Section and Subsection decoding for ZIP Electronics board

24 March 1999
Wayne Johnson

section DAC/csr name	subsection DAC/csr ID	NAME	DAC/csr address	section decoding
Module Info			xx00	zip logic
section 0	RESET		xx0F	zip logic
Sense Bias DAC	DAC 0	Sensor Bias A	xx1A	zip logic
section 1	DAC 1	Sensor Bias B	xx1B	zip logic
	DAC 2	Sensor Bias C	xx1C	zip logic
	DAC 3	Sensor Bias D	xx1D	zip logic
Squid Bias DAC	DAC 0	Squid Bias A	xx2A	zip logic
section 2	DAC 1	Squid Bias B	xx2B	zip logic
	DAC 2	Squid Bias C	xx2C	zip logic
	DAC 3	Squid Bias D	xx2D	zip logic
Squid Lock DAC	DAC 0	Squid Lock A	xx3A	zip logic
section 3	DAC 1	Squid Lock B	xx3B	zip logic
	DAC 2	Squid Lock C	xx3C	zip logic
	DAC 3	Squid Lock D	xx3D	zip logic
Squid Gain DAC	DAC 0	Squid Gain A	xx4A	zip logic
section 4	DAC 1	Squid Gain B	xx4B	zip logic
	DAC 2	Squid Gain C	xx4C	zip logic
	DAC 3	Squid Gain D	xx4D	zip logic
Squid Driver DAC	DAC 0	Squid Driver A	xx5A	zip logic
section 5	DAC 1	Squid Driver B	xx5B	zip logic
	DAC 2	Squid Driver C	xx5C	zip logic
	DAC 3	Squid Driver D	xx5D	zip logic
Q Driver DAC	DAC 0	Q O Driver	xx60	zip logic
section 6	DAC 1	Q I Driver	xx61	zip logic
	DAC 2	Test Voltage	xx62	zip logic
	DAC 3			zip logic
Q Bias DAC	DAC 0	Q O Bias	xx70	zip logic
section 7	DAC 1	Q I Bias	xx71	zip logic
	DAC 2	LED Bias	xx72	zip logic
	DAC 3	Zap Voltage	xx73	zip logic
unused section 8 to D				
CSR	CSR0	Sense Bias CSR	xxE0	zip logic
section E	CSR1	Squid CSR	xxE1	zip logic
	CSR2	Squid Driver CSR	xxE2	zip logic
	CSR3	LED CSR	xxE3	zip logic
	CSR4	Q CSR	xxE4	zip logic
	CSR5	CSR5	xxE5	zip logic
Test section F		Test Pulse	xxFF	zip logic

The Module is decoded from bits 8 to11 of the address bus	These sections are decoded from bits 4 to 7 of the address bus	These subsections are decoded from bits 0 to 3 of the address bus

Outputs are from logic #1: zip_logic3.vhd

data buss bit and matching csr bit name

Data Bit	Sense Bias CSR	Squid CSR	Squid Driver CSR	LED CSR	Q CSR	CSR5	Module ID	Data Bit
BD0	EXT_ENA_A	FB_C/O_SQ_A	**CHA_GAIN0**	WIDTH_0	***Q_O_GAIN0***	TRIG_ENA	Serial Number bit 0	BD0
BD1	EXT_ENA_B	FB_C/O_SQ_B	**CHA_GAIN1**	WIDTH_1	***Q_O_GAIN1***		Serial Number bit 1	BD1
BD2	EXT_ENA_C	FB_C/O_SQ_C	**CHA_GAIN2**	WIDTH_2	***Q_O_GAIN2***		Serial Number bit 2	BD2
BD3	EXT_ENA_D	FB_C/O_SQ_D	**CHA_POLARITY**	WIDTH_3	***Q_O_POLARITY***		Serial Number bit 3	BD3
BD4	***HEAT_PULSE_A***	***SQ_POL_A_-***	**CHB_GAIN0**	WIDTH_4	***Q_I_GAIN0***		Serial Number bit 4	BD4
BD5	***HEAT_PULSE_B***	***SQ_POL_B_-***	**CHB_GAIN1**	WIDTH_5	***Q_I_GAIN1***		Serial Number bit 5	BD5
BD6	***HEAT_PULSE_C***	***SQ_POL_C_-***	**CHB_GAIN2**	WIDTH_6	***Q_I_GAIN2***		Serial Number bit 6	BD6
BD7	***HEAT_PULSE_D***	***SQ_POL_D_-***	**CHB_POLARITY**	WIDTH_7	***Q_I_POLARITY***		Serial Number bit 7	BD7
BD8		TEST_ENA_A	**CHC_GAIN0**	RATE_0	Q_O_ENA		Board Version bit 0	BD8
BD9		TEST_ENA_B	**CHC_GAIN1**	RATE_1	Q_I_ENA		Board Version bit 1	BD9
BD10		TEST_ENA_C	**CHC_GAIN2**	RATE_2			Board Version bit 2	BD10
BD11		TEST_ENA_D	**CHC_POLARITY**	RATE_3			Board Version bit 3	BD11
BD12	***ZAP_ENA_A***	PRE_INT_MON_A	**CHD_GAIN0**	RATE_4	LED_1_ENA		Board Type bit 0	BD12
BD13	***ZAP_ENA_B***	PRE_INT_MON_B	**CHD_GAIN1**	RATE_5	LED_2_ENA		Board Type bit 1	BD13
BD14	***ZAP_ENA_C***	PRE_INT_MON_C	**CHD_GAIN2**	RATE_6	LED_CTRL_0		Board Type bit 2	BD14
BD15	***ZAP_ENA_D***	PRE_INT_MON_D	**CHD_POLARITY**	RATE_7	LED_CTRL_1		Board Type bit 3	BD15

