Rev	Revision Description	Drawn	Date	Appvd	Date
ENG	Initial Specification	BE/CH	2/1100		
E01	Added .dat file to text document and Jedec files checked into Matrix object	CH	2/18/00		
E02	Added pages 31-53	CH	3/24/00		
E03	Added pages 54-67	CH	3/29/00		
E04	Added porten information from M. Lovejoy	CH	4/5/00		
E05	Added additional fuse map file.	CH	5/25/00		
E06	Replaced .dat files	CH	6/2/00		
E07	Include XCR3032XL	EH	4/29/01		
E08	Remove Program – Verify command, Corrected INIT pause times, Added Verify pause times, Added pause after ERASE	RD	7/10/01		
E09	Remove Gray Code Address Table to separate file	EH	8/29/01		

The Fuse Maps for this family is contained in Excel Spead Sheet files named PZ256.xls, PZ128.xls and 64xpla3JEDfm.xls Copies of these spread sheets are checked into the Matrix object for this specification.

XILINX®

Specification for Programming Device XCR3032XL, XCR3064XL, XCR3128XL and XCR3256XL

DRAWN BY **BE**	DATE 2/11/00				
TOLERANCES .X .XX .XXX	CHECKED BY	DATE	SIZE	DRAWING NUMBER	REV.
	ENG APPROVAL	DATE	**A**	**SPD0010**	E09
	MFG APPROVAL	DATE			
Unless otherwise specified, dimensions are in inches.			SCALE	SHEET 1 OF	51

1. Introduction

This document pertains to the following devices and packages:

3032XL	VQ44, PC44, CS48
3064XL	VQ44, VQ100
3128XL	VQ100, TQ144
3256XL	TQ144, PQ208

The device programming and verification procedures are similar to those used with standard non-volatile (NV) memories. Memory cells can be erased only as a single array, while programming of groups of individual cells is performed. Note that after successful completion of an erase operation, all cells in the device are in the logical “1” state. Erase and program operations are performed via the JTAG port and IEEE 1149.1 sixteen state TAP controller.

The Xilinx software generates device-programming files in JEDEC format. The JEDEC file also contains information specific to each device, which will be compared to the device being programmed. The manufacturer’s code to identify Xilinx (or in some cases Philips) as the manufacturer and the product code to identify the device are read via the Device Identification Register of the IEEE Standard 1149.1. Please refer to tables in a later section for codes.

2. Features

2.1 Erase

The device is electrically erasable. The algorithm has to perform a check on the device to ensure all bits are erased before allowing programming. Note that erased cells are verified to logical 1 state.

2.2 Addressing

The device is addressed with a 9-bit address for the 3256 device, an 8-bit address for both the 3128 and the 3064 devices and a 7-bit address for the 3032 device. The MSB selects upper (logical 1) or lower (logical 0) for a given address. The address, less MSB, is indexed with a Gray code count starting from zero and counting sequentially up to a maximum Gray code count. However, some bits of some addresses are “don't cares”; the fuse map for each device is available as an Excel spreadsheet. The legal device addresses are contained in the device specific Add.dat file.

2.3 Operations

The device has several independent operations: Program, Erase, Verify, and Initialize. To use these instructions, an enable command (ISP_ENABLE) must be executed via the JTAG controller. Furthermore, after performing the desired independent operations, a disable command (ISP_DISABLE) must be executed to properly configure all registers when returning to test-logic-reset state in the JTAG controller. Program is used for selectively changing groups of NV-cells within the device. Erase is the procedure used to erase all cells in the device in one operation. Verify is used to read NV-cell content. Initialize is to force configuration of the device to that loaded in the NV-memory without power cycling.

2.4 Signature String

The programmer or host computer must support a mode for reading and displaying the data stored in the user storage area. Note that this storage area does not conform to IEEE 1149.1

signature specifications accessed by the 1149.1 Standard USERCODE instruction. The size of the user storage is shown in the Add.dat file. Note that the number of bits varies for XPLA3 family devices. Because the signature string is not part of the JEDEC file, programming of these bytes is an independent operation from the operation of programming configuration bits and can be performed at anytime.
Note that the user electronic signature string is not in the JEDEC file, but rather an arbitrary bit string defined by the customer.

2.5 Device Security

The device supports a read security feature, which protects the design from being copied. A secured device may still be erased and reprogrammed.

3. Special Instructions

3.1 Device/ File Checksum Calculation

Contained in each JEDEC file is a fuse checksum (C-Field), which should be used to represent the file checksum. The checksum is to be implemented according to JEDEC Standard Number 3A. **The same method must be used to calculate the device checksum and the two should match.**

Transmission checksum according to JEDEC Standard Number 3A can be implemented as well.

3.2 Compatibility Checks

3.2.1 Adapter Type

The adapter should contain an electronically readable code for identification. The programming algorithm must check the adapter ID for compatibility with the target device. The JEDEC design file also contains information specific to the device pin count and fuse size, both of which must be compared to the adapter in use. The pin count, fuse size, and packages for each device are listed in the corresponding Add.dat.

4. Programming Sequence

The device programming sequence, illustrated in Figure 1, begins by verifying that the device design file and programming algorithm match the installed programmer adapter. This check is accomplished by first comparing the programmer adapter ID to all acceptable adapter IDs in the programmer algorithm and next to the fuse count and pin count contained in the JEDEC file. If a mismatch occurs, display message **A: "Incompatible Adapter Or File**" for current algorithm and terminate the programming sequence.

4.1 Read Manufacturer's Code ID

Read the manufacturer's code in the device contained in the Device Identification Register defined in IEEE 1149.1 Standard. This register is read via ISP port with the IEEE 1149.1 IDCODE instruction. Note that the manufacturer code can be Xilinx and in some case Philips; the code translations are listed in Table 3. of this document. The next step is to verify that the device part number listed in the JEDEC file is a valid part number for the manufacturer

determined from the Device Identification Register. If the device part number is not valid for the device manufacturer, display the message **B:** "**Manufacturer's Code Error"** and terminate the programming sequence.

4.2 Read Product Code ID

Read the product codes, both architecture and number of macrocells codes, of the device contained in the Device Identification Register defined in IEEE 1149.1 Standard. This register is read via ISP port with the IEEE 1149.1 IDCODE instruction. The code translations are listed in Table 4. of this document. The next step is to verify that the device part number listed in the JEDEC file is a correct for the architecture and number of macrocells determined from the Device Identification Register. If the device part number is not correct for the device architecture and number of macrocells, display the message **C: "Product Code Error"** and terminate the programming sequence.

4.3 Device Erase Check

Verify that all NV-cells are in the unprogrammed state ('1'). Bulk erase check is performed with ISP verify command (ISP_VERIFY) for each address. Recall that an ISP_ENABLE instruction must be executed prior to sequentially shifting in addresses, executing verify operation and shifting out data for actual data comparison. If any of the NV-cells are programmed ('0'), display the message **E: "Device Not Erased"** and allow the option to erase the device. The device has to be bulk erased prior to programming. If the device fails, display message **F: "Device Failed To Erase"** and terminate the programming sequence.

A. Device Programming

At this point the actual programming cycle begins. The address and data are first loaded before a short programming pulse is applied to the device. This programming procedure should follow the verify procedure where an ISP_ENABLE command was executed first in the procedure and an ISP_DISABLE should not be part of the procedure. The number of address and data bits varies from device to device. Specific address information of all address and number of data bits can be found in the Add.dat file of each device. Programming is performed utilizing the sixteen-state TAP controller, hence, timing is insured.

4.5 Post-Program Verify

After programming all the addresses of the device, perform data verification, as illustrated in Figure 2. This verification procedure should follow the verify and programming procedure where an ISP_ENABLE command was executed first in the procedure and an ISP_DISABLE should not be part of the procedure. If any cell fails to verify, display the message **G: "Device Failed To Program"** and terminate the programming sequence. Note that this operation can be performed at any time.

4.6 Secure Device

Following the programming and post-program verify sequence, the algorithm should query the operator to secure the device, as illustrated in Figure 2. If the user elects not to secure the

device, display the message **H: "Device Not Secured"** and terminate the programming sequence with an ISP_DISABLE instruction. If the user chooses to secure the device, program the security bit (see Figure 4), display the message **D: "Device Secured"** and terminate the programming sequence with an ISP_DISABLE instruction. If the device fails to verify secure, display the message **J: Device Failed To Secure"** and terminate the programming sequence with an ISP_DISABLE instruction. This procedure should follow the programming sequence without an ISP_DISABLE command being included until completing the secure query and response to query.

If a device is Read Secured, the signature string, manufacturer's code and product code can still be read. All other data can not be read. Table 1 shows which operations may be performed after a device is Read Secured.

Note that the device is not Read Secured after programming the security bit until either an ISP init instruction is executed or the power is cycled.

Table 1

Read Secure*

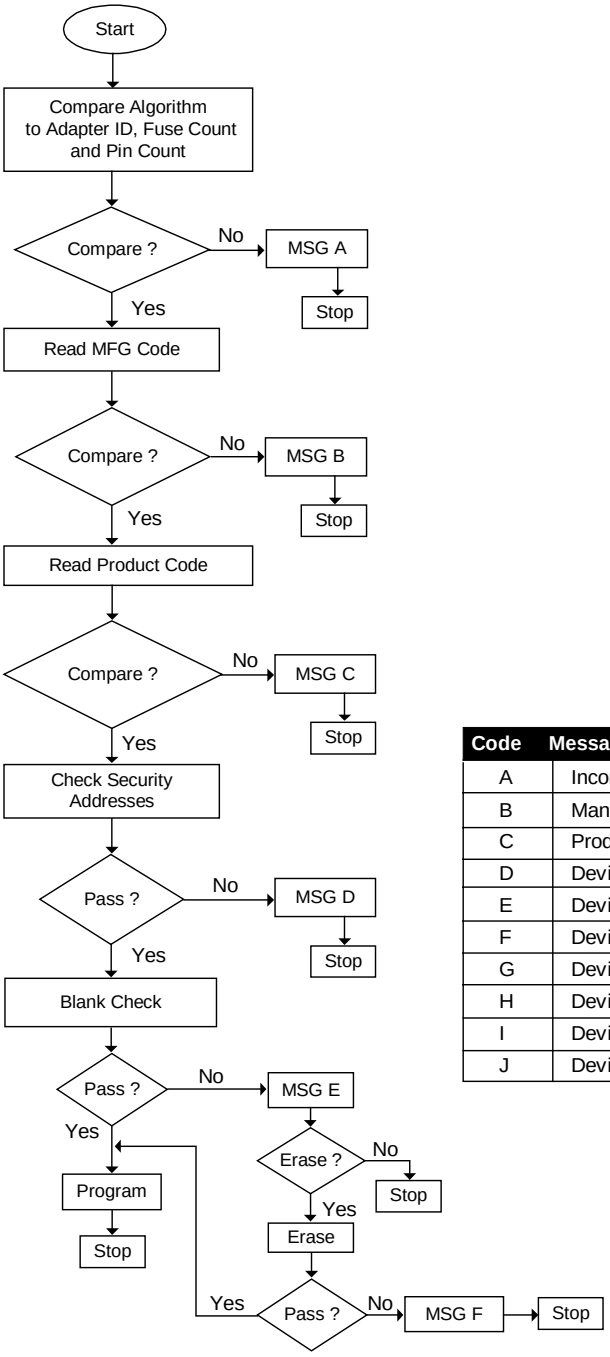
Operation	Valid
Program	No
Erase	Yes
Verify	No
Blank Check	No
Signature String	Yes
Mfg/Product Code	Yes

***** See Add.dat file for security bit address.

5. Other Device Operations

5.1 Device Bulk Erase

To erase the device, follow the flow in Figure 3 and the timing according to Figure ***XXXX***. Note that all NV-memory (user configuration bits and signature string) are erased with this operation. Recall that an ISP_ENABLE instruction must be executed prior to executing the erase operation. If verification of any logic one (1) state of any bit fails after performing the erase operation, then display the message **"Device Failed to Erase** and terminate the programming sequence with an ISP_DISABLE instruction.



