Stellar Instruments

SC1 Controller & Display System

General User's Manual

Contents

Section 1 – General

The SC1 controller and display is a two part system designed for general OEM control applications such as process machine control and commercial appliance control. The system is comprised of a micro-computer based control board that is factory firmware programmed to perform user required logic, timing and general control functions using both analog and digital inputs. This board, referred to as the base unit, can be one of several stock hardware configurations available from Stellar or can be a custom configuration to suit an OEM's particular requirements. **The base unit may be used stand-alone or with the available display unit.**

The display unit is an intelligent one-quarter VGA (320x240) touch screen display that serves as the interface between the user and the control board in applications where operator interaction with the control system is required. A unique feature of the display is its ability to be custom configured by the user to provide the buttons, readouts, pictures, etc., appropriate to the application.

The next several sections will describe some of the available units in detail.

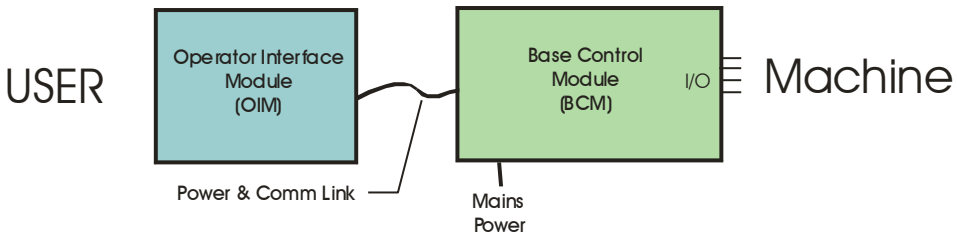
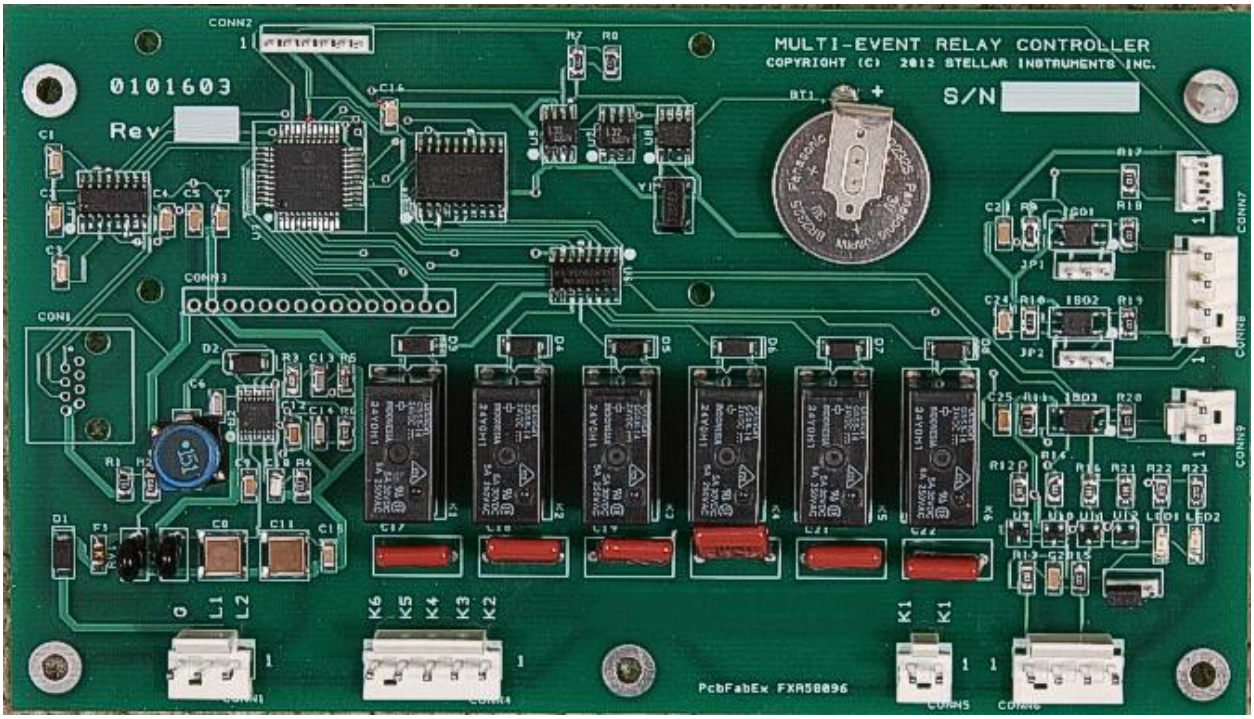


Figure 1) Basic System Block Diagram

Section 2 – Base Unit A (Multi-Event Programmable Timer)



Key Features

- Seven Independently Programmable Relay Outputs
- Real Time Clock – Timing is Programmable by Time of Day.
- User Programmed Time Sequences
- 24VAC Power
- Timing is Easily Programmed Using Standard PC and Host Programming Software or Available Display Unit
- Standard Seven Day Time Cycle
- Micro-Controller Based --- Programmed to Customer Applications for OEMs

GENERAL DESCRIPTION

The Multi-Output Event Timer is a user configured, real time clock based, timer that controls seven relay contact outputs. The standard configuration of the event timer allows each of the seven outputs to be cycled on and off over a time period in virtually any timing arrangement. Cycle granularity is normally one

minute (this is the minimum on or off time) but may be factory reprogrammed to smaller time increments if required.

