PRODUCT DESCRIPTION

Functional Description

The BtV system configuration block diagram is shown in Figure 1 and the BtV2210 block diagram is shown in Figure 2.

Figure 1. BtV Configuration

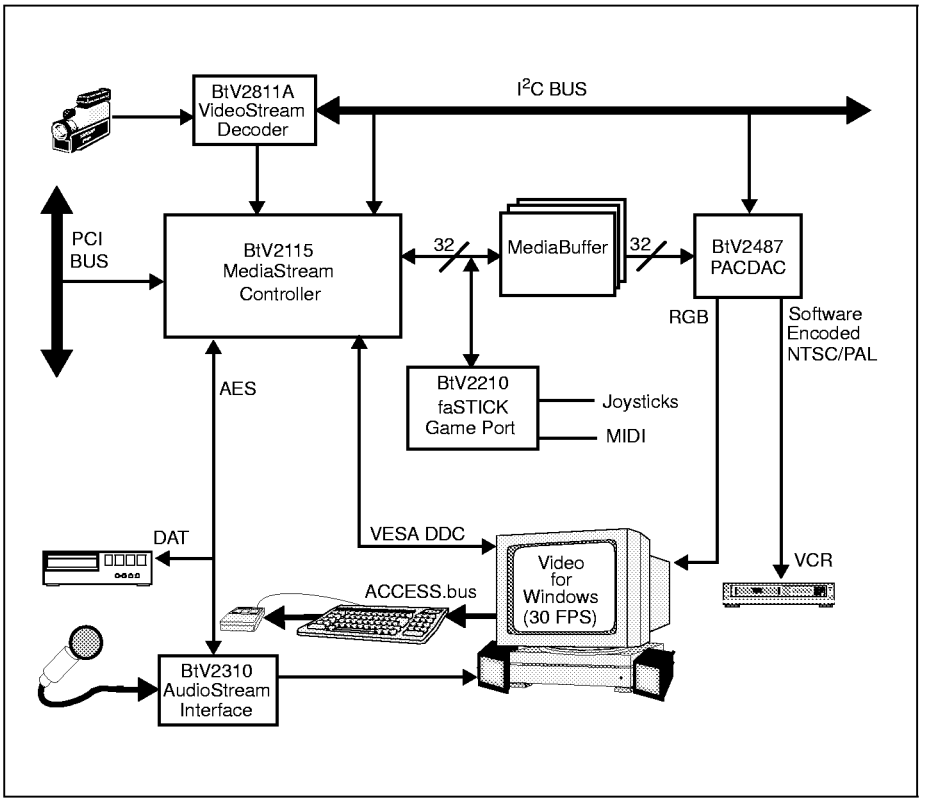
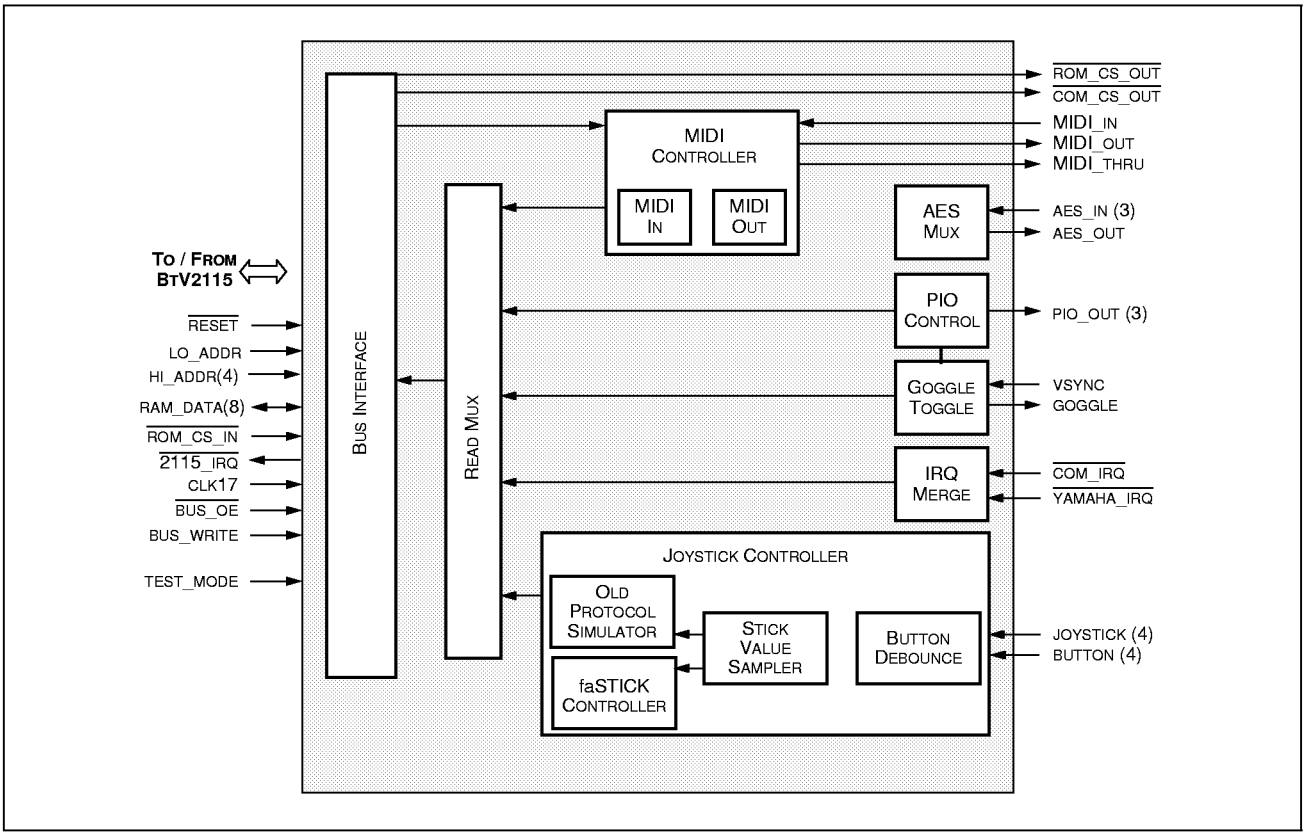