Code	Messages
A	Incompatible Adapter or File for Current Algorithm
B	Manufacturer's Code Error
C	Product Code Error
D	Device Secured
E	Device Not Blank
F	Device Failed To Erase
G	Device Failed To Program
H	Device Not Secured
I	Device Failed To Verify
J	Device Failed To Secure

Figure 1. Overall Programming Sequence

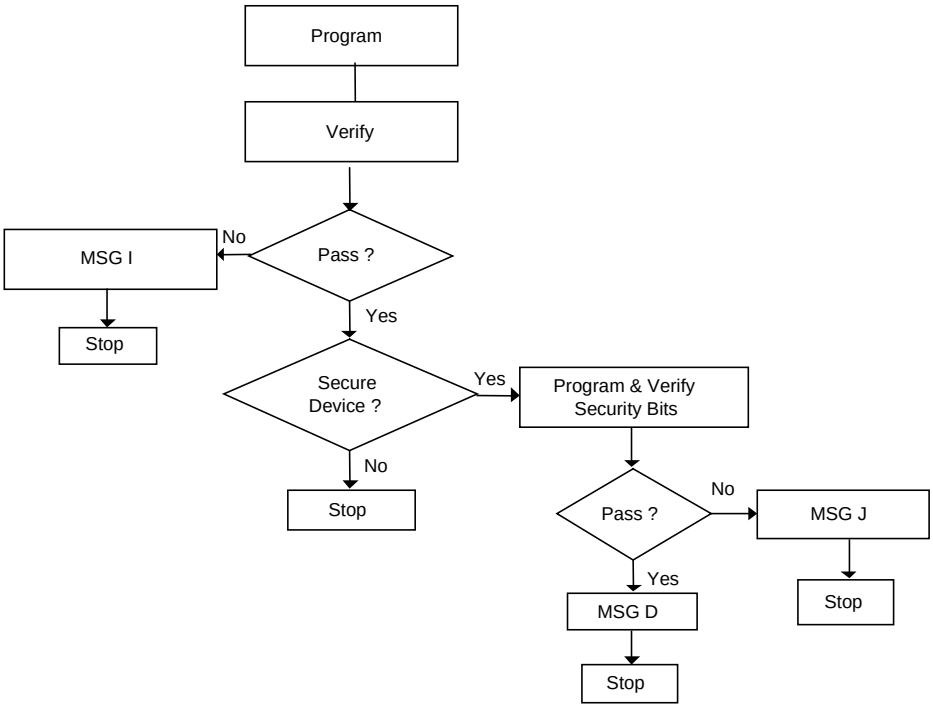


Figure 2. Programming Flow

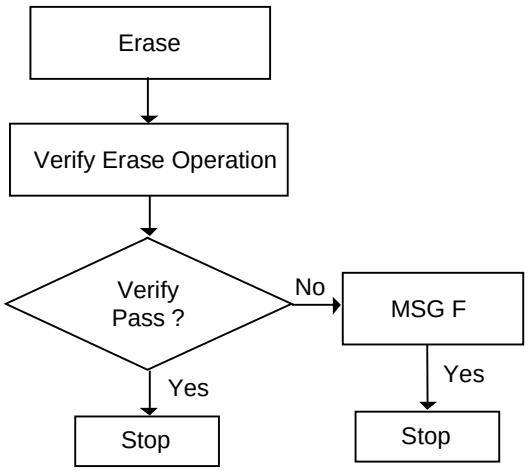


Figure 3. Erase Flow

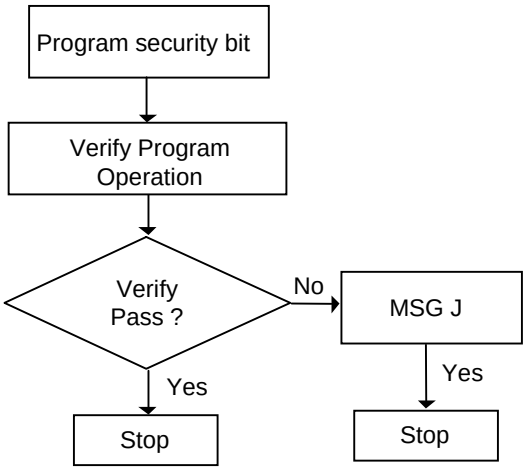


Figure 4. Secure Flow

Table 2. Common DC Programming and Erase Characteristics T_A = 25°C ± 5°C

Symbol	Description	Limits		
		Min	Max	Units
I_{IL}	Input Leakage		10	µA
I_{CC}	V_{CC} Supply Current		80	mA
V_{IL}	Low-Level Input Voltage	0	0.8	V
V_{IH}	High-Level Input Voltage	2.0	5.5	V
V_{OL}	Low-Level Output Voltage		0.4	V
V_{OH}	High-Level Output Voltage	2.4		V
V_{CCBNK}	V_{CC}	3.0	3.6	V

Note: Xilinx recommends that the mean be used whenever possible.

Table 3. Manufacturer ID codes

Manufacturer	ID code
Xilinx	00001001001
Philips	00000010101

Note: The Philips ID Code only applies to the original 3 die released from the Philips Fab (XCR3064XL, XCR3128XL, XCR3256XL). All newly released devices (after Dec 01) will only use the Xilinx ID Code.

Table 4. Device ID codes

Product	Package	Package Code	Architecture Code	Macrocell Count Code
3032XL	VQ44	110	010	000001
3032XL	PC44	101	010	000001
3032XL	CS48	100	010	000001
3064XL	VQ44	110	010	000100
3064XL	VQ100	001	010	000100
3128XL	VQ100	001	010	001000
3128XL	TQ144	011	010	001000
3256XL	TQ144	011	010	010100
3256XL	PQ208	100	010	010100

Table 5. AC Programming Specifications

Symbol	Description	Limits		
		Min	Max	Units
T_{PWPGM}	Program Pulse Width	10	11	ms
T_{ERASE}	Erase Pulse Width	100	110	ms

3032XL Programming Signal Definitions

	VQ44	PC44	CS48
Function	Pin Number	Pin Number	Pin Number
A3/TDI	1 (TDI)	7 (TDI)	B1 (TDI)
A4	2	8	C2
A5	3	9	C1
PORT_EN	4 (PE)	10 (PE)	C3 (PE)
A6	5	11	D3
A7	6	12	D1
A8/TMS	7 (TMS)	13 (TMS)	D2 (TMS)
A9	8	14	E1
VDDE	9	15	E2
A10	10	16	F1
A11	11	17	G1
A12	12	18	E4
A13	13	19	F2
A14	14	20	G2
A15	15	21	F3
GND	16	22	E3
VDDI	17	23	G4
B15	18	24	F4
B14	19	25	G5
B13	20	26	F5
B12	21	27	G6
B11	22	28	G7
B10	23	29	F7
GND	24	30	E6
B9	25	31	E7
B8 / TCK	26 (TCK)	32 (TCK)	E5 (TCK)
B7	27	33	D7
B6	28	34	D6
VDDE	29	35	C7
B5	30	36	C6
B4	31	37	D4
B3 / TDO	32 (TDO)	38 (TDO)	B7 (TDO)
B2	33	39	B6
B1	34	40	A6
B0	35	41	C5
GND	36	42	A5
IN3/GCLK3	37	43	B5
IN2/GCLK2	38	44	A4
IN1/GCLK1	39	1	B4
IN0/GCLK0	40	2	A3
VDDI	41	3	B3
A0	42	4	A2
A1	43	5	A1
A2	44	6	C4
CS48 No connect Package balls: B2,G3,F6,A7			

3064XL Programming Signal Definitions

	VQ100	VQ44		VQ100	VQ44
Function	Pin Number	Pin Number	Function	Pin Number	Pin Number
B7	100		C7	52	
VDDE	3		C6	54	
B8/TDI	4 (TDI)	1 (TDI)	C5	56	
B9	6	2	C4	57	
B10	8	3	C3	58	23
B11	9		GNDE	59	24
B12	10		C2	60	
PORT_EN	11 (PE)	4 (PE)	C1	61	25
B13	12	5	C0/TCK	62 (TCK)	26 (TCK)
B14	13	6	A15	63	
B15	14		A14	64	27
D0/TMS	15 (TMS)	7 (TMS)	A13	65	28
D1	16	8	VDDE	66	29
D2	17		A12	67	
VDDE	18	9	A11	68	
D3	19	10	A10	69	30
D4	20	11	A9	71	31
D5	21		A8/TDO	73 (TDO)	32 (TDO)
D6	23		GNDE	74	
D7	25		A7	75	33
GNDE	26		A6	76	
D8	29	12	A5	79	
D9	30	13	A4	80	
D10	31	14	A3	81	
D11	32	15	VDDE	82	
D12	33		A2	83	
VDDE	34		A1	84	34
D13	35		A0	85	35
D14	36		GNDE	86	36
D15	37		GNDE	86	36
GNDE	38	16	CLK3/IN3	87	37
GNDE	38	16	CLK2/IN2	88	38
VDDI	39	17	CLK1/IN1	89	39
VDDI	39	17	CLK0/IN0	90	40
C15	40		VDDI	91	41
C14	41		VDDI	91	41
C13	42		B0	92	42
GNDE	43		B1	93	43
C12	44	18	B2	94	44
C11	45	19	GNDE	95	
C10	46	20	B3	96	
C9	47	21	B4	97	
C8	48	22	B5	98	
VDDE	51		B6	99	

100 TQFP No Connects:1-2, 5, 7, 22, 24, 27-28, 49-50, 53, 55, 70, 72, 77-78

3128XL Programming Signal Definitions

	TQ144	VQ100		TQ144	VQ100
Function	Pin Number	Pin Number	Function	Pin Number	Pin Number
VDDE		3	G1	55	37
GNDE	3		G0	56	
F1/TDI	4 (TDI)	4 (TDI)	GNDI	57	38
F2	5	5	GNDI	57	38
F3	6	6	VDDI	58	39
F4	7	7	VDDI	58	39
F5	8	8	GNDE	59	
F6	9	9	D0	60	
F10	10	10	D1	61	40
F11	11		D2	62	41
F12	12		D3	63	42
PORT_EN	13 (PE)	11 (PE)	GNDE	64	43
F13	14	12	D4	65	44
F14	15	13	D5	66	45
F15	16	14	D6	67	46
GNDE	17		D10	68	47
H0	18		D11	69	48
H1/TMS	20 (TMS)	15 (TMS)	D12	70	49
H2	21	16	D13	71	50
H3	22	17	D14	72	
H4	23		VDDE	73	
VDDE	24	18	D15	74	
H5	25	19	VDDE	76	51
H6	26	20	C15	77	
H10	27	21	C14	78	52
H11	28	22	C13	79	53
H12	29	23	C12	80	54
H13	30	24	C11	81	55
H14	31	25	C10	82	56
H15	32		C6	83	57
GNDE	33	26	C5	84	58
G15	37		GNDE	85	59
G14	38		C4	86	
G13	39	27	C3	87	60
G12	40	28	C2	88	61
G11	41	29	C1/TCK	89 (TCK)	62 (TCK)
G10	42	30	C0	90	
G6	44	31	A15	91	
G5	45	32	A14	92	63
G4	46	33	A13	93	64
VDDE	50	34	A12	94	65
VDDI	51		VDDE	95	66
GNDI	52		A11	96	
G3	53	35	A10	97	67
G2	54	36	A6	98	68

	TQ144	VQ100		TQ144	VQ100
Function	Pin Number	Pin Number	Function	Pin Number	Pin Number
A5	99	69	CLK3/IN3	125	87
A4	100	70	CLK2/IN2	126	88
A3	101	71	CLK1/IN1	127	89
A2	102	72	CLK0/IN0	128	90
A1/TDO	104 (TDO)	73 (TDO)	GNDI	129	
GNDE	105	74	VDDI	130	91
A0	106		E15	131	
B0	107	75	E14	132	92
B1	109	76	E13	133	93
B2	110	77	E12	134	94
B3	111	78	GNDE	135	95
B4	112	79	E11	136	
B5	113	80	E10	137	
B6	114	81	E6	138	96
VDDE	115	82	E5	139	97
B10	116	83	E4	140	98
B11	117	84	E3	141	99
B12	118	85	E2	142	100
B13	119		E1	143	1
B14	120		VDDE	144	
B15	121		E0	1	2
VDDI	123		F0	2	
GNDI	124	86			
144 LQFP No Connects:	19, 34-36, 43, 47-49, 75, 103, 108, 122				

3256XL Programming Signal Definitions

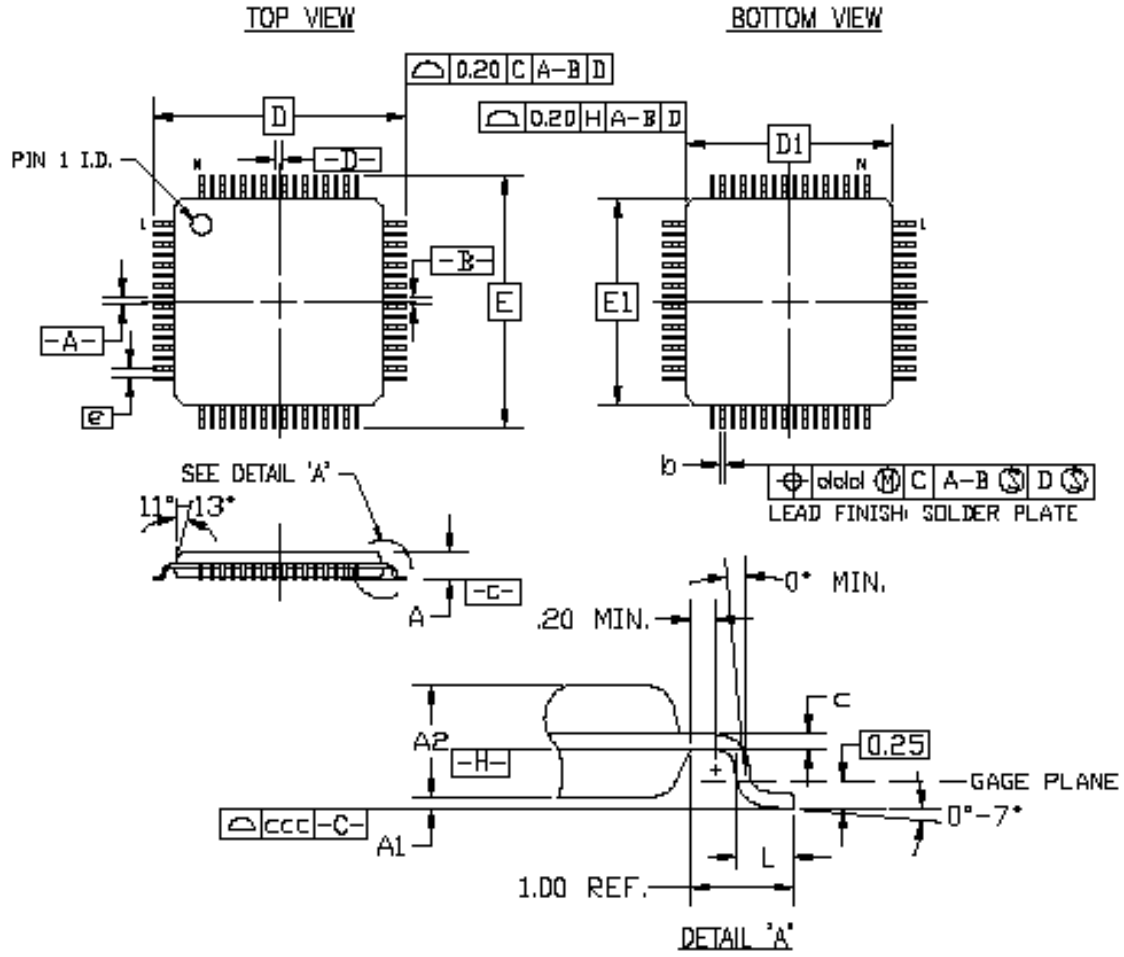
	PQ208	TQ144		PQ208	TQ144
Function	Pin No.	Pin No.	Function	Pin No.	Pin No.
I1	154	1	P11	112	32
I0	153	2	GNDE		33
VDDE			P12	111	
GNDE	152	3	P13	110	34
J0/TDI[0]	151	4 (TDI)	P14	109	35
J1	150		P15	108	36
J2	149	5	VDDE	107	
J3	148	6	GNDE		
J4	147	7	O15	102	37
J11	146	8	O14	101	
J12	145		O13	100	38
J13	144	9	O12	99	39
VDDE	143		O11	98	
J14	142	10	O4	97	40
J15	141	11	O3	96	41
L0	140		O2	95	42
L1	139		GNDE	94	
L2	138	12	O1	93	43
PORT_EN		13	O0	92	44
L3	137	14	M15	91	45
L4	136	15	M14	90	
L11	135	16	M13	89	46
GNDE	134	17	M12	88	47
L12	133		M11	87	48
L13	132	18	M4	86	49
L14	131	19	VDDE	85	50
L15	130		M3	84	
N0/TMS[0]	129	20 (TMS)	VDDI	83	51
N1	128		GNDI	82	52
N2/TMS[1]	127 (TMS)	21	M2	81	53
N3	126	22	M1	80	54
VDDE	125		M0	79	
N4	124	23	F0	78	
VDDE		24	F1	77	55
N11	123	25	F2	76	56
N12	122		GNDI	75	57
N13	121	26	GNDI	75	57
N14	120	27	VDDI	74	58
N15	119	28	VDDI	74	58
P0	118		F3	73	
P1	117		GNDE	72	59
PORT_EN	116		F4	71	60
P2	115	29	F11	70	61
P3	114	30	F12	69	62
P4	113	31	GNDE		