Timing cycles are easily programmed using a standard PC computer with an RS232 serial port and the user-friendly graphical interface host software. Cycle programming may also be accomplished with an available display unit.

The event timer requires 24vac power, and the six relay outputs are configured to switch the 24volt power into the customer's loads. The relay contacts can handle up to three amps ac.

Two transistor outputs are provided for operating indicator lamps.

Two optically isolated inputs are also provided as operator inputs for issuing special commands to the event timer such as manual override or cycle disable. These inputs are designed to accept 24vac or dc and may be used with external pushbuttons, key-switches or similar devices.

A 0-5Vdc analog input is provided to accept an array of analog sensors.

The event timer is nominally supplied as an open printed circuit board with 0.156" header style connectors, but can be factory packaged and terminated to suit specific customer requirements.

SPECIFICATIONS

POWER: 24VAC +/- 20% @ 55mA (Plus Output Draw on 24VAC Relay Outputs)

DIGITAL INPUTS: Three Optically Isolated Nominal 24V AC or DC Actuation

ANALOG INPUT: 0-4Vdc Range

OUTPUTS: Seven Relay Contact Outputs – 24VAC Activated Upon Contact Closure --- 3 Amp Rating. Five Contact Outputs Configured as 24VAC Powered (Form A) Contacts and One Contact Output Configured as Dry Form A

Two Transistor Outputs – Capable of Sourcing or Sinking 10mA per output

SERIAL PROGRAM INTERFACE: RS232

ENVIRONMENTAL: -20 to +60 °C 5-95% RH non-condensing

SIZE: 7” x 4” x 1.2” (L x W x H)