Figure 2. BtV2210 Block Diagram



BtV2210 is comprised of the following three main components.

- Joystick controller
- MIDI controller
- Merge Logic
  - -- PIO pins
  - -- IRQ glue
  - -- ROM chip select
  - -- COM chip select
  - -- Goggle toggle
  - -- AES Mux

Each of these components is discussed in the following sections.

Joystick Controller

The joystick controller connects to 4 joystick inputs (JOYSTICK[3:0]) and 4 button inputs (BUTTON[3:0]). The joystick setting is calculated based on the time it takes to charge the external joystick capacitor, which is charged via the joystick resistance value. The joystick controller then uses this value (maximum 16-bit) to emulate a conventional joystick port or to provide the full set of values very quickly via the BtV-faSTICK protocol. The button values are provided directly in the compatible protocol mode or debounced when given via the BtV-faSTICK protocol. Refer to "Joystick Data Register" on page 8 for protocol information.

The joystick setting is continuously calculated and the resulting values are stored internally. When in BtV-faSTICK mode, these values are returned to the CPU as digital values. In compatible mode, the values are translated into pulse widths in order to be compatible with existing software.

MIDI Controller

The MIDI controller provides a complete MPU-401 compatible MIDI port for UART mode. The MIDI input controller captures MIDI data on the MIDI_IN pin and decodes the serial data stream to present the 8-bit values to the CPU. The MIDI output port takes an 8-bit value from the CPU and generates the MIDI serial data stream on the MIDI-OUT pin. The MIDI_THRU pin is simply of copy of MIDI_IN after synchronization and glitch removal.

Merge Logic

The merge logic is simply the glue that ties the game port interface to the MediaStream Controller. Each component is explained in the following paragraphs.

PIO Pins

The PIO pins enable programming control for external devices. Pins PIO_OUT[2:0] provide the external control. GOGGLE provides a fourth programmable pin if GOGGLE[3] = 0 (goggle disabled). The PIO pins default to logic high at reset, allowing them to be used for chip select to external devices. Since goggle is disabled at reset, the fourth PIO pin also defaults to logic high.

IRQ Glue

The IRQ glue logic combines external IRQ inputs ($\overline{\text{YAMAHA\_IRQ}}$) from a Yamaha sound chip, COM port chip $\overline{\text{COM\_IRQ}}$ (typically a modem), and the MIDI controller and presents them on a single IRQ pin ($\overline{\text{2115\_IRQ}}$) to the MediaStream controller.

ROM Chip Select

The ROM chip select output ($\overline{\text{ROM\_CS\_OUT}}$) is an active low chip select for the MediaStream BIOS ROM. It is not simply a copy of the input $\overline{\text{ROM\_CS\_IN}}$, rather it is a copy of the input that is not active during MPU-401, COM port, or joystick accesses.

COM Chip Select

The COM chip select output ($\overline{\text{COM\_CS\_OUT}}$) is an active low chip select for an external COM port device. It is not simply a copy of the input $\overline{\text{ROM\_CS\_IN}}$, rather it is a copy of the input that is only active during COM port access.

Goggle Toggle

The goggle toggle logic drives the goggle output (GOGGLE) to logic level 0 during reset. The value is subsequently toggled from 0 to 1 or 1 to 0 on each subsequent vsync falling edge. Once the goggle value is correlated to the current even/odd field being displayed, it will maintain that relationship until the CRT display mode is changed.

AES Mux

The AES multiplexing logic provides a 3:1 digital multiplexor to merge three input AES streams. Typically the streams consist of an input stream from the BtV2811 VideoStream Decoder (AES_IN[0]), an input stream from an external AES source such as a DAT tape (AES_IN[1]), and an input stream from a second AES source such as an internal CD-ROM drive (AES_IN[2]).

BtV2210 Pin List Summary

Table 1 provides a summary of the BtV2210 pin list. The overbar above a signal indicates active-LOW. For a pin layout diagram, refer to Figure 3 on page 7.

