Technical Manual

for

SBC488/OEM

Single Board Computer with the IEEE 488 Interface

by

TIDAL

TIDAL ENGINEERING
2 Emery Avenue
Randolph, NJ  07869

tel: 973-328-1181
fax: 973-328-2302
email: tidaleng@gti.net

By: Craig Borax
Date: 4 November 1996
Rev. A
Forward

The SBC488/OEM technical manual covers the operation and maintenance for the SBC488. There also is another version of the SBC488 called the SBC488/TEST that supports the board in test applications. In test applications, the SBC488 acts as a test controller and is used to create small to medium sized automated test systems. The SBC488/TEST manual covers both operation, maintenance and software development.
# Table of Contents

1.0 **OVERVIEW** ................................................................................................................................. 1

1.1 **PART NUMBER BREAKDOWN** .............................................................................................. 3

2.0 **SPECIFICATIONS** ...................................................................................................................... 4

  2.1 **THE IEEE 488 INTERFACE** ................................................................................................. 4
  2.2 **RS-232 INTERFACE** ............................................................................................................ 4
  2.3 **OPTIONAL ANALOG TO DIGITAL CONVERTER (ADC) SPECIFICATIONS** .. 4
  2.4 **OPTIONAL DIGITAL TO ANALOG CONVERTER (DAC) SPECIFICATIONS** .................. 5
  2.5 **OPTIONAL DIGITAL INPUT/OUTPUT (I/O) SPECIFICATIONS** ............................................. 5
  2.6 **SMARTCore(TM) SPECIFICATIONS** .................................................................................... 6
  2.7 **INDICATORS AND CONTROLS** ............................................................................................ 7
  2.8 **PIN DESCRIPTIONS** ............................................................................................................. 8
  2.9 **PHYSICAL DESCRIPTION** .................................................................................................... 10

3.0 **FUNCTIONAL DESCRIPTION** .................................................................................................. 11

  3.1 **IEEE 488 INTERFACE** ........................................................................................................ 11
  3.2 **RS-232 INTERFACE** .......................................................................................................... 13
  3.3 **OPTIONAL ANALOG TO DIGITAL CONVERTER (ADC)** .................................................. 14
  3.4 **OPTIONAL DIGITAL TO ANALOG CONVERTER (DAC)** .................................................. 15
  3.5 **OPTIONAL DIGITAL INPUT/OUTPUT (I/O)** ........................................................................ 17
  3.6 **SMARTCore FEATURES** ..................................................................................................... 18
  3.7 **POWER SUPPLY FEATURES** ............................................................................................. 19

4.0 **OPERATING INSTRUCTIONS** .................................................................................................. 21

  4.1 **UNPACKING** .......................................................................................................................... 21
  4.2 **SETUP AND CONFIGURATION** ............................................................................................. 21
  4.3 **ELECTRICAL TESTING USING IEEE 488 COMMUNICATIONS** ........................................ 21
    4.3.1 **Equipment Required** .................................................................................................. 21
    4.3.2 **Procedure for using the SBCwin program with IEEE 488 communications** ........ 21
    4.3.3 **Procedure for using IOtech Quicktest Program** .......................................................... 22
  4.4 **ELECTRICAL TESTING USING RS-232 COMMUNICATIONS** ........................................... 24
    4.4.1 **Equipment Required** .................................................................................................. 24
    4.4.2 **Procedure for using SBCwin program with RS-232 communications** ................... 24
    4.4.3 **Procedure for using the Terminal Program** ................................................................. 25

5.0 **PROGRAMMING** ....................................................................................................................... 27

  5.1 **DEVICE SPECIFIC COMMANDS** ........................................................................................ 27
  5.2 **IEEE 488 EVENT PROCESSING** ......................................................................................... 28
  5.3 **IEEE 488 STANDARD COMMANDS** ............................................................................... 30
  5.4 **PROGRAM EXAMPLE USING IEEE 488 STANDARD COMMANDS** ........................... 32

6.0 **TROUBLESHOOTING** ................................................................................................................. 32

7.0 **DRAWINGS AND SCHEMATICS** .............................................................................................. 35

  7.1 **SBC488 PARTS LIST** ............................................................................................................ 35
  7.2 **SCHEMATICS AND DRAWINGS** ........................................................................................ 35
List of Figures

Figure 1, SBC488 Block Diagram.................................................................2
Figure 2, SmartCore Block Diagram ...........................................................6
Figure 3, IEEE 488 Address Switch ............................................................7
Figure 4, Indicator Lights............................................................................7
Figure 5, Miscellaneous Control Schematic .................................................13
Figure 6, IEEE 488 and RS-232 Interface Schematics.................................14
Figure 7, Analog to Digital Converter Schematic .........................................15
Figure 8, Digital to Analog Converter Schematic .........................................16
Figure 9, Digital I/O ...................................................................................18
Figure 10, SmartCore Block Diagram..........................................................19
Figure 11, SBC488 Power Supply Diagram .................................................20
Figure 12, ESE and ESR Generation ............................................................29
Figure 13, Status Byte Generation ...............................................................30
Figure 14, SBC488 Schematic Diagram......................................................37
Figure 15, SBC488 Component Layout.........................................................38
## SAFETY

### WARNING

High voltage AC and DC exist on the SBC488 printed circuit board. Do not touch the unit with power on. Failure to observe this warning could result in serious injury or death.

### Copyright Notice

This documentation, firmware and all printed circuit board artworks are copyrighted with all rights reserved by Tidal Engineering, Whippany, NJ USA. No part of this product may be copied, reproduced or transmitted by any means without prior written permission from Tidal Engineering. It is unlawful to copy EPROM contents without permission. Copyright © Tidal Engineering, 1996. All Rights Reserved.

### Trademark Notices

SmartCore and Z-World Engineering are trademarks of Z-World, Inc. Windows is a registered trademark of Microsoft Corporation. IOtech is a trademark of IOtech, Inc.
1.0 OVERVIEW

The SBC488 is a single board computer that incorporates a variety of input and output capabilities beginning with two communication interfaces.

The SBC488 gets its name from the IEEE 488 bus which is also known as the General Purpose Interface Bus (GPIB). The IEEE 488 bus allows up to 15 instruments within a localized area to communicate with each other. Data is sent through the bus in parallel a byte (8 bits) at a time. The data consists of device dependent (commands specific to the SBC488) and generic interface messages (commands common to all IEEE 488 devices). Each device is assigned and responds to a unique address. The address is set by a five position address dipswitch that is read by the SBC488 at power up.

The SBC488 implements only the Talker/Listener functions defined by the IEEE 488 standard. The SBC488 therefore, acts as a slave to the bus controller. It responds to data request commands and setpoint commands. Another SBC488 model, the SBC488/TEST can perform talker/listener and IEEE 488 bus controller functions and is used to control small to medium sized automated test systems. See Figure 1, the SBC488 block diagram.

In addition to the IEEE 488 interface the SBC488 also supports RS-232 communications at 9600 Baud. The RS-232 interface is an asynchronous serial interface. Commands are received and data is sent by the SBC488 over the RS-232 interface. Either the IEEE 488 or the RS-232 interface can control the SBC488. This provides the end user with the flexibility of multiple communication options.

The optional analog input on the SBC488 consists of an eleven channel, 12 bit Analog to Digital Converter (ADC). The ADC is isolated from the MicroProcessor Unit (MPU) and the communication interfaces with high speed opto-isolators affording a 2500 VAC isolation barrier.

The optional analog output on the SBC488 consists of four, 12 bit Digital to Analog Converters (DAC). The DACs are also isolated from the MPU and the communication interfaces.

The SBC488’s optional digital I/O is capable of 32 bits of bi-directional input and output. The optional four 8 bit ports are also optically isolated.