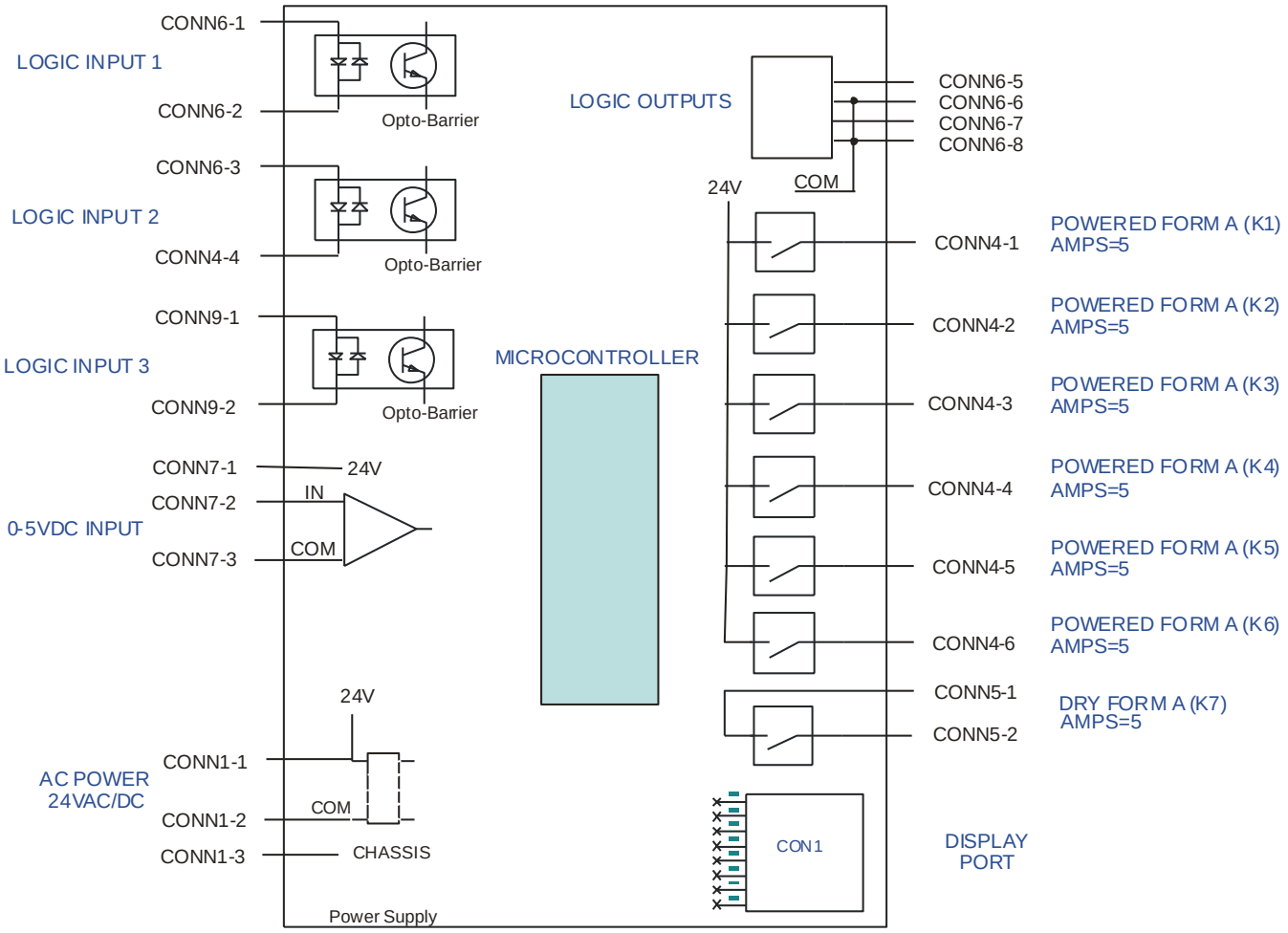


Figure 2.1 Base Unit A Pin-Out and Block Diagram

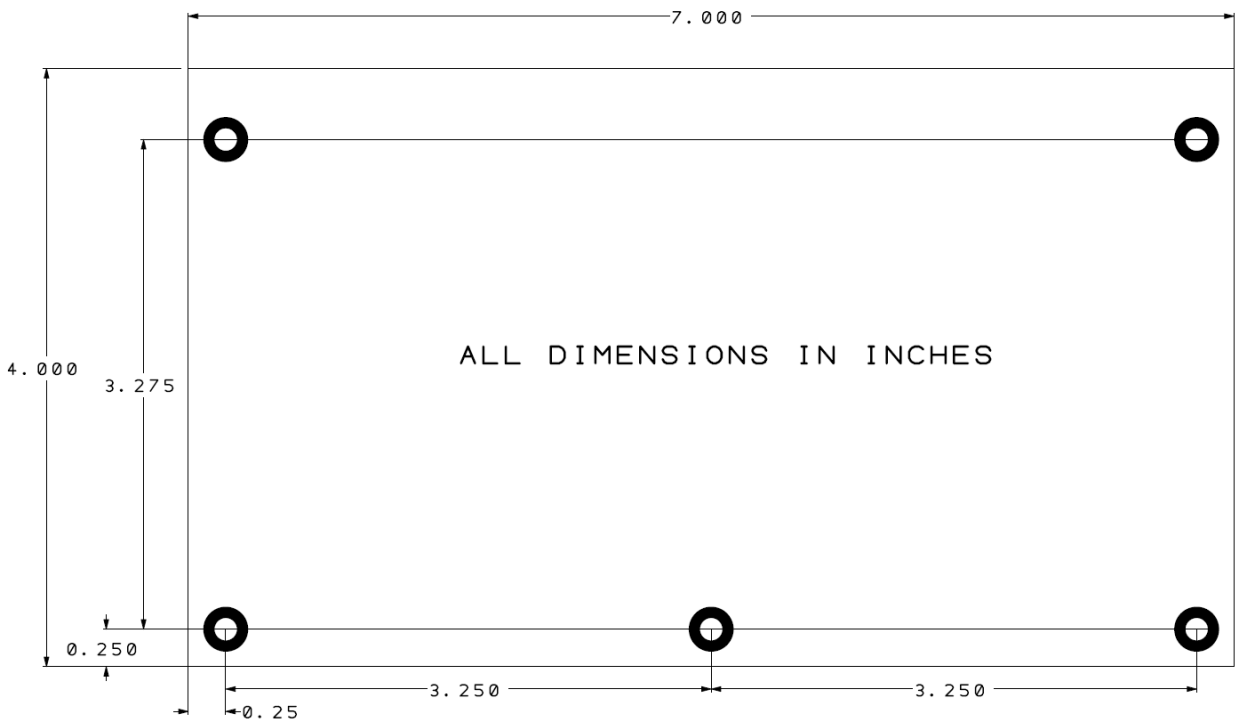


Figure 2.2 Base Unit A Mechanical Mounting

Section 3 – Base Unit B (Logic Controller with Analog Inputs)

Key Features

- Four Independently Programmable Relay Outputs
- Three Optically Isolated Logic Inputs
- Five “Pull Down” Logic Outputs
- Three Analog Inputs (Configured for 1000 ohm RTDs)
- Real Time Clock Available
- 120/240VAC Power
- User Settings Easily Programmed Using Standard PC and Host Programming Software or Available Display Unit
- Pulse Width Modulated (PWM) Logic Level Output
- Micro-Controller Based --- Programmed to Special Customer Applications for OEMs

3.1 General Description

The BCM hardware is contained on a single PC board and is designed to mount internally within the sealing machine that is physically, electrically and environmentally protected. The board will be X.X” x X.X” (TBD) and will mount via a series of peripheral holes and standoffs.

