



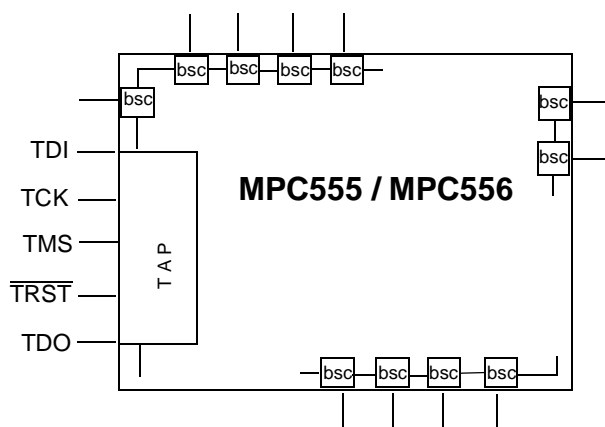
## SECTION 22

### IEEE 1149.1-COMPLIANT INTERFACE (JTAG)

The MPC555 / MPC556 includes dedicated user-accessible test logic that is fully compatible with the *IEEE 1149.1-1990 Standard Test Access Port and Boundary Scan Architecture*. Problems associated with testing high-density circuit boards have led to development of this standard under the sponsorship of the Test Technology Committee of IEEE and the Joint Test Action Group (JTAG). The MPC555 / MPC556 supports circuit-board test strategies based on this standard.

This section is intended to be used with the supporting IEEE 1149.1-1990 standard. The scope of this description includes those items required by the standard to be defined and, in certain cases, provides additional information specific to the implementation. For internal details and applications of the standard, refer to the IEEE 1149.1-1990 document.

An overview of the JTAG pins on the MPC555 / MPC556 is shown in [Figure 22-1](#).

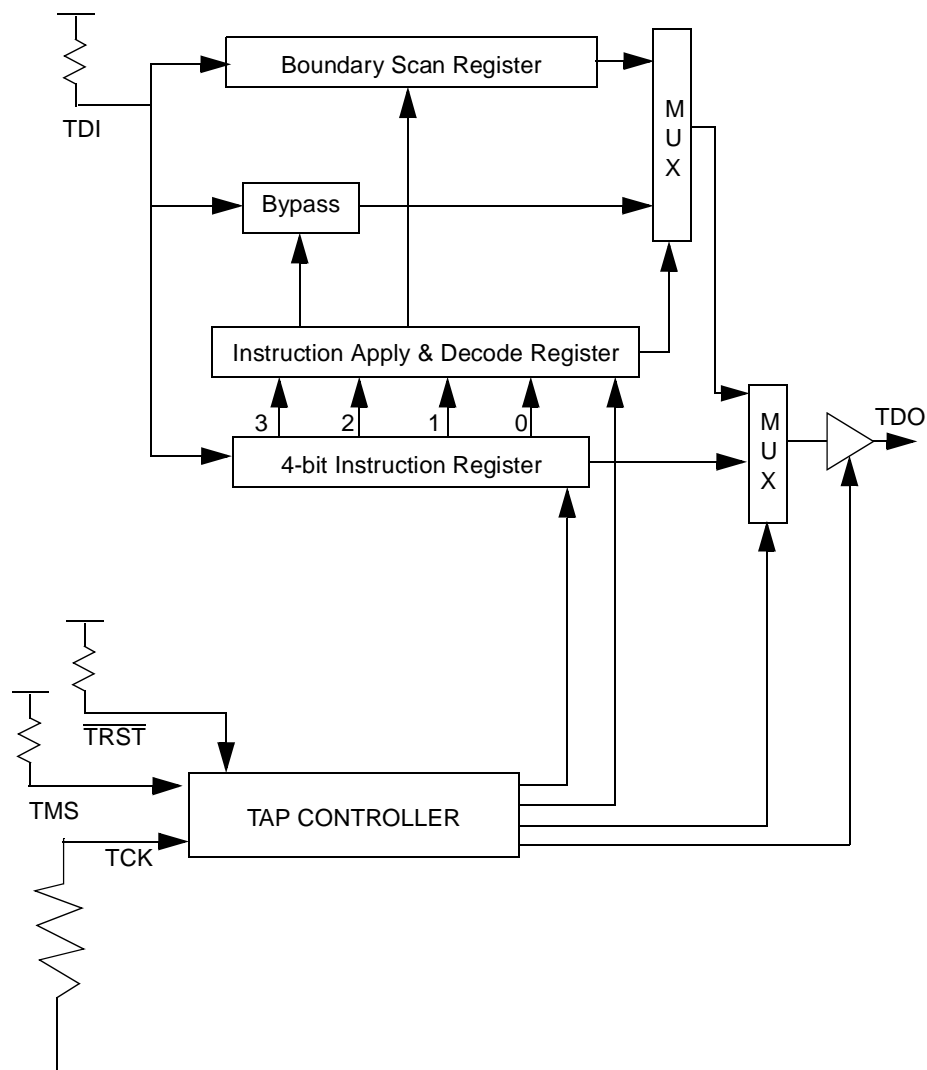


**Figure 22-1 JTAG Pins**

Boundary scan cells (BSC) are placed at the digital boundary of the chip (normally the package pins). The boundary scan cells are chained together to form a boundary scan register (BSR). The data is serially shifted in through the serial port (TDI) and serially shifted out through the output port (TDO).

#### 22.1 JTAG Interface Block Diagram

A block diagram of the MPC555 / MPC556 implementation of the IEEE 1149.1-1990 test logic is shown in [Figure 22-2](#).



**Figure 22-2 Test Logic Block Diagram**

## 22.2 JTAG Signal Descriptions

The MPC555 / MPC556 has five dedicated JTAG pins, which are described in [Table 22-1](#). The TDI and TDO scan ports are used to scan instructions as well as data into the various scan registers for JTAG operations. The scan operation is controlled by the test access port (TAP) controller, which in turn is controlled by the TMS input sequence.

To enable JTAG on reset for board test, bit 11 (DGPC select JTAG pins) and bit 16 (PRPM peripheral mode enable) of the reset configuration word should be held high during the rising edge of reset (see [7.5.2 Hard Reset Configuration Word](#)). These

need to be configurable on the user board to allow JTAG test of a board. To allow normal operation of the board these bits need to be low in the reset configuration word.