Table 1. BtV2210 Pin LIst Summary *(1 of 2)*

	Pin Count	Signal Name	I/O	Description
Total Bus Interface Pins = 21	1	$\overline{\text{RESET}}$	I	Resets all BtV2210 functions before execution
	1	LO_ADDR	I	1-bit odd/even address for MIDI controller, Internal Registers. Normally connected to BtV2115 RAM_DATA[8]
	4	HI_ADDR	I	4-bit high order address for decode. Normally connected to BtV2115 RAM_DATA[27:24]. 1000 = COM 1010 = joystick 1011 = BtV2210 Internal Registers 1100 = MIDI all others = ROM
	8	RAM_DATA	I/O	8-bit bi-directional data bus
	1	$\overline{\text{ROM\_CS\_IN}}$	I	Input chip enable
	1	$\overline{\text{2115\_IRQ}}$	O	Interrupt output to BtV2115 controller
	1	CLK17	I	16.9344MHz input clock
	1	$\overline{\text{BUS\_OE}}$	I	Bus output enable
	1	BUS_WRITE	I	1 = bus read 0 = bus write
	1	TEST_MODE	I	Test enable 1 = test mode 0 = normal mode
	1	$\overline{\text{YAMAHA\_IRQ}}$	I	Interrupt input from Yamaha chip
COM Port pins = 2	1	$\overline{\text{COM\_CS\_OUT}}$	O	COM port chip select output
	1	$\overline{\text{COM\_IRQ}}$	I	COM port interrupt input
Total ROM Pins =1	1	$\overline{\text{ROM\_CS\_OUT}}$	O	Decoded ROM chip select output

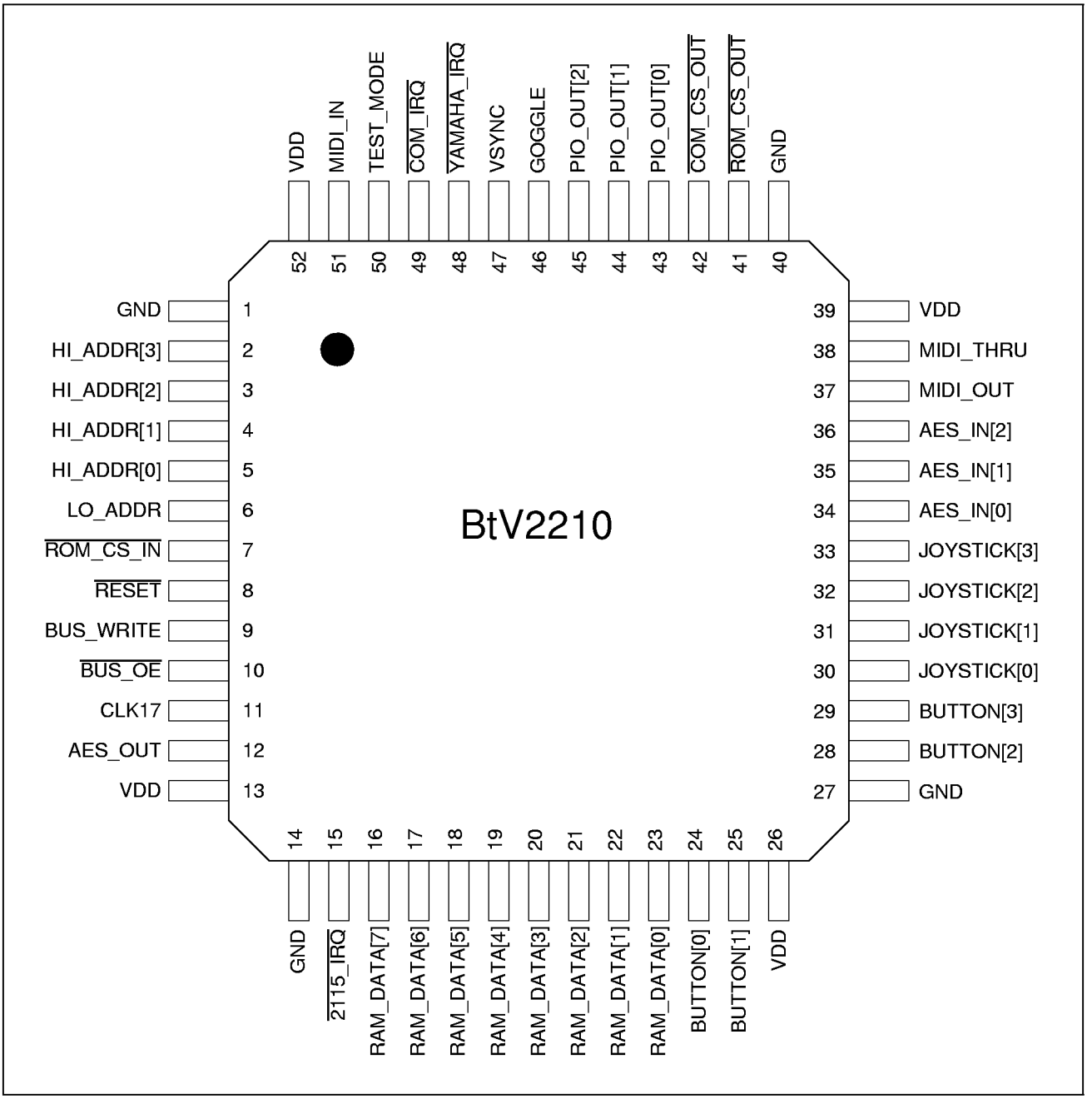
Table 1. BtV2210 Pin LIst Summary *(2 of 2)*

