Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science
6.111 - Introductory Digital Systems Laboratory

CPLD Module

Created 09/20/99 by D. Seth
Revised 01/31/01 by D. Troxel

The CPLD module comprises of four interconnected Cypress 374I CPLD's that can be accessed via the kit's NuBus interface and 50-pin connectors. The state of the I/O lines of the NuBus interconnects are displayed on the HEX LED's while the I/O lines of the 50 pin connector extend directly to the inputs of the Logic Analyzer.

The diagram below shows the architecture of the module and its interface to the kit.

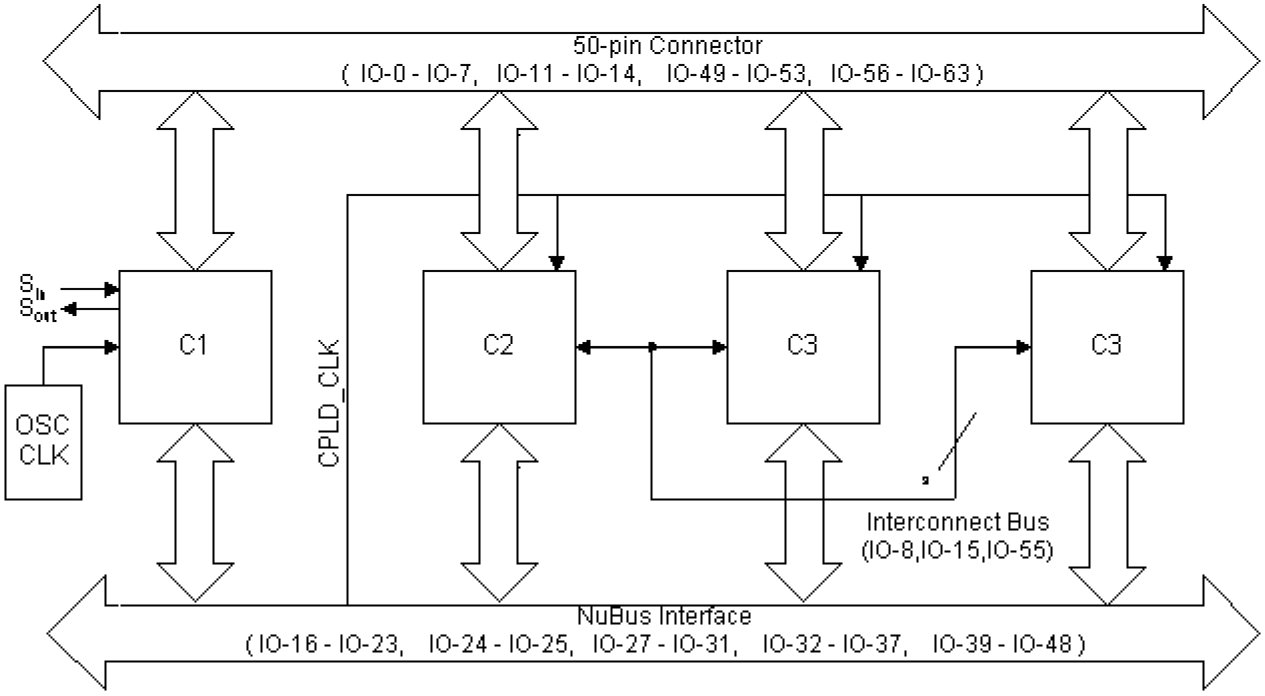
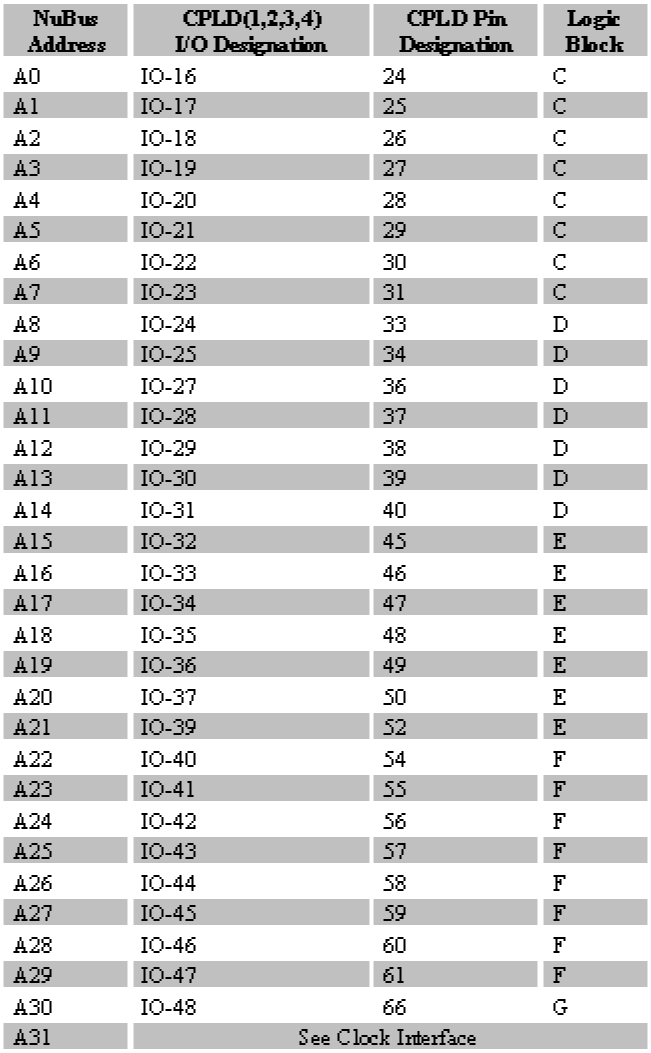

50-pin Connector
(IO-0 - IO-7, IO-11 - IO-14,   IO-49 - IO-53, IO-56 - IO-63)
S_{in}
S_{out}
OSC
CLK
C1
CPLD_CLK
C2
C3
C3
3
Interconnect Bus
(IO-8,IO-15,IO-55)
NuBus Interface
(IO-16 - IO-23,   IO-24 - IO-25, IO-27 - IO-31,   IO-32 - IO-37,   IO-39 - IO-48)

Figure 1: System Diagram

The major components of the module can be categorized into the NuBus Interface, the 50-pin connector Interface, the Interconnect Bus, the Serial Interface, the Programming Interface and the Clocking Scheme.

NuBus Interface

31 I/O pins of each CPLD are interconnected and extended to the NuBus connector. For example, IO-0 of all 4 CPLD's are tied together to NuBus Address A0. Table 1 elaborates these interconnections.