The SBC488 incorporates an MPU module from Z-World Engineering called a SmartCore. The SmartCore features a Z180 MPU running at 9.216 Mhz.
The SBC488 supports software development using the optional Dynamic C environment available from Z-World Engineering. With the software development option and Dynamic C, a programmer can make modifications to the generic firmware of the SBC488. This allows changes to the standard command set and the incorporation of additional commands. The software development option includes a 128K EPROM with the Dynamic C monitor installed.
1.1 Part Number Breakdown

The part number for the SBC488/OEM determines the installed options. The part number is created as follows:

<table>
<thead>
<tr>
<th>SBC488/OEM</th>
<th>-xDA</th>
<th>x.xx</th>
<th>-ADx.x</th>
<th>-xDIG</th>
<th>-DEV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model number</td>
<td>no. of DAC channels</td>
<td>DAC Option voltage reference</td>
<td>ADC Option voltage reference</td>
<td>No. of Digital Ports</td>
<td>Software Develop. Option</td>
</tr>
<tr>
<td>SBC488/OEM</td>
<td>blank</td>
<td>blank</td>
<td>blank</td>
<td>blank</td>
<td>blank</td>
</tr>
<tr>
<td>-1DA</td>
<td>2.50</td>
<td>-AD2.50</td>
<td>-1DIG</td>
<td>-DEV</td>
<td></td>
</tr>
<tr>
<td>-2DA</td>
<td>4.10</td>
<td>-AD4.10</td>
<td>-2DIG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-3DA</td>
<td>5.00</td>
<td>-AD5.00</td>
<td>-3DIG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-4DA</td>
<td>8.20</td>
<td>-AD8.20</td>
<td>-4DIG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-5DA</td>
<td>10.0</td>
<td>-AD10.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-6DA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Part number example:

SBC488/OEM-4DA2.50-AD2.50-3DIG

This part number is for an SBC488 single board computer with the OEM firmware and the following hardware options:

- Four DACs with a 0-2.5 VDC full scale range.
- Eleven channel ADC with a 0-2.5 VDC full scale range.
- Three digital I/O ports, 8 bits each for a total of 24 I/O.
2.0 Specifications

2.1 The IEEE 488 Interface

Specification: IEEE 488.1 and IEEE 488.2 (partial)
Controller: NEC uPD7210 or National Instruments 7210
Controller Clock: 8 Mhz
Transceivers: 75160 and 75162 (Texas Instruments)
Address Capability: 0 to 30
SRQ generation: ESE,SRE masks (See section 5, programming)

2.2 RS-232 Interface

RS-232 Communications
Baud Rate: 9600 Baud
Data Size: 8 bit
Parity: None
Stop bits: 1
Connector Configuration: 9 Pin D-Sub, IBM Pinout

2.3 Optional Analog to Digital Converter (ADC) Specifications

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels</td>
<td>11</td>
</tr>
<tr>
<td>Resolution</td>
<td>12 bit</td>
</tr>
<tr>
<td>ADC P/N</td>
<td>TLC2543</td>
</tr>
<tr>
<td>Linearity Error</td>
<td>+/-1 LSB max.</td>
</tr>
<tr>
<td>Gain Error</td>
<td>+/-1 LSB max.</td>
</tr>
<tr>
<td>Offset Error</td>
<td>+/-1.5 LSB max.</td>
</tr>
</tbody>
</table>

Voltage Reference

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference Voltage Options</td>
<td>2.50, 4.1 and 5.0 VDC</td>
</tr>
<tr>
<td>Reference P/N</td>
<td>LM4040-x</td>
</tr>
<tr>
<td>Reference Accuracy</td>
<td>+/- 0.5% max.</td>
</tr>
<tr>
<td>Temperature Coeff.</td>
<td>+/- 100 ppm/°C</td>
</tr>
</tbody>
</table>

Isolation: 2500 VAC (Between CPU and I/O)
### 2.4 Optional Digital to Analog Converter (DAC) Specifications

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels</td>
<td>6 Max.</td>
</tr>
<tr>
<td>Resolution</td>
<td>12 bits</td>
</tr>
<tr>
<td>DAC P/N</td>
<td>LTC1257</td>
</tr>
<tr>
<td>Offset Error</td>
<td>+/-5mV max.</td>
</tr>
<tr>
<td>Total Error</td>
<td>+/-2 LSB max.</td>
</tr>
<tr>
<td>Output Impedance</td>
<td>300 ohm max.</td>
</tr>
<tr>
<td>Rated load</td>
<td>2 ma</td>
</tr>
<tr>
<td>Temperature Coeff.</td>
<td>+/- 0.02 LSB/°C max.</td>
</tr>
</tbody>
</table>

### Voltage Reference

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference Voltage</td>
<td>2.50, 4.1, 5.0, 8.2 and 10.0 VDC</td>
</tr>
<tr>
<td>Reference P/N</td>
<td>LM4040-x</td>
</tr>
<tr>
<td>Reference Accuracy</td>
<td>+/- 0.5% max.</td>
</tr>
<tr>
<td>Temperature Coeff.</td>
<td>+/- 100 ppm/°C</td>
</tr>
</tbody>
</table>

Isolation: 2500 VAC (Between CPU and I/O)

### 2.5 Optional Digital Input/Output (I/O) Specifications

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of ports</td>
<td>4 eight bit ports (24 bits total)</td>
</tr>
<tr>
<td>Type</td>
<td>Quasi Bi-directional</td>
</tr>
<tr>
<td>Port P/N</td>
<td>Phillips PCF8574</td>
</tr>
<tr>
<td>Output Voltage</td>
<td>5.0 VDC typical.</td>
</tr>
<tr>
<td>Input Voltage Range</td>
<td>-0.5 to 5.5 VDC</td>
</tr>
<tr>
<td>Output Current Low</td>
<td>10 ma Min.</td>
</tr>
<tr>
<td>Output Current High</td>
<td>30 uA Min.</td>
</tr>
<tr>
<td>Input Voltage Low</td>
<td>1.5 VDC Max.</td>
</tr>
<tr>
<td>Input Voltage High</td>
<td>3.5 VDC Min.</td>
</tr>
</tbody>
</table>

Isolation: 2500 VAC (Between CPU and I/O)
2.6 **SmartCore**(tm) **Specifications**

The SBC488 incorporates an MPU module from Z-World Engineering that provides a compact and powerful central processor core module (Figure 2, SmartCore block diagram). The SmartCore(tm) features:

- Z180 CPU running at 9.216 Mhz.
- Static Ram (SRAM), 32K to 128K for data storage (with optional battery backup).
- Erasable Programmable Read Only Memory (EPROM), 32K to 128K for program storage.
- Electrically Erasable PROM (EEPROM), 512 bytes, for non-volatile storage without a battery.
- Real-Time Clock (RTC), for time calculations.
- Reset and power failure supervisor to provide reset to the MPU if power is out of range and to switch in the battery back-up if there is a battery installed.
- Serial interface for program development option, Port 1.
- Serial interface for the application, Port 0.
- Two programmable timers used as interval timers.
- Two Direct Memory Access (DMA) channels for high speed transfers to internal serial ports and external I/O.
- Memory decoding for external devices (/CS1-/CS6).

![SmartCore Block Diagram](image)

**Figure 2, SmartCore Block Diagram**
2.7 Indicators and Controls

a. Address switch
The IEEE 488 address is programmed via the 5 position dip switch on the rear of the SBC488. The address is configured as follows:

![Diagram of Address Switch]

Figure 3, IEEE 488 Address Switch
Note: Cycle SBC488 power after changing dip switch setting.

b. Indicator Lights
The SBC488 uses five indicator lights to indicate the status of the communication channel. The IEEE 488 interface and RS-232 interface share the indicator lights. The light’s functions are similar for both communication channels.