3.2 POWER INPUT – AC power to the base unit is line voltage and is 120/208/240vac +/-10%. BCM AC power Input voltage is configured via pluggable jumpers.

3.3 Relay Outputs – There are four relay outputs and they are all dry contacts with the following ratings:

- Contact #1 – 1 Form A Contact – 30amp resistive, 2HP@ 120vac/2HP@240vac. This contact controls the electro-magnet.
- Contact #2 – 1 Form A Contact – 5amp resistive, 1/8HP@ 120/240vac
- Contact #3 – 1 Form A Contact – 5amp resistive, 1/8HP@ 120/240vac. This is a 1/20th HP DC motor.
- Contact #4 – 1 Form A Contact – 5amp resistive, 1/4HP@ 120/240vac. This can be up to a ¼ HP motor.

All relay contacts are "snubbed" with a line rated capacitor to enhance contact life.

3.4 Open Collector Outputs – There are four open collector outputs, each with a current sinking capability of 50ma. A 5vdc voltage supply pin will be provided in conjunction with each open collector output to facilitate operation of external solid state relays or contactors. Commutating diodes will be used in conjunction with the outputs to protect circuitry from inductive transients created by the external relays.

3.5 Analog PWM Output – There is one PWM output. It is hardware configured as an open collector output. The frequency and modulation characteristics are configured to meet the needs of a particular application.

3.6 Logic Inputs – There are two optically isolated inputs that will accept a 5-15vdc level. These inputs can be factory jumpered to be externally activated from a 5-15vdc source or be "pulled down" to ground by an external contact. A 5vdc voltage supply pin will be provided with each logic input for use as a logic activation source.

3.7 RTD Inputs – There are three analog inputs, each configured to accommodate two-wire, 1000ohm platinum RTDs. Each input will have a temperature range from 32-500degF. The electronic accuracy will be within +/- 5degF (does not include RTD sensor). The analog inputs are twelve bit resolution and should allow for +/- 1degF resolution.

3.8 Communication/Power Port – An RJ45 connector will be provided to link the BCM to the OIM; this connector will carry 12vdc power, ground and RS232 communication between the system modules.

Figure 3.1 is a combination block and pinout diagram for the base unit B.