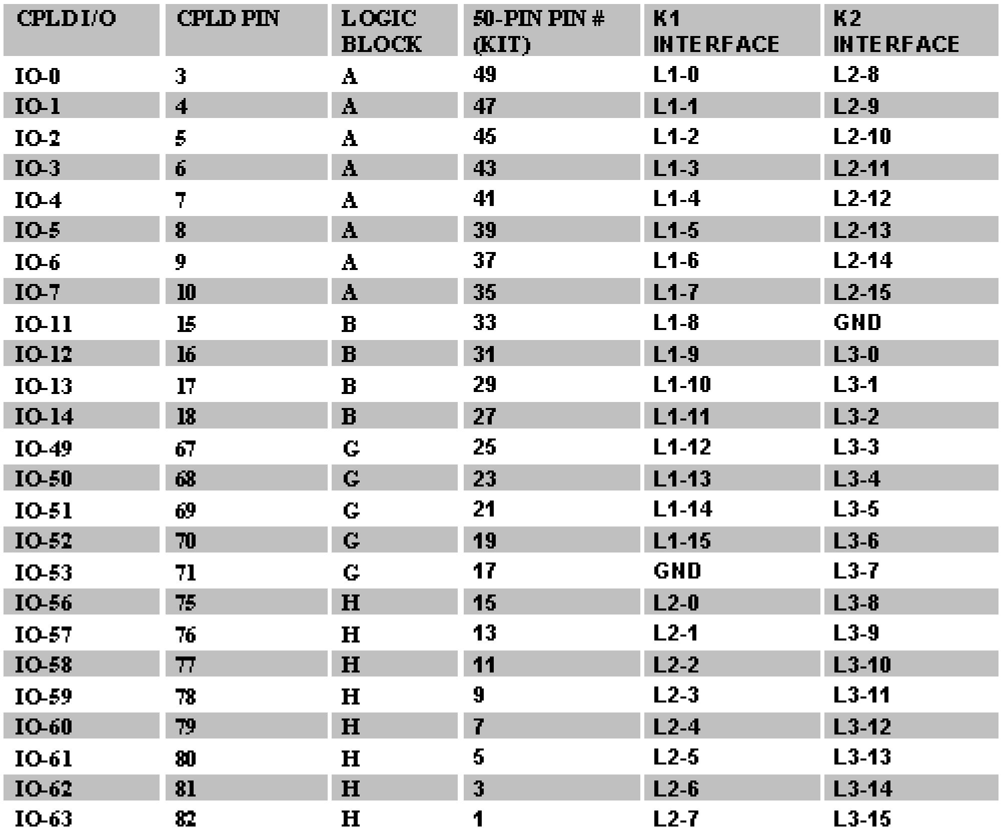


NuBus Address	CPLD(1,2,3,4) I/O Designation	CPLD Pin Designation	Logic Block
A0	IO-16	24	C
A1	IO-17	25	C
A2	IO-18	26	C
A3	IO-19	27	C
A4	IO-20	28	C
A5	IO-21	29	C
A6	IO-22	30	C
A7	IO-23	31	C
A8	IO-24	33	D
A9	IO-25	34	D
A10	IO-27	36	D
A11	IO-28	37	D
A12	IO-29	38	D
A13	IO-30	39	D
A14	IO-31	40	D
A15	IO-32	45	E
A16	IO-33	46	E
A17	IO-34	47	E
A18	IO-35	48	E
A19	IO-36	49	E
A20	IO-37	50	E
A21	IO-39	52	E
A22	IO-40	54	F
A23	IO-41	55	F
A24	IO-42	56	F
A25	IO-43	57	F
A26	IO-44	58	F
A27	IO-45	59	F
A28	IO-46	60	F
A29	IO-47	61	F
A30	IO-48	66	G
A31	See Clock Interface		

Table 1: NuBus Interface

50-Pin Connector Interface

25 I/O pins are interconnected among the 4 CPLD's and the 50-pin connector. The signals of the 50-pin connector interface directly to the logic analyzer via the connector on the kit. Table 2 provides details of these interconnections. Logic Blocks are partitions internal to the CPLD chip. This information, at times, can be useful during device fitting.



CPLD I/O	CPLD PIN	LOGIC BLOCK	50-PIN PIN # (KIT)	K1 INTERFACE	K2 INTERFACE
IO-0	3	A	49	L1-0	L2-8
IO-1	4	A	47	L1-1	L2-9
IO-2	5	A	45	L1-2	L2-10
IO-3	6	A	43	L1-3	L2-11
IO-4	7	A	41	L1-4	L2-12
IO-5	8	A	39	L1-5	L2-13
IO-6	9	A	37	L1-6	L2-14
IO-7	10	A	35	L1-7	L2-15
IO-11	15	B	33	L1-8	GND
IO-12	16	B	31	L1-9	L3-0
IO-13	17	B	29	L1-10	L3-1
IO-14	18	B	27	L1-11	L3-2
IO-49	67	G	25	L1-12	L3-3
IO-50	68	G	23	L1-13	L3-4
IO-51	69	G	21	L1-14	L3-5
IO-52	70	G	19	L1-15	L3-6
IO-53	71	G	17	GND	L3-7
IO-56	75	H	15	L2-0	L3-8
IO-57	76	H	13	L2-1	L3-9
IO-58	77	H	11	L2-2	L3-10
IO-59	78	H	9	L2-3	L3-11
IO-60	79	H	7	L2-4	L3-12
IO-61	80	H	5	L2-5	L3-13
IO-62	81	H	3	L2-6	L3-14
IO-63	82	H	1	L2-7	L3-15

Table 2: 50-pin Connector Interface

Note: If you use K1, then IO-53 must not be used. Similarly, if you use K2, then IO-11 must not be used.

Serial Interface

The CPLD Module supports an RS-232 interface via the DB-9 Male connector. The received serial data from Pin 2 of the DB-9 is brought to logic levels via inversion from a MAX 233 and is made available on Pin 12 of CPLD 1. The data to be transmitted is presented at Pin 73 of CPLD 1, which appears on Pin 3 of the DB-9 after being inverted by the MAX 233. Please see */mit/6.111/vhdl/serial/* for VHDL code that emulates a receiver and a transmitter.

Clock Interface

Figure 2 below presents the clocking scheme used by the module.