	Pin Count	Signal Name	I/O	Description
Total MIDI Interface Pins = 3	1	MIDI_IN	I	MIDI input
	1	MIDI_OUT	O	MIDI output
	1	MIDI_THRU	O	MIDI throughput
Total Joystick Pins = 8	4	JOYSTICK	I	4 analog connections to joystick
	4	BUTTON	I	4 button inputs
Total Goggle Toggle Pins = 2	1	VSYNC	I	Vertical sync input to toggle the goggle output
	1	GOGGLE	O	Toggles between 0 and 1 on each subsequent vsync falling edge
Total AES Mux Pins = 4	3	AES_IN	I	3 AES inputs
	1	AES_OUT	O	Output of the AES mux
Total PIO Pins = 3	3	PIO_OUT	O	Programmed I/O pins for software to use to control external devices
Total Signal Pins = 44				
Total Non-signal Pins = 8	4	GND		Ground
	4	VDD		± 5V Power Supply
Total BtV2210 Pins = 52				

BtV2210 Pin Layout

Figure 3 shows the pin assignments for the BtV2210. Refer to Table 1 on page 5 for a descriptive summary of each pin.

Figure 3. BtV2210 Pin Layout



External Register Definitions

The BtV2210 external registers are defined in the following paragraphs.

Joystick Data Register

name: JOY_DATA
index: 201h, read/write
address: GUI_BASE + 0x7A0000 +201
size: 8 bit
function: The Joystick Data Register functions under two protocols: compatible and BtV-faSTICK. At reset, the compatible protocol instructs the Joystick Data Register to function as a standard PC Game Port. Both protocols are described in the paragraphs below.

Compatible Protocol

This is the default protocol. Writing to the Joystick Data Register resets the pulse width counter to zero and restarts the pulse width counter. Subsequent reads return a single byte, formatted as shown in Table 2.

Table 2. Joystick Data Register Reset Values

Bit	Value
0	Joystick 0 X value
1	Joystick 0 Y value
2	Joystick 1 X value
3	Joystick 1 Y value
4	Button 0 level
5	Button 1 level
6	Button 2 level
7	Button 3 level

The joystick values start at 1, and over time switch to 0. The time it takes to switch to a 0 indicates the setting of the joystick. The CPU polls the register to determine the joystick setting. The button values are raw values and need to be debounced by the CPU.

The joystick interface automatically switches into compatible protocol whenever old software accesses the Joystick Data Register.

BtV-faSTICK Protocol

The BtV-faSTICK protocol provides the full set of button and joystick values very quickly. To switch in or out of the BtV-faSTICK protocol, the CPU performs four quick writes in succession to the joystick port. The writes are then followed by nine reads to obtain the button values and the eight joystick parameters. Each read to IO address 0x0201 increments the internal read index. Read registers are specified by the value in the index register, as shown in Table 3.

Table 3. Indexing to Read Registers

Index Value	Read Register Contents
0	Debounced button values. See Table 4 for bit definitions
1	High order byte of 16 bit X count for stick 0
2	Low order byte of 16 bit X count for stick 0
3	High order byte of 16 bit Y count for stick 0
4	Low order byte of 16 bit Y count for stick 0
5	High order byte of 16 bit X count for stick 1
6	Low order byte of 16 bit X count for stick 1
7	High order byte of 16 bit Y count for stick 1
8	Low order byte of 16 bit Y count for stick 1

Table 4. Index 0 Bit Contents

Bit	Value
0	1
1	1
2	1
3	1
4	Button 0 level - digitally debounced
5	Button 1 level - digitally debounced
6	Button 2 level - digitally debounced
7	Button 3 level - digitally debounced

MIDI Data Register

name: MIDI_DATA
index: 330h, read/write
address: GUI_BASE + 0x7C0000 + 0x330
size: 8 bit
function: During UART mode, the MIDI Data Register contains the most recent byte of data that has been decoded from the MIDI_IN pin (refer to "MIDI Command Register" on page 10). When the MIDI controller switches between non-UART mode and UART mode, the command acknowledge byte (FEh) is read from this register. Data to go on the MIDI_OUT pin is written to the MIDI Data Register. The MIDI Command Register contains the status bits that tell the CPU when it is safe to read and write this register. Even though a four-deep FIFO exists on both the input and output sides of this register, the status bits must be checked before every read or write even when the FIFO is empty.

MIDI Command Register

name: MIDI_CMD
index: 331h, read/write
address: GUI_BASE + 0x7C0000 + 0x0331
size: 8 bit
function: When read, the MIDI Command Register provides two bits of status to the processor. Refer to Table 5 for bit content definition.

Table 5. MIDI Command Bit Contents

Bit	Description
7	Data-in Available bit (active low)
6	Ready to Receive Data bit (active low)
5:0	Not used

MIDI_CMD[7] is the Data-in Available bit. It is active low; that is, when set to 0 the MIDI port has data available for the CPU. When set to 1, the MIDI port does not have data available for the CPU. As soon as the MIDI controller receives a byte of data from MIDI_IN, this bit is set to 0. The CPU then accesses the MIDI Data Register, which is fed by a four-deep FIFO. The CPU must read the data from the MIDI Data Register before its buffer is full. This bit is also tied to the MIDI_IRQ line. Assuming the MIDI_IN line has continuous data, the CPU has 4x320µs to read the first byte of data before it gets overwritten by the next byte of data.