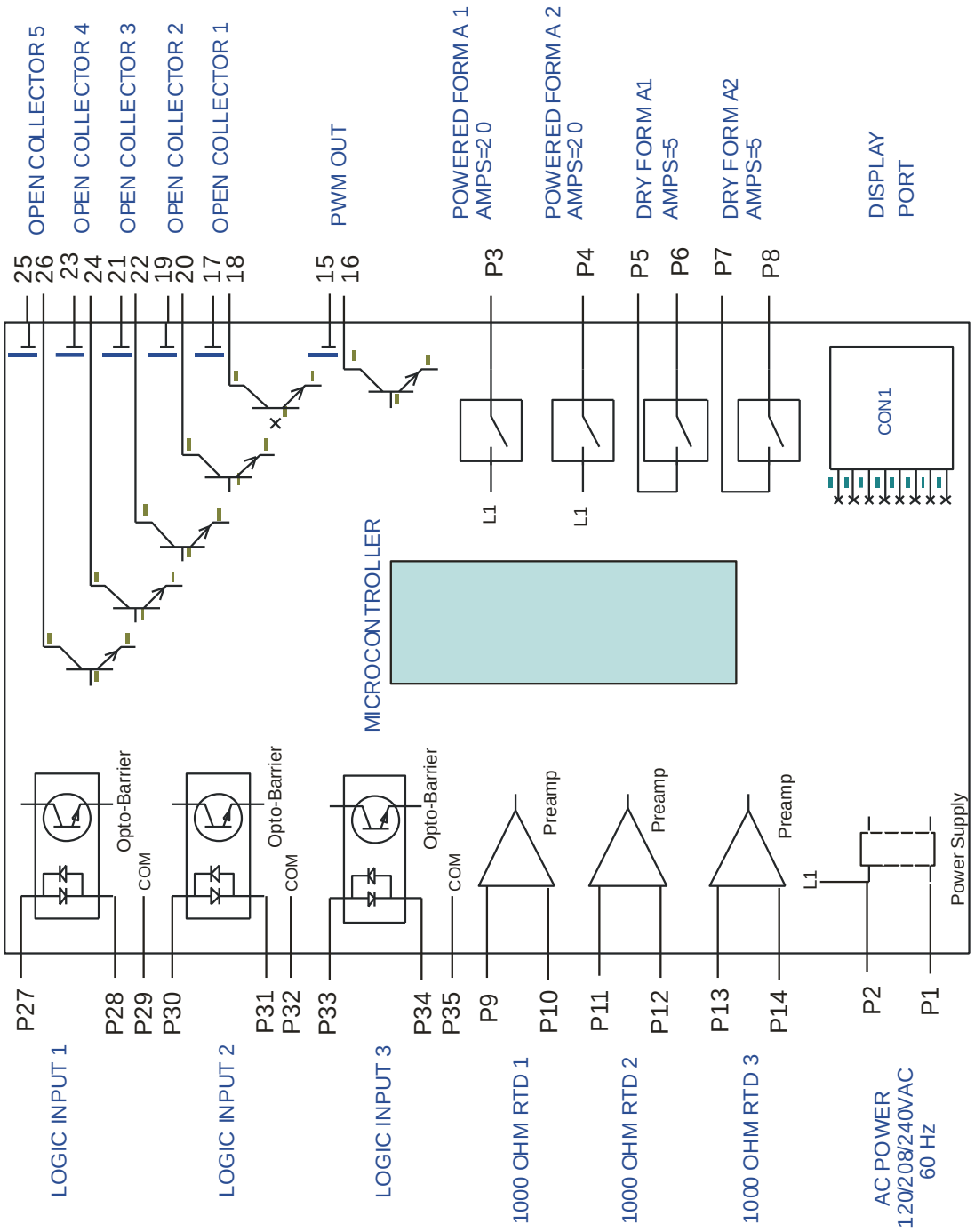


Figure 3.1 Base Unit B Pin-Out and Block Diagram