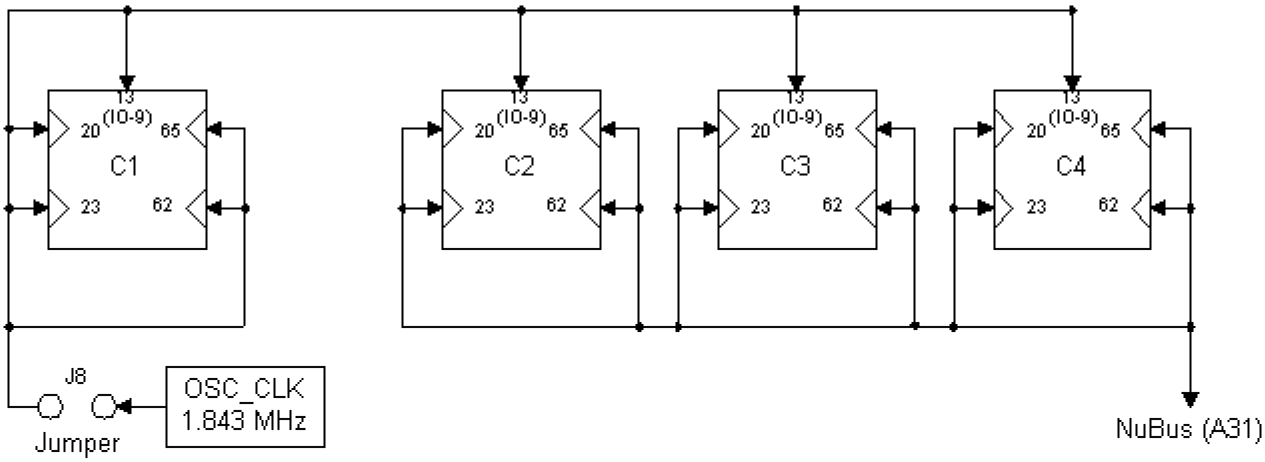


Figure 2: Clocking Scheme

The following are some clocking strategies that can be utilized:

1. C1 sources its clock from the oscillator by enabling jumper J8, while C [2,3 *or* 4] can buffer this signal (IO-9) to any connection on the NuBus, e.g. A30. A short circuit can then be created between this interconnection, i.e. A30 and A31 by installing a jumper wire.
2. C [2,3 or 4] buffer A31 to IO-9 thereby clocking C1. In this scenario, jumper J8 is removed and an external clock is supplied to A31.
3. Jumper J8's strategic location makes it possible to insert a wire within the jumper and plug the other end into NuBus A31. If this is done, a termination resistor (discussed ahead) may be required between A31 and GND located on the NUBus connector.
4. Make C1 use a separate clock, while C [2,3, and 4] share their clock from A31.

Irrespective of the clocking strategy, it is always important to consider the effects of clock skew.

Interconnect Bus

A 3-bit bus exists between CPLD 2, 3, and 4. This bus does not have an external interface, i.e. it cannot be accessed via the NuBus or the 50-pin connector interface. The details of these interconnections are:

CPLD I/O	CPLD Pin	Logic Block
[C2,C3 and C4 only]		
IO-8	12	B
IO-15	19	B
IO-55	73	G

Programming

The module supports the In-System Re-Programming of CPLD's. The following steps are the programming procedure:

Disable **all** clocks to **all** CPLD's. Cypress admits to a bug in the 374I CPLD in which the programming is incorrect if the input clock is running. This means removing jumper J8 and the input to A31. This also applies to the CPLD's not being programmed. For instance, if you want to program CPLD # 2 only, Jumper J8 must be removed to avoid corrupting CPLD # 1. All CPLDs are supplied with +12V when any CPLD is programmed.

Insert the 10-pin ribbon connector from the programmer into the blue 10-pin socket. The ***right orientation is critical - check the key.*** The correct orientation is so that the key of the cable is in the slot of the socket on the board. Pin 1 of the cable must mate with Pin 1 of the connector as shown in Figure 5.

Select the proper jumper configuration from the table below. A '1' means connected while a blank signifies 'open'.

Description	C4	C3	C2	C1	J1	J2	J3	J4	J5	J6	J7
C1 only				1	1						
C2 only			1			1	1				
C3 only		1						1	1		
C4 only	1									1	1
C1,C2							1				
C1,C2,C3									1		
C1,C2,C3,C4	1	1	1	1							1

Table 3: Jumper Selections

The program to use for the CPLD board in the 6.111 kit is `isr374`. Thisprogram may be called with one to eight arguments.

Another program, `isrconf`, can be used with an argument naming a Cypress style configuration file. Information on Cypress style configuration files is in the acco binder alongside of the CPLD programming computers in the digital lab.

The arguments for isr374 determine the number of CPLDs to be processed. The jumper configuration on the board specifies both the number and which CPLDs are to be processed. This must agree with the arguments you use for the command `isr374`.

See Table 3 for information on the jumpers. Normally, there is a jumper installed in position 7 on the right of the board and this specifies that all four of the CPLDs are used. To program only one CPLD, the one on the right, one would use the command:

```
isr374 b b <filename> b
```

An argument may consist of a keyword or a filename specifier. The filename specifier may or may not end in .jed but the file must have a .jed suffix. Keywords are case independent and may be abbreviated by their first letter. The keywords are `erase, e, bypass, b, n, verify, v, program,` and `p`. The keyword, `n`, is a synonym for bypass or `b`. It is used in Cypress style configuration files. Both verify and program require a filename as the next argument. You may omit the program keyword if you like. File names can either be absolute or relative and, of course, cannot be equal to one of the above keywords or its initial letter abbreviation.

The isr374 program will output the configuration file it has generated from your arguments and also the following instructions. Follow them.

Connect the ISR programmer (10 pin 3M) to the board in your kit. Make sure the key is correct! Turn the kit DC power on. Pull the switch on the board forward towards the front of the kit and check that the red light is on. Ensure that no clocks are connected. The left hand jumper should be removed and no wires plugged into A31.

Type a `return` to begin programming.

Push SW1 forward to enable the green LED and to connect the clock(s) before using the CPLD.

Test Programs

We have two programs to test the board. They are intended to verify the operation of the CPLD as well as the connections between the 50-pin connector and the logic analyzer.

Driver: This program loads a 31 bit counter that is displayed on the NuBus LED's.

Reader: This program reads the first 25 I/O lines of the NuBus connector and extends them to the 50-pin connector .

The output of the first 25 bits of the NuBus can therefore be seen on the Logic Analyzer. The remaining 6 bits on the NuBus can be directly probed by the scope.