**Table 22-1 JTAG Interface Pin Descriptions**

Signal Name	Input/ Output (I/O)	Internal Pull-Up/ Pull-Down Provided	Description
TDI	Input	Pull-up	Test data input pin. Sampled on the rising edge of TCK. Has a pull-up resistor.
TDO	Output	None	Test data output pin. Actively driven during the Shift-IR and Shift-DR controller states. Changes on the falling edge of TCK. Can be placed in high-impedance state.
TMS	Input	Pull-up	Test mode select pin. Sampled on the rising edge of TCK to sequence the test controller's state machine. Has a pull-up resistor.
TCK	Input	Pull-down	Test clock input to synchronize the test logic. Has a pull-down resistor.
$\overline{\text{TRST}}$	Input	Pull-up	TAP controller asynchronous reset. Provides initialization of the TAP controller and other logic as required by the standard. Has a pull-up resistor.

## 22.3 Operating Frequency

The TCK frequency must be between 5 MHz and 10 MHz. This pin is internally driven to a low value when disconnected.

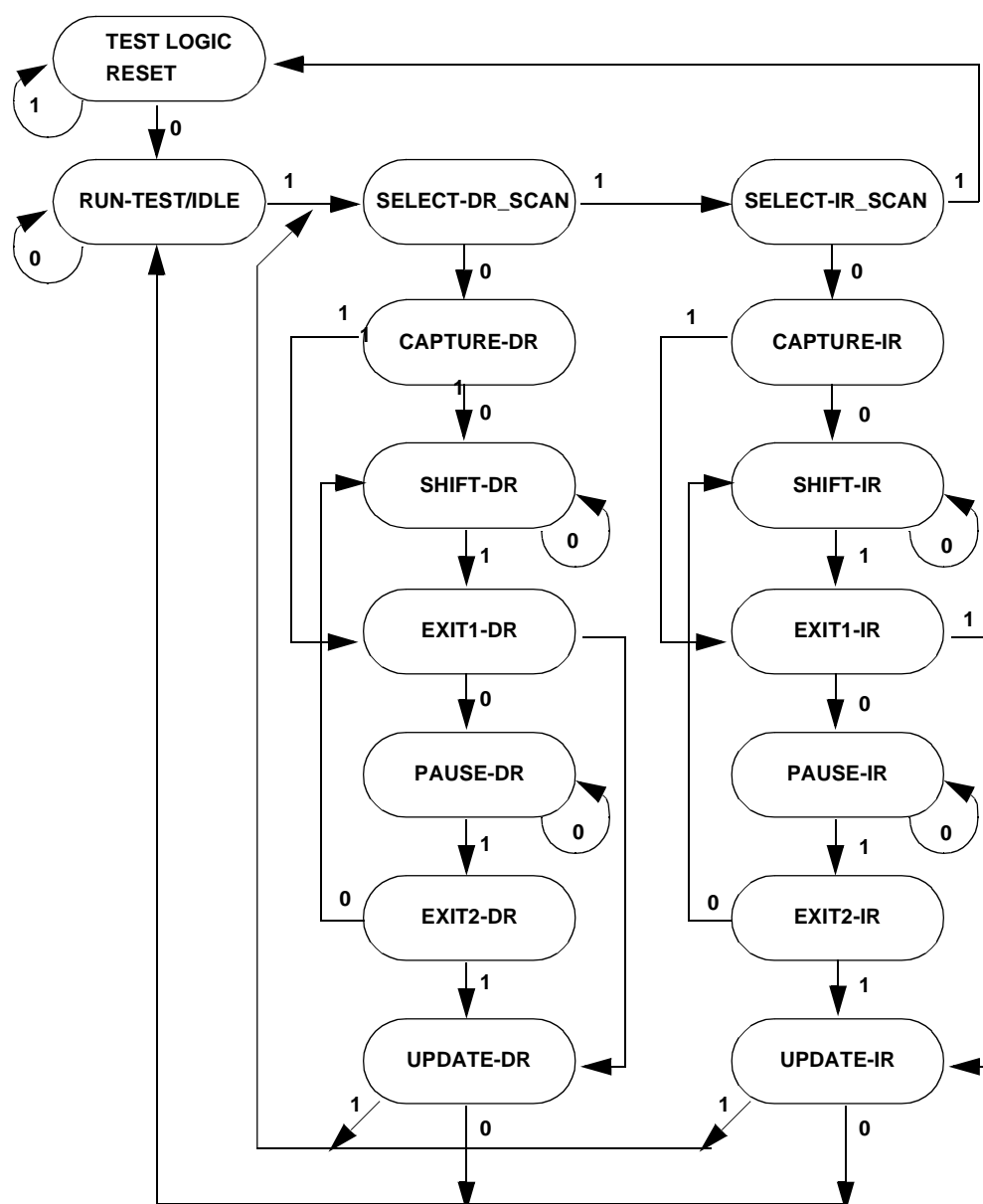
## 22.4 TAP Controller

$\overline{\text{TRST}}$  is used to reset the TAP controller asynchronously. The  $\overline{\text{TRST}}$  pin ensures that the JTAG logic does not interfere with the normal operation of the chip. This pin is optional in the JTAG specification.

The TAP controller changes state either on the rising edge of TCK or when  $\overline{\text{TRST}}$  is asserted.

The TDO signal remains in a high-impedance state except during the Shift-DR or Shift-IR controller states. During these controller states, TDO is updated on the falling edge of TCK.

The TAP controller states are designed to meet the IEEE 1149.1 standard. Refer to [Figure 22-3](#).



**Figure 22-3 TAP Controller State Machine**

## 22.5 Instruction Register

The MPC555 / MPC556 JTAG implementation includes the public instructions (EXTEST, SAMPLE/PRELOAD, and BYPASS), and also supports the CLAMP instruction. One additional public instruction (HI-Z) provides the capability for disabling all device output drivers. The MPC555 / MPC556 includes a 4-bit instruction register without parity consisting of a shift register with four parallel outputs. Data is transferred from the shift register to the parallel outputs during the update-IR controller state. The four bits are used to decode the five unique instructions listed in [Table 22-2](#).