	PQ208	TQ144		PQ208	TQ144
Function	Pin No.	Pin No.	Function	Pin No.	Pin No.
F13	68	63	VDDE	23	
GNDE		64	C11	22	92
F14	67		C4	21	93
F15	66	65	C3	20	94
H0	65	66	VDDE		95
H1	64	67	C2	19	96
VDDE	63		C1	18	97
H2	62	68	C0	17	98
H3	61	69	A15	16	
H4	60		A14	15	
H11	59	70	GNDE	14	
H12	58		A13	13	99
H13	57	71	A12	12	100
H14	56		A11	11	101
H15	55	72	A4	10	102
VDDE		73	A3	9	103
GNDE	50		A2/TDO[0]	8	104 (TDO)
G15	49		A1	7	
G14	48	74	GNDE		105
G13	47	75	A0	6	106
VDDE		76	VDDE	5	
G12	46		B0	4	107
G11	45	77	B1	3	108
G4	44	78	B2	206	
G3	43	79	B3	205	
G2	42	80	B4	204	109
VDDE	41		B11	203	110
G1	40		B12	202	111
G0	39	81	B13	201	
E15	38		GNDE	200	
E14	37	82	B14	199	112
E13	36	83	B15	198	113
E12	35		D0	197	114
E11	34	84	VDDE		115
GNDE		85	D1	196	116
E4	33	86	D2	195	117
GNDE	32		D3	194	
E3	31	87	D4	193	118
E2/TCK[1]	30 (TCK)	88	D11	192	119
E1	29		VDDE	191	
E0/TCK[0]	28	89 (TCK)	D12	190	120
C15	27		D13/TDO[1]	189 (TDO)	121
C14	26	90	D14	188	
C13	25	91	D15	187	122
C12	24		VDDI	186	123

	PQ208	TQ144		PQ208	TQ144
Function	Pin No.	Pin No.	Function	Pin No.	Pin No.
GNDI	185	124	K4	172	137
GNDI	185	124	K3	171	
CLK3/IN3	184	125	K2	170	138
CLK2/IN2	183	126	K1	169	
CLK1/IN1	182	127	K0	168	
CLK0/IN0	181	128	I15	167	139
GNDI	180	129	I14	166	140
VDDI	179	130	VDDE	165	
K15	178	131	I13	164	141
K14	177	132	I12	163	142
K13/TDI[1]	176 (TDI)	133	I11	162	
K12	175	134	I4	161	143
GNDE	174	135	VDDE		144
K11	173	136	I3	160	
	208 PQFP No Connects: 1, 2, 51, 52, 53, 54, 103, 104, 105, 106, 155, 156, 157, 158, 207, 208				

.

VQFP Packages - VQ44, VQ64, VQ100



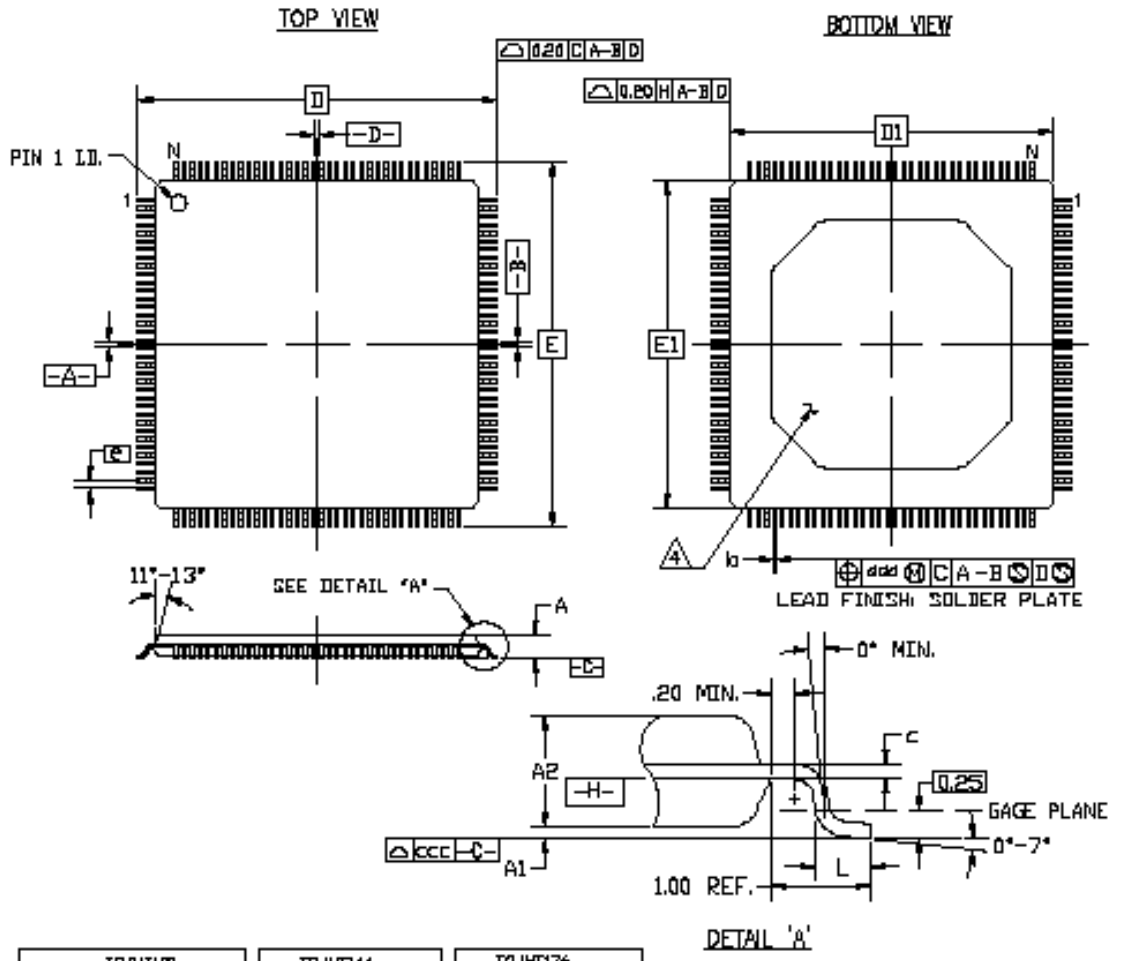
SYMBOL	VQ44 MILLIMETERS			VQ64 MILLIMETERS			VQ100 MILLIMETERS		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	---	---	1.20	---	---	1.20	---	---	1.20
A_1	0.05	---	0.15	0.05	0.10	0.15	0.05	0.10	0.15
A_2	0.95	1.00	1.05	0.95	1.00	1.05	0.95	1.00	1.05
D/E	12.00 BSC			12.00 BSC			16.00 BSC.		
D_1/E_1	10.00 BSC			10.00 BSC.			14.00 BSC.		
b	0.30	0.37	0.45	0.17	0.22	0.27	0.17	0.22	0.27
c	0.09	---	0.20	0.09	---	0.20	0.09	---	0.20
e	0.80 BSC.			0.50 BSC.			0.50 BSC.		
L	0.45	0.60	0.75	0.45	0.60	0.75	0.45	0.60	0.75
ccc	---	---	0.10	---	---	0.08	---	---	0.08
ddd	---	---	0.20	---	---	0.08	---	---	0.08
N	44			64			100		
REF.	JEDEC MS-026-ACB			JEDEC MS-026-ACD			JEDEC MS-026-AED		

NOTES:

1. ALL DIMENSIONS AND TOLERANCES CONFORM TO ANSI Y14.5M-1982.
2. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE MOLD PROTRUSION SHALL NOT EXCEED 0.25mm PER SIDE.
3. THE TOP OF PACKAGE MAY BE SMALLER THAN THE BOTTOM OF PACKAGE BY 0.15mm.

44, 64, 100-PIN PLASTIC VERY THIN QFP (VQ44, VQ64, VQ100)

TQFP/HTQFP Packages - TQ100, TQ144, TQ176, HT100, HT144, HT176



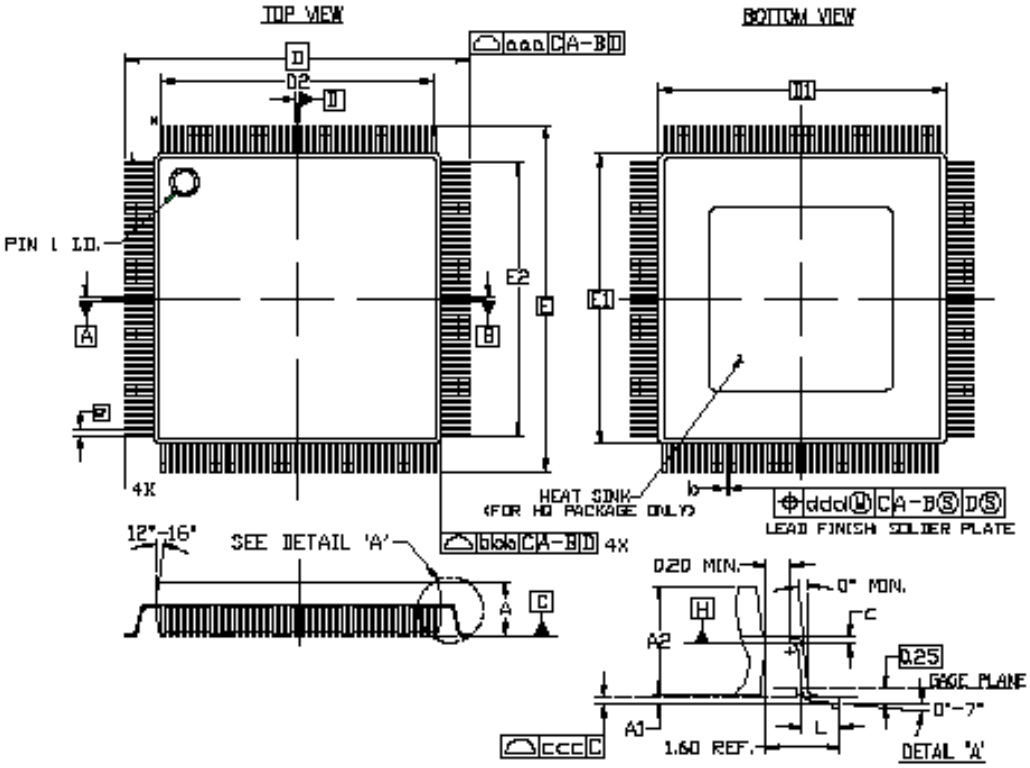
SYMBOL	TQ/HT100 MILLIMETERS			TQ/HT144 MILLIMETERS			TQ/HT176 MILLIMETERS		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	---	---	1.60	---	---	1.60	---	---	1.60
A1	0.05	---	0.15	0.05	0.10	0.15	0.05	0.10	0.15
A2	1.35	1.40	1.45	1.35	1.40	1.45	1.35	1.40	1.45
D/E	16.00 BSC			22.00 BSC			26.00 BSC		
D1/E1	14.00 BSC			20.00 BSC			24.00 BSC		
L	0.45	0.60	0.75	0.45	0.60	0.75	0.45	0.60	0.75
e	0.50 BSC			0.50 BSC			0.50 BSC		
b	0.17	0.22	0.27	0.17	0.22	0.27	0.17	0.22	0.27
c	0.09	---	0.20	0.09	---	0.20	0.09	---	0.20
ccc	---	---	0.08	---	---	0.08	---	---	0.08
ddd	---	---	0.08	---	---	0.08	---	---	0.08
N	100			144			176		
REF.	JEDEC MS-026-BED			JEDEC MS-026-BFB			JEDEC MS-026-BGA		

NOTE:

1. ALL DIMENSIONS AND TOLERANCES CONFORM TO ANSI Y14.5-1982
2. DIMENSION D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE MOLD PROTRUSION SHALL NOT EXCEED 0.25mm PER SIDE.
3. PACKAGE TOP DIMENSION MAY BE SMALLER THAN THE BOTTOM DIMENSION BY 0.15mm.
4. THE SAME PACKAGE DIMENSIONS APPLY FOR THERMALLY ENHANCED PRODUCTS, HEAT SINK IS ADDED. THE PACKAGE CODE IS "HT".