The above files along with an empty pre-numbered VHDL file (374sample.vhd) can be found at */mit/6.111/cpld/sources/test/*

Termination

The diagram below shows a high frequency square wave when observed with an analog scope:

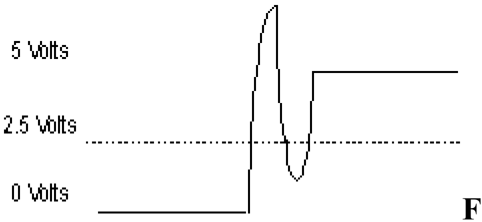


Figure 3: Ringing

Clearly, if this signal were to be a clock-input to a device, there would be some serious problems. All signals used as clocks must be verified to see if termination is necessary. Inserting a resistor on the protoboard between the signal and ground can reduce this swing. A 100 resistor is ideal for most cases.

Note that with a 100 termination, the signal's logic high is now about 3.8 Volt, which satisfies the requirements of a TTL compatible device. If you feel the need to terminate a bus, ask the front desk for a resistor network. Its logic diagram is as follows:

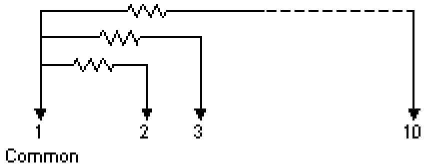


Figure 4: Resistor Network

Common Errors

- If you use K1 of the kit to interface with the module, then IO-53 must not be used. Similarly, if K2 is used, then IO-11 must not be used. Refer to Table 2, which lists these pins as `GND'.
- The 50-pin connector and the programming 10-pin connector must be inserted in the right way. The diagram below explains the proper alignment of Pin 1.

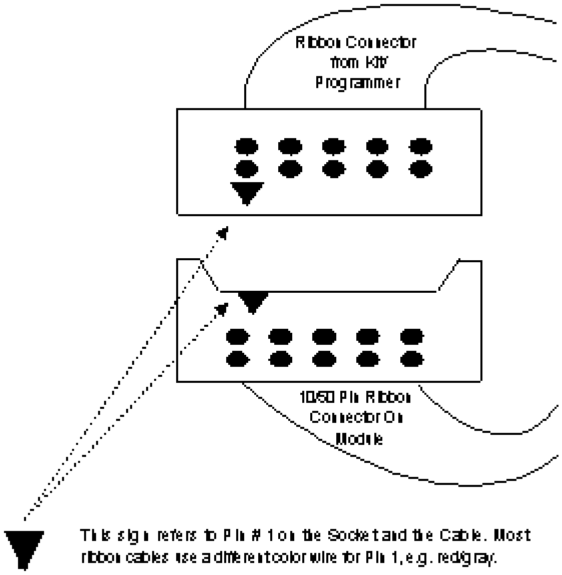

Ribbon Connector from Kit/ Programmer
10/50 Pin Ribbon Connector On Module
This sign refers to Pin # 1 on the Socket and the Cable. Most ribbon cables use a different color wire for Pin 1, e.g. red/gray.

Figure 5: Ribbon Connector Insertion

- All clocks must be disabled before commencing programming.
- Critical signals such as clock sources may require termination. Verify this with a scope.
- Don't forget to push SW1 backward upon the completion of programming.

About the Cypress CYC374I CPLD

The 128 macrocells in the 84 pin CY7C374i are divided between eight logic blocks. Each logic block includes 16 macrocells, a 72 x 86 product term array, and an intelligent product term allocator. You can refer to *(cy7c374i)* for more details.