**Table 22-2 Instruction Decoding**

Code				Instruction
B3	B2	B1	B0	
0	0	0	0	EXTEST
0	0	0	1	SAMPLE/PRELOAD
0	X	1	X	BYPASS
0	1	0	0	HI-Z
0	1	0	1	CLAMP and BYPASS

The parallel output of the instruction register is reset to all ones in the test-logic-reset controller state. Note that this preset state is equivalent to the BYPASS instruction.

During the capture-IR controller state, the parallel inputs to the instruction shift register are loaded with the CLAMP command code.

### 22.5.1 EXTEST

The external test (EXTEST) instruction selects the 346-bit boundary scan register. EXTEST also asserts internal reset for the MPC555 / MPC556 system logic to force a predictable beginning internal state while performing external boundary scan operations.

By using the TAP, the register is capable of scanning user-defined values into the output buffers, capturing values presented to input pins and controlling the output drive of three-state output or bi-directional pins. For more details on the function and use of EXTEST, refer to the scan chain document.

### 22.5.2 SAMPLE/PRELOAD

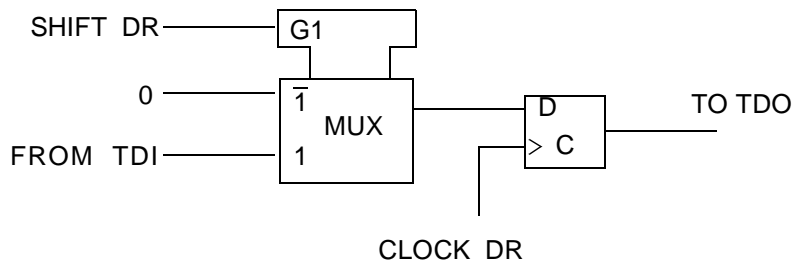
The SAMPLE/PRELOAD instruction initializes the boundary scan register output cells prior to selection of EXTEST. This initialization ensures that known data will appear on the outputs when entering the EXTEST instruction. The SAMPLE/PRELOAD instruction also provides a means to obtain a snapshot of system data and control signals.

#### NOTE

Since there is no internal synchronization between the scan chain clock (TCK) and the system clock (CLKOUT), the user must provide some form of external synchronization to achieve meaningful results.

### 22.5.3 BYPASS

The BYPASS instruction selects the single-bit bypass register as shown in [Figure 22-4](#). This creates a shift register path from TDI to the bypass register and, finally, to TDO, circumventing the 463-bit boundary scan register. This instruction is used to enhance test efficiency when a component other than the MPC555 / MPC556 becomes the device under test.



**Figure 22-4 Bypass Register**

When the bypass register is selected by the current instruction, the shift register stage is set to a logic zero on the rising edge of TCK in the capture-DR controller state. Therefore, the first bit to be shifted out after selecting the bypass register will always be a logic zero.

#### 22.5.4 CLAMP

The CLAMP instruction selects the single-bit bypass register as shown in [Figure 22-4](#), and the state of all signals driven from system output pins is completely defined by the data previously shifted into the boundary scan register (e.g., using the SAMPLE/PRELOAD instruction).

#### 22.5.5 HI-Z

The HI-Z instruction is provided as a manufacturer's optional public instruction to prevent having to backdrive the output pins during circuit-board testing. When HI-Z is invoked, all output drivers, including the two-state drivers, are turned off (i.e., high impedance). The instruction selects the bypass register.

#### 22.6 Restrictions

The MPC555 / MPC556 provides flexible control of external signals using the boundary scan register and EXTEST or CLAMP instructions. As a result, the circuit board test environment must be designed to avoid signal contention which may result in device destruction.

#### 22.7 Low-Power Stop Mode

The MPC555 / MPC556 features a low-power stop mode. The interaction of the scan chain interface with low-power stop mode is as follows:

1. The TAP controller must be in the test-logic-reset state to either enter or remain in the low-power stop mode. Leaving the TAP controller in the test-logic-reset state negates the ability to achieve low-power, but does not otherwise affect device functionality.

2. The TCK input is not blocked in low-power stop mode. To consume minimal power, the TCK input should be externally connected to VDD or ground, although TCK pin is internally connected to ground.
3. The TMS, TDI and  $\overline{\text{TRST}}$  pins include on-chip pullup resistors. In low-power stop mode, these three pins should remain either unconnected or connected to VDD to achieve minimal power consumption.



#### NOTE

For proper reset of the scan chain test logic, the best approach is to assert  $\overline{\text{TRST}}$  at power on reset ( $\overline{\text{PORESET}}$ ).

### 22.8 Non-IEEE 1149.1-1990 Operation

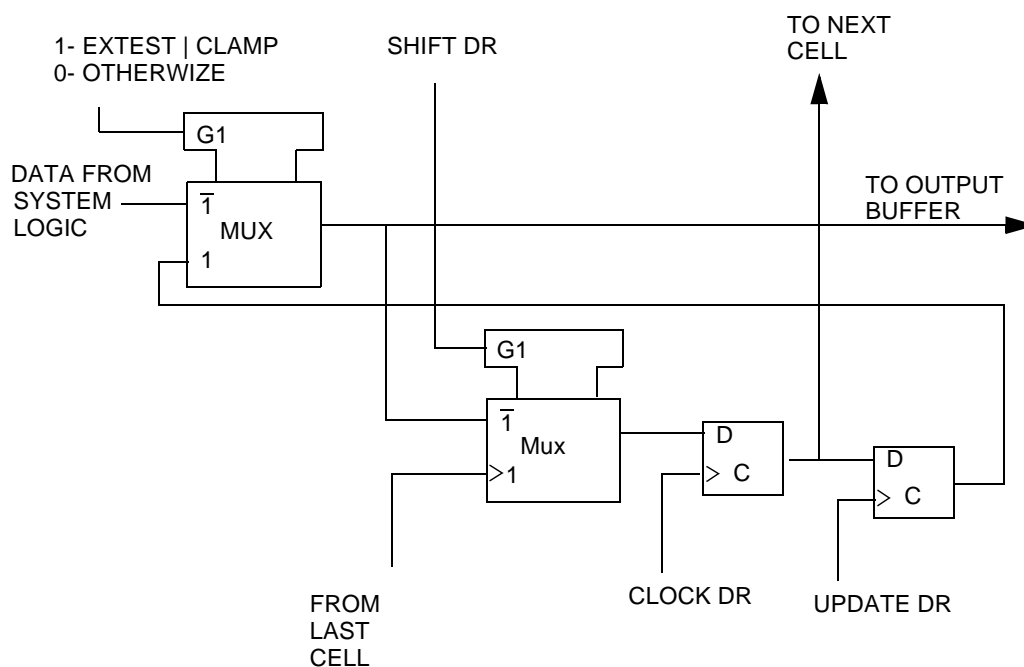
In non-IEEE 1149.1-1990 operation, the IEEE 1149.1-1990 test logic must be kept transparent to the system logic by forcing and holding the TAP controller into the **test-logic-reset** controller state. There are two methods of forcing and holding the controller to this state. The first is to assert the  $\overline{\text{TRST}}$  signal, forcing the TAP into the **test-logic-reset** controller state. The second is to provide at least five TCK pulses with TMS held high.