100, 144, 176-PIN TQFP/HEAT SINK TQFP (TQ/HT100, 144, 176)

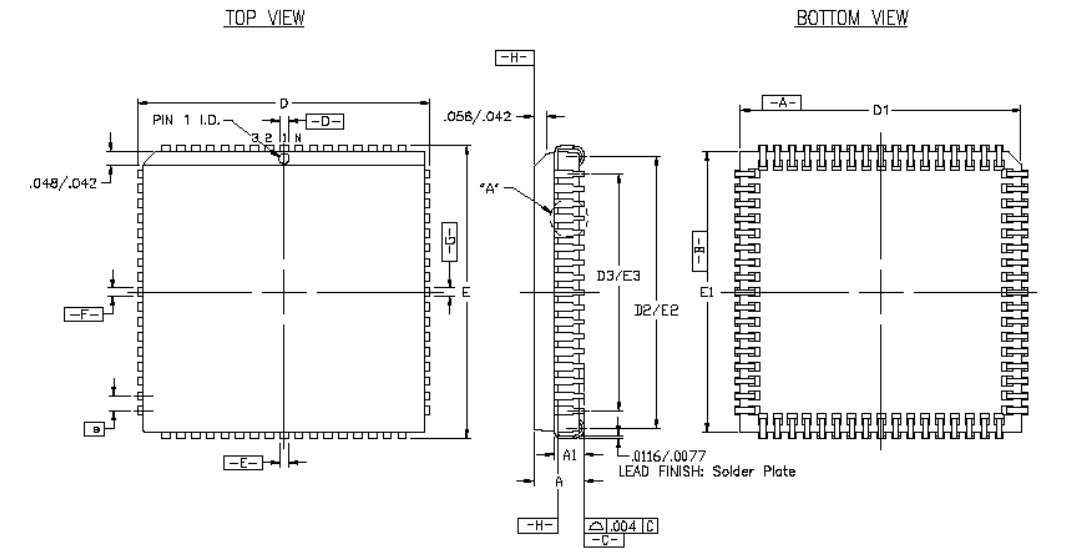
PQ/HQFP Packages - PQ44, PQ160, PQ208, PQ240, HQ160, HQ208, HQ240



	PQ44			PQ/HQ160			PQ/HQ208			PQ/HQ240		
	MILLIMETERS			MILLIMETERS			MILLIMETERS			MILLIMETERS		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	---	2.15	2.35	---	3.70	4.10	---	3.70	4.10	---	3.70	4.10
A_1	0.05	---	0.25	0.25	0.33	0.50	0.25	0.33	0.50	0.25	0.38	0.50
A_2	1.95	2.00	2.10	3.20	3.40	3.60	3.20	3.40	3.60	3.20	3.40	3.60
D/E	13.20 BSC			31.20 BSC			30.60 BSC			34.60 BSC		
D_1/E_1	10.00 BSC			28.00 BSC			28.00 BSC			32.00 BSC		
D_2/E_2	8.00 REF.			25.35 REF.			25.50 REF.			29.50 REF.		
L	0.73	0.88	1.03	0.73	0.88	1.03	0.50	0.60	0.75	0.50	0.60	0.75
e	0.80 BSC.			0.65 BSC.			0.50 BSC.			0.50 BSC.		
b	0.30	---	0.45	0.22	---	0.40	0.17	0.22	0.27	0.17	---	0.27
c	0.13	---	0.23	0.13	---	0.23	0.09	---	0.20	0.09	---	0.20
aaa	---	---	0.25	---	---	0.25	---	---	0.25	---	---	0.25
bbb	---	---	0.20	---	---	0.20	---	---	0.20	---	---	0.20
ccc	---	---	0.10	---	---	0.10	---	---	0.08	---	---	0.08
ddd	---	---	0.20	---	---	0.13	---	---	0.08	---	---	0.08
N	44			160			208			240		
REF.	JEDEC MS-022-AB			JEDEC MS-022-DDL			JEDEC MO-143-FA-1			JEDEC MO-143-GA		

NOTES:
1. ALL DIMENSIONS AND TOLERANCES CONFORM TO ANSI Y14.5M-1994.
2. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE MOLD PROTRUSION SHALL NOT EXCEED 0.25mm PER SIDE.
3. PACKAGE TOP DIMENSIONS MAY BE SMALLER THAN THE BOTTOM DIMENSIONS BY 0.20mm.
4. THE SAME PACKAGE DIMENSIONS APPLY FOR THERMALLY ENHANCED PRODUCTS. HEAT SINK IS ADDED. THE PACKAGE CODE IS "HQ".

44, 160, 208, 240-PIN PQFP/HEAT SINK PQFP (PQ44, PQ/HQ160, 208, 240)



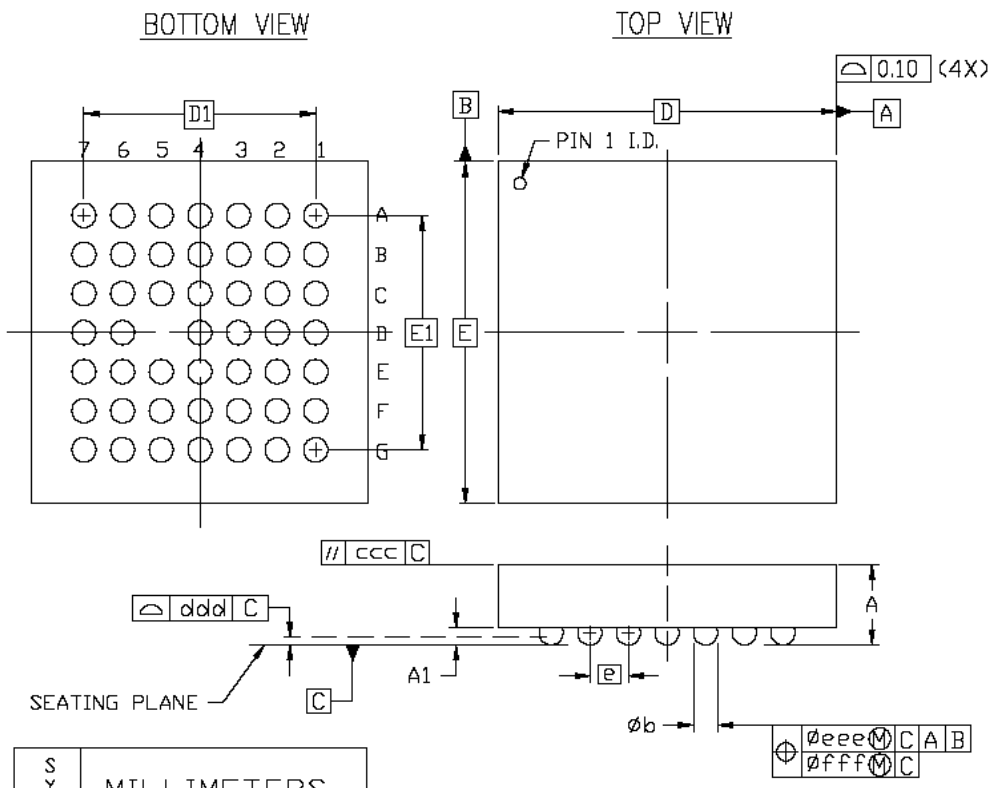
SYMBOL	INCHES									
	MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	MIN.	MAX.
A	.165	.180	.165	.180	.165	.180	.165	.200	.165	.200
A_1	.090	.120	.099	.110	.090	.120	.090	.130	.090	.130
D/E	.385	.395	.485	.495	.685	.695	.985	.995	1.185	1.195
D_1/E_1	.350	.356	.450	.456	.650	.656	.950	.958	1.150	1.158
D_2/E_2	.290	.330	.390	.430	.590	.630	.890	.930	1.090	1.130
D_3/E_3	.200 REF.		.300 REF.		.500 REF.		.800 REF.		1.000 REF.	
e	.050 BSC		.050 BSC		.050 BSC		.050 BSC		.050 BSC	
N	20		28		44		68		84	

NOTES:

1. ALL DIMENSIONS AND TOLERANCES CONFORM TO ANSI Y14.5M-1982.
2. DIMENSIONS 'D1' AND 'E1' DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 PER SIDE.
3. 'N' IS NUMBER OF TERMINALS.
4. CONFORM TO JEDEC MO-047
5. TOP OF PACKAGE MAY BE SMALLER THAN BOTTOM BY .010".

20, 28, 44, 68 and 84–PIN PLCC (PC20 THRU PC84)

Chip Scale Package - CS48



SYMBOL	MILLIMETERS		
	MIN.	NOM.	MAX.
A	1.30	---	1.80
A_1	0.35	0.40	0.45
D/E	7.00 BSC		
D_1/E_1	4.80 BSC		
e	0.80 BSC		
øb	0.45	0.50	0.55
ccc	---	---	0.10
ddd	---	---	0.10
eee	---	---	0.15
fff	---	---	0.08
M	7		

NOTES:

1. ALL DIMENSIONING AND TOLERANCING CONFORM TO ANSI Y14.5M-1994
2. SYMBOL "M" IS THE PIN MATRIX SIZE.

48-BALL CHIP SCALE BGA (CS48)

THE ATTACHED FILE CONTAINS ALL ABSOLUTE ADDRESSES A8 TO A0 IN BINARY
FOR THE XILINX XCR3256XL DEVICE.
ADDRESS BITS A7-A0 ARE A GRAY CODE COUNT OF ADDRESSES AND
A8 SELECTS UPPER (1) OR LOWER (0) ROW AT THE GIVEN ADDRESS.

DEVICE PACKAGES AND PIN COUNTS

1. TQ144
2. PQ208
3. CS280

DEVICE FUSE COUNT

1. TQ144 - 115868
2. PQ208 - 115868
3. CS280 - 115868

ERASE ADDRESSES (FOR BULK ERASE)

No address required for bulk erase;
erase is performed with ISP command.

Note: When reading an erased device, logical one ('1') is
the correct logical value. Logical zero ('0') is
the correct logical value for a programmed bit.

OPERATING MODES

Programming and erasing is possible via ISP only.

READ SECURITY ADDRESS
==========

A security feature is available which when programmed, disables access
to the information contained in the Non-volatile Eprom array. Any operation
reading the user array should not be allowed. A 'read secured' device
can still be erased and only then, reprogrammed.

While checking the device for read security before an operation, if the
following address is programmed, consider the device read secured.
A logical one ('1') if read permitted or a logical zero ('0') if
read secured.

Note: The jedec file DOES NOT contain a fuse for this security bit.
The security bit must be set with an independent operation to the address shown below.

1. Upper address: 110111000 bit: D0

PORTEN ADDRESS
==========

A feature is available which when programmed, disables the JTAG port and converts the port pins to fully-functional I/Os. This occurs upon subsequent power-cycling or execution of a "reinit" ISP command. See datasheet for discussion of reclaiming JTAG port on a device programmed with port pins as I/Os.

While checking the device for configuration of the JTAG port pins before an operation, if the following address is programmed, consider the device to have the JTAG port disabled and fully-functional I/Os enabled.
A logical one ('1') if JTAG port is enabled or a logical zero ('0') if JTAG port is disabled and fully-functional I/Os are enabled.

Note: The jedec file DOES contain a fuse for this porten bit.
The porten bit must be set to zero with an independent operation to the address shown below.

1. Upper address: 110111000 bit: D1

MANUFACTURER'S CODE
==========

Available via ISP only

PRODUCT AND VERSION CODE
==========

Available via ISP only

SIGNATURE STRING ADDRESSES
==========

The signature string is a user definable code which can be loaded at anytime with appropriate programming equipment. The programmer should allow reading and displaying of the string. These data are erased with ISP erase operation and programmed with the same procedure used for ordinary

configuration bits. The user should be able to define the
string during a programming session.

These signature string addresses and bit locations are as follows:
1. Lower address: 010111001 bits: D5-19, D66-111, D208-D303
2. Upper address: 110111001 bits: D5-19, D66-111, D208-D303

****

THE ATTACHED FILE CONTAINS ALL ABSOLUTE ADDRESSES A7 TO A0 IN BINARY
FOR THE XILINX XCR3128XL DEVICE.
ADDRESS BITS A6-A0 ARE A GRAY CODE COUNT OF ADDRESSES AND
A7 SELECTS UPPER (1) OR LOWER (0) ROW AT THE GIVEN ADDRESS.

DEVICE PACKAGES AND PIN COUNTS

1. VQ100
2. TQ144

DEVICE FUSE COUNT

1. VQ100 - 52009 (0-52008)
2. TQ144 - 52009 (0-52008)

ERASE ADDRESSES (FOR BULK ERASE)

No address required for bulk erase;
erase is performed with ISP command.

Note: When reading an erased device, logical one ('1') is
the correct logical value. Logical zero ('0') is
the correct logical value for a programmed bit.

OPERATING MODES

Programming and erasing is possible via ISP only.

READ SECURITY ADDRESS
==========

A security feature is available which when programmed, disables access
to the information contained in the Non-volatile Eprom array. Any operation
reading the user array should not be allowed. A 'read secured' device
can still be erased and only then, reprogrammed.

While checking the device for read security before an operation, if the

following address is programmed, consider the device read secured.
A logical one ('1') if read permitted or a logical zero ('0') if
read secured.

Note: The jedec file DOES NOT contain a fuse for this security bit.
The security bit must be set with an independent operation to the
address shown below.

1. Upper address: 11011100 bit: D0

PORTEN ADDRESS
==========

A feature is available which when programmed, disables the JTAG port
and converts the port pins to fully-functional I/Os. This occurs
upon subsequent power-cycling or execution of a "reinit" ISP command.
See datasheet for discussion of reclaiming JTAG port on a device
programmed with port pins as I/Os.