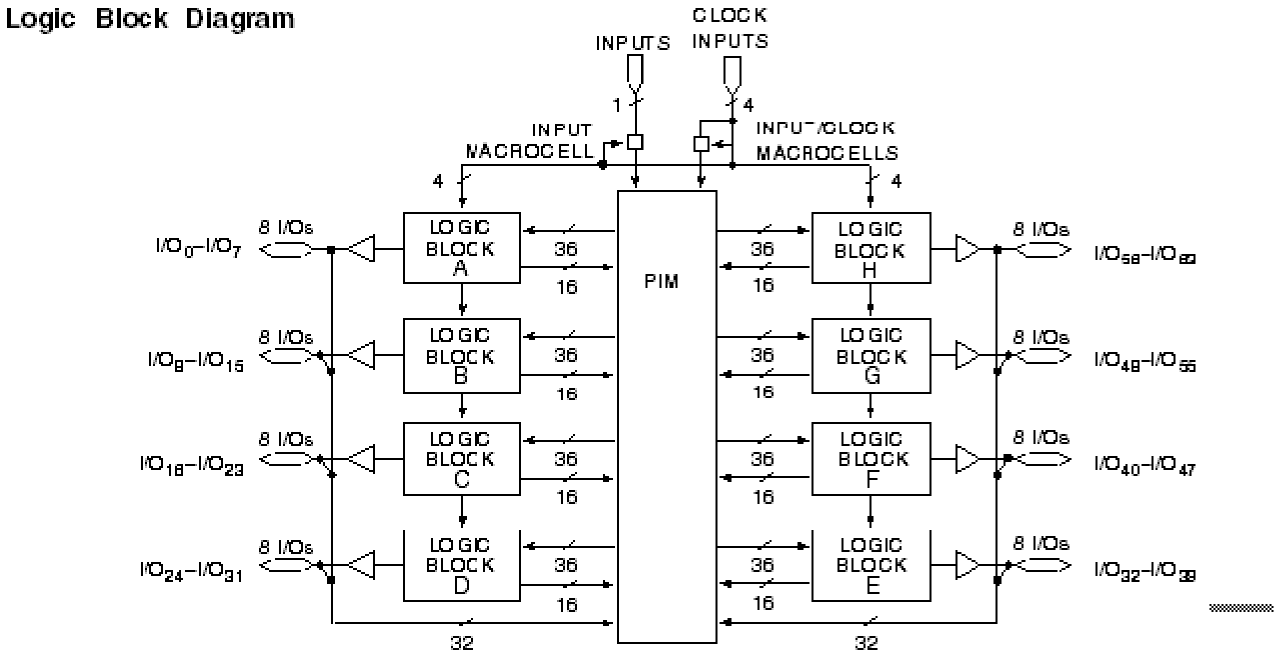


Figure 6: Logic Block Diagram

The pin layout of the CYC374I CPLD is

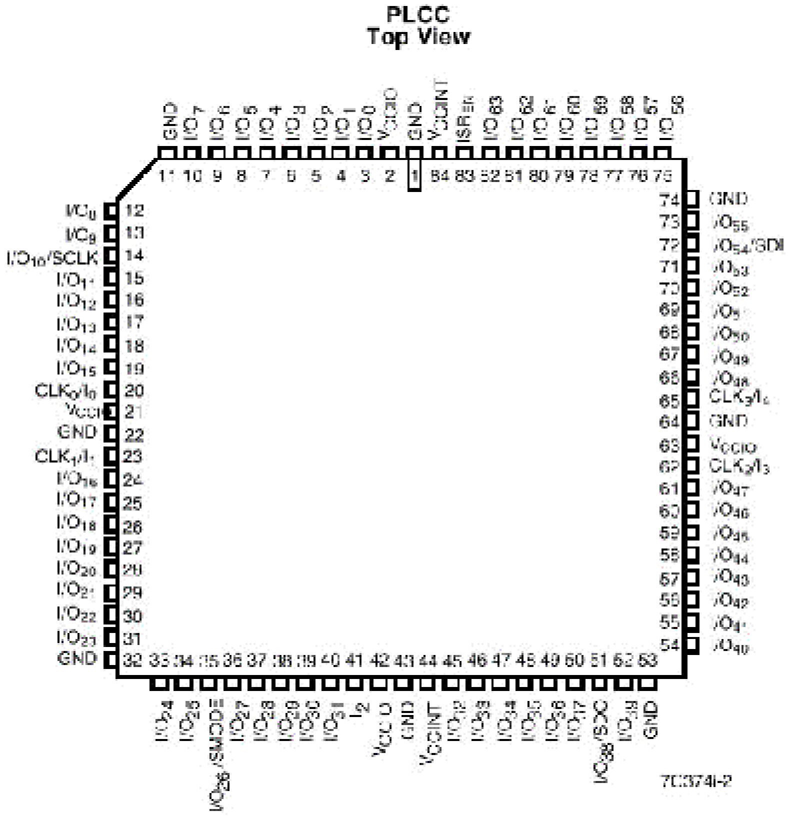
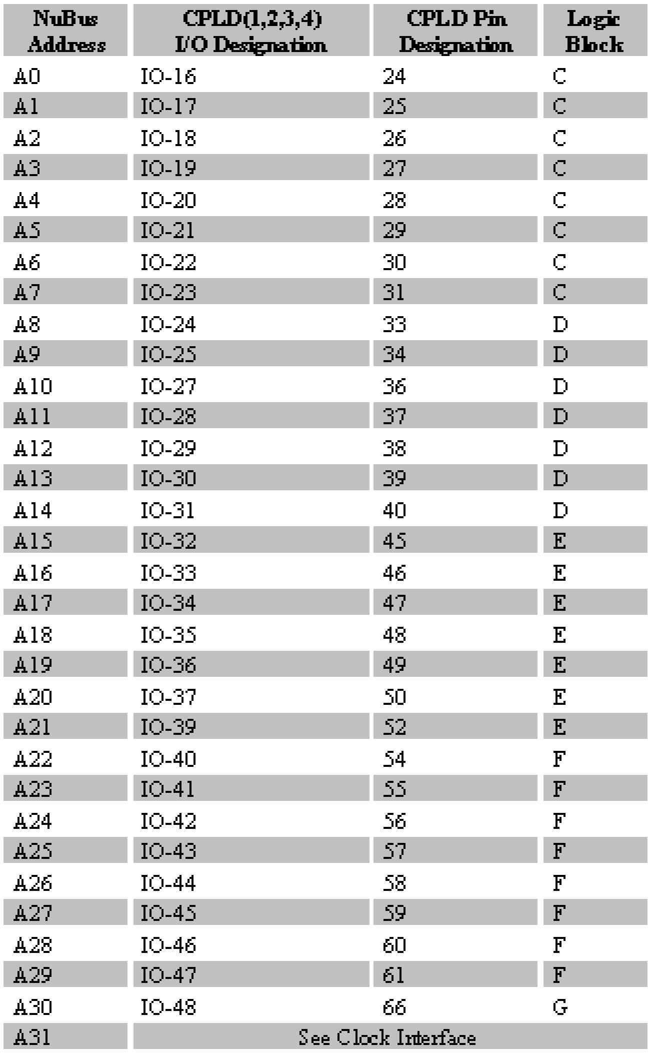


Figure 7: Pin Layout

NuBus Interface

31 I/O pins of each CPLD are interconnected and extended to the NuBus connector. For example, IO-0 of all 4 CPLD's are tied together to NuBus Address A0. Table 1 elaborates these interconnections.

