. When PB7 is used as $\overline{IRQ}$ with the Interval Timer it should be programmed as an input. If PB7 is enabled by A3 and an interrupt occurs PB7 will go low. When the Timer is read prior to the interrupt flag being set, the number of time intervals remaining will be read, i.e., 51, 50, 49, etc.

When the Timer has counted down to 0 0 0 0 0 0 0 0 an interrupt will occur on the next count and the counter will read 1 1 1 1 1 1 1 1. After interrupt, the timer register decrements at a divide by "1" rate of the system clock. If after interrupt, the Timer is read and a value of 1 1 1 0 0 1 1 1 is read, the time since interrupt is 28T. The value read is in two's complement.

Value Read = 1 1 1 0 0 1 0 0
Complement = 0 0 0 1 1 0 1 1
ADD 1 = 0 0 0 1 1 1 0 0 = 28.

Thus, to arrive at the total elapsed time, merely do a two's complement and add to the original time written into the Timer. Again, assume time written as 0 0 1 1 0 1 0 0 (= 52). With a divide by 8, total time to interrupt is (52 x 8) + 1 = 417T. Total elapsed time would be 416T + 28T = 444T, assuming the value read after interrupt was 1 1 1 0 0 1 0 0.

After an interrupt, whenever the Timer is written or read the interrupt is reset. However, the reading of the Timer at the same time the interrupt occurs will not reset the interrupt flag.

Figure 5 illustrates an example of interrupt.

Figure 4. Basic Elements of Interval Timer

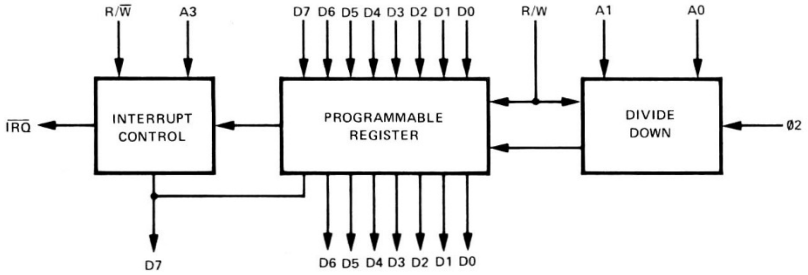
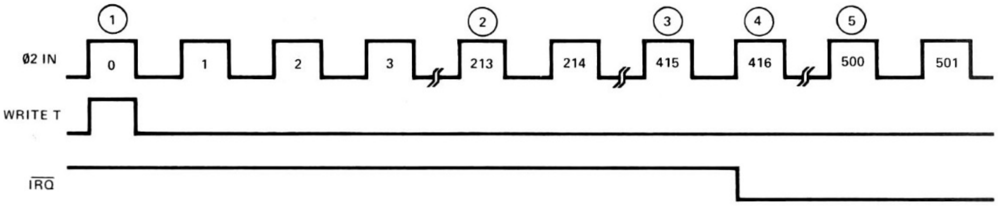


Figure 5. Timer Interrupt Timing



1. Data written into Interval Timer is 0 0 1 1 0 1 0 0 = 52_{10}
2. Data in Interval Timer is 0 0 0 1 1 0 0 1 = 25_{10}

$$52 - \frac{213}{8} - 1 = 52\text{-}26\text{-}1 = 25$$

3. Data in Interval Timer is 0 0 0 0 0 0 0 0 = 0_{10}

$$52 - \frac{415}{8} - 1 = 52\text{-}51\text{-}1 = 0$$

4. Interrupt has occured at ϕ_2 pulse #416
   Data in Interval Timer = 1 1 1 1 1 1 1 1
5. Data in Interval Timer is 1 0 1 0 1 1 0 0
   two's complement is 0 1 0 1 0 1 0 0 = 84_{10}
   84 + (52 x 8) = 500_{10}

When reading the Timer after an interrupt, A3 should be low so as to disable the $\overline{IRQ}$ output. This is done so as to avoid future interrupts until another Write timer operation.