While checking the device for configuration of the JTAG port pins
before an operation, if the following address is programmed, consider
the device to have the JTAG port disabled and fully-functional I/Os
enabled.
A logical one ('1') if JTAG port is enabled or a logical zero ('0') if
JTAG port is disabled and fully-functional I/Os are enabled.

Note: The jedec file DOES contain a fuse for this porten bit.
The porten bit must be set to zero with an independent operation
to the address shown below.

1. Upper address: 11011100 bit: D1

MANUFACTURER'S CODE
==========

Available via ISP only

PRODUCT AND VERSION CODE
==========

Available via ISP only

SIGNATURE STRING ADDRESSES
==========

The signature string is a user definable code which can

be loaded at anytime with appropriate programming equipment.
The programmer should allow reading and displaying of
the string. These data are erased with ISP erase operation
and programmed with the same procedure used for ordinary
configuration bits. The user should be able to define the
string during a programming session.

These signature string addresses and bit locations are as follows:
1. Lower address: 01011101 bits: D5-19, D47-73, D170-265
2. Upper address: 11011101 bits: D5-19, D47-73, D170-265

****

THE ATTACHED FILE CONTAINS ALL ABSOLUTE ADDRESSES A7 TO A0 IN BINARY
FOR THE XILINX XCR3064XL DEVICE.
ADDRESS BITS A6-A0 ARE A GRAY CODE COUNT OF ADDRESSES AND
A7 SELECTS UPPER (1) OR LOWER (0) ROW AT THE GIVEN ADDRESS.

DEVICE PACKAGES AND PIN COUNTS

1. VQ44
2. VQ100

DEVICE FUSE COUNT

1. VQ44 - 24481 (0-24480)
2. VQ100 - 24481 (0-24480)

ERASE ADDRESSES (FOR BULK ERASE)

No address required for bulk erase;
erase is performed with ISP command.

Note: When reading an erased device, logical one ('1') is
the correct logical value. Logical zero ('0') is
the correct logical value for a programmed bit.

OPERATING MODES

Programming and erasing is possible via ISP only.

READ SECURITY ADDRESS
==========

A security feature is available which when programmed, disables access
to the information contained in the Non-volatile Eprom array. Any operation

reading the user array should not be allowed. A 'read secured' device can still be erased and only then, reprogrammed.

While checking the device for read security before an operation, if the following address is programmed, consider the device read secured.
A logical one ('1') if read permitted or a logical zero ('0') if read secured.

Note: The jedec file DOES NOT contain a fuse for this security bit.
The security bit must be set with an independent operation to the address shown below.

1. Upper address: 11011100 bit: D0

PORTEN ADDRESS
==========

A feature is available which when programmed, disables the JTAG port and converts the port pins to fully-functional I/Os. This occurs upon subsequent power-cycling or execution of a "reinit" ISP command. See datasheet for discussion of reclaiming JTAG port on a device programmed with port pins as I/Os.

While checking the device for configuration of the JTAG port pins before an operation, if the following address is programmed, consider the device to have the JTAG port disabled and fully-functional I/Os enabled.
A logical one ('1') if JTAG port is enabled or a logical zero ('0') if JTAG port is disabled and fully-functional I/Os are enabled.

Note: The jedec file DOES contain a fuse for this porten bit.
The porten bit must be set to zero with an independent operation to the address shown below.

1. Upper address: 11011100 bit: D1

MANUFACTURER'S CODE
==========

Available via ISP only

PRODUCT AND VERSION CODE
==========

Available via ISP only

SIGNATURE STRING ADDRESSES
==========

The signature string is a user definable code which can
be loaded at anytime with appropriate programming equipment.
The programmer should allow reading and displaying of
the string. These data are erased with ISP erase operation
and programmed with the same procedure used for ordinary
configuration bits. The user should be able to define the
string during a programming session.

These signature string addresses and bit locations are as follows:
1. Lower address: 01011101 bits: D5-9, D75-122
2. Upper address: 11011101 bits: D5-9, D75-122

****

THE ATTACHED FILE CONTAINS ALL ABSOLUTE ADDRESSES A6 TO A0 IN BINARY
FOR THE XILINX XCR3032XL DEVICE.
ADDRESS BITS A5-A0 ARE A GRAY CODE COUNT OF ADDRESSES AND
A6 SELECTS UPPER (1) OR LOWER (0) ROW AT THE GIVEN ADDRESS.

DEVICE PACKAGES AND PIN COUNTS

1. VQ44
2. PC44
3. CS48

DEVICE FUSE COUNT

1. VQ44 - 11529 (0-11528)
2. PC44 - 11529 (0-11528)
3. CS48 - 11529 (0-11528)

ERASE ADDRESSES (FOR BULK ERASE)

No address required for bulk erase;
erase is performed with ISP command.

Note: When reading an erased device, logical one ('1') is
the correct logical value. Logical zero ('0') is
the correct logical value for a programmed bit.

OPERATING MODES

Programming and erasing is possible via ISP only.

READ SECURITY ADDRESS
==========

A security feature is available which when programmed, disables access to the information contained in the Non-volatile Eprom array. Any operation reading the user array should not be allowed. A 'read secured' device can still be erased and only then, reprogrammed.

While checking the device for read security before an operation, if the following address is programmed, consider the device read secured.
A logical one ('1') if read permitted or a logical zero ('0') if read secured.

Note: The jedec file DOES NOT contain a fuse for this security bit.
The security bit must be set with an independent operation to the address shown below.

1. Upper address: 1101110 bit: D0

PORTEN ADDRESS
==========

A feature is available which when programmed, disables the JTAG port and converts the port pins to fully-functional I/Os. This occurs upon subsequent power-cycling or execution of a "reinit" ISP command. See datasheet for discussion of reclaiming JTAG port on a device programmed with port pins as I/Os.

While checking the device for configuration of the JTAG port pins before an operation, if the following address is programmed, consider the device to have the JTAG port disabled and fully-functional I/Os enabled.
A logical one ('1') if JTAG port is enabled or a logical zero ('0') if JTAG port is disabled and fully-functional I/Os are enabled.

Note: The jedec file DOES contain a fuse for this porten bit.
The porten bit must be set to zero with an independent operation to the address shown below.

1. Upper address: 1101110 bit: D1

MANUFACTURER'S CODE
==========

Available via ISP only

PRODUCT AND VERSION CODE
==========

Available via ISP only

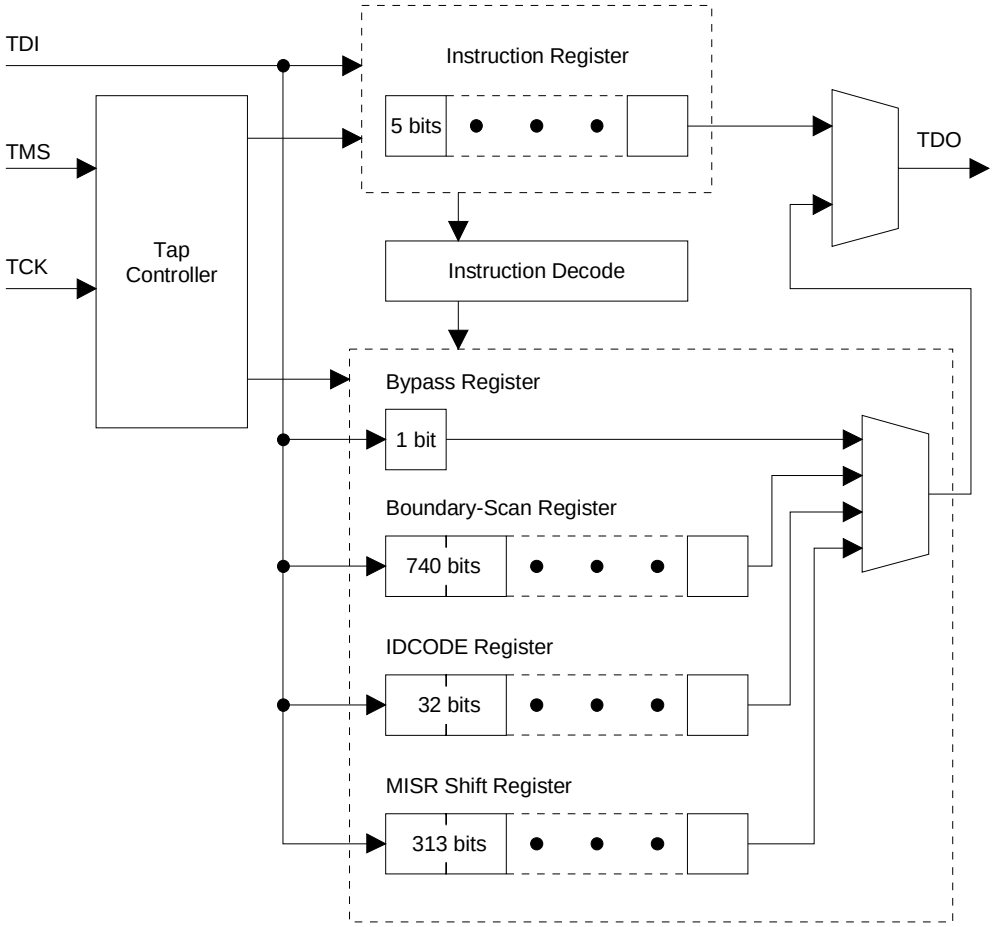
SIGNATURE STRING ADDRESSES
==========

The signature string is a user definable code which can be loaded at anytime with appropriate programming equipment. The programmer should allow reading and displaying of the string. These data are erased with ISP erase operation and programmed with the same procedure used for ordinary configuration bits. The user should be able to define the string during a programming session.

These signature string addresses and bit locations are as follows:
1. Lower address: 0101111 bits: D5-9, D66-113
2. Upper address: 1101111 bits: D5-9, D66-113

****

Code	*Instruction*	*Register used*
00000	Extest	Boundary-Scan
00001	Idcode	Ident
00010	Sample	Boundary-Scan
00011	Intest	Boundary-Scan
00100	Stctest	Boundary-Scan
00101	High-Z	Bypass
00110	Clamp	Bypass
00111	Write	ISP Shift
01000	EOTF	ISP Shift
01001	Enable	ISP Shift
01010	Erase	ISP Shift
01011	Program	ISP Shift
01100	Verify	ISP Shift
01101	Init	Bypass
01110	Read	ISP Shift
10000	Disable	ISP Shift
10001	Testmode	Bypass
11111	Bypass	Bypass


TDI
TMS
TCK
Tap Controller
Instruction Register
5 bits
Instruction Decode
TDO
Bypass Register
1 bit
Boundary-Scan Register
740 bits
IDCODE Register
32 bits
MISR Shift Register
313 bits

JTAG Register Description

Register	# of bits	Register Description
Instruction Register	5	The Instruction Register is a shift-register-based design which allows an instruction to be shifted into a device. The instruction shifted into the register is latched at the completion of the shifting process. The instruction is used to select the BST or ISP operation and/or the data register to be accessed. The parallel output from the Instruction Register is latched to ensure that the BST and ISP logic is protected from the transient data patterns that will occur in its shift-register stages as new instruction data is entered.
Bypass Register	1	The Bypass Register contains a single shift-register stage and is used to provide a minimum length serial path between the TDI and TDO pins of a component when no test or program operation of that component is required. This allows more rapid movement of test/program data to and from other components on a circuit pack that are required to perform test/program operations.
Boundary Scan Register	740	The Boundary-Scan Register allows testing of circuitry external to the CPLD and also permits the system signals flowing into and out of the CPLD logic to be sampled and examined without causing interference with the normal operation of the CPLD logic. The Boundary-Scan Register is a long shift register composed of all the Boundary-Scan cells at the pins of the device.
IDCODE Register	32	This register must consist of a 32 bit shift-register, parallel-in and serial out. The register contains the following information: Bit(s) — Usage 0 — 1 - pre-defined 1-11 — Manufacturing Identity 12-27 — Part Number 28-31 — Version
ISP Shift Register	9 address bits 304 data bits 313 bits total	Used to address the EEPROM row and contains the data that is being written into or read from the EEPROM array.

JTAG Pin Description

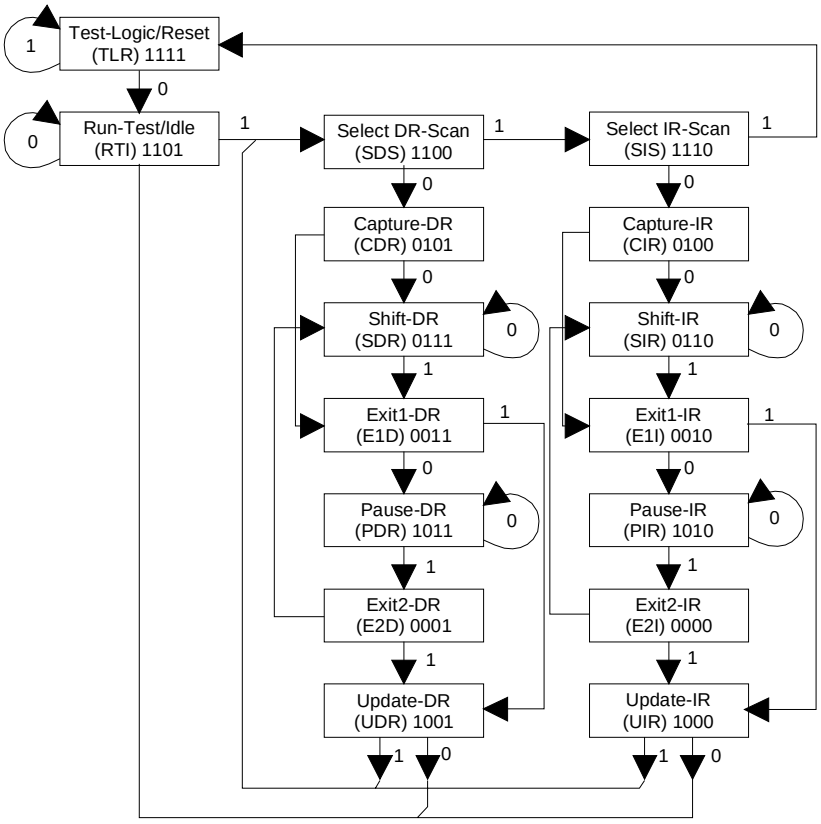
Pins	Name	Description
TCK	Test Clock Output	Clock pin to shift the serial data and instructions in and out of the TDI and TDO pins, respectively. TCK is also used to clock the TAP Controller state machine.
TMS	Test Mode Select	Test mode select pin selects the JTAG instruction mode. TMS should be driven high during user mode operation.
TDI	Test Data Input	Serial input pin for instructions and test data. Data is shifted in on the rising edge of TCK.
TDO	Test Data Output	Serial output pin for instructions and test data. Data is shifted out on the falling edge of TCK. The signal is tri-stated if data is not being shifted out of the device.