The best approach is to connect a pull down resistor to  $\overline{\text{TRST}}$ , or to connect it to  $\overline{\text{PORESET}}$  with a resistor. If boundary scan is required, the JTAG controller should drive  $\overline{\text{TRST}}$  to the negated state ("1" value) following  $\overline{\text{PORESET}}$ .

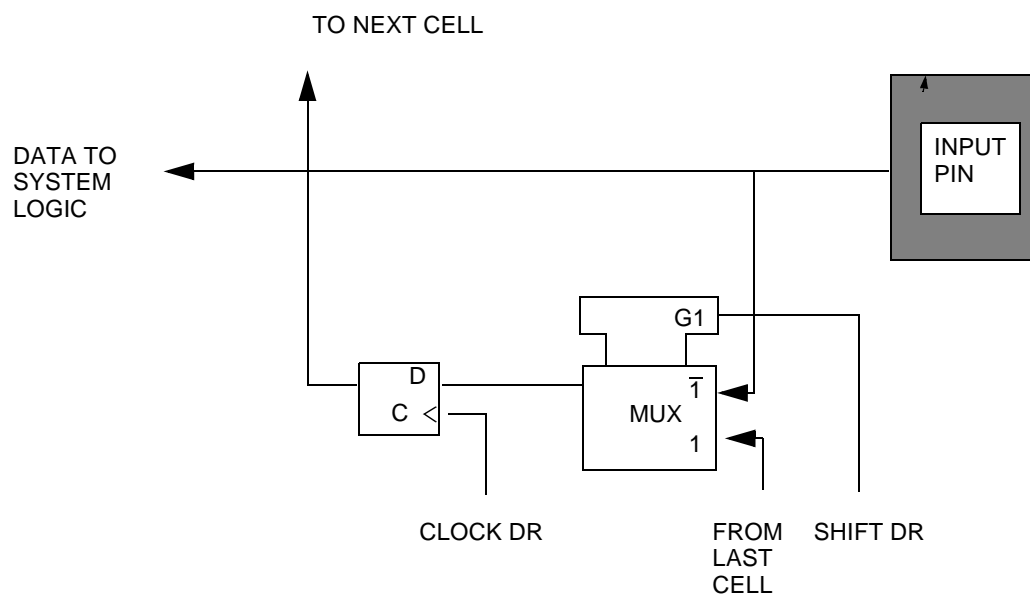
### 22.9 Boundary Scan Register

The MPC555 / MPC556 scan chain implementation has a 346-bit boundary scan register. This register contains bits for all device signal and clock pins and associated control signals. The XTAL, EXTAL and XFC pins are associated with analog signals and are not included in the boundary scan register.

An IEEE-1149.1 compliant boundary scan register has been included on the MPC555 / MPC556. This 346-bit boundary scan register can be connected between TDI and TDO when EXTEST or SAMPLE/PRELOAD instructions are selected. This register is used for capturing signal pin data on the input pins, forcing fixed values on the output signal pins, and selecting the direction and drive characteristics (a logic value or high impedance) of the bidirectional and three-state signal pins. **Figure 22-5** through **Figure 22-8** depict the various cell types.