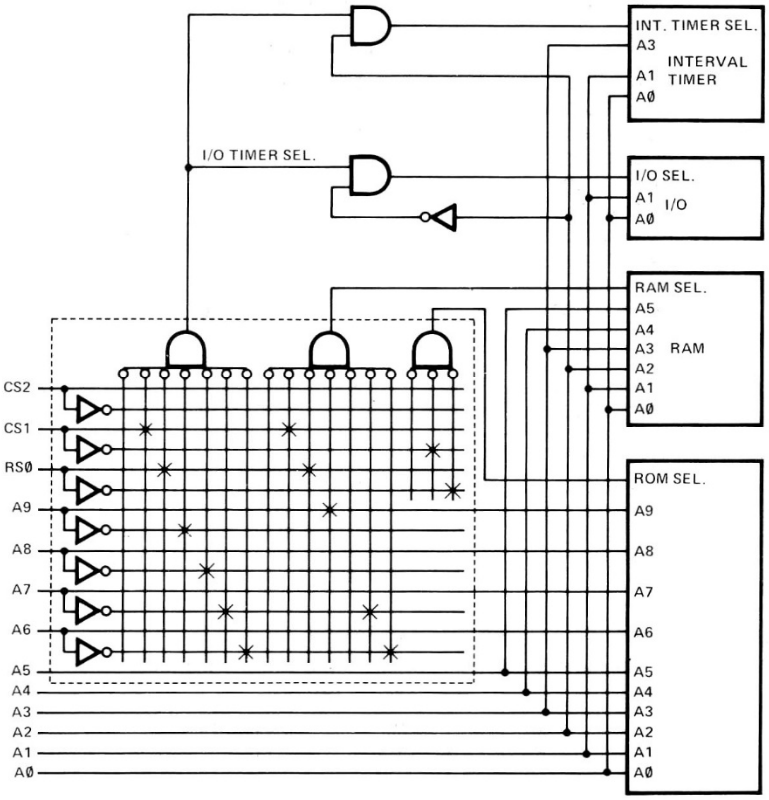
Addressing

Because the address decode matrix is maskable the SY6530 offers many variations to the user. RAM, ROM and the I/O — Interval Timer block may be enabled individually by any combination of A6-A9 plus RS, CS1 and CS2 (refer to Figure 6 for a typical configuration). Because CS1 and CS2 are mask options and act independently neither, either, or both may be masked as Chip Selects or Port B lines.

One-Chip Addressing

Figure 6 illustrates a 1-chip system for the SY6530, and Figure 8 details address decoding.

Figure 6. SY6530 One Chip Address Encoding Scheme



A. X indicates mask programming

i.e. ROM select = CS1•RSO

RAM select = $\overline{\text{CS1}}$•$\overline{\text{RSO}}$•$\overline{\text{A9}}$•A7•A6

I/O TIMER SELECT = $\overline{\text{CS1}}$•$\overline{\text{RSO}}$•A9•A8•A7•A6

B. Notice that A8 is a don't care for RAM select

C. **The CS2/PB5 pin functions as PB5 in this example.**

Seven Chip Addressing

In the 7-chip system the objective would be to have 7K of contiguous ROM, with RAM in low order memory. The 7K of ROM could be placed between addresses 65,535 and 1024. For this case, assume A13, A14, and A15 are all 1 when addressing ROM, and 0 when addressing RAM or I/O. This would place the 7K ROM between addresses 65,535 and 58,367. The 2 lines designated as chip-select or I/O would be mask programmed as chip select. RS would be connected to address line A10. CS1 and CS2 would be connected to address lines A11 and A12 respectively. See Figure 7.

I/O Register — Timer Addressing

The previous two examples have illustrated how to address the ROM, RAM and the general I/O Register — Timer Block. A0 thru A3 specify which of the four I/O registers are selected and select the modes of operation for the Timer. Figure 8 illustrates the internal decoding of these address bits and their function.

Address line A2 selects I/O or Timer. If I/O-Timer Select is enabled and A2 is low the I/O registers are selected and bits A0 and A1 are decoded to select the individual register.

During a write when I/O-Timer Select is enabled and A2 is high the Timer is selected. Bits A0 and A1 select the ÷ by rate (the data lines should at this time have the count value to be written), and A3 determines if PB7 is to act as an $\overline{\text{IRQ}}$ output.