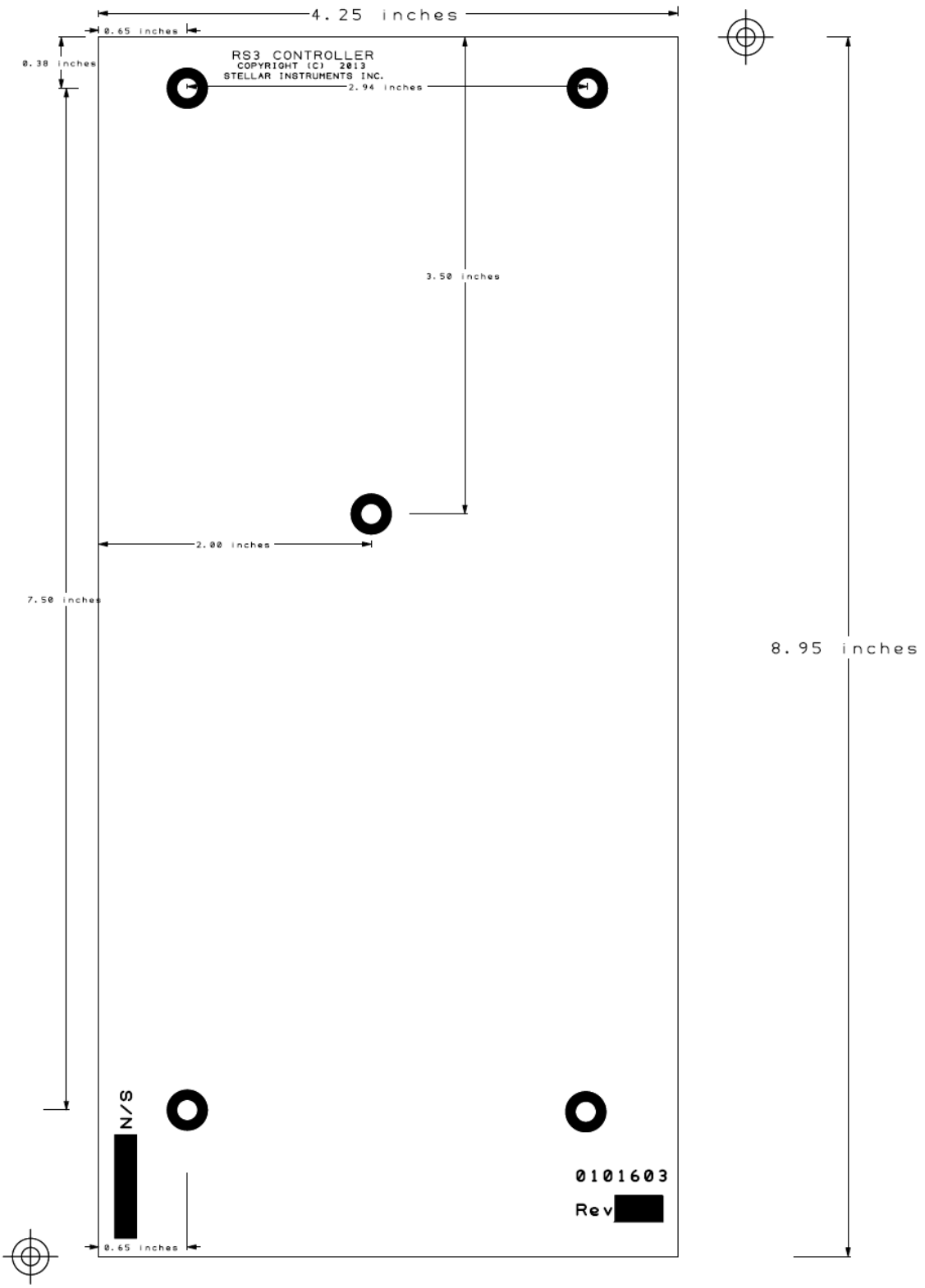
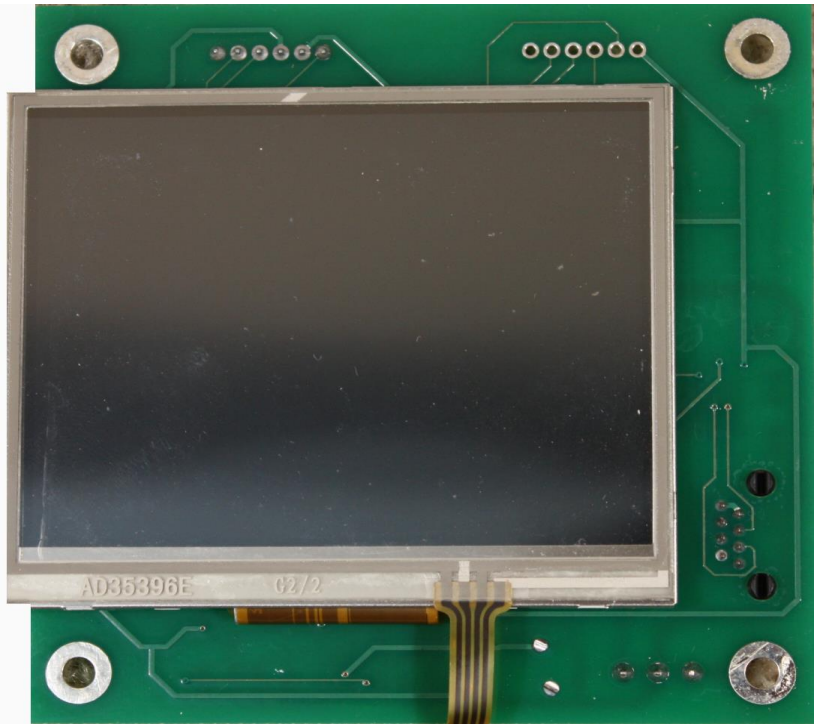