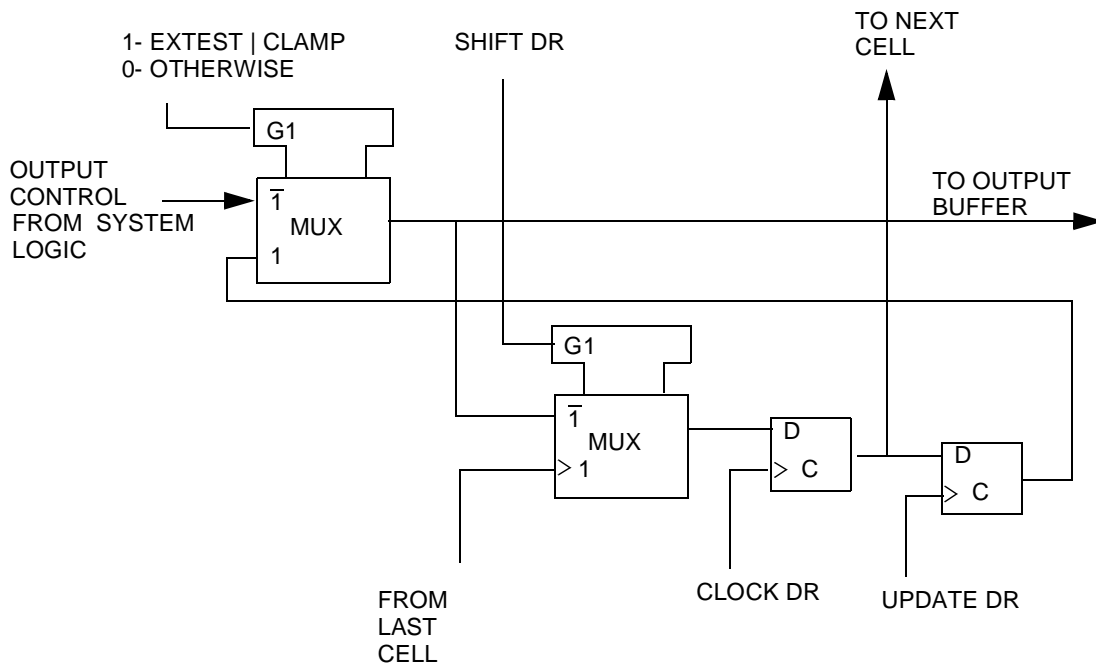


**Figure 22-5 Output Pin Cell (O.pin)**

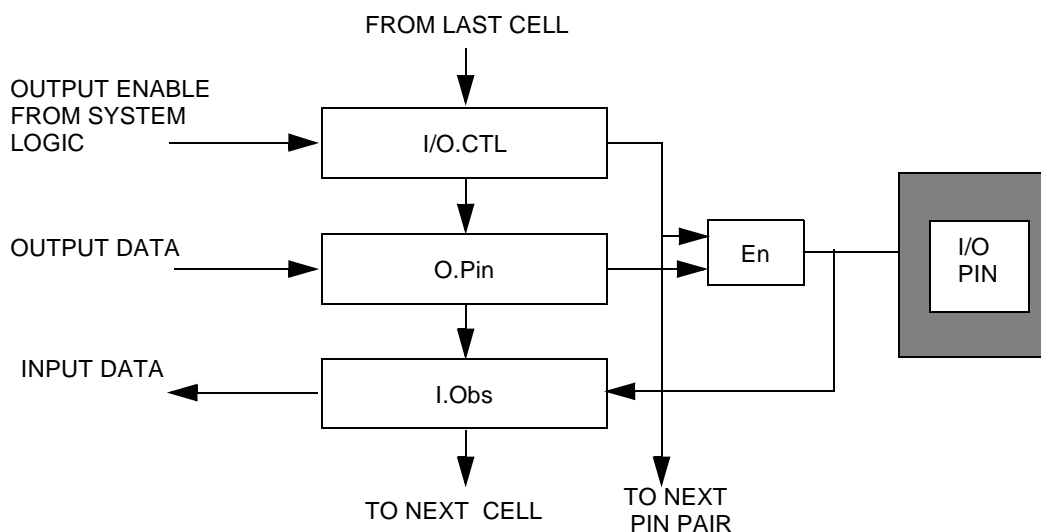


**Figure 22-6 Observe-Only Input Pin Cell (I.Obs)**





**Figure 22-7 Output Control Cell (IO.CTL)**



**Figure 22-8 General Arrangement of Bidirectional Pin Cells**

The key to using the boundary scan register is knowing the boundary scan bit order and the pins that are associated with them. Below in [Table 22-3](#) is the bit order starting

from the TDI input and going to the TDO output. This table uses the “long” pin names of the MPC555 / MPC556. See [Table 2-6](#) for a translation of the long names to the short names found on the pinout information in [Figure 2-2](#).



The first column in the table defines the bit's ordinal position in the boundary scan register. The shift register cell nearest TDI (i.e., first to be shifted in) is defined as bit 1; the last bit to be shifted in is 345.

The second column references one of the three MPC555 / MPC556 cell types depicted in [Figure 22-5](#) through [Figure 22-8](#), which describe the cell structure for each type.

The third column lists the pin name for all pin-related cells or defines the name of bi-directional control register bits. The fourth column lists the pin type. The last column indicates the associated boundary scan register control bit for bi-directional output pins.

Bi-directional pins include two scan cells for data (IO.Cell) as depicted in [Figure 22-8](#). These bits are controlled by the cell shown in [Figure 22-7](#). The value of the control bit controls the output function of the bidirectional pin. One or more bidirectional data cells can be serially connected to a control cell.

**Table 22-3 Boundary Scan Bit Definition**

Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
0	i.obs	b_cnrx0	i	—
1	o.pin	b_cntx0	o	—
2	IO.PIN	b_tpuch0	io	g251.ctl
3	IO.ctl	g251.ctl	—	—
4	IO.PIN	b_tpuch1	io	g252.ctl
5	IO.ctl	g252.ctl	—	—
6	IO.PIN	b_tpuch2	io	g253.ctl
7	IO.ctl	g253.ctl	—	—
8	IO.PIN	b_tpuch3	io	g254.ctl
9	IO.ctl	g254.ctl	—	—
10	IO.PIN	b_tpuch4	io	g255.ctl
11	IO.ctl	g255.ctl	—	—
12	IO.PIN	b_tpuch5	io	g256.ctl
13	IO.ctl	g256.ctl	—	—
14	IO.PIN	b_tpuch6	io	g257.ctl
15	IO.ctl	g257.ctl	—	—
16	IO.PIN	b_tpuch7	io	g258.ctl

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
17	IO.ctl	g258.ctl	—	—
18	IO.PIN	b_tpuch8	io	g259.ctl
19	IO.ctl	g259.ctl	—	—
20	IO.PIN	b_tpuch9	io	g260.ctl
21	IO.ctl	g260.ctl	—	—
22	IO.PIN	b_tpuch10	io	g261.ctl
23	IO.ctl	g261.ctl	—	—
24	IO.PIN	b_tpuch11	io	g262.ctl
25	IO.ctl	g262.ctl	—	—
26	IO.PIN	b_tpuch12	io	g263.ctl
27	IO.ctl	g263.ctl	—	—
28	IO.PIN	b_tpuch13	io	g264.ctl
29	IO.ctl	g264.ctl	—	—
30	IO.PIN	b_tpuch14	io	g265.ctl
31	IO.ctl	g265.ctl	—	—
32	IO.PIN	b_tpuch15	io	g266.ctl
33	IO.ctl	g266.ctl	—	—
34	IO.PIN	b_t2clk	io	g267.ctl
35	IO.ctl	g267.ctl	—	—
36	IO.PIN	a_t2clk	io	g268.ctl
37	IO.ctl	g268.ctl	—	—
38	IO.PIN	a_tpuch0	io	g269.ctl
39	IO.ctl	g269.ctl	—	—
40	IO.PIN	a_tpuch1	io	g302.ctl
41	IO.ctl	g302.ctl	—	—
42	IO.PIN	a_tpuch2	io	g303.ctl
43	IO.ctl	g303.ctl	—	—
44	IO.PIN	a_tpuch3	io	g304.ctl
45	IO.ctl	g304.ctl	—	—
46	IO.PIN	a_tpuch4	io	g305.ctl
47	IO.ctl	g305.ctl	—	—