Supported Low Level JTAG Boundary-Scan Commands

Instruction (Instr. Code) ***Register Used***	**Description**
BYPASS (11111) *Bypass Register*	Required. Places the 1 bit bypass register between the TDI and TDO pins, which allows the BST data to pass synchronously through the selected device to adjacent devices during normal device operation without affecting the operation of the IC.
EXTEST (00000) *Boundary-Scan Register*	Required. Puts the IC into external mode. Forces data externally from the boundary scan outputs and receives data externally to the boundary scan inputs.
IDCODE (00001) *Boundary-Scan Register*	Optional. Selects the IDCODE register and places it between TDI and TDO, allowing the IDCODE to be serially shifted out of TDO. The IDCODE instruction permits blind interrogation of the components assembled onto a circuit pack. Thus, in circumstances where the component population may vary, it is possible to determine what components exist in a product.
Sample /Preload (00010) *Boundary-Scan Register*	Required. Allows IC to remain in normal mode and selects the boundary scan register to be connected between TDI and TDO. Allows sampling data entering and leaving the device. It allows preload data into the boundary scan register before executing the EXTEST instruction.
INTEST (00011) *Boundary-Scan Register*	Optional. Puts the IC in internal mode and selects the boundary scan register to be connected between TDI and TDO. It allows to drive data into the CORE logic of the IC from the boundary scan inputs and to receive CORE logic output data into the boundary scan outputs.
STCTEST (00100) *Boundary-Scan Register*	Optional. The STCTEST instruction is used by the Philips design community to check the integrity of the JTAG Boundary-Scan structure. This command will be for internal Philips use only and will not be advertised to customers.
HIGHZ (00101) *Bypass Register*	Optional. The HIGHZ instruction places the component in a state in which all of its system logic outputs are placed in an inactive drive state (e.g., high impedance). In this state, an in-circuit test system may drive signals onto the connections normally driven by a component output without incurring the risk of damage to the component. The HIGHZ instruction also forces the Bypass Register between TDI and TDO.
CLAMP (00110) *Bypass Register*	Optional. Forces data externally from the boundary scan outputs and selects the one bit bypass register to be connected between TDI and TDO. The data into the boundary scan register can be preloaded using sample/preload.

Supported Low Level ISP Commands

WRITE (00111) *ISP Sift Register*	To configure a specified address of the SRAM. Selects ISP register to be connected between TDI and TDO. (PRIVATE INSTRUCTION)
EOTF (01000) *ISP Shift Register*	Enters ISP mode in on-the-fly mode, which allows the IC to be in normal mode while the IC EEPROM is re-programmed. (PRIVATE INSTRUCTION)
ENABLE (01001) *ISP Shift Register*	Enters normal ISP mode, puts IC in disable mode.
ERASE (01010) *ISP Shift Register*	Bulk erases the IC EEPROM.
VERIFY (01100) *ISP Shift Register*	Reads the contents of a specified address of the IC EEPROM.
INIT (01101) *Bypass Register*	Starts initialization process, moves data from the EEPROM to SRAM.
READ (01110) *ISP Shift Register*	Reads a specific address from the SRAM. If security bit is programmed, SRAM access is blocked. (PRIVATE INSTRUCTION)
DISABLE (10000) *ISP Shift Register*	Leaves the ISP mode.
TESTMODE (10001) *Bypass Register*	Puts the IC into test mode. (PRIVATE INSTRUCTION)
PROGRAM (01011) **ISP**	Program the data in the ISP shift register into the addressed EEPROM row.



JTAG TAP Controller

3.1.2 JTAG TAP Controller

The TAP Controller is a synchronous finite state machine that responds to changes at the TMS and TCK signals of the TAP and controls the sequence of operations of the circuitry defined by the 1149.1 IEEE Standard. This TAP Controller is implemented in the Philips CPLDs.

Test-Logic/Reset: The BST and ISP logic is disabled so that normal operation of the on-chip system logic (i.e. in response to stimuli received through the system pins only) can continue unimpeded. This is achieved by initializing the instruction register to contain the IDCODE instruction. No matter what the original state of the controller, it will enter *Test-Logic-Reset* when TMS is held high for at least 5 rising edges of TCK. The controller remains in this state while TMS is high. Note that the TAP controller will be forced to the *Test-Logic-Reset* controller state at power-up.

Run-Test/Idle: All of the instructions supported by Philips CPLDs do not cause functions to execute in the *Run-Test/Idle* controller state. Thus, all BST and ISP data registers selected by the current instruction shall retain their previous state (i.e. Idle). The instruction does not change while the TAP controller is in this state.

Select-DR-Scan: This is a temporary controller state in which all BST and ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state.

Select-IR-Scan: This is a temporary controller state in which all BST and ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state.

Capture-DR: In this controller state, data may be parallel loaded into the BST data registers selected by the current instruction on the rising edge of TCK. If a BST data register selected by the current instruction does not have parallel input, or if capturing is not required for the selected test, then the register retains its previous state. The instruction does not change while the TAP controller is in this state.

Shift-DR: In this controller state, the BST or ISP data register connected between TDI and TDO as a result of the current instruction shifts data one stage towards its serial output on each rising edge of TCK. BST or ISP data registers that are selected by the current instruction, but are not placed in the serial path, retain their previous state. The instruction does not change while the TAP controller is in this state.

Exit1-DR: This is a temporary state. All BST or ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state.

Pause-DR: This controller state allows shifting of the BST or ISP data register in the serial path between TDI and TDO to be temporarily halted. All BST or ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state.

Exit2-DR: This is a temporary state. All BST or ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state.

Update-DR: Some BST data registers may be provided with a latched parallel output to prevent changes at the parallel output while data is shifted in the associated shift-register path in response to certain instructions (e.g. EXTEST). Data is latched onto the parallel output of these BST data registers from the shift-register path on the falling edge of TCK in the *Update-DR* controller state. The data held at the latched parallel outputs should not change. The instruction does not change while the TAP controller is in this state.

Capture-IR: In this controller state, the shift-register contained in the instruction register loads a pattern of fixed logic values on the rising edge of TCK. In addition, design-specific data may be loaded into shift-register stages that are not required to be set to fixed values. BST or ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state.

Shift-IR: In this controller state, the shift-register contained in the instruction register is connected between TDI and TDO and shifts data one stage towards its serial output on each rising edge of TCK. BST or ISP data registers that are selected by the current instruction, but are not placed in the serial path, retain their previous state. The instruction does not change while the TAP controller is in this state.

Exit1-IR: This is a temporary state. BST or ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state and the instruction register retains its state.

Pause-IR: This controller state allows shifting of the instruction register to be temporarily halted. BST or ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state and the instruction register retains its state.

Exit2-IR: This is a temporary state. BST or ISP data registers selected by the current instruction retain their previous state. The instruction does not change while the TAP controller is in this state and the instruction register retains its state.

Update-IR: The instruction shifted into the instruction register is latched onto the parallel output from the shift-register path on the falling edge of TCK in the *Update-IR* controller state. Once the new instruction has been latched, it becomes the current instruction. BST or ISP data registers selected by the current instruction retain their previous state.

Support the following High Level ISP and JTAG Boundary-Scan Instructions:

ISP Commands (ENABLE):

Bulk_Erase	Erase the EEPROM array.
Blank_Check	Verify that the EEPROM array has been erased.
Program	Program the EEPROM array.
Verify	Verify the EEPROM array.
Pr_Security	Program the Security Bit.
Rd_Security	Read the Security Bit.
Pr_UES	Program the UES.
Usercode	Read the UES.
Program_Verify	Simultaneously Program/Verify the EEPROM.

JTAG Commands:

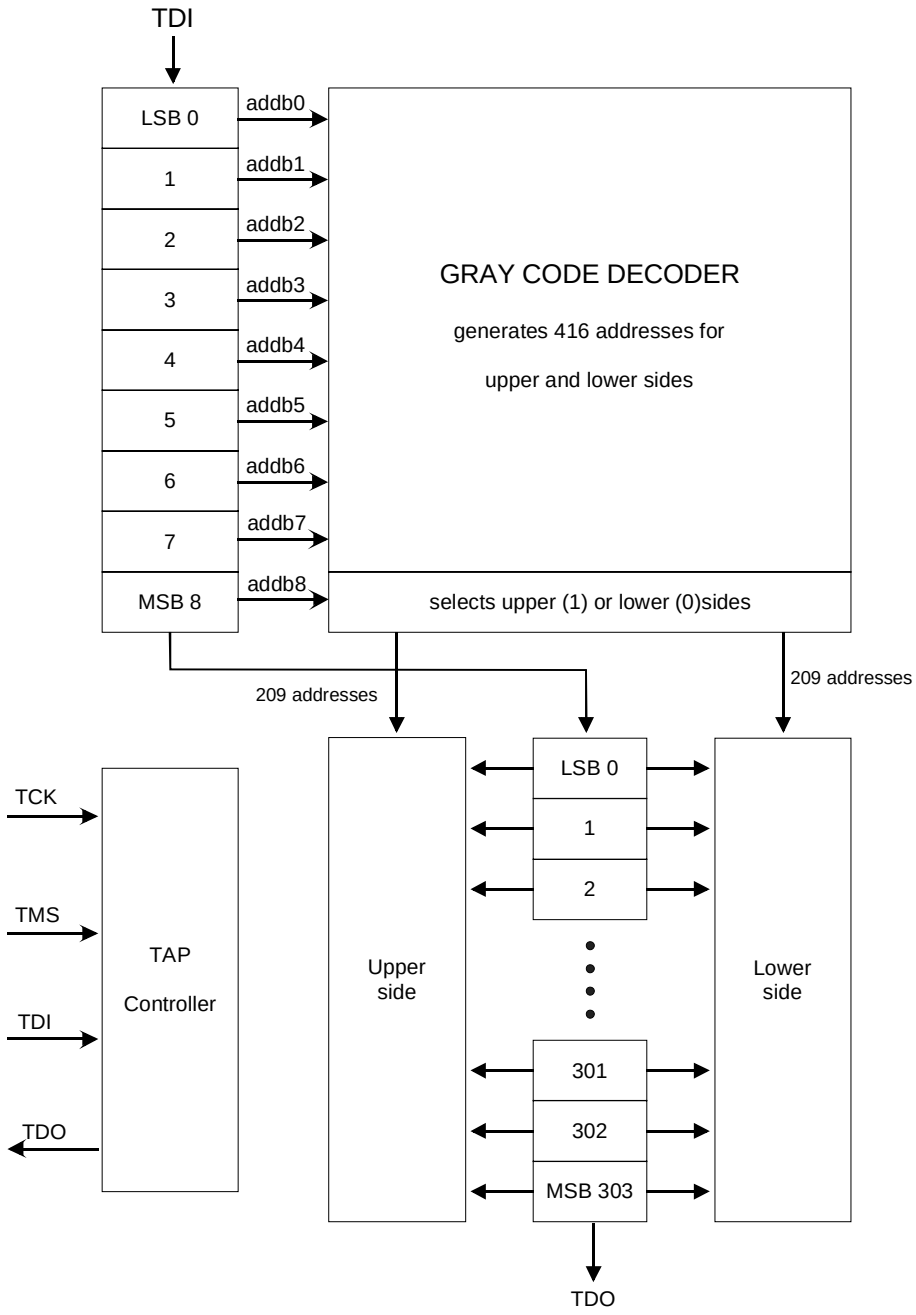
Bypass	Connect TDO to TDO.
Idcode	Read the devices ID code.
Precondition Outputs	Allows the user to specify outputs

High Level ISP and JTAG Boundary-Scan Instructions:

- ISP Commands (ENABLE):

- Bulk_Erase - Erase the EEPROM array.
- Blank_Check - Verify that the EEPROM array has been erased.
- Program - Program the EEPROM array.
- Verify - Verify the EEPROM array.
- Pr_Security - Program the Security Bit.
- Rd_Security - Read the Security Bit.
- Pr_UES - Program the UES.
- Usercode - Read the UES.
- Program_Verify - Simultaneously Program/Verify the EEPROM.
- JTAG Commands:
- Bypass - Connect TDI to TDO.
- Idcode - Read the devices ID code.
- Precondition Outputs - User specified outputs

XPLA3 256 MACROCELL SHIFT REGISTER ARCHITECTURE



XPLA3 256 MACROCELL DEVICE PROGRAMMING EXAMPLE

Bulk_Erase

The Bulk_Erase command erases the entire EEPROM array.