![Diagram of Indicator Lights]

Figure 4, Indicator Lights
Indicator light functions for IEEE 488 communications

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>TALK</td>
<td>This light indicates that the IEEE 488 bus has requested data from the SBC488 and that the data is being sent.</td>
</tr>
<tr>
<td>LISTEN</td>
<td>This light indicates that IEEE 488 bus has put the SBC488 in the listen mode by sending its listen address.</td>
</tr>
<tr>
<td>SRQ</td>
<td>The SRQ or Service Request light indicates that a condition has occurred in the SBC488 that requires service. The logic of SRQ generation is programmable over the IEEE 488 and is explained in section 5.2.</td>
</tr>
<tr>
<td>ERROR</td>
<td>This light is lit when a command error occurs. The light will go out when the next valid command is received.</td>
</tr>
<tr>
<td>TEST</td>
<td>This light is not used in this version</td>
</tr>
</tbody>
</table>

Indicator light functions for RS-232 communications

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>TALK</td>
<td>This light indicates that the RS-232 bus has requested data from the SBC488 and that the data is being sent.</td>
</tr>
<tr>
<td>LISTEN</td>
<td>This light indicates that RS-232 bus is sending data to the SBC488.</td>
</tr>
<tr>
<td>SRQ</td>
<td>This light is not used.</td>
</tr>
<tr>
<td>ERROR</td>
<td>This light is lit when a command error occurs. The light will go out when the next valid command is received.</td>
</tr>
<tr>
<td>TEST</td>
<td>This light is not used.</td>
</tr>
</tbody>
</table>

2.8 Pin Descriptions

a. P1 IEEE 488 Connector

<table>
<thead>
<tr>
<th>Pin</th>
<th>Name/Description</th>
<th>Pin</th>
<th>Name/Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DIO1/Data line</td>
<td>13</td>
<td>DIO5/Data line</td>
</tr>
<tr>
<td>2</td>
<td>DIO2/Data line</td>
<td>14</td>
<td>DIO6/Data line</td>
</tr>
<tr>
<td>3</td>
<td>DIO3/Data line</td>
<td>15</td>
<td>DIO7/Data line</td>
</tr>
<tr>
<td>4</td>
<td>DIO4/Data line</td>
<td>16</td>
<td>DIO8/Data line</td>
</tr>
<tr>
<td>5</td>
<td>EOI/End or Identify</td>
<td>17</td>
<td>REN/Remote Enable</td>
</tr>
<tr>
<td>6</td>
<td>DAV/Data Valid</td>
<td>18</td>
<td>DAV Gnd</td>
</tr>
<tr>
<td>7</td>
<td>NRFD/Not Ready For Data</td>
<td>19</td>
<td>NRFD Gnd</td>
</tr>
<tr>
<td>8</td>
<td>NDAC/Not Data Accepted</td>
<td>20</td>
<td>NDAC Gnd</td>
</tr>
<tr>
<td>9</td>
<td>IFC/Interface Clear</td>
<td>21</td>
<td>IFC Gnd</td>
</tr>
<tr>
<td>10</td>
<td>SRQ/Service Request</td>
<td>22</td>
<td>SRQ Gnd</td>
</tr>
<tr>
<td>11</td>
<td>ATN/Attention</td>
<td>23</td>
<td>ATN Gnd</td>
</tr>
<tr>
<td>12</td>
<td>Shield</td>
<td>24</td>
<td>Ground</td>
</tr>
</tbody>
</table>

b. P3 DAC and ADC Connector
<table>
<thead>
<tr>
<th>Pin</th>
<th>I/O</th>
<th>Name/Description</th>
<th>Voltage Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O</td>
<td>CH0/Channel 0 output</td>
<td>0 to DAC VREF*4095/4096</td>
</tr>
<tr>
<td>2</td>
<td>O</td>
<td>CH1/Channel 1 output</td>
<td>0 to DAC VREF*4095/4096</td>
</tr>
<tr>
<td>3</td>
<td>O</td>
<td>CH2/Channel 2 output</td>
<td>0 to DAC VREF*4095/4096</td>
</tr>
<tr>
<td>4</td>
<td>O</td>
<td>CH3/Channel 3 output</td>
<td>0 to DAC VREF*4095/4096</td>
</tr>
<tr>
<td>5</td>
<td>O</td>
<td>CH4/Channel 3 output</td>
<td>0 to DAC VREF*4095/4096</td>
</tr>
<tr>
<td>6</td>
<td>O</td>
<td>CH5/Channel 3 output</td>
<td>0 to DAC VREF*4095/4096</td>
</tr>
<tr>
<td>23</td>
<td>I</td>
<td>CH0/Channel 0 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>25</td>
<td>I</td>
<td>CH1/Channel 1 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>26</td>
<td>I</td>
<td>CH2/Channel 2 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>21</td>
<td>I</td>
<td>CH3/Channel 3 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>19</td>
<td>I</td>
<td>CH4/Channel 4 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>17</td>
<td>I</td>
<td>CH5/Channel 5 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>15</td>
<td>I</td>
<td>CH6/Channel 6 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>22</td>
<td>I</td>
<td>CH7/Channel 7 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>11</td>
<td>I</td>
<td>CH8/Channel 8 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>13</td>
<td>I</td>
<td>CH9/Channel 9 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>18</td>
<td>I</td>
<td>CH10/Channel 10 input</td>
<td>0 to ADC VREF*4095/4096</td>
</tr>
<tr>
<td>10</td>
<td>I</td>
<td>Ground Reference</td>
<td>Reference</td>
</tr>
</tbody>
</table>

c. P4 RS-232 Interface Connector

<table>
<thead>
<tr>
<th>Pin</th>
<th>I/O</th>
<th>Name/Description</th>
<th>Voltage Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I</td>
<td>Gnd</td>
<td>Reference</td>
</tr>
<tr>
<td>2</td>
<td>I</td>
<td>RX/Receive Data</td>
<td>-8 to +8 VDC</td>
</tr>
<tr>
<td>3</td>
<td>O</td>
<td>TX/Transmit Data</td>
<td>-8 to +8 VDC</td>
</tr>
<tr>
<td>4</td>
<td>I</td>
<td>CTS/Clear to Send</td>
<td>-8 to +8 VDC</td>
</tr>
<tr>
<td>5</td>
<td>O</td>
<td>RTS/Request to Send</td>
<td>-8 to +8 VDC</td>
</tr>
<tr>
<td>7</td>
<td>I</td>
<td>Gnd</td>
<td>Ref</td>
</tr>
</tbody>
</table>
d. P6 Power

<table>
<thead>
<tr>
<th>Pin</th>
<th>Name/Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>120 VAC +/- 20%</td>
</tr>
<tr>
<td>2</td>
<td>GND</td>
</tr>
<tr>
<td>3</td>
<td>120 VAC +/- 20%</td>
</tr>
</tbody>
</table>

g. P9 Digital I/O Connector

<table>
<thead>
<tr>
<th>Pin</th>
<th>I/O</th>
<th>Name/Description</th>
<th>Voltage Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O</td>
<td>+5 VDC Power</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>2</td>
<td>I/O</td>
<td>D1-LSB Data 0 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>3</td>
<td>I/O</td>
<td>D1 Data 1 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>10</td>
<td>I/O</td>
<td>D2-LSB Data 0 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>11</td>
<td>I/O</td>
<td>D2- Data 1 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>18</td>
<td>I/O</td>
<td>D3-LSB Data 0 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>19</td>
<td>I/O</td>
<td>D3 Data 1 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>26</td>
<td>I/O</td>
<td>D4-LSB Data 0 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>27</td>
<td>I/O</td>
<td>D4- Data 1 I/O</td>
<td>+5 VDC</td>
</tr>
<tr>
<td>34</td>
<td>O</td>
<td>Power Return</td>
<td>0 VDC</td>
</tr>
</tbody>
</table>

2.9 Physical Description

Dimensions: 5.0” x 9.0” (127mm x 229mm)
Weight: 0.74 lb. (330 gm)
Power Requirements: 120 VAC +/- 20%, 6 VA, 4 W
Operating temperature: 0 to 70 °C.
3.0 Functional Description

3.1 IEEE 488 Interface

The IEEE 488 bus, originally developed by Hewlett Packard, is designed to connect test equipment to computers and computer peripherals. It allows up to 15 devices within a localized area to communicate with each other. Data is sent by one device (the talker) and received by any number of listeners. The bus controller sends commands to the devices on the bus. The talker/listener devices can send data and receive data and commands.

Each device has a unique address that it responds to. The address of a device is usually set by a group of dip switches on the device that are read at power up.

The interface consists of eight data lines (DIO0-DIO8) and eight handshake/control lines. The handshaking lines are DAV, NRFD, NDAC. The control lines are IFC, ATN, SRQ, REN and EOI.

The DAta Valid (DAV) line tells the recipient of the data that it is valid. The Not Ready For Data (NRFD) line tells the sender that at least one device is not ready for data. The NDAC line tells the sender that at least one device has not accepted the data.

The bus controller uses the ATN line to tell the devices on the bus that it is sending a command. When the ATN line is true, all devices listen for the command coming over the bus. When the ATN is false, only addressed devices will participate.

The Service ReQuest line (SRQ) allows any device on the bus to request service and “interrupt” the controller.