The addressing of the ROM select, RAM select and I/O timer select lines would be as follows:

Figure 7. SY6530 Seven Chip Addressing Scheme

		CS2	CS1	RS				
		A12	A11	A10	A9	A8	A7	A6
SY6530 #1,	ROM SELECT	0	0	1	X	X	X	X
	RAM SELECT	0	0	0	0	0	0	0
	I/O TIMER	0	0	0	1	0	0	0
SY6530 #2,	ROM SELECT	0	1	0	X	X	X	X
	RAM SELECT	0	0	0	0	0	0	1
	I/O TIMER	0	0	0	1	0	0	1
SY6530 #3,	ROM SELECT	0	1	1	X	X	X	X
	RAM SELECT	0	0	0	0	0	1	0
	I/O TIMER	0	0	0	1	0	1	0
SY6530 #4,	ROM SELECT	1	0	0	X	X	X	X
	RAM SELECT	0	0	0	0	0	1	1
	I/O TIMER	0	0	0	1	0	1	1
SY6530 #5,	ROM SELECT	1	0	1	X	X	X	X
	RAM SELECT	0	0	0	0	1	0	0
	I/O TIMER	0	0	0	1	1	0	0
SY6530 #6,	ROM SELECT	1	1	0	X	X	X	X
	RAM SELECT	0	0	0	0	1	0	1
	I/O TIMER	0	0	0	1	1	0	1
SY6530 #7,	ROM SELECT	1	1	1	X	X	X	X
	RAM SELECT	0	0	0	0	1	1	0
	I/O TIMER	0	0	0	1	1	1	0

* RAM select for SY6530 #5 would read = $\overline{A12} \bullet \overline{A11} \bullet \overline{A10} \bullet \overline{A9} \bullet A8 \bullet \overline{A7} \bullet \overline{A6}$

Figure 8. Addressing Decode for I/O Register and Timer

ADDRESSING DECODE

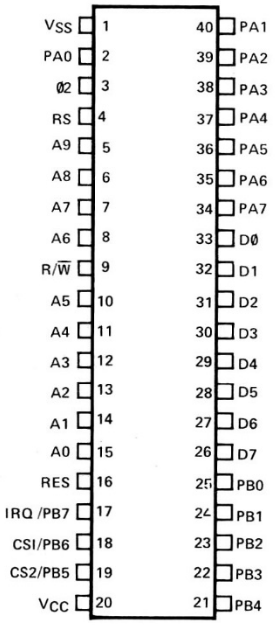
	ROM SELECT	RAM SELECT	I/O-TIMER SELECT	R/$\overline{W}$	A3	A2	A1	A0
READ ROM	1	0	0	1	X	X	X	X
WRITE RAM	0	1	0	0	X	X	X	X
READ RAM	0	1	0	1	X	X	X	X
WRITE DDRA	0	0	1	0	X	0	0	1
READ DDRA	0	0	1	1	X	0	0	1
WRITE DDRB	0	0	1	0	X	0	1	1
READ DDRB	0	0	1	1	X	0	1	1
WRITE ORA	0	0	1	0	X	0	0	0
READ ORA	0	0	1	1	X	0	0	0
WRITE ORB	0	0	1	0	X	0	1	0
READ ORB	0	0	1	1	X	0	1	0
WRITE TIMER								
÷ 1T	0	0	1	0	*	1	0	0
÷ 8T	0	0	1	0	*	1	0	1
÷ 64T	0	0	1	0	*	1	1	0
÷ 1024T	0	0	1	0	*	1	1	1
READ TIMER	0	0	1	1	*	1	X	0
READ INTERRUPT FLAG	0	0	1	1	X	1	X	1

X = Don't care condition

* A_3 = 1 Enables IRQ to PB7

A_3 = 0 Disables IRQ to PB7

Pin Configuration



Programming Instructions

The SY6530 utilizes computer aided techniques to manufacture and test custom ROM patterns. The pattern and address coding is supplied to Synertek in any of several formats.

1) 2708-type EPROMs.
2) Synertek data card formats.
3) Other input formats, providing they can be translated into one of the above.