Register outputs that are PLAIN text are outputs of the latching register: logic #2: csr_logic3.vhd

Register outputs that are BOLD text are outputs that are pulses from logic #3: ziprelay2.vhd

Register outputs that are bold and italized text are outputs that are pulses from logic #4: chargeamps.vhd

Module Info ID bits are read only. These are shorts to +VCC or GND to set the condition

Data bits to the DACs set the output voltage level

DAC output voltage calculation:
Vout=Vrefl +((Vrefh-Vrefl) * N)/4096
N= is digital code in decimal

POLARITY	2	1	0	GAIN
0	0	0	0	1
0	0	0	1	1.43
0	0	1	0	2
0	0	1	1	5
0	1	0	0	10
0	1	0	1	14.3
0	1	1	0	20
0	1	1	1	50
1	0	0	0	-1
1	0	0	1	-1.43
1	0	1	0	-2
1	0	1	1	-5
1	1	0	0	-10
1	1	0	1	-14.3
1	1	1	0	-20
1	1	1	1	-50

NOTE:
ALL SIGNAL NAMES ARE HIGH TRUE

LED_CTRL_1	LED_CTRL_0	
0	0	OFF
0	1	SINGLE
1	0	MULTI
1	1	ON

CHIP AND DECODED OUTPUTS TO ZIP BOARD CIRCUITS

These are the names of the chips, the project name that the code is found in and the outputs by name that they decode for the ZIP board

project name:chargeamps	project name: zipchips	project name:ziprelay2	project name: addrlogic
CHARGE AMP	CSR_LOGIC3	ZIPRELAY2	ZIP_LOGIC3
qo_gain_1	qo_ena	chA_gain_1	moduleid
qo_gain_10	qi_ena	chA_gain_10	sense_bias_dac
qo_gain_1_143	led_1_ena	chA_gain_1_143	squid_bias_dac
qo_gain_2_5	led_2_ena	chA_gain_2_5	squid_lock_dac

Section and Subsection decoding for ZIP Electronics board

24 March 1999
Wayne Johnson

qo_gain_1_2
qo_gain_143_5
qo_pol_pos
qo_pol_neg
qi_gain_1
qi_gain_10
qi_gain_1_143
qi_gain_2_5
qi_gain_1_2
qi_gain_143_5
qi_pol_pos
qi_pol_neg
zap_ena_A
zap_ena_B
zap_ena_C
zap_ena_D
heat_pulse_A
heat_pulse_B
heat_pulse_C
heat_pulse_D
sq_pol_A_pos
sq_pol_A_neg
sq_pol_B_pos
sq_pol_B_neg
sq_pol_C_pos
sq_pol_C_neg
sq_pol_D_pos
sq_pol_D_neg
clockena

ext_ena_A
ext_ena_B
ext_ena_C
ext_ena_D
fb_c/o_sq_A
fb_c/o_sq_B
fb_c/o_sq_C
fb_c/o_sq_D
test_ena_A
test_ena_B
test_ena_C
test_ena_D
pre_int_mon_A
pre_int_mon_B
pre_int_mon_C
pre_int_mon_D
trig_ena
led_pulse_out
clockena

READ ONLY INFO
module configuration

chA_gain_1_2
chA_gain_143_5
chA_pol_pos
chA_pol_neg
chB_gain_1
chB_gain_10
chB_gain_1_143
chB_gain_2_5
chB_gain_1_2
chB_gain_143_5
chB_pol_pos
chB_pol_neg
chC_gain_1
chC_gain_10
chC_gain_1_143
chC_gain_2_5
chC_gain_1_2
chC_gain_143_5
chC_pol_pos
chC_pol_neg
chD_gain_1
chD_gain_10
chD_gain_1_143
chD_gain_2_5
chD_gain_1_2
chD_gain_143_5
chD_pol_pos
chD_pol_neg
moduleaddr
clockena

squid_gain_dac
squid_driver_dac
q_driver_dac
q_bias_dac
sense_bias_csr
squid_csr
squid_driver_csr
led_csr
q_csr
csr5
rw
rst
testpulse
buffer_read
buffer_write
moduleselect

sense_bias_csr_ena
squid_csr_ena
squid_driver_csr_ena
led_csr_ena
q_csr_ena
csr5_ena

NOTE:
These _ena outputs are for test purposes and have no functional connection to the board

CONTROL INPUTS FROM ZIP_LOGIC

sense_bias_csr
squid_csr
q_csr
testpulse

sense_bias_csr
squid_csr
q_csr
csr5
led_csr
moduleid
testpulse

squid_driver_csr
moduleselect

CONTROL INPUTS FROM CSR_LOGIC3

trig_ena

CONTROL INPUTS FROM OTHER SOURCES

read
write
poweron
trigplus_minus

read
write
rst
poweron
trigplus_minus

read
write
rst
poweron

read
write
slot(4 downto 0)
addr(15 downto 0)