The End Or Identify line (EOI) can be asserted by the talker when the last data byte has been put on the bus. This is optional and is set up in the IEEE 488 bus controller. The SBC488 requires that the EOI is asserted during the last byte of a data transfer.

The IEEE 488 interface in the SBC488 is managed by an NEC uPD7210 type Application Specific Integrated Circuit (ASIC). The IEEE 488 Integrated Circuit (IC), U81 is connected to the SmartCore’s 8 bit data bus (See Figure 5). In addition, U81 is connected to the least significant three bits of the address bus. These address lines are used by the CPU to address the eight read and write registers that control the U81’s operation. Also, U81 connects to the SmartCore’s /CS1, /IORD, /IOWR and /INT1 lines. Note that the “/” before the signal name means that the signal is active low.

The Chip Select 1 (/CS1) signal is an active low signal that tells the IEEE 488 IC that it is being addressed.
The Input Output Read (/IORD) line is an active low signal that tells the IEEE 488 IC that the MicroProcessor Unit (MPU) is reading data from one of its eight registers. The IEEE 488 IC then enables its data bus drivers and puts the data from the selected register on the bus.

The Input Output Write (/IOWR) line is an active low signal that tells the IEEE 488 IC that the MPU is writing data to it. The IEEE 488 IC then reads the data bus and stores the data in the register selected by the three address lines.

U81 also requires an external clock that is provided by an 8 Mhz CMOS oscillator IC, U110.

The IEEE 488 IC (U81) is interfaced to the IEEE 488 bus through two application specific GPIB transceivers U82 (75160) and U83 (75162). The data transceiver (U82) is a bi-directional high speed tristate driver/receiver. The control transceiver (U83) serves as the interface to the control lines (DAV, NRFD, NDAC, IFC, ATN, SRQ, REN, EOI). U83 has both tristate drivers and open collector drivers as determined by the IEEE 488 standard.

The IEEE 488 interface connector, P1 is a shielded connector specifically designed for the IEEE 488 bus.

The interface between the SmartCore and U81 is interrupt driven. That is, once U81 is programmed at initialization, it is only serviced when data is received or requested by the bus. The U81 interrupt line, U81-11, which is normally low, will go high after the communication begins on the bus and low after the MPU services the interrupt and reads the appropriate register. The SmartCore /INT1 (U100-21) pin is used for this signal.

The IEEE 488 address is programmed by DSW1 (See figure 5) and defined by the IEEE standard to be between 0 and 30. The SBC488 interprets an address of 31 to mean that RS-232 control is being used. The data buffer/input port U71 (74HC541) allows the MPU to read the address set on the switch. The MPU then programs U81 at power up for that device address.
3.2 RS-232 Interface

In addition to the IEEE 488 interface the SBC488 also supports RS-232 communications at 9600 Baud.

The Z180 Microprocessor has two integrated UART (Universal Asynchronous Receiver/Transmitter) ports, 0 and 1. The UART channel 0 is used by the SBC488 application. The channel 1 UART is used for software development with the -DEV option. The UART signals are passed through the RS-232 transceiver U85 (See Figure 6). U85 has an onboard charge pump circuit to generate the +/- 9 volts required by...
the RS-232 interface. The RS-232 UART port is interrupt driven and generates an internal interrupt when it requires attention.

The RS-232 interface connects externally through a 9 Pin D-Sub-miniature connector (P4) PCB mounted on the SBC488. Note that the CTS signal (RTS at the IBM) should be driven true to enable communications.

Figure 6, IEEE 488 and RS-232 Interface Schematics

### 3.3 Optional Analog to Digital Converter (ADC)

The analog input on the SBC488 consists of an eleven channel, 12 bit ADC. The ADC is isolated from the MPU and the communication interfaces with high speed opto-isolators affording a 2500 VAC isolation barrier.

The ADC IC, U90, interfaces via a four wire serial interface that simplifies optical isolation. The interface signals consist of:

- a. Serial Clock (U90-18)
b. Chip Select (U90-15)
c. Data Out (U90-16)
d. Data In (U90-17)

These digital signals are manipulated by the MPU via the 8 bit addressable latch, U61 and are isolated by U41 opto-isolator (See Figure 5). The Data Out pin of the ADC is isolated by U44 opto-isolator and is read by the 8 bit input port.

The ADC reference inputs are provided by a precision voltage reference U22. The ADC converter’s full scale voltage can be changed to 2.5, 4.096 or 5.0 VDC by substituting the appropriate version of U22. Note that the ADC reference can only accept 5.0 VDC max. whereas the DACs can accept up to 10 VDC. The ADC inputs can accept higher voltages, however, by configuring a voltage divider with RN9-RN11 and optional RN31.

The ADC’s analog inputs are protected by diode clamps to +5 VDC and ground and by input resistor networks RN9-RN11. The input resistors in conjunction with capacitor networks CN1 and CN2 provide a low pass filter to reduce input noise on the ADC.

Figure 7, Analog to Digital Converter Schematic

3.4 Optional Digital to Analog Converter (DAC)

The analog output on the SBC488 consists of up to six 12 bit DACs. The DACs are isolated from the MPU and the communication interfaces.

The DAC ICs, U11-U16, interface via a four wire serial interface. The interface signals consist of:

a. Serial Clock (U11-1)
b. Load DAC (U11-3)
c. Serial Data In (U11-2)
d. Serial Data Out (U11-4)

These digital signals are manipulated by the MPU via the 8 bit addressable latch, U61 and are isolated by U42 opto-isolator.

The DAC reference inputs are provided by U21.

The DAC’s analog outputs are protected by an output resistor network, RN8.

Figure 8, Digital to Analog Converter Schematic
3.5 Optional Digital Input/Output (I/O)

The SBC488’s optional digital I/O provides up to 32 bits of bi-directional digital I/O. The four 8 bit ports are isolated from the MPU and the communication interfaces with high speed opto-isolators affording a 2500 VAC isolation barrier.

The digital I/O port IC’s, U51-U53, are remote 8 bit I/O expanders for the I²C bus. Their I²C interface is a two wire serial interface. (See Figure 9, Digital I/O)

The interface signals consist of:

a. Serial Clock SCL (U51-14)
b. Serial Data SDA (U51-15)

The I²C interface incorporates a serial addressing method whereby data is written to or read from a specific port among up to eight ports that are connected on the I²C bus. The port address is configured by three pins on each chip, A0 (Pin 1), A1 (Pin 2) and A2 (Pin 3). These three address pins allow up to eight devices to be individually identified on the same serial bus. In the SBC488, U51 is at address 0, U52 is at address 1, U53 is at address 2 and U54 is at address 3.

Each pin on these ports can sink up to 10 ma and source up to 30 uA. Additional source current can be provided by the optional pull-up resistor networks RN21 thru RN24. The ports are quasi bi-directional which means they can be used as inputs or outputs. The MPU can read any pin by first setting the port to high, which engages a high impedance pull-up and then reading the pin.

These chips also provide an interrupt output that occurs on any input pin change. This feature is not used in the SBC488 firmware though it is wired up for future use.

The digital signals, SDA and SCL are manipulated by the MPU via the 8 bit addressable latch, U61 and are isolated by opto-isolator U47 (See Figures 5 and 9). The interrupt and read data are isolated by U48 opto-isolator. The read data is connected to 8 bit input port U71. The interrupt signal connects to /INT0 (U100-22) on the SmartCore.
3.6 Smart Core Features

The SmartCore(tm) features a Z180 CPU running at 9.216 Mhz. (Figure 10, SmartCore block diagram).

The SmartCore onboard memory consists of:

- a. SRAM, 32K (128K for -DEV option) Battery backup optional.
- b. EPROM, 32K (128K for -DEV option)
- c. EEPROM, 512 bytes.

Other SmartCore peripherals are:

- d. Real-Time Clock
- e. Reset and Power Failure Supervisor.
- f. Serial Interface for program development.
- g. Serial Interface for the application.
h. Two programmable timers.
   i. Two DMA channels.
   j. Memory decoding for external devices (/CS1-/CS6).

![SmartCore Block Diagram](image)

**Figure 10, SmartCore Block Diagram**

### 3.7 Power Supply Features

The SBC488 features an off line switching power supply. The power supply provides power to the computer interface circuitry and to the I/O sections. The schematic for the power supply is included as figure 11 below.