Figure 3.2 Base Unit B Mechanical Mounting

Section 4— OPERATOR INTERFACE MODULE (OIM)



4.1 General Information – The OIM has a ¼ VGA (320 x 240 pixel) color liquid crystal display and a touch screen for operator command entry. The display is also LED backlit. The display is powered from any base module (12-24vdc), and contains all power conversion necessary to power the LCD screen and the LED backlighting. The display has its own microcontroller and SD card mass storage memory and allows menu driven calibration, setup and operation via various user created background screens and context sensitive buttons. The OIM will electrically connect to a base module via a standard eight conductor Ethernet cable terminated with RJ45 connectors. The programming and setup of this interface will be discussed later in this section

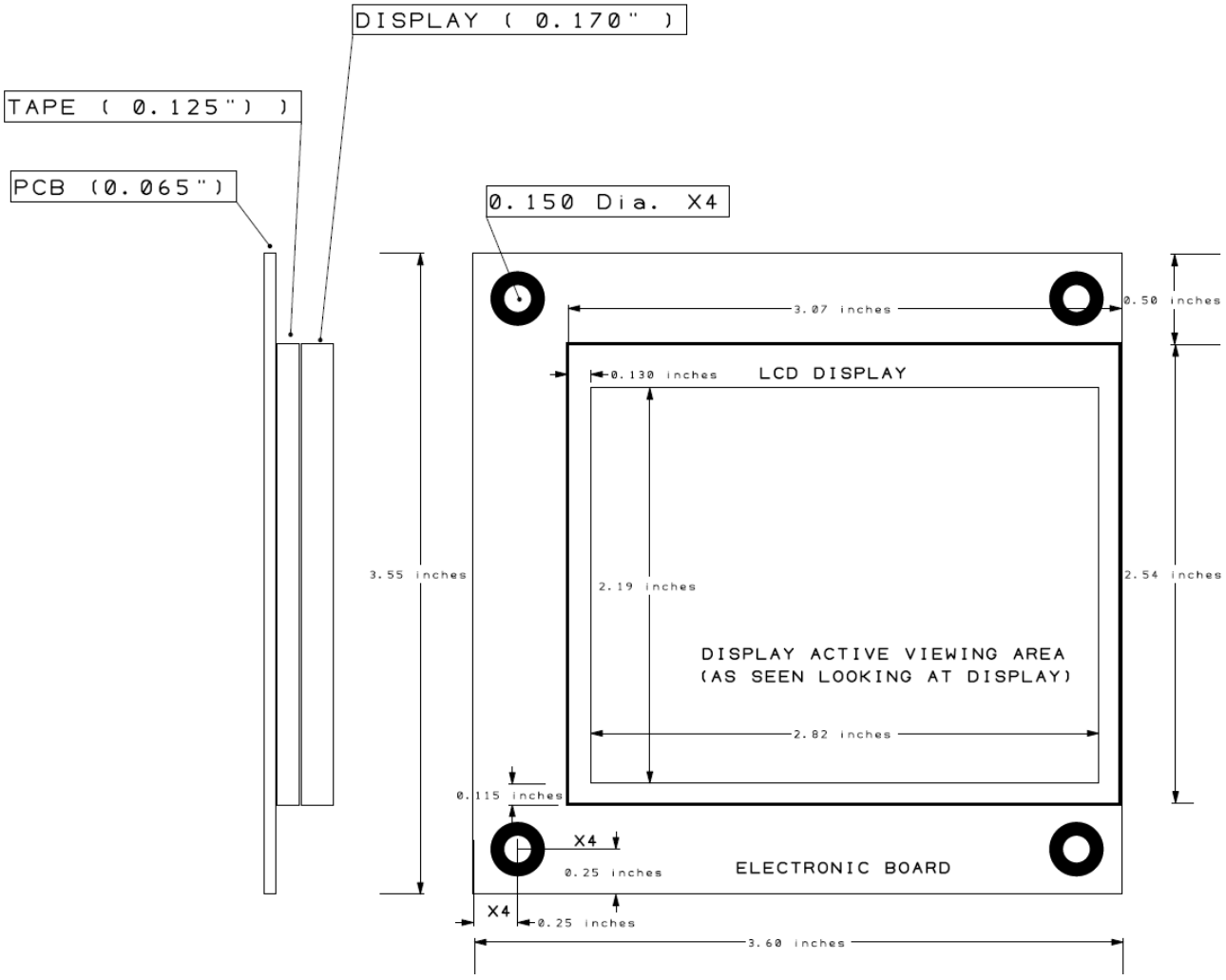


Figure 2.3 Operator Interface Module Mechanical Diagram

4.2 Display Programming

Important Note: Display creation requires some familiarity with graphics programs such as Windows Paint, CorelDRAW, etc. If the user does not have a fundamental knowledge of a graphics application program, Stellar Instruments may be contracted to create the customer's user screens.

4.2.1 General – As noted earlier, the display is a ¼ VGA-320 by 240 pixel touchscreen display. The display is a "smart" device and allows the user substantial flexibility in configuring the display format and touchscreen functions

to suit the application. The user interface will generally consist of a series of screens to be displayed.

It is suggested that a story board or flow chart containing all the desired screens and their ordering be created prior to creating each individual screen. Then a 320 x 240 image for each screen should be created to be saved in bitmap (.bmp) format.

Each screen is an image that may contain areas that can be given special properties during user programming and will become **buttons** or **windows**. Buttons are used for operator initiated actions (e.g. fetch a variable from the base unit and display it, alter a base unit parameter, go to another display screen, etc.) and windows are used to display information (data pulled from the base unit).

Programming of button and window functions is facilitated with a Stellar supplied "form" in Excel format. There is a specific form for each base unit allowing the user to implement buttons and windows from the user created images. The form will allow the user to specify a name for the screen he is about to create, give the next desired screens a names (assuming there are next screens) and give the previous screen a name (again assuming there is a previous screen).

Note: Screen file names must be seven characters or less, not including the ".bmp" extension.

The user will then define button and window locations and sizes for each of the buttons/windows on each image using an (x,y) coordinate system based upon Windows drawing conventions where (0,0) is the top-left of the screen and (320,240) is the bottom-right. For example a button defined by (155,115), (165,125) represents a 10x10 pixel button located in the middle of the screen. These definitions will tell the display controller how to react to the touchscreen and where to display data in a particular window location.

4.2.2 Screen Image Creation – The screen is defined as the entire 320 x 240 area of the display. The default screen is a blank screen that will be the default screen in the graphics program being used. The user may want to create a more pleasing screen to serve as a background into which he will insert his buttons and windows. A typical method for screen creation is to use Windows "Paint". The

user would establish a 320 x 240 pixel screen in “Paint” and then fill the screen with whatever picture is desired. This may be anything from a blank white screen to an image, or in other words, anything that can be imported into “Paint”. It is worth noting that any graphics program may be used for screen creation as long as the result can be saved in .bmp format. The title of this file will be entered in the box labeled “**This Screen**” on the supplied form. If there will be buttons on the screen that will cause the display to jump to other screens, then they should be entered into the appropriate “**Next Screen**” and “**Previous**” screen boxes on the form. Remember to save each screen image upon completion noting that each screen will have a unique name and a .bmp file extension. Browser buttons are also available to the right of the screen name entry boxes to facilitate location of the desired screen bitmap file.