Bulk_Erase Command Flow

Command	Sequence
Bulk_Erase	1. Ensure device is in test-logic/reset state 2. shift in the ENABLE instruction 3. shift in the ERASE instruction 4. pause to allow voltages to settle 5. execute the instruction (erase the contents of the EEPROM array) 6. pause to allow voltages to disccharge 7. shift in the INIT instruction 8. execute the instruction (activate the contents of the EEPROM array) 9. shift in the DISABLE instruction 10. execute the instruction (activate the contents of the EEPROM array)

Bulk_Erase Detailed Algorithm

Step	Transition Conditions	TAP State	CPLD Event Description	Programmer Action
0	TMS = 1	Test-Logic/Reset	BEGIN	BEGIN
Loop0	TMS = 1, TCK = ↑	Test-Logic/Reset	Ensure device in Test-Logic/Reset State	Loop 5 times
1	TMS = 0, TCK = ↑	Run-Test/Idle		
2	TMS = 1, TCK = ↑	Select DR-Scan		
3	TMS = 1, TCK = ↑	Select IR-Scan		
4	TMS = 0, TCK = ↑	Capture-IR		
5	TMS = 0, TCK = ↑	Shift-IR		
Loop1	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1001 (Enable)
6	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Enable MSB)
7	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register ; Set enable flip-flop	
8	TMS = 1, TCK = ↑	Select DR-Scan		
9	TMS = 1, TCK = ↑	Select IR-Scan		
10	TMS = 0, TCK = ↑	Capture-IR		
11	TMS = 0, TCK = ↑	Shift-IR		
Loop2	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bit (0-3)	TDI = 0101 (Erase)
12	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Erase MSB)
13	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
14	TMS = 1, TCK = ↑	Select DR-Scan		
15	TMS = 0, TCK = ↑	Capture-DR		
16	TMS = 1, TCK = ↑	Exit1-DR		
17	TMS = 0, TCK = ↑	Pause-DR		Loop 20 us
18	TMS = 1, TCK = ↑	Exit2-DR		
19	TMS = 1, TCK = ↑	Update-DR		
20	TMS = 0, TCK = ↑	Run-Test/Idle	Execute: Erase the device	Loop for 100 ms
21	TMS = 1, TCK = ↑	Select DR-Scan		
22	TMS = 0, TCK = ↑	Capture-DR		
23	TMS = 1, TCK = ↑	Exit1-DR		
Loop 3	TMS = 0, TCK = ↑	Pause-DR	Discharge hgh voltages	Loop 5ms
24	TMS = 1, TCK = ↑	Exit2-DR		

25	TMS = 1, TCK = ↑	Update-DR		
26	TMS = 1, TCK = ↑	Select DR-Scan		
27	TMS = 1, TCK = ↑	Select IR-Scan		
28	TMS = 0, TCK = ↑	Capture-IR		
29	TMS = 0, TCK = ↑	Shift-IR		
Loop4	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1011 (Init)
30	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Init MSB)
31	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
32	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	Loop for 200us
33	TMS = 1, TCK = ↑	Select DR-Scan		
34	TMS = 1, TCK = ↑	Select IR-Scan		
35	TMS = 0, TCK = ↑	Capture-IR		
36	TMS = 0, TCK = ↑	Shift-IR		
Loop5	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 0000 (Disable)
37	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 1 (Disable MSB)
38	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
39	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	
40	TMS = 1, TCK = ↑	Select DR-Scan		
41	TMS = 1, TCK = ↑	Select IR-Scan		
42	TMS = 1, TCK = ↑	Test-Logic/Reset	One negative edge clock	
43	TMS = 1,	Test-Logic/Reset	In user mode	DONE

XPLA3 256 MACROCELL DEVICE PROGRAMMING EXAMPLE

Blank_Check

The Blank_Check command verifies that the entire EEPROM array has been erased.

Blank_Check Command Flow

Command	Sequence
Blank_Check	1. ensure device is in Test-Logic/Reset state 2. shift in the ENABLE instruction 3. 3. shift in the VERIFY instructionshift in the address of the EEPROM row being verified. 4. Pause 20us to allow voltages to settle 5. execute the command (this transfers the row data into the ISP Shift Register) 6. shift out the data from the ISP Shift Register 7. compare the shifted-out data to the expected data 8. Repeat step 3 though 7 until all EEPROM rows have been checked 9. shift in the INIT instruction 10. execute the instruction (activate the contents of the EEPROM array) 11. shift in the DISABLE instruction 12. execute the instruction (activate the contents of the EEPROM array)

Blank_Check Detailed Algorithm

Step	Transition Conditions	TAP State	CPLD Event Description	Programmer Action
0	TMS = 1	Test-Logic/Reset	BEGIN	BEGIN
Loop0	TMS = 1, TCK = ↑	Test-Logic/Reset	Ensure device in Test-Logic/Reset	Loop 5 times
1	TMS = 0, TCK = ↑	Run-Test/Idle		
2	TMS = 1, TCK = ↑	Select DR-Scan		
3	TMS = 1, TCK = ↑	Select IR-Scan		
4	TMS = 0, TCK = ↑	Capture-IR		
5	TMS = 0, TCK = ↑	Shift-IR		
Loop1	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1001 (Enable)
6	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Enable MSB)
7	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register ;	
8	TMS = 1, TCK = ↑	Select DR-Scan		
9	TMS = 1, TCK = ↑	Select IR-Scan		
10	TMS = 0, TCK = ↑	Capture-IR		
11	TMS = 0, TCK = ↑	Shift-IR		
Loop2	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bit (0-3)	TDI = 0011 (Verify)
12	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Verify MSB)
13	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
14	TMS = 1, TCK = ↑	Select DR-Scan		
15	TMS = 0, TCK = ↑	Capture-DR		
16	TMS = 0, TCK = ↑	Shift-DR		
Loop4	TMS = 0, TCK = ↑	Shift-DR	Shift first address bits (8-1)	TDI = address (MSB, ...)
17	TMS = 1, TCK = ↑	Exit1-DR	Shift first address bit 0	TDI = address (LSB)
Loop3	TMS = 0, TCK = ↑	Pause-DR	Pause to allow voltages to settle	Loop for 20us
Loop3	TMS = 1, TCK = ↑	Exit2-DR	Load address in Address Latch	
Loop3	TMS = 1, TCK = ↑	Update-DR	Load the Shift Register.	
Loop3	TMS = 0, TCK = ↑	Run-Test/Idle		
Loop3	TMS = 1, TCK = ↑	Select DR-Scan		
Loop3	TMS = 0, TCK = ↑	Capture-DR		
Loop3	TMS = 0, TCK = ↑	Shift-DR	Shift out data bit 303	TD0 = data bit 303; TDI=1

Loop5	TMS = 0, TCK = ↑	Shift-DR	Shift out data bits (302 - 0)	TD0 = data bits (302-0);TDI=1
Loop5	TMS = 0, TCK = ↑	Shift-DR	Shift in next address bits (8-1)	
Loop3	TMS = 1, TCK = ↑	Exit1-DR	Shift in next address bit 0	Compare Data
18	**Execute loop3 until all addresses have been loaded. Load dummy address after last data read.**			
19	TMS = 0, TCK = ↑	Pause-DR		
20	TMS = 1, TCK = ↑	Exit2-DR		
21	TMS = 1, TCK = ↑	Update-DR		
22	TMS = 0, TCK = ↑	Run-Test/Idle		
23	TMS = 1, TCK = ↑	Select DR-Scan		
24	TMS = 1, TCK = ↑	Select IR-Scan		
25	TMS = 0, TCK = ↑	Capture-IR		
26	TMS = 0, TCK = ↑	Shift-IR		
Loop4	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1011 (Init)
27	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Init MSB)
28	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
29	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	Loop for 200us
30	TMS = 1, TCK = ↑	Select DR-Scan		
31	TMS = 1, TCK = ↑	Select IR-Scan		
32	TMS = 0, TCK = ↑	Capture-IR		
33	TMS = 0, TCK = ↑	Shift-IR		
Loop5	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 0000 (Disable)
34	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 1 (Disable MSB)
35	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
36	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	
37	TMS = 1, TCK = ↑	Select DR-Scan		
38	TMS = 1, TCK = ↑	Select IR-Scan		
39	TMS = 1, TCK = ↑	Test-Logic/Reset	One negative edge clock	
40	TMS = 1,	Test-Logic/Reset	In user mode	DONE

XPLA3 256 MACROCELL DEVICE PROGRAMMING EXAMPLE

Program

The Program command programs the user's data into the EEPROM array.

Program Command Sequence

Command	Sequence
Program	1. Ensure device is in Test-Logic/Reset state 2. shift in the ENABLE instruction 3. shift in the PROGRAM instruction 4. shift in the address and data for the EEPROM row being programmed. 5. execute the command (Program the data into the selected EEPROM row) 6. Repeat steps 4 and 6 until all EEPROM rows have been programmed 7. shift in the INIT instruction 8. execute the instruction (activate the contents of the EEPROM array) 9. shift in the DISABLE instruction 10. execute the instruction (activate the contents of the EEPROM array)

Program Detailed Algorithm

Step	Transition Conditions	TAP State	CPLD Event Description	Programmer Action
0	TMS = 1	Test-Logic/Reset	BEGIN	BEGIN
Loop0	TMS = 1, TCK = ↑	Test-Logic/Reset	Ensure device in Test-Logic/Reset State	Loop 5 times
1	TMS = 0, TCK = ↑	Run-Test/Idle		
2	TMS = 1, TCK = ↑	Select DR-Scan		
3	TMS = 1, TCK = ↑	Select IR-Scan		
4	TMS = 0, TCK = ↑	Capture-IR		
5	TMS = 0, TCK = ↑	Shift-IR		
Loop1	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1001 (Enable)
6	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Enable MSB)
7	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register; Set Enable Flip-flop	
8	TMS = 1, TCK = ↑	Select DR-Scan		
9	TMS = 1, TCK = ↑	Select IR-Scan		
10	TMS = 0, TCK = ↑	Capture-IR		
11	TMS = 0, TCK = ↑	Shift-IR		
Loop2	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1101 (Program)
12	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Program MSB)
13	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
Loop3	TMS = 1, TCK = ↑	Select DR-Scan		
Loop3	TMS = 0, TCK = ↑	Capture-DR		
Loop3	TMS = 0, TCK = ↑	Shift-DR		
Loop4	TMS = 0, TCK = ↑	Shift-DR	Shift in Data bits 303-0	TDI = data bits (303 - 0)
Loop5	TMS = 0, TCK = ↑	Shift-DR	Shift in Address bit 8-1	TDI = address bits (8-1)
Loop3	TMS = 1, TCK = ↑	Exit1-DR	Shift in Address bit 0	TDI = address bit 0 (LSB)
Loop3	TMS = 0, TCK = ↑	Pause-DR	Wait for voltages to settle	Loop for 20 us
Loop3	TMS = 1, TCK = ↑	Exit2-DR		
Loop3	TMS = 1, TCK = ↑	Update-DR		
Loop3	TMS = 0, TCK = ↑	Run-Test/Idle	Program data in EEPROM	Wait 10 ms.
	Execute loop3 418 times to PROGRAM the entire device.			
14	TMS = 1, TCK = ↑	Select DR-Scan		
15	TMS = 1, TCK = ↑	Select IR-Scan		
16	TMS = 0, TCK = ↑	Capture-IR		
17	TMS = 0, TCK = ↑	Shift-IR		
Loop4	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1011 (Init)
18	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Init MSB)
19	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
20	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	Loop for 200 us
21	TMS = 1, TCK = ↑	Select DR-Scan		
22	TMS = 1, TCK = ↑	Select IR-Scan		
23	TMS = 0, TCK = ↑	Capture-IR		
24	TMS = 0, TCK = ↑	Shift-IR		
Loop5	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 0000 (Disable)
25	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 1 (Disable MSB)
26	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
27	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	

28	TMS = 1, TCK = ↑	Select DR-Scan		
29	TMS = 1, TCK = ↑	Select IR-Scan		
30	TMS = 1, TCK = ↑	Test-Logic/Reset	One negative edge clock	
31	TMS = 1,	Test-Logic/Reset	In user mode	DONE

XPLA3 256 MACROCELL DEVICE PROGRAMMING EXAMPLE

Verify

The Verify command verifies that the user's data in the EEPROM array matches the data contained in the JEDEC file.

Verify Command Sequence

Command	Sequence
Verify	1. ensure device is in Test-Logic/Reset state 2. shift in the ENABLE instruction 3. shift in the VERIFY instruction 4. shift in the address of the EEPROM row being verified. 5. Pause 20us for voltages to settle 6. execute the command (this transfers the row data into the ISP Shift Register) 7. shift out the data from the ISP Shift Register 8. compare the shifted-out data to the expected data 9. Repeat step 4 though 8 until all EEPROM rows have been verified 10. shift in the INIT instruction 11. execute the instruction (activate the contents of the EEPROM array) 12. shift in the DISABLE instruction 13. execute the instruction (activate the contents of the EEPROM array)

Verify Command Detailed Algorithm

Step	Transition Conditions	TAP State	CPLD Event Description	Programmer Action
0	TMS = 1	Test-Logic/Reset	BEGIN	BEGIN
Loop0	TMS = 1, TCK = ↑	Test-Logic/Reset	Ensure device in Test-Logic/Reset	Loop 5 times
1	TMS = 0, TCK = ↑	Run-Test/Idle		
2	TMS = 1, TCK = ↑	Select DR-Scan		
3	TMS = 1, TCK = ↑	Select IR-Scan		
4	TMS = 0, TCK = ↑	Capture-IR		
5	TMS = 0, TCK = ↑	Shift-IR		
Loop1	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1001 (Enable)
6	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Enable MSB)
7	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register ; Set enable flip-flop	
8	TMS = 1, TCK = ↑	Select DR-Scan		
9	TMS = 1, TCK = ↑	Select IR-Scan		
10	TMS = 0, TCK = ↑	Capture-IR		
11	TMS = 0, TCK = ↑	Shift-IR		
Loop2	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bit (0-3)	TDI = 0011 (Verify)
12	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Verify MSB)
13	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
14	TMS = 1, TCK = ↑	Select DR-Scan		
15	TMS = 0, TCK = ↑	Capture-DR		
16	TMS = 0, TCK = ↑	Shift-DR		
Loop4	TMS = 0, TCK = ↑	Shift-DR	Shift first address bits (8-1)	TDI = address (MSB, ...)
17	TMS = 1, TCK = ↑	Exit1-DR	Shift first address bit 0	TDI = address (LSB)