MIDI_CMD[6] is the Ready to Receive Data bit. It is active low; that is, when set to 0 the MIDI port is ready to receive data. When set to 1, the MIDI port is not ready to receive another byte of data. The MIDI controller contains four levels of buffering. So while one byte is being sent, additional bytes can be written from the CPU and stored in a four-deep FIFO buffer. As long as the active-low Ready to Receive Bit is active (0), the FIFO is not full and the CPU can write to it. Keeping the register filled will maintain a constant data stream on the MIDI_OUT pin. To keep the MIDI_OUT line running continuously, the CPU will need to write to the MIDI Data Register every 320µs with a 1280µs maximum spacing. Start-up is the only time that two bytes can be written.

Data written to the MIDI Command Register is interpreted as a command. The MIDI controller recognizes the following two commands.

Command	Action
3Fh	Enter UART mode (MIDI enable)
FFh	Exit UART mode (MIDI disable)

After either an "Enter UART mode" or an "Exit UART mode" command is sent, the CPU must check the Data-in Available bit. When it becomes active (0), the command acknowledge byte is available in the MIDI Data Register. This byte must be read before incoming MIDI data arrives. The command acknowledge byte value should always be FEh, indicating a successful UART mode transition.

Internal Register Definitions

BtV2210 has a group of internal registers. All are accessed through a pair of registers: the Index Address Register and the Index Data Register. Setting the corresponding value in the Index Address Register provides direct access to the BtV2210 register thru the BtV2210 Index Data Register.

Index Address Register

name: INDEX_ADDR[2:0]
address: GUI_BASE + 0x7B0000, read/write
size: 3 bit
function: The Index Address Register is a 3-bit register that holds the address of the register that will be accessed through the Index Data Register. Table 6 defines the Index Address Register.

Table 6. Index Address Register

Index	Register Selected
0	Chip Revision Register
1	Interrupt Enable Register
2	Interrupt Status Register
3	Goggle Control Register
4	PIO Control Register
5	AES MUX Control Register
6:255	Reserved

Index Data Register

name: INDEX_DATA[7:0]
address: GUI_BASE + 0x7B0001, read/write
size: 8 bit
function: The Index Data Register is an 8-bit register that provides the window into all of the BtV2210 internal registers. Once the Index Address Register is loaded, subsequent reads and writes to the Index Data Register provide read and write access to the corresponding internal register.

Chip Revision Register

name: REVID[7:0]
address: Index Address 0, read only
size: 8 bit
function: The Chip Revision Register provides the revision level of BtV2210, is currently 01h.

Interrupt Enable Register

name: INT_ENABLE[3:0]
address: Index Address 1, read/write
size: 4 bit
function: Writing 1's to this register enables the corresponding interrupt condition to generate an interrupt on the BtV2210 interrupt pin. The values read back are the interrupt enable bits (***not*** the interrupt status). Disabling the interrupt bit does not affect the interrupt status; it only affects whether that particular interrupt condition will drive the interrupt pin from BtV2210 to the MediaStream controller. At power-on reset all bits are set to 0, disabling all interrupts. The Interrupt Enable Register is defined in Table 7.

Table 7. Interrupt Enable Register Content

Bit	Description
0	Enable COM Port Interrupt
1	Enable MIDI Data Ready Interrupt
2	Enable Yamaha (OPL) Interrupt
3	Enable MIDI Acknowledge Interrupt

Interrupt Status Register

name: INT_STATUS[3:0]
index: Index Address 2, read only
size: 4 bit
function: This register provides the interrupt status bits. If set to 1, the interrupt condition is occurring. The value of the interrupt status is ***not*** dependent on the interrupt enable register values. The Interrupt Status Register is defined in Table 8.

Table 8. Interrupt Status Register Content

Bit	Description
0	COM Port Interrupt
1	MIDI Data Ready Interrupt
2	Yamaha (OPL) Interrupt
3	MIDI Acknowledge Interrupt

Goggle Control Register

name: GOGGLE[3:0]
index: Index Address 3, read/write
size: 4 bit
function: This register controls the goggle toggle output. Typically, the goggle toggle output is "toggled" on each falling edge of the VSYNC input. The Goggle Control Register is defined in Table 9.

Table 9. Goggle Control Register

Bit	Description
0	Goggle status (read only). Available as long as goggle is enabled (bit 3=1) and provides the current value of the goggle output pin, including the effect of the invert bit (see bit 1).
1	Invert goggle status output. 1 = Allows software to flip the goggle output bit, enabling matching of the goggle status to the current field displayed. Since the goggle status flips every field, software must read the goggle status at some point and correlate the goggle value with the displayed field (even/odd). 0 = Do not invert (default)
2	Toggle on rising/falling edge of goggle VSYNC. 1 = Goggle output toggles on the falling edge of the incoming VSYNC. 0 = Goggle output toggles on the falling edge of the incoming VSYNC (def).
3	Enable GOGGLE pin 1 = Enable goggle output. 0 = Disable goggle output (def). The goggle output pin follows the PIO control register (See "PIO Control Register" below.) When goggle is disabled, the values in bits 1 and 2 are ignored and the value read from bit 0 is meaningless.