The power supply features:
- 2500 VAC isolation
- 3 outputs +5, +5 and +15 VDC
- EMI suppression
- Overload Protection
Figure 11, SBC488 Power Supply Diagram
4.0 Operating Instructions

4.1 Unpacking
Call the carrier immediately upon receipt if the shipping carton is damaged. Then inspect the SBC488 for damaged or loose components. If there is any damage contact your sales agent to obtain Return Material Authorization (RMA).

4.2 Setup and Configuration
The SBC488 is ready to operate without any calibration or software setup. To test the SBC488’s optional ADC and DAC functionality on the bench the analog inputs can be looped back to the analog outputs. This test procedure is described in 4.3 and 4.4 below.

4.3 Electrical Testing Using IEEE 488 Communications
The SBC488 can be tested using either the supplied SBCwin program (only compatible with IOtech’s IEEE 488 cards and drivers) or any IEEE 488 test facility such as IOtech’s windows based QuickTest program.

4.3.1 Equipment Required
a. IBM PC or compatible.
b. IEEE 488 adapter card and software.
c. Oscilloscope (optional).
d. Fluke 77 DVM or equal.
e. IEEE 488 cable.

4.3.2 Procedure for using the SBCwin program with IEEE 488 communications
a. Set the address dip switch on the SBC488 to the IEEE 488 address desired. See Section 2.7 for the address switch configuration.

b. Connect AD0 thru AD5 inputs to DA0 thru DA2 outputs to loop back analog signals if these options are installed (See figures 7 and 8).

c. Install or verify installation of IEEE 488 card and driver software. Refer to the IOtech manual.

d. Connect the IEEE 488 cable from the IBM PC compatible to the SBC488.

e. Apply power to the SBC488. The SBC488’s LEDs will cycle for a short time to indicate that it started properly.

f. Turn on the IBM PC compatible and load the IEEE 488 driver (DRVR488W.EXE) and the SBCwin program.

g. Using IOtech’s CONFIG.EXE program, create a logical instrument named SBC488 with the IEEE 488 address which was
previously set on the SBC488's address dip switch. Configure
the device for CR, LF and EOI Terminators.

h. Click on the SBCwin icon to start the SBCwin program.

i. Choose the IEEE 488 interface from the interface selection
pop up window. Then enter the address that is setup on the
SBC488’s dip switch in the address pop up window and click OK.
The main screen will appear and the communications light will be
green.

j. Push the DAC button for each of the installed DAC’s and enter
a value from 0 to 4095. Check the readback on each ADC channel
against the DAC setting. Measure the DAC output and compare it
against the calculated value using the formulas in Section 2.

k. Individually ground (P9 pin 34) each pin corresponding to
each optional digital I/O port (See Figure 9) and verify the
readback on the SBCwin screen.

l. Set and reset each bit of each installed digital I/O port
while observing the appropriate pin using a multimeter (see
figure 9).

4.3.3 Procedure for using IOtech Quicktest Program

a. Set the address dip switch on the SBC488 to the IEEE 488
address desired. See Section 2.7 for the address switch
configuration.

b. Connect AD0 thru AD5 inputs to DA0 thru DA2 outputs to loop
back analog signals if these options are installed (See figures
7 and 8).

c. Install or verify installation of IEEE 488 card and driver
software. Refer to the IOtech manual.

d. Connect the IEEE 488 cable from the IBM PC compatible to the
SBC488.

e. Apply power to the SBC488. The SBC488’s LEDs will cycle for
a short time to indicate that it started properly.

f. Turn on the IBM PC compatible and load the IEEE 488 driver
(DRVR488W.EXE) and IOtech’s QuickTest program.

g. Using IOtech’s CONFIG.EXE program, create a logical
instrument named SBC488 with the IEEE 488 address which was
previously set on the SBC488’s address dip switch. Configure
the device for CR, LF and EOI Terminators.

h. Type the command “*IDN?” in the output data window and then
click on the Send button. Note: type the text inside the
quotation marks.
i. Click on the Read button and observe the data returned. The data should read \textit{SBC488,REVxx} where \textit{xx} is the revision of the firmware.

j. If DAC0 is installed, enter the command \texttt{"DA0,4095"} and click the send button.

k. Read the voltage on DAC Port 0, (P3 pin 1) with respect to ground (P3-9) with the DVM. The voltage should read full scale, i.e. DAC\_VREF*4095/4096.

l. If the ADC option is installed, enter the command \texttt{AD?0} and click the Send button.

m. Click the Read button and observe the data returned. The data should read \texttt{xxxx} where the actual voltage is \texttt{xxxx*DAC\_VREF/4096}. \texttt{xxxx} can range from 0 to 4095. This corresponds to 0 to ADC\_VREF VDC on the ADC converter.

n. Repeat steps j. thru m. for the other six ADC and DAC channels using the following commands:

   - \texttt{"DA1,4095"} and \texttt{"AD?1"}
   - \texttt{"DA2,4095"} and \texttt{"AD?2"}
   - \texttt{"DA3,4095"} and \texttt{"AD?3"}
   - \texttt{"DA4,4095"} and \texttt{"AD?4"}
   - \texttt{"DA5,4095"} and \texttt{"AD?5"}

   Note: Use only on channels with the optional DAC’s installed.

Follow steps p-w below only for installed digital I/O ports.

o. Enter the command \texttt{"D0,128"} and click the Send button.

p. Read the voltages on I/O Port 0, P9 pins 2 thru 9 with respect to P9-34 (Gnd). The voltages should read as follows.

\begin{tabular}{cccccccc}
Pin 9 & Pin 8 & Pin 7 & Pin 6 & Pin 5 & Pin 4 & Pin 3 & Pin 2 \\
3-5 VDC & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{tabular}

q. Enter the command \texttt{"D?"} and click the Send button.

r. The SBC488 will respond with \texttt{www,xxx,yyy,zzz} where these are the decimal representations (0-255) of each 8 bit port value.

s. Enter the command \texttt{"D0,64"} and click the Send button.

t. Read the voltages on I/O Port 0, P9 pins 2 thru 9 with respect to P9-10 (Gnd). The voltages should read as follows.

\begin{tabular}{cccccccc}
Pin 9 & Pin 8 & Pin 7 & Pin 6 & Pin 5 & Pin 4 & Pin 3 & Pin 2 \\
0 & 3-5 VDC & 0 & 0 & 0 & 0 & 0 & 0
\end{tabular}
w. Repeat steps o. thru t. for Port 1 “D1,xxx” thru Port 3 “D3,xxx”.

4.4 Electrical Testing Using RS-232 Communications
The SBC488 can be bench tested using either the supplied SBCwin program or any Terminal program such as the one included with Microsoft Windows (tm).

4.4.1 Equipment Required
   a. IBM PC or compatible with Serial port.
   b. Oscilloscope (optional).
   c. Fluke 77 DVM or equal.
   d. RS-232 Cable 9 pin (SBC488 side) to computer serial port.

4.4.2 Procedure for using SBCwin program with RS-232 communications
   a. Set the address dip switch on the SBC488 to 31. See Section 2.7 for the address switch configuration.
   b. Connect A0 thru AD5 inputs to DA0 thru DA2 outputs to loop back analog signals if these options are installed (See figures 7 and 8).
   c. Install or verify installation of serial card and driver software.
   d. Connect the RS-232 cable from the IBM PC compatible to the SBC488.
   e. Apply power to the SBC488. The SBC488’s LEDs will cycle for a short time to indicate that it started properly.
   f. Turn on the IBM PC compatible and load the SBCwin program.
   h. Click on the SBCwin icon to start the SBCwin program.
   i. Choose the RS-232 interface from the interface selection pop up window. Then enter the port that the SBC488 is connected to in the address pop up window and click OK. The main screen will appear and the communications light will be green.
   j. Push the DAC button for each of the installed DAC’s and enter a value from 0 to 4095. Check the readback on each ADC channel against the DAC setting. Measure the DAC output and compare it against the calculated value using the formulas in Section 2.
   k. Individually ground (P9 pin 34) each pin corresponding to each optional digital I/O port (See Figure 9) and verify the readback on the SBCwin screen.
   l. Set and reset each bit of each installed digital I/O port while observing the appropriate pin using a multimeter (see figure 9).
4.4.3 Procedure for using the Terminal Program

a. Set the SBC488 address dip switch to 31. See Section 2.7 for the address switch configuration.