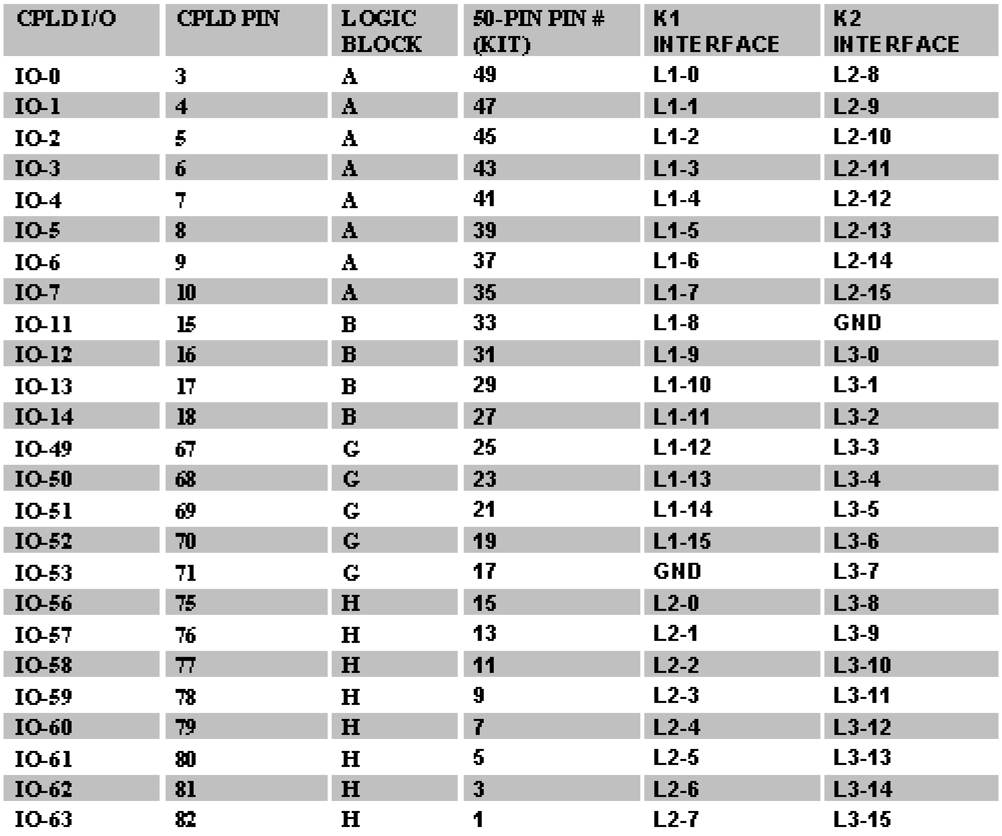


NuBus Address	CPLD(1,2,3,4) I/O Designation	CPLD Pin Designation	Logic Block
A0	IO-16	24	C
A1	IO-17	25	C
A2	IO-18	26	C
A3	IO-19	27	C
A4	IO-20	28	C
A5	IO-21	29	C
A6	IO-22	30	C
A7	IO-23	31	C
A8	IO-24	33	D
A9	IO-25	34	D
A10	IO-27	36	D
A11	IO-28	37	D
A12	IO-29	38	D
A13	IO-30	39	D
A14	IO-31	40	D
A15	IO-32	45	E
A16	IO-33	46	E
A17	IO-34	47	E
A18	IO-35	48	E
A19	IO-36	49	E
A20	IO-37	50	E
A21	IO-39	52	E
A22	IO-40	54	F
A23	IO-41	55	F
A24	IO-42	56	F
A25	IO-43	57	F
A26	IO-44	58	F
A27	IO-45	59	F
A28	IO-46	60	F
A29	IO-47	61	F
A30	IO-48	66	G
A31	See Clock Interface		

Table 1: NuBus Interface

50-Pin Connector Interface

25 I/O pins are interconnected among the 4 CPLD's and the 50-pin connector. The signals of the 50-pin connector interface directly to the logic analyzer via the connector on the kit. Table 2 provides details of these interconnections. Logic Blocks are partitions internal to the CPLD chip. This information, at times, can be useful during device fitting.



CPLD I/O	CPLD PIN	LOGIC BLOCK	50-PIN PIN # (KIT)	K1 INTERFACE	K2 INTERFACE
IO-0	3	A	49	L1-0	L2-8
IO-1	4	A	47	L1-1	L2-9
IO-2	5	A	45	L1-2	L2-10
IO-3	6	A	43	L1-3	L2-11
IO-4	7	A	41	L1-4	L2-12
IO-5	8	A	39	L1-5	L2-13
IO-6	9	A	37	L1-6	L2-14
IO-7	10	A	35	L1-7	L2-15
IO-11	15	B	33	L1-8	GND
IO-12	16	B	31	L1-9	L3-0
IO-13	17	B	29	L1-10	L3-1
IO-14	18	B	27	L1-11	L3-2
IO-49	67	G	25	L1-12	L3-3
IO-50	68	G	23	L1-13	L3-4
IO-51	69	G	21	L1-14	L3-5
IO-52	70	G	19	L1-15	L3-6
IO-53	71	G	17	GND	L3-7
IO-56	75	H	15	L2-0	L3-8
IO-57	76	H	13	L2-1	L3-9
IO-58	77	H	11	L2-2	L3-10
IO-59	78	H	9	L2-3	L3-11
IO-60	79	H	7	L2-4	L3-12
IO-61	80	H	5	L2-5	L3-13
IO-62	81	H	3	L2-6	L3-14
IO-63	82	H	1	L2-7	L3-15

Table 2: 50-pin Connector Interface

Note: If you use K1, then IO-53 must not be used. Similarly, if you use K2, then IO-11 must not be used.

Serial Interface

The CPLD Module supports an RS-232 interface via the DB-9 Male connector. The received serial data from Pin 2 of the DB-9 is brought to logic levels via inversion from a MAX 233 and is made available on Pin 12 of CPLD 1. The data to be transmitted is presented at Pin 73 of CPLD 1, which appears on Pin 3 of the DB-9 after being inverted by the MAX 233. Please see */mit/6.111/vhdl/serial/* for VHDL code that emulates a receiver and a transmitter.

Clock Interface

Figure 2 below presents the clocking scheme used by the module.

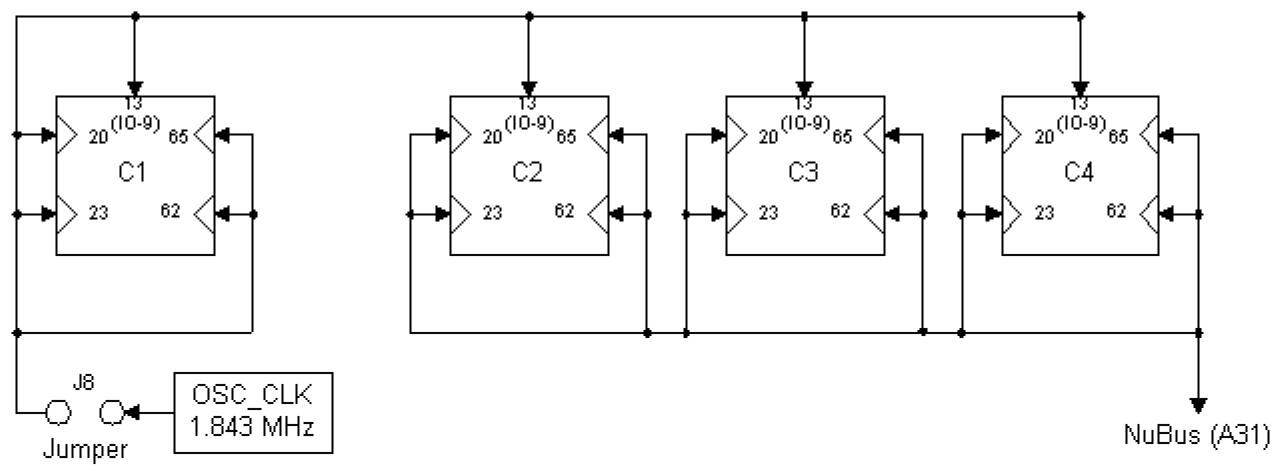


Figure 2: Clocking Scheme

The following are some clocking strategies that can be utilized:

1. C1 sources its clock from the oscillator by enabling jumper J8, while C [2,3 *or* 4] can buffer this signal (IO-9) to any connection on the NuBus, e.g. A30. A short circuit can then be created between this interconnection, i.e. A30 and A31 by installing a jumper wire.
2. C [2,3 or 4] buffer A31 to IO-9 thereby clocking C1. In this scenario, jumper J8 is removed and an external clock is supplied to A31.
3. Jumper J8's strategic location makes it possible to insert a wire within the jumper and plug the other end into NuBus A31. If this is done, a termination resistor (discussed ahead) may be required between A31 and GND located on the NUBus connector.
4. Make C1 use a separate clock, while C [2,3, and 4] share their clock from A31.