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
48	IO.PIN	a_tpuch5	io	g306.ctl
49	IO.ctl	g306.ctl	—	—
50	IO.PIN	a_tpuch6	io	g307.ctl
51	IO.ctl	g307.ctl	—	—
52	IO.PIN	a_tpuch7	io	g308.ctl
53	IO.ctl	g308.ctl	—	—
54	IO.PIN	a_tpuch8	io	g309.ctl
55	IO.ctl	g309.ctl	—	—
56	IO.PIN	a_tpuch9	io	g310.ctl
57	IO.ctl	g310.ctl	—	—
58	IO.PIN	a_tpuch10	io	g311.ctl
59	IO.ctl	g311.ctl	—	—
60	IO.PIN	a_tpuch11	io	g312.ctl
61	IO.ctl	g312.ctl	—	—
62	IO.PIN	a_tpuch12	io	g313.ctl
63	IO.ctl	g313.ctl	—	—
64	IO.PIN	a_tpuch13	io	g314.ctl
65	IO.ctl	g314.ctl	—	—
66	IO.PIN	a_tpuch14	io	g315.ctl
67	IO.ctl	g315.ctl	—	—
68	IO.PIN	a_tpuch15	io	g316.ctl
69	IO.ctl	g316.ctl	—	—
70	i.obs	a_an0_anw_pqb0	i	—
71	i.obs	a_an1_anx_pqb1	i	—
72	i.obs	a_an2_any_pqb2	i	—
73	i.obs	a_an3_anz_pqb3	i	—
74	i.obs	a_an48_pqb4	i	—
75	i.obs	a_an49_pqb5	i	—
76	IO.PIN	a_an50_pqb6	io	g333.ctl
77	IO.ctl	g333.ctl	—	—
78	IO.PIN	a_an51_pqb7	io	g334.ctl

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
79	IO.ctl	g334.ctl	—	—
80	IO.PIN	a_an52_ma0_pqa0	io	g335.ctl
81	IO.ctl	g335.ctl	—	—
82	IO.PIN	a_an53_ma1_pqa1	io	g336.ctl
83	IO.ctl	g336.ctl	—	—
84	IO.PIN	a_an54_ma2_pqa2	io	g337.ctl
85	IO.ctl	g337.ctl	—	—
86	IO.PIN	a_an55_pqa3	io	g338.ctl
87	IO.ctl	g338.ctl	—	—
88	IO.PIN	a_an56_pqa4	io	g339.ctl
89	IO.ctl	g339.ctl	—	—
90	IO.PIN	a_an57_pqa5	io	g340.ctl
91	IO.ctl	g340.ctl	—	—
92	IO.PIN	a_an58_pqa6	io	g341.ctl
93	IO.ctl	g341.ctl	—	—
94	IO.PIN	a_an59_pqa7	io	g342.ctl
95	IO.ctl	g342.ctl	—	—
96	i.obs	b_an0_anw_pqb0	i	—
97	i.obs	b_an1_anx_pqb1	i	—
98	i.obs	b_an2_any_pqb2	i	—
99	i.obs	b_an3_anz_pqb3	i	—
100	i.obs	b_an48_pqb4	i	—
101	i.obs	b_an49_pqb5	i	—
102	IO.PIN	b_an50_pqb6	io	g349.ctl
103	IO.ctl	g349.ctl	—	—
104	IO.PIN	b_an51_pqb7	io	g350.ctl
105	IO.ctl	g350.ctl	—	—
106	IO.PIN	b_an52_ma0_pqa0	io	g351.ctl
107	IO.ctl	g351.ctl	—	—
108	IO.PIN	b_an53_ma1_pqa1	io	g352.ctl
109	IO.ctl	g352.ctl	—	—

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
110	IO.PIN	b_an54_ma2_pqa2	io	g353.ctl
111	IO.ctl	g353.ctl	—	—
112	IO.PIN	b_an55_pqa3	io	g354.ctl
113	IO.ctl	g354.ctl	—	—
114	IO.PIN	b_an56_pqa4	io	g355.ctl
115	IO.ctl	g355.ctl	—	—
116	IO.PIN	b_an57_pqa5	io	g356.ctl
117	IO.ctl	g356.ctl	—	—
118	IO.PIN	b_an58_pqa6	io	g357.ctl
119	IO.ctl	g357.ctl	—	—
120	IO.PIN	b_an59_pqa7	io	g358.ctl
121	IO.ctl	g358.ctl	—	—
122	i.obs	etrig2	i	—
123	i.obs	etrig1	i	—
124	IO.PIN	mda11	io	g365.ctl
125	IO.ctl	g365.ctl	—	—
126	IO.PIN	mda12	io	g366.ctl
127	IO.ctl	g366.ctl	—	—
128	IO.PIN	mda13	io	g367.ctl
129	IO.ctl	g367.ctl	—	—
130	IO.PIN	mda14	io	g368.ctl
131	IO.ctl	g368.ctl	—	—
132	IO.PIN	mda15	io	g369.ctl
133	IO.ctl	g369.ctl	—	—
134	IO.PIN	mda27	io	g370.ctl
135	IO.ctl	g370.ctl	—	—
136	IO.PIN	mda28	io	g371.ctl
137	IO.ctl	g371.ctl	—	—
138	IO.PIN	mda29	io	g372.ctl
139	IO.ctl	g372.ctl	—	—
140	IO.PIN	mda30	io	g403.ctl