b. Connect AD0 thru AD5 inputs to DA0 thru DA2 outputs to loop back analog signals if these options are installed (See figures 7 and 8).

c. Install or check installation of serial card.

d. Connect the RS-232 cable to the SBC488.

e. Apply power to the SBC488. The SBC488’s LEDs will cycle for a short time to indicate that it started properly.

f. Turn on the IBM compatible PC.

g. Run the terminal setup program and configure the port to 9600 baud, no Parity, 8 bit data and 1 stop bit.

h. Type the command “*IDN?” in the terminal area and then press return.

i. The SBC488 should return SBC488,REVxx where xx is the revision of the firmware.

j. If the DAC0 option is installed, enter the command “DA0,4095” and press return.

k. Read the voltage on DAC Port 0, P3 pin 1 with respect to P3-34 with the DVM. The voltage should read full scale.

l. If the ADC option is installed, enter the command “AD?0” and press return.

m. The SBC488 should return xxxx where the output voltage is xxxx*FS_ADC/4096. xxxx can range from 0 to 4095 corresponding to 0 to full scale on the ADC.

n. Repeat steps i. thru m. for the other five optional ADC and DAC channels using the commands:

   “DA1,4095” and “AD?1”
   “DA2,4095” and “AD?2”
   “DA3,4095” and “AD?3”
   “DA4,4095” and “AD?4”
   “DA5,4095” and “AD?5”

o. If the Digital I/O option is installed, enter the command “D0,128” and press return.

Follow steps p-w below only for installed digital I/O ports.
p. Read the voltages on I/O Port 0, P9 pins 2 thru 9 with respect to P9-34 (Gnd). The voltages should read as follows.

<table>
<thead>
<tr>
<th>Pin 9</th>
<th>Pin 8</th>
<th>Pin 7</th>
<th>Pin 6</th>
<th>Pin 5</th>
<th>Pin 4</th>
<th>Pin 3</th>
<th>Pin 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-5 VDC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

q. Enter the command “D?” and press return.

r. The SBC488 will respond with www,xxx,yyy,zzz where these are the decimal representations (0-255) of each 8 bit port value.

s. Enter the command “D0,64” and press return.

t. Read the voltages on I/O Port 0, P9 pins 2 thru 9 with respect to P9-10 (Gnd). The voltages should read as follows.

<table>
<thead>
<tr>
<th>Pin 9</th>
<th>Pin 8</th>
<th>Pin 7</th>
<th>Pin 6</th>
<th>Pin 5</th>
<th>Pin 4</th>
<th>Pin 3</th>
<th>Pin 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3-5 VDC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

w. Repeat steps o. thru t. for Port 1 “D1,xxx” thru Port 3 “D3,xxx”.
5.0 Programming

5.1 Device Specific Commands

The SBC488 is addressable via the IEEE 488 interface and the RS-232 interface. The SBC488 assumes IEEE 488 communications if the address switch is set in the range of 0 to 30 (valid IEEE 488 address) and RS-232 communication for a setting of 31. The device specific commands are implemented for both interfaces. The IEEE 488 commands listed in Section 5.2 are only available over the IEEE 488 interface (with the exception of *IDN?). Example programs use IOtech’s DRIVER488/DRV driver syntax but should be helpful as a guide for other drivers.

The following are the device specific commands:

a. Set Digital to Analog Converter (DAC)

\texttt{DAy,xxxx} \ where \ xxxx \ is \ desired \ output \ voltage \ multiplied \ by \ \texttt{FS\_Volts*xxxx/4096}. \ xxxx \ can \ range \ from \ 0 \ to \ 4095 \ corresponding \ to \ 0 \ to \ \texttt{DAC\_VREF \ VDC*4095/4096} \ on \ the \ DAC. \ y \ is \ the \ DAC \ channel \ and \ is \ valid \ from \ 0 \ to \ 5, \ assuming \ that \ 5 \ DAC’s \ are \ installed. \ Response: \ None

QuickBasic Example using DAC set commands

\begin{verbatim}
'QuickBasic Example using DAC set commands
OPEN "\DEV\IEEEOUT" FOR OUTPUT AS #1
OPEN "\DEV\IEEEIN" FOR OUTPUT AS #2
'ASSUME SBC488 IS AT ADDRESS 9
'ASSUME DAC VREF AND ADC VREF ARE BOTH 2.5 VDC
FS_VOLTS=16 'FULL SCALE VOLTAGE IS 16 VDC
FS_AMPS=320 'FULL SCALE CURRENT IS 320 ADC
V_SET=5.5 'VOLTAGE SETTING IS 5.5 VDC
I_SET=100 'CURRENT SETTING IS 100 AMPS
OV_SET=6.5 'SET OVERVOLTAGE TO 6.5 VOLTS
PRINT#1, "OUTPUT9;DA0,"+RIGHT$(STR$(V_SET*8192/(2.5*FS_VOLTS)),4)
PRINT#1, "OUTPUT9;DA1,"+RIGHT$(STR$(I_SET*8192/(2.5*FS_AMPS)),4)
PRINT#1, "OUTPUT9;DA2,"+RIGHT$(STR$(OV_SET*8192/(2.5*FS_VOLTS)),4)
\end{verbatim}

b. Read Analog to Digital Converter (ADC)

\texttt{AD?y} \ Response: \ returns \ xxxx. \ Actual \ output \ voltage \ is \ xxxx*FS\_Volts/4096. \ xxxx \ can \ range \ from \ 0 \ to \ 4095 \ corresponding \ to \ F.S. \ on \ ADC. \ y \ can \ be \ 0 \ to \ 10. \ This \ assumes \ that \ the \ ADC \ option \ is \ installed.

c. Read Digital Input/Output (I/O)

\texttt{D?} \ Response: \ returns \ www,xxx,yyy,zzz \ where \ these \ are \ the \ decimal \ representations \ (0-255) \ of \ each \ 8 \ bit \ port.
d. Write Digital I/O

\textbf{Dy,xxx} where \textit{xxx} is a decimal number representing the binary weighted value of the port and \textit{y} is the port number. \textit{y} is valid only from 0 to 3 and only when the corresponding optional port is installed.

Response: None

QuickBasic program using Digital I/O

```
'QuickBasic Example using digital I/O

OPEN "\DEV\IEEEOUT" FOR OUTPUT AS #1
OPEN "\DEV\IEEEIN" FOR OUTPUT AS #2
'ASSUME SBC488 IS AT ADDRESS 9
'D?1 COMMAND POSITIONS
START=1
STOP=2
REM_SHUTDOWN=4
OV_RESET=8
'D?0 COMMAND BIT POSITIONS
C_MODE=1 'current mode
V_MODE=2 'volt mode
SS_MODE=4 'soft-start mode
OT=8 'thermal fault
PH_BAL=16 'phase balance fault
PGM_LINE=32 'program line fault = 32
OV=64 'over-voltage fault
STDBY=128 'standby mode

PRINT#1, "OUTPUT9;D1,"+STR$(START) 'HIT START
PRINT#1, "OUTPUT9;D1,0" 'RELEASE START
PRINT#1, "OUTPUT9;D1,"+STR$(STOP) 'HIT START
PRINT#1, "OUTPUT9;D1,0" 'RELEASE STOP
PRINT#1, "OUTPUT9,D?,0"
INPUT #2, R$
IF VAL(R$) AND C_MODE THEN PRINT "CURRENT MODE"
IF VAL(R$) AND V_MODE THEN PRINT "VOLTAGE MODE"
IF VAL(R$) AND SS_MODE THEN PRINT "SOFT START MODE"
IF VAL(R$) AND OT THEN PRINT "OVER TEMPERATURE FAULT"
IF VAL(R$) AND PH_BAL THEN PRINT "PHASE BALANCE FAULT"
IF VAL(R$) AND PGM_LINE THEN PRINT "PROGRAM LINE FAULT"
IF VAL(R$) AND OV THEN PRINT "OVER VOLTAGE FAULT"
IF VAL(R$) AND STDBY THEN PRINT "STANDBY MODE"
```

5.2 IEEE 488 Event Processing

All of the device specific commands in the previous section are initiated by the IEEE 488 bus controller. The SBC488 however, like all IEEE 488 compatible device, can also initiate activity by generating a service request. This is done by activating the SRQ line. To accommodate this feature, the bus controller must be programmed to handle service request, usually in an interrupt handling sub-routine.