PIO Control Register

name: PIO_CNTL[3:0]
index: Index Address 4, read/write
size: 4 bit
function: The PIO control bits are tied to 4 output pins. The software directly controls the 4 output pins for driving, for example: a tuner that has a 3-line interface. The 4th pin is shared with the goggle toggle function. If goggle toggle is enabled, PIO_CNTL[3] is a don't care (see Table 9, bit 3).

Reads to this register are the values written. The value read for bit 3 is ***not*** the value of the goggle toggle output. The power-on reset value is 0x0F.

AES Mux Control Register

name: AES_MUX[1:0]
index: Index Address 5, read/write
size: 2 bit
function: The AES Mux Control Register controls the AES multiplexor. The power-on reset value is 00. The AES Mux Control Register is defined in Table 10.

Table 10. AES Mux Control Register Contents

2-Bit Value	Input Selection
00	Select AES input 0
01	Select AES input 1
10	Select AES input 2
11	Select AES input 0

Bus Interface Timing Parameters

The BtV2210 synchronizes all inputs to CLK17. So all timing parameters are relative to the rising or falling edge of the ROM_CS_IN pin.

Table 11. BtV2210 Output Specifications for DRAM Signals

Parameter Description	Symbol	min (ns)	max (ns)
ADDRESS setup to $\overline{\text{ROM\_CS\_IN}}$	ASCGAME	5	-
$\overline{\text{ROM\_CS\_IN}}$ setup to $\overline{\text{WE}}$	CSWGAME	5	-
$\overline{\text{ROM\_CS\_IN}}$ hold from $\overline{\text{WE}}$	CHWGAME	0	-
RAM_DATA hold from $\overline{\text{ROM\_CS\_IN}}$	DHCGAME	0	-
$\overline{\text{ROM\_CS\_IN}}$ pulse width	CGAME	130	-
$\overline{\text{WE}}$ pulse width	WEGAME	130	-
$\overline{\text{ROM\_CS\_IN}}$ setup to $\overline{\text{BUS\_OE}}$	CSOEGAME	0	-
$\overline{\text{ROM\_CS\_IN}}$ hold from $\overline{\text{BUS\_OE}}$	CHOEGAME	0	-
$\overline{\text{ROM\_CS\_IN}}$ precharge	CPGAME	70	-
RAM_DATA setup to $\overline{\text{ROM\_CS\_IN}}$	DSCGAME	5	-
$\overline{\text{ROM\_CS\_IN}}$ setup to $\overline{\text{WE}}$	CSWGAME	5	-
$\overline{\text{ROM\_CS\_IN}}$ hold from $\overline{\text{WE}}$	CHWGAME	10	-
ADDRESS hold from $\overline{\text{ROM\_CS\_IN}}$	AHCGAME	0	-
READ DATA setup	DSROM	20	-
READ DATA hold	DHROM	0	20

BtV2210 Timing Diagrams

Figure 4 shows the BtV2210 read cycle; Figure 5 shows the write cycle.