Synertek Data Card Format

A. The format for the first and all succeeding records, except for the last record in a file is as follows:

;N_1N_0 $A_3A_2A_1A_0$ $(D_1D_0)_1$ $(D_1D_0)_2$ $X_3X_2X_1X_0$

where:

1. All characters (N,A,D,X) are the ASCII characters 0 through F, each representing a hexadecimal digit.
2. ; is a record mark indicating the start of a record.
3. N_1N_0 = the number of bytes of data in this record (in hexadecimal). Each pair of hexadecimal characters (D_1D_0) represents a single byte in the record.
4. $A_3A_2A_1A_0$ = the hexadecimal starting address for the record. A_3 represents address bits 15 through 12, etc. The 8-bit byte represented by $(D_1D_0)_1$ is stored in address $A_3A_2A_1A_0$; $(D_1D_0)_2$ is stored in $(A_3A_2A_1A_0)$ + 1, etc.
5. (D_1D_0) = two hexadecimal digits representing an 8-bit byte of data. (D_1 = high order 4 binary bits and D_0 = low-order 4 bits). A maximum of 18 (Hex) or 24 (decimal) bytes of data per record is permitted.
6. $X_3X_2X_1X_0$ = record check sum. This is the hexadecimal sum of all characters in the record, including N_1N_0 and $A_3A_2A_1A_0$ but excluding the record mark and the check sum characters. To generate the check sum, each byte of data (represented by two ASCII characters), is treated as 8 binary bits. The binary sum of these 8-bit bytes is truncated to 16 binary bits (4 hexadecimal digits) and is then represented in the record as four ASCII characters ($X_3X_2X_1X_0$).

; 00 $C_3C_2C_1C_0$ $X_3X_2X_1X_0$

1. 00 = zero bytes of data in this record. This identifies this as the final record in a file.
2. $C_3C_2C_1C_0$ = the total number of records (in hexadecimal) in this file, including the last record.
3. $X_3X_2X_1X_0$ = check sum for this record.

C. Example

The following example illustrates the exact format of the hex interface file in both listing and punched paper tape form.

```
;18F000CA86004C00F0FDF9212D21FF292DBF2161F5F7FF657D677D0D40
;18F018E564672DFD7575E50000CF4112F800925198D200539192F20C98
;18F03008DB02880810DE12D894189AC2830E9800FBB6232F087F650AA5
;18F048036E20EF2FA58D4465E8FDF93DE775EF257FB520ED64657C0DEB
;18F0607F11D05A1EDF0250B0DAFE009252909912DB108A0298DE080C0D
;18F078D95058DF82D2D79A00ED65E68724EE05212764A5F5BDA9050E2C
;18F090EC20FF652525246933213F20FF31293B7E18D65042DE40500A92
;18F0A81E5E5B02534A53DE4A9B189259969F589E5E92DF52DE9E9A0CA2
;18F0C000B3268D2400EF6765E7A0B5606725217D20AF35EDF5202F0C08
;18F0D8692525342B35256CDF12F2785FFF547FD2E2D6525BDF5A720D26
;10F0F012DB020F1A1ABF86D2DA9ADAC8DECA1B0A12
;00000B000B
```

Additional Pattern Information

In addition to the ROM data patterns, it is necessary to provide the information outlined below.

CUSTOMER NAME
CUSTOMER PART NO.
CUSTOMER CONTACT (NAME)
CUSTOMER TELEPHONE NO.
CS1/PB6 (ENTER "CS1" OR "PB6")
CS2/PB5 (ENTER "CS2" OR "PB5")
PULL-UP RESISTOR ON PB7 ("YES" OR "NO")
LOGIC FORMAT ("POS" OR "NEG")

DEVICE ADDRESSING (Enter "H" for High, "L" for Low, or "N" for don't care)

	RS	CS1	CS2	A9	A8	A7	A6
ROM SELECT				▨	▨	▨	▨
RAM SELECT							
I/O TIMER SELECT							

Send Information To:

Synertek – ROM
P.O. Box 552
Santa Clara, CA 95052

SY6530 Customer Specification Form

1. Date.

2. Customer name.

3. Customer part no.
   (maximum 10 digits)

4. Synertek "C" number.

5. Customer Contact.

6. Customer phone number

7. Chip Select Code
   (Check one square in each block)

CS1	
PB6	

CS2	
PB5	

PULL UP	YES	
ON PB7	NO	

8. ROM/RAM/I-O SELECTS (Specify H or L or N (don't care) in each box.

	RS	CS1	CS2	A9	A8	A7	A6
ROM SELECT				N	N	N	N
RAM SELECT							
I/O SELECT							

9. Customer's Input

   Punched Cards □
   Punched Tape □

10. Data Format

    MOS Technology □
    Intel Hex □
    Intel BPNF □
    Binary □

11. Logic Format

    Positive □
    Negative □

12. Verification Status

    Hold □
    Not Required □