The IEEE 488 standard defines a method for status reporting. This method uses the IEEE 488.1 status byte (See Figure 13). Three bits of this byte are defined as follows:

- \texttt{RQS} => Service Request bit
- \texttt{ESB} => Event Status bit
MAV => Message Available bit

The RQS bit is set when the SBC488 has requested service via the SRQ control line.

The Event Status Bit (ESB) is set when one of the events defined in the standard Event Status Register (ESR) has occurred. The ESR is masked by the Event Status Enable register (ESE) so that the user can mask specific or all events from causing an ESB and subsequent SRQ. (See Figures 12 and 13)

The Message Available bit (MAV) is set when a message is available in the SBC488 output buffer.

Figure 12, ESE and ESR Generation
5.3 IEEE 488 Standard Commands

a. Read Event Status Register (See Figure 12)

*ESR?
Response: Decimal number weighted as follows:

1 => Operation Complete  
2 => not used (Request Control)  
4 => not used (Query Error)  
8 => Device Dependent Error  
16 => Execution Error  
32 => Command Error  
64 => not used (User Request)  
128 => Power on, 1 after power on

Note:
1. Power on bit is set every time the SBC488 is reset. It can be used to detect a power outage or SBC488 reset.
2. All bits are cleared after the *ESR? command is executed.

b. Read Event Status Enable mask (See Figure 12)

*ESE?
Response: Reads the *ESE mask. The mask is bitwise anded with ESR and enables ESR generation. This command uses the same format as *ESR? above.
c. Set Event Status Enable mask (See Figure 12)

*ESE,xxx
Response: Sets the *ESE mask. Mask bitwise enables ESR generation. Same format as *ESR? above.

d. Read Status byte (See Figure 12)

*STB?
Response: Decimal number weighted as follows:

1 => N/A
2 => N/A
4 => N/A
8 => N/A
16 => MAV, Message Available
32 => ESB, Event Status Bit
64 => RQS, Request Service
128 => N/A

e. Read Status Register Enable mask (See Figure 12)

*SRE?
Response: xxx Mask bitwise enables ESR generation. Decimal number weighted as follows

1 => N/A
2 => N/A
4 => N/A
8 => N/A
16 => MAV, Message Available
32 => ESB, Event Status Bit
64 => N/A
128 => N/A

f. Set Status Register Enable mask (See Figure 12)

*SRE,xxx
The same as format as *SRE? above.

g. Read SBC Part and serial number

*IDN?
Response: SBC488,REVxx where xx is the revision of the firmware.
5.4 Program Example using IEEE 488 Standard Commands

QuickBasic Program Example using IEEE 488 Standard Commands

`QuickBasic Example using digital I/O`

OPEN "\DEV\IEEEOUT" FOR OUTPUT AS #1
OPEN "\DEV\IEEEIN" FOR OUTPUT AS #2
`ASSUME SBC488 IS AT ADDRESS 9`
`SRE BIT POSITIONS`
MAV=16 'MAV, Message Available
ESB=32 'ESB, Event Status Bit
`ESE BIT POSITIONS`
DDE=8 'Device Dependent Error
EE=16 'Execution Error
COMERR=32 'Command Error
PON=128 'Power on, 1 after power on
`GET SBC488 VERSION NUMBER`
PRINT#1, "OUTPUT9,*IDN?"
INPUT #2, R$
PRINT "FIRMWARE VERSION NUMBER "+R$
`ENABLE MAV AND ESB IN SRE AND READ BACK`
PRINT#1, "OUTPUT9,*SRE,"+STR$(MAV+ESB)
PRINT#1, "OUTPUT9,*SRE?"
INPUT #2, R$
PRINT "SERVICE REQUEST ENABLE REGISTER IS SET TO"+R$
`ENABLE PON AND COMERR IN SRE AND READ BACK`
`POWER ON AND COMMAND ERRORS WILL PRODUCE SERVICE REQUESTS`
PRINT#1, "OUTPUT9,*ESE,"+STR$(PON+COMER)
PRINT#1, "OUTPUT9,*ESE?"
INPUT #2, R$
PRINT "EVENT STATUS ENABLE REGISTER IS SET TO"+R$

6.0 TROUBLESHOOTING

<table>
<thead>
<tr>
<th>FAULT CONDITION</th>
<th>PROBABLE CAUSE</th>
<th>CHECK, REMEDY</th>
</tr>
</thead>
<tbody>
<tr>
<td>No IEEE 488 Communications</td>
<td>Address errors</td>
<td>SBC488 Dip switch.</td>
</tr>
<tr>
<td>Note 3</td>
<td></td>
<td>Valid addresses are 0 thru 30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Control program address</td>
</tr>
<tr>
<td></td>
<td>Termination</td>
<td>Check control program termination, CR, LF and EOI</td>
</tr>
<tr>
<td>IEEE 488 Cable</td>
<td>Replace Cable</td>
<td></td>
</tr>
<tr>
<td>SBC488 Power Supplies</td>
<td>Check power supplies on SBC488 (U7)</td>
<td></td>
</tr>
<tr>
<td>SmartCore Failure</td>
<td>Check data bus for activity</td>
<td></td>
</tr>
<tr>
<td>No IEEE 488 Communications (Continued)</td>
<td>7210 Chip failure</td>
<td>Check Interrupt line out of 7210 during transmission.</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>No RS-232 Communications</td>
<td>Address errors</td>
<td>Check Interrupt line out of 7210 during transmission.</td>
</tr>
<tr>
<td>No RS-232 Communications</td>
<td>Address errors</td>
<td>SBC488 Dip switch. Valid address is 31.</td>
</tr>
<tr>
<td></td>
<td>Note 3</td>
<td>Terminal program set for 9600,N,8,1</td>
</tr>
<tr>
<td>RS-232 Cable</td>
<td>Format error</td>
<td>Replace Cable</td>
</tr>
<tr>
<td>SBC488 Power Supplies</td>
<td>Format error</td>
<td>Check power supplies on SBC488 (U7).</td>
</tr>
<tr>
<td>SmartCore Failure</td>
<td>RS-232 transceiver Chip failure</td>
<td>Check data bus for activity</td>
</tr>
<tr>
<td></td>
<td>RS-232 transceiver Chip failure</td>
<td>Check U85 RX and TX lines during transmission</td>
</tr>
<tr>
<td>DAC out voltage error</td>
<td>DAC power supply</td>
<td>Check DAC power supply voltage (U1 and U2)</td>
</tr>
<tr>
<td>ADC Readback error</td>
<td>DAC power supply</td>
<td>Check DAC power supply voltage (U1 and U2)</td>
</tr>
<tr>
<td>ADC Readback error</td>
<td>DAC power supply</td>
<td>Check ADC power supply voltage (U5 and U3)</td>
</tr>
<tr>
<td>ADC Readback error</td>
<td>Opto-Isolator failure</td>
<td>Check U41 and U42 outputs while commanding voltage change.</td>
</tr>
<tr>
<td>note 1</td>
<td>Opto-Isolator failure</td>
<td>Check U43, U46 and U45 outputs while reading voltage over bus (AD?0).</td>
</tr>
<tr>
<td>ADC Readback error</td>
<td>Voltage reference</td>
<td>Check U22 for 2.50 VDC</td>
</tr>
<tr>
<td>note 1</td>
<td>ADC IC failure</td>
<td>Replace U90</td>
</tr>
<tr>
<td>Digital Readback error</td>
<td>Control software</td>
<td>Check data for errors</td>
</tr>
<tr>
<td>Digital Readback error</td>
<td>Control software</td>
<td>Check data for errors</td>
</tr>
<tr>
<td>Digital Readback error</td>
<td>Digital power supply</td>
<td>Check power supply voltage (U6)</td>
</tr>
<tr>
<td>Digital Readback error</td>
<td>Digital power supply</td>
<td>Check power supply voltage (U6)</td>
</tr>
<tr>
<td>Digital Readback error</td>
<td>Opto-Isolator failure</td>
<td>Check U48 and U47 outputs while reading digital over bus (D?0).</td>
</tr>
<tr>
<td>Digital Readback error</td>
<td>Opto-Isolator failure</td>
<td>Check U48 and U47 outputs while reading digital over bus (D?0).</td>
</tr>
<tr>
<td>Digital Readback error</td>
<td>Control software</td>
<td>Check data for errors</td>
</tr>
<tr>
<td>Digital set error</td>
<td>Control software</td>
<td>Check data for errors</td>
</tr>
<tr>
<td>Digital set error</td>
<td>Port IC failure</td>
<td>Replace port IC U51, U52 or U53</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>Voltage levels</td>
<td>Verify input voltage levels are valid (See section 2)</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>Digital power supply</td>
<td>Check power supply voltage (U6)</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>Digital power supply</td>
<td>Check power supply voltage (U6)</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>Opto-Isolator failure</td>
<td>Check U48 and U47 outputs while reading digital over bus (D?0).</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>Opto-Isolator failure</td>
<td>Check U48 and U47 outputs while reading digital over bus (D?0).</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>Control software</td>
<td>Check data for errors</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>Port IC failure</td>
<td>Replace port IC U51, U52 or U53</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>load current</td>
<td>Verify load current levels are valid (See</td>
</tr>
<tr>
<td>Digital set error (Cont’d)</td>
<td>note 2</td>
<td>Section 2)</td>
</tr>
</tbody>
</table>
Notes:

1. ADC input voltages outside of specified input range may cause the IC to latch and fail.

2. Digital output ports may require a pull up resistor for certain loads to increase the source current available. RN21 thru RN24 are optional pull-up resistors that are socketed for easy installation.

3. Always cycle SBC488 power after changing dip switch setting.
7.0 DRAWINGS

7.1 SBC488 Parts List

The SBC488 Parts List is on following page.

7.2 Drawings

The SBC488 overall schematic, Figure 14 follows. The SBC488 component layout is on Figure 15.
<table>
<thead>
<tr>
<th>ITEM</th>
<th>REV QTY</th>
<th>REF DES</th>
<th>VALUE</th>
<th>VOLT</th>
<th>MPE P/N</th>
<th>DESCRIPTION</th>
<th>MFG PART NUMBER</th>
<th>MANUFACTURER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>BT1</td>
<td>CR2032</td>
<td>3V</td>
<td>LITHIUM BATTERY, 20 MM</td>
<td>CR2032</td>
<td>CR2032</td>
<td>PANASONIC</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>C1</td>
<td>100</td>
<td>35V</td>
<td></td>
<td></td>
<td>ECE-AIVSP101 (P1231)</td>
<td>PANASONIC</td>
</tr>
<tr>
<td>3 B</td>
<td>25</td>
<td>C11,C13-C33,C38-C40</td>
<td>0.1</td>
<td>50V</td>
<td></td>
<td></td>
<td>11146</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>C41-C43</td>
<td>330</td>
<td>16V</td>
<td></td>
<td></td>
<td>ECE-A1CFS331 (P1213)</td>
<td>PANASONIC</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>C51</td>
<td>0.1</td>
<td>1000V</td>
<td></td>
<td></td>
<td>ECQ-E2A104MW</td>
<td>PANASONIC</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>C52-C53</td>
<td>0.01</td>
<td>1000V</td>
<td></td>
<td></td>
<td>ECQ-U2A103MM</td>
<td>PANASONIC</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>C61</td>
<td>47</td>
<td>25V</td>
<td></td>
<td></td>
<td>12216</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>C81</td>
<td>33</td>
<td>250V</td>
<td></td>
<td></td>
<td>ECE-A2E6330</td>
<td>PANASONIC</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>C91</td>
<td>220 PF</td>
<td>1000V</td>
<td></td>
<td></td>
<td>10843</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>CN1-CN2</td>
<td>0.1</td>
<td>50V</td>
<td>CAPACITOR NETWORK, 8 PIN SIP</td>
<td>201CH7X7R104R5</td>
<td>DALE</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>D1</td>
<td>1N5230B</td>
<td>4.7V</td>
<td></td>
<td></td>
<td>10990</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>D21</td>
<td>1N4148</td>
<td></td>
<td></td>
<td></td>
<td>10978</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>D41-D42</td>
<td>1N5819</td>
<td></td>
<td></td>
<td></td>
<td>11291</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>D61</td>
<td>HER102</td>
<td></td>
<td></td>
<td></td>
<td>11125</td>
<td></td>
</tr>
<tr>
<td>15 B</td>
<td>6</td>
<td>DB1-DB6</td>
<td>GP06M</td>
<td></td>
<td></td>
<td></td>
<td>11144</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>L1</td>
<td>47 MM</td>
<td></td>
<td>COMMON MODE CHOKE</td>
<td></td>
<td>104502</td>
<td>MAGNAPOWER</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>L2</td>
<td>27U</td>
<td></td>
<td></td>
<td></td>
<td>5800-270</td>
<td>J.W. MILLER</td>
</tr>
<tr>
<td>18</td>
<td>5</td>
<td>LED0-LED4</td>
<td>L20321</td>
<td></td>
<td>LED, RED, RT ANGLE</td>
<td></td>
<td>5600F1</td>
<td>INDUSTRIAL DEVICES INC</td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>P1</td>
<td>553811-1</td>
<td></td>
<td>CONNECTOR, 24 PIN, IEEE 488</td>
<td></td>
<td>AMP</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>P13</td>
<td>929834-01</td>
<td></td>
<td>CONNECTOR, HEADER, 10 POS, SIL</td>
<td>3M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td>P14</td>
<td>929836</td>
<td>11229</td>
<td>CONNECTOR, 10 PIN HEADER, DIL</td>
<td>3 M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>2</td>
<td>P22/P23</td>
<td>929836</td>
<td>11229</td>
<td>CONNECTOR, 34 PIN HEADER, DIL</td>
<td>3 M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>1</td>
<td>P3</td>
<td>929836</td>
<td>11229</td>
<td>CONNECTOR, 24 PIN HEADER, DIL</td>
<td>3 M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>P4</td>
<td>P09S-21</td>
<td>p09s-21</td>
<td>CONNECTOR, 9 PIN D-SUB, RT ANGLE</td>
<td>DK: 309P-ND OR A158</td>
<td>NORCOMP OR AMP (DUAL)</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>P6</td>
<td>26-48-103</td>
<td></td>
<td>3 POSITION CONNECTOR, .156&quot;</td>
<td></td>
<td>26-48-1035</td>
<td>MOLEX</td>
</tr>
<tr>
<td>26</td>
<td>1</td>
<td>P9</td>
<td>929836</td>
<td>11229</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>1</td>
<td>Q2</td>
<td>1N2222A</td>
<td></td>
<td></td>
<td></td>
<td>11200</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>1</td>
<td>R1</td>
<td>2.2K</td>
<td></td>
<td></td>
<td></td>
<td>10040</td>
<td></td>
</tr>
<tr>
<td>29 B</td>
<td>2</td>
<td>R12-R13</td>
<td>10.0K</td>
<td>1%</td>
<td></td>
<td></td>
<td>10356</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>6</td>
<td>R14-R19</td>
<td>10K</td>
<td></td>
<td></td>
<td></td>
<td>10048</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>5</td>
<td>R21-R25</td>
<td>1K</td>
<td></td>
<td></td>
<td></td>
<td>10036</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>2</td>
<td>R30-R31</td>
<td>56K</td>
<td></td>
<td></td>
<td></td>
<td>10057</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>2</td>
<td>R41-R42</td>
<td>680</td>
<td></td>
<td></td>
<td></td>
<td>10034</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>R51</td>
<td>220</td>
<td></td>
<td></td>
<td></td>
<td>10028</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>1</td>
<td>R71</td>
<td>6.8</td>
<td></td>
<td></td>
<td></td>
<td>10010</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>1</td>
<td>R81</td>
<td>56</td>
<td></td>
<td></td>
<td></td>
<td>10033</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>1</td>
<td>R91</td>
<td>1K</td>
<td>1/2W</td>
<td></td>
<td></td>
<td>10125</td>
<td></td>
</tr>
</tbody>
</table>
Figure 14, SBC488 Schematic Diagram
Figure 15, SBC488 Component Layout