Figure 4. Read Cycle

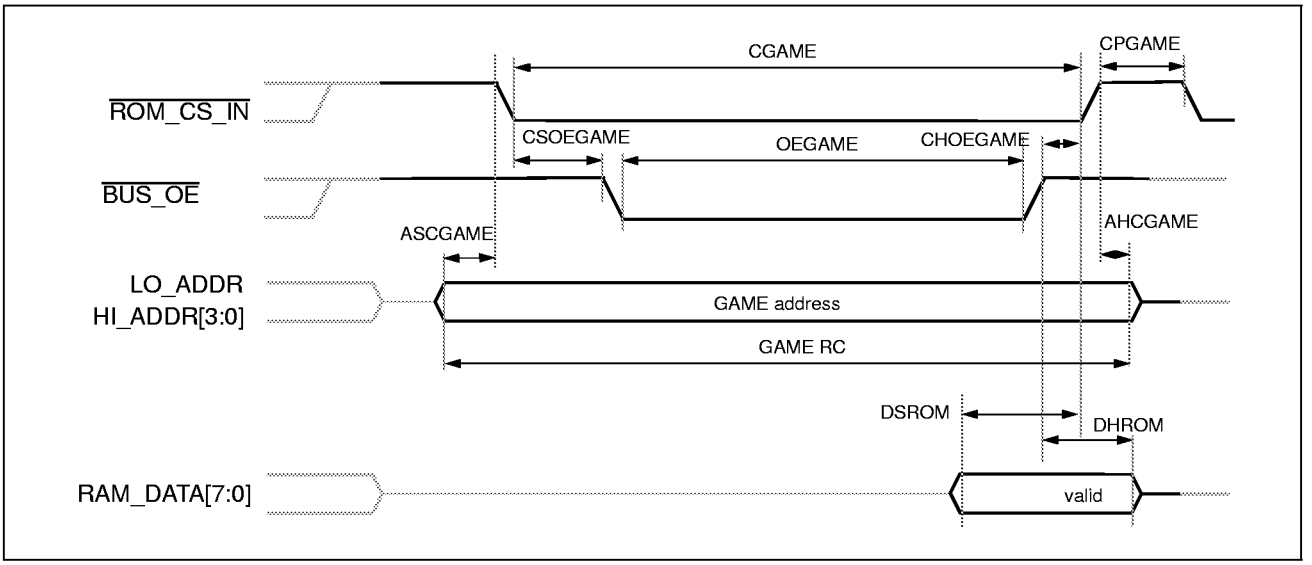
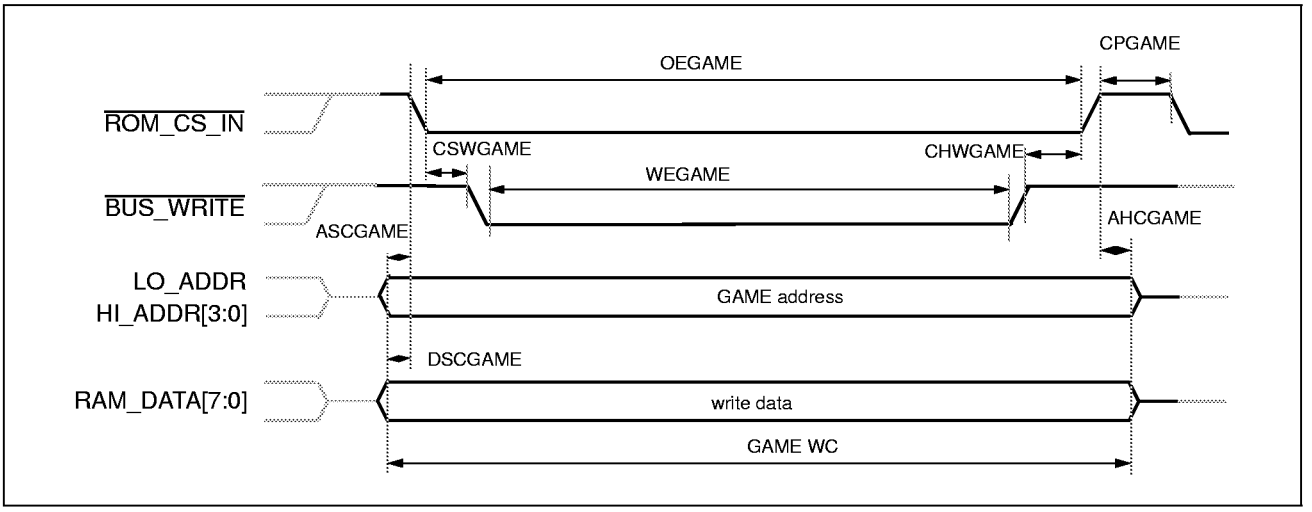


Figure 5. Write Cycle



Output Driver Load Limits

Table 12 lists the maximum recommended load limits for maintaining clean signal transitions. Depending on the receiving device, some applications can operate acceptably beyond these limits.

Table 12. Maximum BtV2210 Load Limits

Signal	Max Capacitance (pF)
RAM_DATA[7:0]	85
MIDI_OUT	50
MIDI_THRU	50
$\overline{2115\_IRQ}$	50
$\overline{COM\_CS\_OUT}$	50
ROM_CS_OUT	50
GOGGLE	50
AES_OUT	50
PIO_OUT[2:0]	50

Absolute Maximum Ratings

Table 13 shows voltage and temperature ranges for the Btv2210.

Table 13. Absolute Maximum Ratings

Parameter	Symbol	Min	Type	Max	Units
VAA, VDD (measured to GND)				7.0	V
Voltage on Any Signal Pin		GND–0.5		VDD+0.5	V
Ambient Operating Temperature	T_A	–55		+125	˚C
Storage Temperature	T_S	–65		+150	˚C
Junction Temperature	T_J			+150	˚C
Vapor Phase Soldering (1 minute)	T_{VSOL}			220	˚C
Stresses above those listed in this table may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those listed in the operational sections of this specification are not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. This device employs high-impedance CMOS devices on all signal pins. It should be handled as an ESD-sensitive device. Voltage values on any signal pin that extend beyond the power supply rails by more than the amount(s) specified above can cause destructive latchup.					

Recommended Operating Conditions

Table 14 shows the recommended operating conditions for the BtV2210.

Table 14. Recommended Operating Conditions

Parameter	Min	Type	Max	Units
Load Capacitance on digital outputs: AES_OUT, $\overline{2115\_IRQ}$, MIDI_OUT, MIDI_THRU, $\overline{ROM\_CS\_OUT}$, $\overline{COM\_CS\_OUT}$, PIO_OUT[2:0], GOGGLE			25	pF
Load Capacitance on IO pins RAM_DATA[7:0]			85	pF
Supply Voltage	4.75	5.00	5.25	V
Junction Temperature T_j	0		125	˚C