Irrespective of the clocking strategy, it is always important to consider the effects of clock skew.

Interconnect Bus

A 3-bit bus exists between CPLD 2, 3, and 4. This bus does not have an external interface, i.e. it cannot be accessed via the NuBus or the 50-pin connector interface. The details of these interconnections are:

CPLD I/O	CPLD Pin	Logic Block
[C2,C3 and C4 only]		
IO-8	12	B
IO-15	19	B
IO-55	73	G

Programming

The module supports the In-System Re-Programming of CPLD's. The following steps are the programming procedure:

Disable **all** clocks to **all** CPLD's. Cypress admits to a bug in the 374I CPLD in which the programming is incorrect if the input clock is running. This means removing jumper J8 and the input to A31. This also applies to the CPLD's not being programmed. For instance, if you want to program CPLD # 2 only, Jumper J8 must be removed to avoid corrupting CPLD # 1. All CPLDs are supplied with +12V when any CPLD is programmed.

Insert the 10-pin ribbon connector from the programmer into the blue 10-pin socket. The ***right orientation is critical - check the key.*** The correct orientation is so that the key of the cable is in the slot of the socket on the board. Pin 1 of the cable must mate with Pin 1 of the connector as shown in Figure 5.

Select the proper jumper configuration from the table below. A '1' means connected while a blank signifies 'open'.

Description	C4	C3	C2	C1	J1	J2	J3	J4	J5	J6	J7
C1 only				1	1						
C2 only			1			1	1				
C3 only		1						1	1		
C4 only	1									1	1
C1,C2							1				
C1,C2,C3									1		
C1,C2,C3,C4	1	1	1	1							1

Table 3: Jumper Selections

The program to use for the CPLD board in the 6.111 kit is `isr374`. This program may be called with one to eight arguments.

Another program, `isrconf`, can be used with an argument naming a Cypress style configuration file. Information on Cypress style configuration files is in the acco binder alongside of the CPLD programming computers in the digital lab.

The arguments for isr374 determine the number of CPLDs to be processed. The jumper configuration on the board specifies both the number and which CPLDs are to be processed. This must agree with the arguments you use for the command `isr374`.

See Table 3 for information on the jumpers. Normally, there is a jumper installed in position 7 on the right of the board and this specifies that all four of the CPLDs are used. To program only one CPLD, the one on the right, one would use the command:

```
isr374 b b <filename> b
```

An argument may consist of a keyword or a filename specifier. The filename specifier may or may not end in .jed but the file must have a .jed suffix. Keywords are case independent and may be abbreviated by their first letter. The keywords are `erase, e, bypass, b, n, verify, v, program,` and `p`. The keyword, `n`, is a synonym for bypass or `b`. It is used in Cypress style configuration files. Both verify and program require a filename as the next argument. You may omit the program keyword if you like. File names can either be absolute or relative and, of course, cannot be equal to one of the above keywords or its initial letter abbreviation.

The isr374 program will output the configuration file it has generated from your arguments and also the following instructions. Follow them.

Connect the ISR programmer (10 pin 3M) to the board in your kit. Make sure the key is correct! Turn the kit DC power on. Pull the switch on the board forward towards the front of the kit and check that the red light is on. Ensure that no clocks are connected. The left hand jumper should be removed and no wires plugged into A31.

Type a `return` to begin programming.

Push SW1 forward to enable the green LED and to connect the clock(s) before using the CPLD.

Test Programs

We have two programs to test the board. They are intended to verify the operation of the CPLD as well as the connections between the 50-pin connector and the logic analyzer.

Driver: This program loads a 31 bit counter that is displayed on the NuBus LED's.

Reader: This program reads the first 25 I/O lines of the NuBus connector and extends them to the 50-pin connector .

The output of the first 25 bits of the NuBus can therefore be seen on the Logic Analyzer. The remaining 6 bits on the NuBus can be directly probed by the scope.

The above files along with an empty pre-numbered VHDL file (374sample.vhd) can be found at */mit/6.111/cpld/sources/test/*

Termination

The diagram below shows a high frequency square wave when observed with an analog scope:

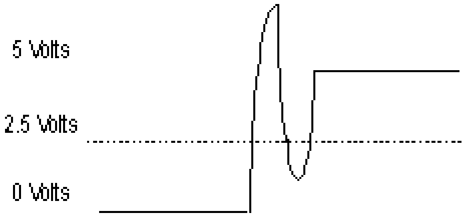


Figure 3: Ringing

Clearly, if this signal were to be a clock-input to a device, there would be some serious problems. All signals used as clocks must be verified to see if termination is necessary. Inserting a resistor on the protoboard between the signal and ground can reduce this swing. A 100 resistor is ideal for most cases.

Note that with a 100 termination, the signal's logic high is now about 3.8 Volt, which satisfies the requirements of a TTL compatible device. If you feel the need to terminate a bus, ask the front desk for a resistor network. Its logic diagram is as follows:

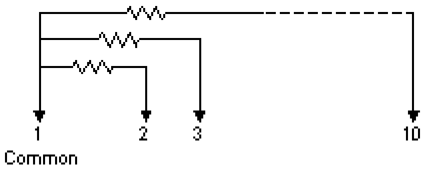


Figure 4: Resistor Network

Common Errors

- If you use K1 of the kit to interface with the module, then IO-53 must not be used. Similarly, if K2 is used, then IO-11 must not be used. Refer to Table 2, which lists these pins as `GND'.
- The 50-pin connector and the programming 10-pin connector must be inserted in the right way. The diagram below explains the proper alignment of Pin 1.