Loop3	TMS = 0, TCK = ↑	Pause-DR	Pause to allow voltages to settle	Loop for 20us
Loop3	TMS = 1, TCK = ↑	Exit2-DR	Load address in Address Latch	
Loop3	TMS = 1, TCK = ↑	Update-DR	Load the Shift Register.	
Loop3	TMS = 0, TCK = ↑	Run-Test/Idle		
Loop3	TMS = 1, TCK = ↑	Select DR-Scan		
Loop3	TMS = 0, TCK = ↑	Capture-DR		
Loop3	TMS = 0, TCK = ↑	Shift-DR	Shift out data bit 303	TD0 = data bit 303; TDI=1
Loop5	TMS = 0, TCK = ↑	Shift-DR	Shift out data bits (302 - 0)	TD0 = data bits (302-0);TDI=1
Loop5	TMS = 0, TCK = ↑	Shift-DR	Shift in next address bits (8-1)	
Loop3	TMS = 1, TCK = ↑	Exit1-DR	Shift in next address bit 0	Compare Data
17	**Execute loop3 until all addresses have been loaded. Load dummy address after last data read.**			
18	TMS = 0, TCK = ↑	Pause-DR		
20	TMS = 1, TCK = ↑	Exit2-DR		
21	TMS = 1, TCK = ↑	Update-DR		
22	TMS = 0, TCK = ↑	Run-Test/Idle		
23	TMS = 1, TCK = ↑	Select DR-Scan		
24	TMS = 1, TCK = ↑	Select IR-Scan		
25	TMS = 0, TCK = ↑	Capture-IR		
26	TMS = 0, TCK = ↑	Shift-IR		
Loop4	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1011 (Init)
27	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Init MSB)
28	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
29	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	Loop for 200 us
30	TMS = 1, TCK = ↑	Select DR-Scan		
31	TMS = 1, TCK = ↑	Select IR-Scan		
32	TMS = 0, TCK = ↑	Capture-IR		
33	TMS = 0, TCK = ↑	Shift-IR		
Loop5	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 0000 (Disable)
34	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 1 (Disable MSB)
35	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
36	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	
37	TMS = 1, TCK = ↑	Select DR-Scan		
38	TMS = 1, TCK = ↑	Select IR-Scan		
39	TMS = 1, TCK = ↑	Test-Logic/Reset	One negative edge clock	
40	TMS = 1,	Test-Logic/Reset	In user mode	DONE

XPLA3 256 MACROCELL DEVICE PROGRAMMING EXAMPLE

Pr_Security

The Pr_Security command programs the security cell of the device.

Pr_Security Command Sequence

Command	Sequence
Pr_Security	1. Ensure device is in the Test-Logic/Reset state 2. shift in the ENABLE instruction 3. shift in the PROGRAM instruction 4. shift in the address of the EEPROM row containing the security bit and shift in the data with the corresponding security bit set to the programming state and the other bits set to the non-programming state. 5. execute the command (Program the data into the selected EEPROM row) 6. shift in the INIT instruction

	7. execute the command (Activate the security bit) 8. shift in the DISABLE instruction 9. execute the instruction

Program Security Bit Detailed Algorithm

Step	Transition Conditions	TAP State	CPLD Event Description	Programmer Action
0	TMS = 1	Test-Logic/Reset	BEGIN	BEGIN
Loop0	TMS = 1, TCK = ↑	Test-Logic/Reset	Ensure device in Test-Logic/Reset State	Loop 5 times
1	TMS = 0, TCK = ↑	Run-Test/Idle		
2	TMS = 1, TCK = ↑	Select DR-Scan		
3	TMS = 1, TCK = ↑	Select IR-Scan		
4	TMS = 0, TCK = ↑	Capture-IR		
5	TMS = 0, TCK = ↑	Shift-IR		
Loop1	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1001 (Enable)
6	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Enable MSB)
7	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register; Set Enable Flip-flop	
8	TMS = 1, TCK = ↑	Select DR-Scan		
9	TMS = 1, TCK = ↑	Select IR-Scan		
10	TMS = 0, TCK = ↑	Capture-IR		
11	TMS = 0, TCK = ↑	Shift-IR		
Loop2	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1101 (Program)
12	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Program MSB)
13	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
14	TMS = 1, TCK = ↑	Select DR-Scan		
15	TMS = 0, TCK = ↑	Capture-DR		
16	TMS = 0, TCK = ↑	Shift-DR		
Loop3	TMS = 0, TCK = ↑	Shift-DR	Shift in Data bits 303-0	TDI = data bits (303 - 0)
Loop4	TMS = 0, TCK = ↑	Shift-DR	Shift in Address bit 8-1	TDI = address bits (8-1)
17	TMS = 1, TCK = ↑	Exit1-DR	Shift in Address bit 0	TDI = address bit 0 (LSB)
18	TMS = 0, TCK = ↑	Pause-DR	Wait for voltages to settle	Loop for 20 us
19	TMS = 1, TCK = ↑	Exit2-DR		
20	TMS = 1, TCK = ↑	Update-DR		
21	TMS = 0, TCK = ↑	Run-Test/Idle	Program data in EEPROM	Wait 10 ms.
22	TMS = 1, TCK = ↑	Select DR-Scan		
23	TMS = 1, TCK = ↑	Select IR-Scan		
24	TMS = 0, TCK = ↑	Capture-IR		
25	TMS = 0, TCK = ↑	Shift-IR		
Loop5	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1011 (Init)
26	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Init MSB)
27	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
28	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	Loop for 200 us
29	TMS = 1, TCK = ↑	Select DR-Scan		
30	TMS = 1, TCK = ↑	Select IR-Scan		
31	TMS = 0, TCK = ↑	Capture-IR		
32	TMS = 0, TCK = ↑	Shift-IR		
Loop6	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 0000 (Disable)
33	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 1 (Disable MSB)

34	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
35	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	
36	TMS = 1, TCK = ↑	Select DR-Scan		
37	TMS = 1, TCK = ↑	Select IR-Scan		
38	TMS = 1, TCK = ↑	Test-Logic/Reset	One negative edge clock	
39	TMS = 1,	Test-Logic/Reset	In user mode	DONE

XPLA3 256 MACROCELL DEVICE PROGRAMMING EXAMPLE

Rd_Security

The Rd_Security command reads the security bit of the device.

Read Security Bit Command Sequence

Command	Sequence
Rd_Security	1. Ensure device in is Test-Logic/Reset state 2. shift in the ENABLE instruction 3. shift in the VERIFY instruction 4. shift in the 304 data bits (all 0's) and illegal EEPROM row address 0xFF. 5. Pause for voltages to settle. 6. execute the command 7. shift out the data from the ISP Shift Register (data = 0 → device secure, data = 1 → device not secure) 8. shift in the INIT instruction 9. execute the command 10. shift in the DISABLE instruction 11. execute the instruction

Read Security Bit Detailed Algorithm

Step	Transition Conditions	TAP State	CPLD Event Description	Programmer Action
0	TMS = 1	Test-Logic/Reset	BEGIN	BEGIN
Loop0	TMS = 1, TCK = ↑	Test-Logic/Reset	Ensure device in Test-Logic/Reset State	Loop 5 times
1	TMS = 0, TCK = ↑	Run-Test/Idle		
2	TMS = 1, TCK = ↑	Select DR-Scan		
3	TMS = 1, TCK = ↑	Select IR-Scan		
4	TMS = 0, TCK = ↑	Capture-IR		
5	TMS = 0, TCK = ↑	Shift-IR		
Loop1	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1001 (Enable)
6	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Enable MSB)
7	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register ; Set Enable Flip-flop	
8	TMS = 1, TCK = ↑	Select DR-Scan		
9	TMS = 1, TCK = ↑	Select IR-Scan		
10	TMS = 0, TCK = ↑	Capture-IR		
11	TMS = 0, TCK = ↑	Shift-IR		
Loop2	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 0011 (Verify LSB)
12	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Verify MSB)
13	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
14	TMS = 1, TCK = ↑	Select DR-Scan		

15	TMS = 0, TCK = ↑	Capture-DR		
16	TMS = 0, TCK = ↑	Shift-DR		
Loop3	TMS = 0, TCK = ↑	Shift-DR	Shift in data bits (303-0)	TDI = All data bits = 0's
Loop4	TMS = 0, TCK = ↑	Shift-DR	Shift in address bits (8-1)	TDI = 11111111 (MSB,...)
17	TMS = 1, TCK = ↑	Exit1-DR	Shift in address bit 0	TDI = 1 (LSB)
18	TMS = 0, TCK = ↑	Pause-DR	Pause for voltages to settle	Loop for 20us
19	TMS = 1, TCK = ↑	Exit2-DR	Load address in Address Latch	
20	TMS = 1, TCK = ↑	Update-DR	Load the Shift Register.	
21	TMS = 0, TCK = ↑	Run-Test/Idle		
22	TMS = 1, TCK = ↑	Select DR-Scan		
23	TMS = 0, TCK = ↑	Capture-DR		
Loop5	TMS = 0, TCK = ↑	Shift-DR	Shift out data bit 303	TDO = 0 → device secure
24	TMS = 1, TCK = ↑	Exit1-DR		
25	TMS = 0, TCK = ↑	Pause-DR		
26	TMS = 1, TCK = ↑	Exit2-DR		
27	TMS = 1, TCK = ↑	Update-DR		
28	TMS = 1, TCK = ↑	Select DR-Scan		
29	TMS = 1, TCK = ↑	Select IR-Scan		
30	TMS = 0, TCK = ↑	Capture-IR		
31	TMS = 0, TCK = ↑	Shift-IR		
Loop6	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1011 (Init)
32	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 0 (Init MSB)
33	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
34	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	Loop for 200 us
35	TMS = 1, TCK = ↑	Select DR-Scan		
36	TMS = 1, TCK = ↑	Select IR-Scan		
37	TMS = 0, TCK = ↑	Capture-IR		
38	TMS = 0, TCK = ↑	Shift-IR		
Loop7	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 0000 (Disable)
39	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 1 (Disable MSB)
40	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
41	TMS = 0, TCK = ↑	Run-Test/Idle	Initialize device with new program data	
42	TMS = 1, TCK = ↑	Select DR-Scan		
43	TMS = 1, TCK = ↑	Select IR-Scan		
44	TMS = 1, TCK = ↑	Test-Logic/Reset	One negative edge clock	
45	TMS = 1,	Test-Logic/Reset	In user mode	DONE

JTAG Instructions

The JTAG commands supported by the ISP PC Parallel Port Programmer are described in the following subsections. These commands are basic JTAG commands which are required by the ISP programmer to control multiple JTAG/ISP devices in a JTAG chain.

Bypass

The Bypass command passes data from TDI to TDO with a one half clock cycle delay. The outputs of the device are not affected by this operation.

BYPASS Command Sequence

Command	Sequence

BYPASS	1. shift in the BYPASS instruction 2. shift in/out data while in the DR-Shift TAP Controller State

BYPASS Detailed Algorithm

Step	Transition Conditions	TAP State	CPLD Event Description	Programmer Action
0	TMS = 1	Test-Logic/Reset	BEGIN	BEGIN
loop0	TMS = 1, TCK = ↑	Test-Logic/Reset	Ensure device in Test-Logic/Reset State	Loop 5 times
1	TMS = 0, TCK = ↑	Run-Test/Idle		
2	TMS = 1, TCK = ↑	Select DR-Scan		
3	TMS = 1, TCK = ↑	Select IR-Scan		
4	TMS = 0, TCK = ↑	Capture-IR		
5	TMS = 0, TCK = ↑	Shift-IR		
loop1	TMS = 0, TCK = ↑	Shift-IR	Shift in instruction bits (0-3)	TDI = 1111 (BYPASS)
6	TMS = 1, TCK = ↑	Exit1-IR	Shift in instruction bit 4	TDI = 1 (BYPASS MSB)
7	TMS = 1, TCK = ↑	Update-IR	Load the Instruction Register	
8	TMS = 1, TCK = ↑	Select DR-Scan		
9	TMS = 0, TCK = ↑	Capture-DR		
10	TMS = 0, TCK = ↑	Shift-DR		
loop2	TMS = 0, TCK = ↑	Shift-DR	Shift in/out Data bits	TDI=TDO (1/2 TCK delay)
11	TMS = 1, TCK = ↑	Exit1-DR	Shift in/out last Data bit	TDI=TDO (1/2 TCK delay)
12	TMS = 1, TCK = ↑	Update-DR		
13	TMS = 1, TCK = ↑	Select DR-Scan		
14	TMS = 1, TCK = ↑	Select IR-Scan		
15	TMS = 1, TCK = ↑	Test-Logic/Reset		
	TMS = 1	Test-Logic/Reset	DONE	DONE

Please note that the device must remain in loop2, "*Shift-DR State*", to remain in the BYPASS mode.

IDcode

The IDcode command reads the IDcode of the device. The outputs of the device are not affected by this operation.

Added to Specification on April 5, 2000.
Information dated 12/23/99

The additional pin for JTAG port enabling (porten) is required to be grounded during the power-up cycle and permits enabling the JTAG port when the port had been reconfigured as I/Os. XPLA 3 devices are shipped with the JTAG port enabled. During device programming, the pins can be left as dedicated JTAG ports or reconfigured as normal I/O pins. If the JTAG ports are reconfigured as I/O pins the change takes effect upon execution of the ISP reinit command and with all subsequent power-up cycles. Note that reconfiguring the JTAG port pins as I/Os, makes these pins non-JTAG ISP functional; however, the porten pin can be used to reclaim these pins for ISP programming by applying Vdd to the

porten pin. This forces the JTAG port pins to be functional for ISP programming; after completing the desired ISP function, bringing porten to grd re-establishes the JTAG port pins as I/O pins if programmed as I/Os. If the JTAG ports are configured during part programming as dedicated JTAG ports (i.e. not normal I/Os) then the porten pin has no effect. External circuitry must be compatible with this dual use of JTAG port pins.