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
141	IO.ctl	g403.ctl	—	—
142	IO.PIN	mda31	io	g404.ctl
143	IO.ctl	g404.ctl	—	—
144	IO.PIN	mpwm0	io	g405.ctl
145	IO.ctl	g405.ctl	—	—
146	IO.PIN	mpwm1	io	g406.ctl
147	IO.ctl	g406.ctl	—	—
148	IO.PIN	mpwm2	io	g407.ctl
149	IO.ctl	g407.ctl	—	—
150	IO.PIN	mpwm3	io	g408.ctl
151	IO.ctl	g408.ctl	—	—
152	IO.PIN	mpwm16	io	g409.ctl
153	IO.ctl	g409.ctl	—	—
154	IO.PIN	mpwm17	io	g410.ctl
155	IO.ctl	g410.ctl	—	—
156	IO.PIN	mpwm18	io	g411.ctl
157	IO.ctl	g411.ctl	—	—
158	IO.PIN	mpwm19	io	g412.ctl
159	IO.ctl	g412.ctl	—	—
160	IO.PIN	mpio32b5	io	g413.ctl
161	IO.ctl	g413.ctl	—	—
162	IO.PIN	mpio32b6	io	g414.ctl
163	IO.ctl	g414.ctl	—	—
164	IO.PIN	mpio32b7	io	g415.ctl
165	IO.ctl	g415.ctl	—	—
166	IO.PIN	mpio32b8	io	g416.ctl
167	IO.ctl	g416.ctl	—	—
168	IO.PIN	mpio32b9	io	g417.ctl
169	IO.ctl	g417.ctl	—	—
170	IO.PIN	mpio32b10	io	g418.ctl
171	IO.ctl	g418.ctl	—	—

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
172	IO.PIN	mpio32b11	io	g419.ctl
173	IO.ctl	g419.ctl	—	—
174	IO.PIN	mpio32b12	io	g420.ctl
175	IO.ctl	g420.ctl	—	—
176	IO.PIN	mpio32b13	io	g421.ctl
177	IO.ctl	g421.ctl	—	—
178	IO.PIN	mpio32b14	io	g422.ctl
179	IO.ctl	g422.ctl	—	—
180	IO.PIN	mpio32b15	io	g423.ctl
181	IO.ctl	g423.ctl	—	—
182	IO.PIN	vf0_mpio32b0	io	g424.ctl
183	IO.ctl	g424.ctl	—	—
184	IO.PIN	vf1_mpio32b1	io	g425.ctl
185	IO.ctl	g425.ctl	—	—
186	IO.PIN	vf2_mpio32b2	io	g426.ctl
187	IO.ctl	g426.ctl	—	—
188	IO.PIN	vfls0_mpio32b3	io	g427.ctl
189	IO.ctl	g427.ctl	—	—
190	IO.PIN	vfls1_mpio32b4	io	g428.ctl
191	IO.ctl	g428.ctl	—	—
192	o.pin	a_cntx0	o	—
193	i.obs	a_cnr0	i	—
194	IO.PIN	pcs0_ss_b_qgpio0	io	g435.ctl
195	IO.ctl	g435.ctl	—	—
196	IO.PIN	pcs1_qgpio1	io	g436.ctl
197	IO.ctl	g436.ctl	—	—
198	IO.PIN	pcs2_qgpio2	io	g437.ctl
199	IO.ctl	g437.ctl	—	—
200	IO.PIN	pcs3_qgpio3	io	g438.ctl
201	IO.ctl	g438.ctl	—	—
202	IO.PIN	miso_qgpio4	io	g439.ctl



**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
203	IO.ctl	g439.ctl	—	—
204	IO.PIN	mosi_qgpio5	io	g440.ctl
205	IO.ctl	g440.ctl	—	—
206	IO.PIN	sck_qgpio6	io	g441.ctl
207	IO.ctl	g441.ctl	—	—
208	i.obs	eck	i	—
209	o.pin	txd1_qgpo1	o	—
210	o.pin	txd2_qgpo2	o	—
211	i.obs	rxid1_qgpi1	i	—
212	i.obs	rxid2_qgpi2	i	—
213	i.obs	epee	i	—
214	o.pin	engclk_buclk	o	—
215	i.obs	extclk	i	—
216	o.pin	clkout	o	—
217	i.obs	poreset_b	i	—
218	i.obs	sreset_b	io	—
219	o.pin	sreset_b	io	g465.ctl
220	IO.ctl	g465.ctl	—	—
221	i.obs	hreset_b	io	—
222	o.pin	hreset_b	io	g466.ctl
223	IO.ctl	g466.ctl	—	—
224	IO.PIN	rstconf_b_texp	io	g467.ctl
225	IO.ctl	g467.ctl	—	—
226	i.obs	irq7_b_modck3	i	—
227	i.obs	irq6_b_modck2	i	—
228	IO.PIN	irq5_b_sgpioc5_modck1	io	g503.ctl
229	IO.ctl	g503.ctl	—	—
230	IO.PIN	data_sgpiod[16]	io	g112.ctl
231	IO.ctl	g112.ctl	—	—
232	IO.PIN	data_sgpiod[17]	io	g112.ctl
233	IO.PIN	data_sgpiod[18]	io	g112.ctl

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
234	IO.PIN	data_sgpiod[14]	io	g111.ctl
235	IO.PIN	data_sgpiod[15]	io	g111.ctl
236	IO.PIN	data_sgpiod[19]	io	g112.ctl
237	IO.PIN	data_sgpiod[20]	io	g112.ctl
238	IO.PIN	data_sgpiod[12]	io	g111.ctl
239	IO.PIN	data_sgpiod[13]	io	g111.ctl
240	IO.PIN	data_sgpiod[21]	io	g112.ctl
241	IO.PIN	data_sgpiod[10]	io	g111.ctl
242	IO.PIN	data_sgpiod[11]	io	g111.ctl
243	IO.PIN	data_sgpiod[22]	io	g112.ctl
244	IO.PIN	data_sgpiod[23]	io	g112.ctl
245	IO.PIN	data_sgpiod[8]	io	g111.ctl
246	IO.ctl	g111.ctl	—	—
247	IO.PIN	data_sgpiod[9]	io	g111.ctl
248	IO.PIN	data_sgpiod[24]	io	g524.ctl
249	IO.ctl	g524.ctl	—	—
250	IO.PIN	data_sgpiod[25]	io	g525.ctl
251	IO.ctl	g525.ctl	—	—
252	IO.PIN	data_sgpiod[6]	io	g110.ctl
253	IO.PIN	data_sgpiod[7]	io	g110.ctl
254	IO.PIN	data_sgpiod[26]	io	g528.ctl
255	IO.ctl	g528.ctl	—	—
256	IO.PIN	data_sgpiod[27]	io	g529.ctl
257	IO.ctl	g529.ctl	—	—
258	IO.PIN	data_sgpiod[4]	io	g110.ctl
259	IO.PIN	data_sgpiod[5]	io	g110.ctl
260	IO.PIN	data_sgpiod[28]	io	g534.ctl
261	IO.ctl	g534.ctl	—	—
262	IO.PIN	data_sgpiod[29]	io	g535.ctl
263	IO.ctl	g535.ctl	—	—
264	IO.PIN	data_sgpiod[2]	io	g110.ctl