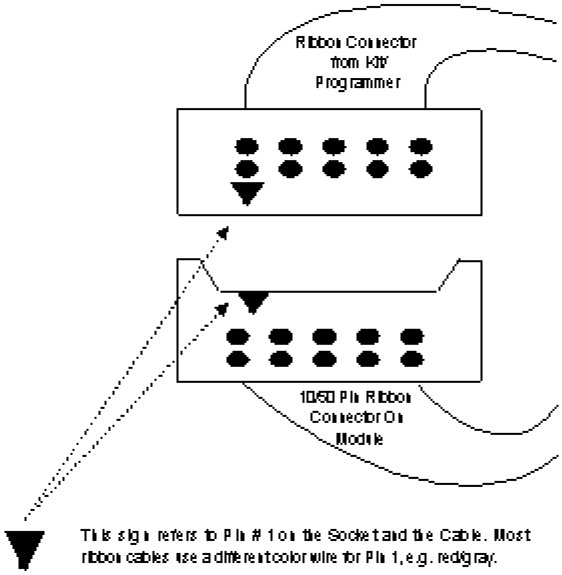

Ribbon Connector from Kit/ Programmer
10/50 Pin Ribbon Connector On Module
This sign refers to Pin # 1 on the Socket and the Cable. Most ribbon cables use a different color wire for Pin 1, e.g. red/gray.

Figure 5: Ribbon Connector Insertion

- All clocks must be disabled before commencing programming.
- Critical signals such as clock sources may require termination. Verify this with a scope.
- Don't forget to push SW1 backward upon the completion of programming.

About the Cypress CYC374I CPLD

The 128 macrocells in the 84 pin CY7C374i are divided between eight logic blocks. Each logic block includes 16 macrocells, a 72 x 86 product term array, and an intelligent product term allocator. You can refer to *(cy7c374i)* for more details.

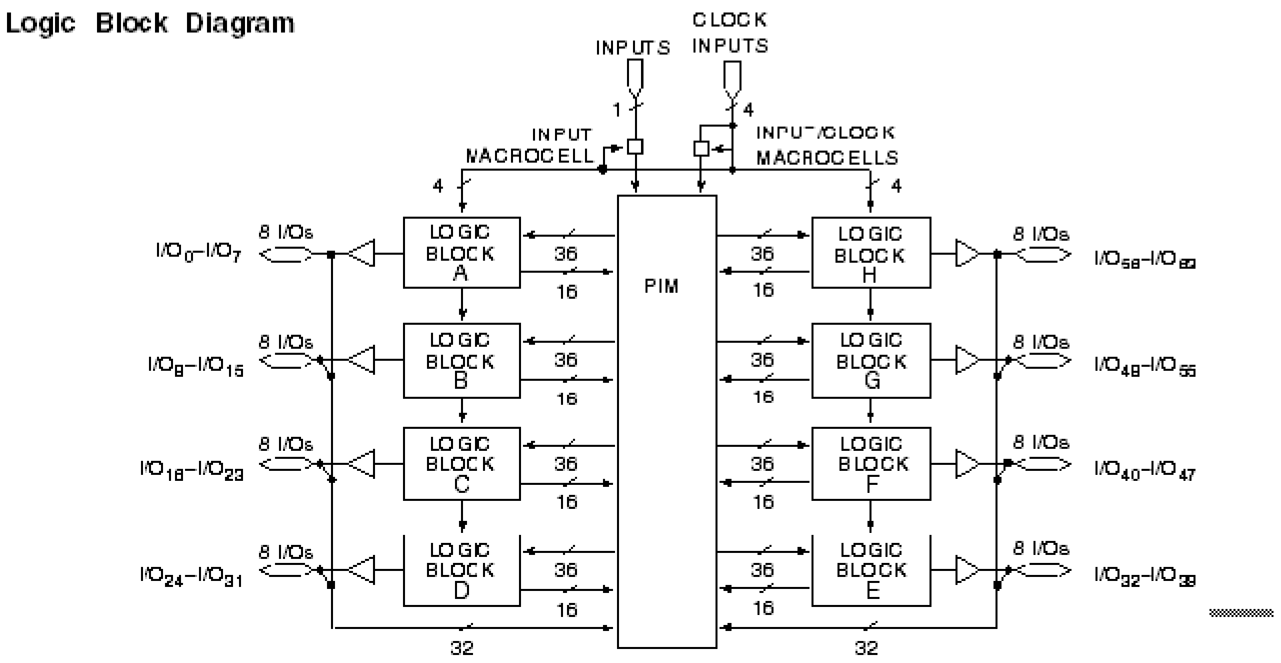


Figure 6: Logic Block Diagram

The pin layout of the CYC374I CPLD is

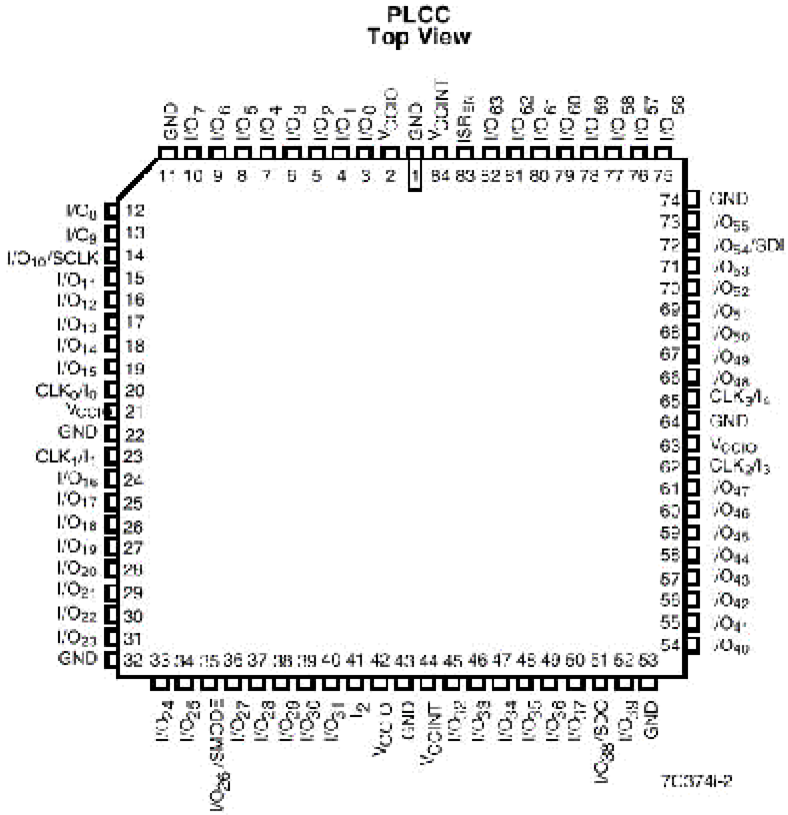


Figure 7: Pin Layout