Target DC Characteristics

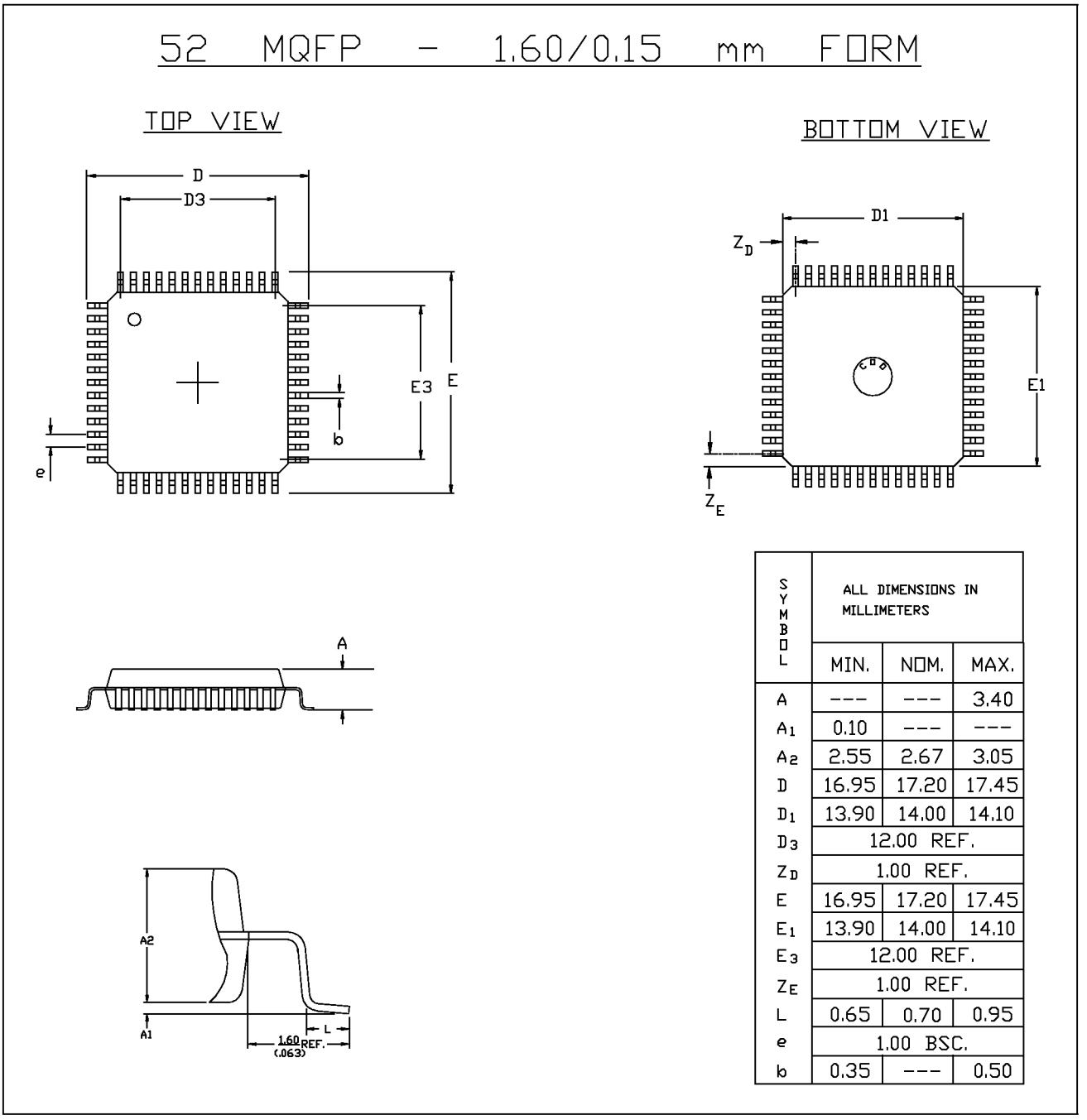
Target DC characteristics are shown in Table 15.

Table 15. Target DC Characteristics

Parameter	Symbol	Min	Type	Max	Units
Digital Inputs					
Input High Voltage	V_{IH}	2.0		V_{DD} +0.5	V
Input Low Voltage	V_{IL}	GND–0.5		0.8	V
Input High Current (V_{in} = 2.4 V)	I_{IH}			10	µA
Input Low Current (V_{in} = 0.4 V)	I_{IL}	–10			µA
Input Capacitance (v = 1 MHz, Vin = 2.4 V)	C_{IN}			7	pF
Hysteresis if applicable			0.3		V
Digital Outputs					
Output High Voltage (I_{OH} = –400 µA)	V_{OH}	2.4			V
Output Low Voltage (I_{OL} = 3.2 mA)	V_{OL}			0.4	V
Three-State Current (0–2.4V)	I_{OZ}			50	µA

Package Mechanical Drawing

Figure 6. 52-Pin PQFP Package Mechanical Drawing



SYMBOL	ALL DIMENSIONS IN MILLIMETERS		
	MIN.	NOM.	MAX.
A	---	---	3.40
A_1	0.10	---	---
A_2	2.55	2.67	3.05
D	16.95	17.20	17.45
D_1	13.90	14.00	14.10
D_3	12.00 REF.		
Z_D	1.00 REF.		
E	16.95	17.20	17.45
E_1	13.90	14.00	14.10
E_3	12.00 REF.		
Z_E	1.00 REF.		
L	0.65	0.70	0.95
e	1.00 BSC.		
b	0.35	---	0.50

Revision History

Table 16. BtV2210 Datasheet Revision History

Revision	Date	Change	Description
A	8/11/95	Initial Release	