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
265	IO.PIN	data_sgpiod[3]	io	g110.ctl
266	IO.PIN	data_sgpiod[30]	io	g540.ctl
267	IO.ctl	g540.ctl	—	—
268	IO.PIN	data_sgpiod[0]	io	g110.ctl
269	IO.ctl	g110.ctl	—	—
270	IO.PIN	data_sgpiod[1]	io	g110.ctl
271	IO.PIN	data_sgpiod[31]	io	g545.ctl
272	IO.ctl	g545.ctl	—	—
273	IO.PIN	addr_sgpioa[29]	io	g102.ctl
274	IO.PIN	addr_sgpioa[25]	io	g102.ctl
275	IO.PIN	addr_sgpioa[26]	io	g102.ctl
276	IO.PIN	addr_sgpioa[27]	io	g102.ctl
277	IO.PIN	addr_sgpioa[28]	io	g102.ctl
278	IO.PIN	addr_sgpioa[24]	io	g102.ctl
279	IO.ctl	g102.ctl	—	—
280	IO.PIN	addr_sgpioa[23]	io	g101.ctl
281	IO.PIN	addr_sgpioa[22]	io	g101.ctl
282	IO.PIN	addr_sgpioa[30]	io	g102.ctl
283	IO.PIN	addr_sgpioa[21]	io	g101.ctl
284	IO.PIN	addr_sgpioa[20]	io	g101.ctl
285	IO.PIN	addr_sgpioa[8]	io	g100.ctl
286	IO.ctl	g100.ctl	—	—
287	IO.PIN	addr_sgpioa[31]	io	g102.ctl
288	IO.PIN	addr_sgpioa[19]	io	g101.ctl
289	IO.PIN	addr_sgpioa[18]	io	g101.ctl
290	IO.PIN	addr_sgpioa[9]	io	g100.ctl
291	IO.PIN	addr_sgpioa[17]	io	g101.ctl
292	IO.PIN	addr_sgpioa[16]	io	g101.ctl
293	IO.ctl	g101.ctl	—	—
294	IO.PIN	addr_sgpioa[10]	io	g100.ctl
295	IO.PIN	addr_sgpioa[15]	io	g100.ctl

**Table 22-3 Boundary Scan Bit Definition (Continued)**



Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
296	IO.PIN	addr_sgpioa[14]	io	g100.ctl
297	IO.PIN	addr_sgpioa[13]	io	g100.ctl
298	IO.PIN	addr_sgpioa[11]	io	g100.ctl
299	IO.PIN	addr_sgpioa[12]	io	g100.ctl
300	IO.PIN	bi_b_sts_b	io	g205.ctl
301	IO.ctl	g205.ctl	—	—
302	IO.PIN	burst_b	io	g131.ctl
303	IO.PIN	bdip_b	io	g207.ctl
304	IO.ctl	g207.ctl	—	—
305	IO.PIN	ta_b	io	g208.ctl
306	IO.ctl	g208.ctl	—	—
307	IO.PIN	ts_b	io	g131.ctl
308	IO.PIN	tsiz1	io	g130.ctl
309	IO.PIN	tsiz0	io	g130.ctl
310	IO.ctl	g130.ctl	—	—
311	IO.PIN	tea_b	io	g214.ctl
312	IO.ctl	g214.ctl	—	—
313	o.pin	oe_b	o	—
314	IO.PIN	rd_wr_b	io	g131.ctl
315	IO.ctl	g131.ctl	—	—
316	o.pin	cs3_b	o	—
317	o.pin	cs2_b	o	—
318	o.pin	cs1_b	o	—
319	o.pin	cs0_b	o	—
320	o.pin	we_b_at[3]	o	—
321	o.pin	we_b_at[2]	o	—
322	o.pin	we_b_at[1]	o	—
323	o.pin	we_b_at[0]	o	—
324	IO.PIN	br_b_vf1_iwp2	io	g227.ctl
325	IO.ctl	g227.ctl	—	—
326	IO.PIN	bg_b_vf0_lwp1	io	g228.ctl

**Table 22-3 Boundary Scan Bit Definition (Continued)**

Bit	Cell Type	Pin/Cell Name	Pin Type	Output CTL Cell
327	IO.ctl	g228.ctl	—	—
328	IO.PIN	bb_b_vf2_iwp3	io	g229.ctl
329	IO.ctl	g229.ctl	—	—
330	IO.PIN	sgpioc7_irqout_b_lwp0	io	g230.ctl
331	IO.ctl	g230.ctl	—	—
332	IO.PIN	irq1_b_rsv_b_sgpioc1	io	g231.ctl
333	IO.ctl	g231.ctl	—	—
334	IO.PIN	irq0_b_sgpioc0	io	g232.ctl
335	IO.ctl	g232.ctl	—	—
336	IO.PIN	irq2_b_cr_b_sgpioc2	io	g233.ctl
337	IO.ctl	g233.ctl	—	—
338	IO.PIN	irq4_b_at2_sgpioc4	io	g234.ctl
339	IO.ctl	g234.ctl	—	—
340	IO.PIN	irq3_b_kr_b_retry_b_sgpio c3	io	g237.ctl
341	IO.ctl	g237.ctl	—	—
342	o.pin	iwp0_vfls0	o	—
343	o.pin	iwp1_vfls1	o	—
344	IO.PIN	sgpioc6_frz_ptr_b	io	g240.ctl
345	IO.ctl	g240.ctl	—	—