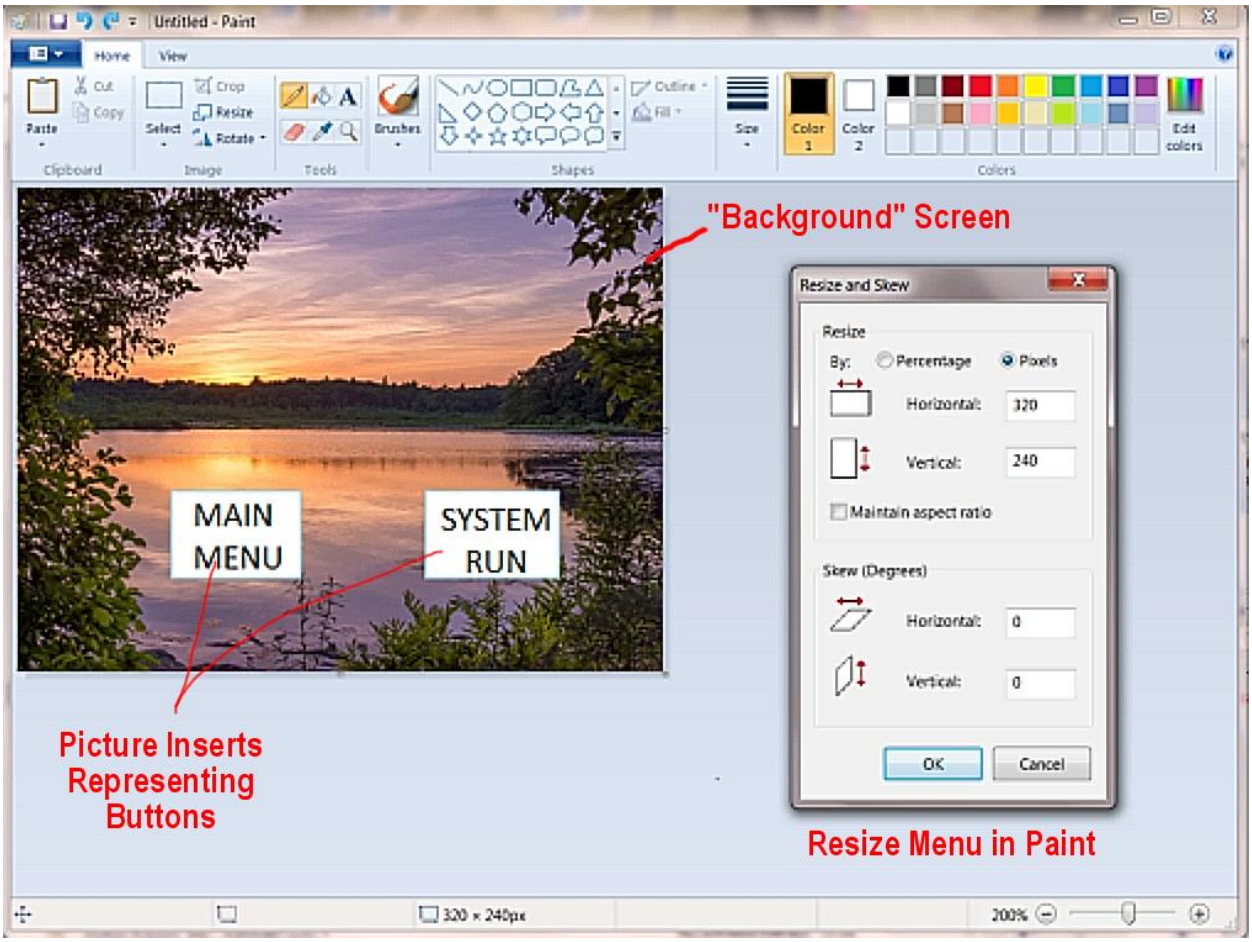


Figure 4.1 Example Display Screen With Two User Buttons - Drawn in Windows Paint

4.2.3 Button Creation and Programming – The user may draw up to ten images representing buttons on each screen image with the buttons also being treated as pictures (i.e. each button may be composed of text, images, etc.). Once all the buttons are drawn, the button coordinate locations must be entered into the Stellar supplied form in the **Screen Configuration Form Button Configuration** table. See Figure 4.2. There will be four column entries into the form (bottom left X&Y and top right X&Y). Another form entry for each button is the association of each button to a function. The **Screen Configuration Form Button Configuration** table has a "Function" column with a pull-down list for each defined button. The pull-down list will contain base unit dependent functions (e.g. increment or decrement a parameter in the base unit, go to another screen, etc.) and functions to allow selection of other screens. The user must obviously have an understanding of the available parameters in his particular base unit. An entry should be filled out in the form for each button image drawn on the screen. The final button configuration entry is an "Index". This entry is used when the button action is to jump to a next screen with the index number being the number of the next screen as entered in section 4.2.2.

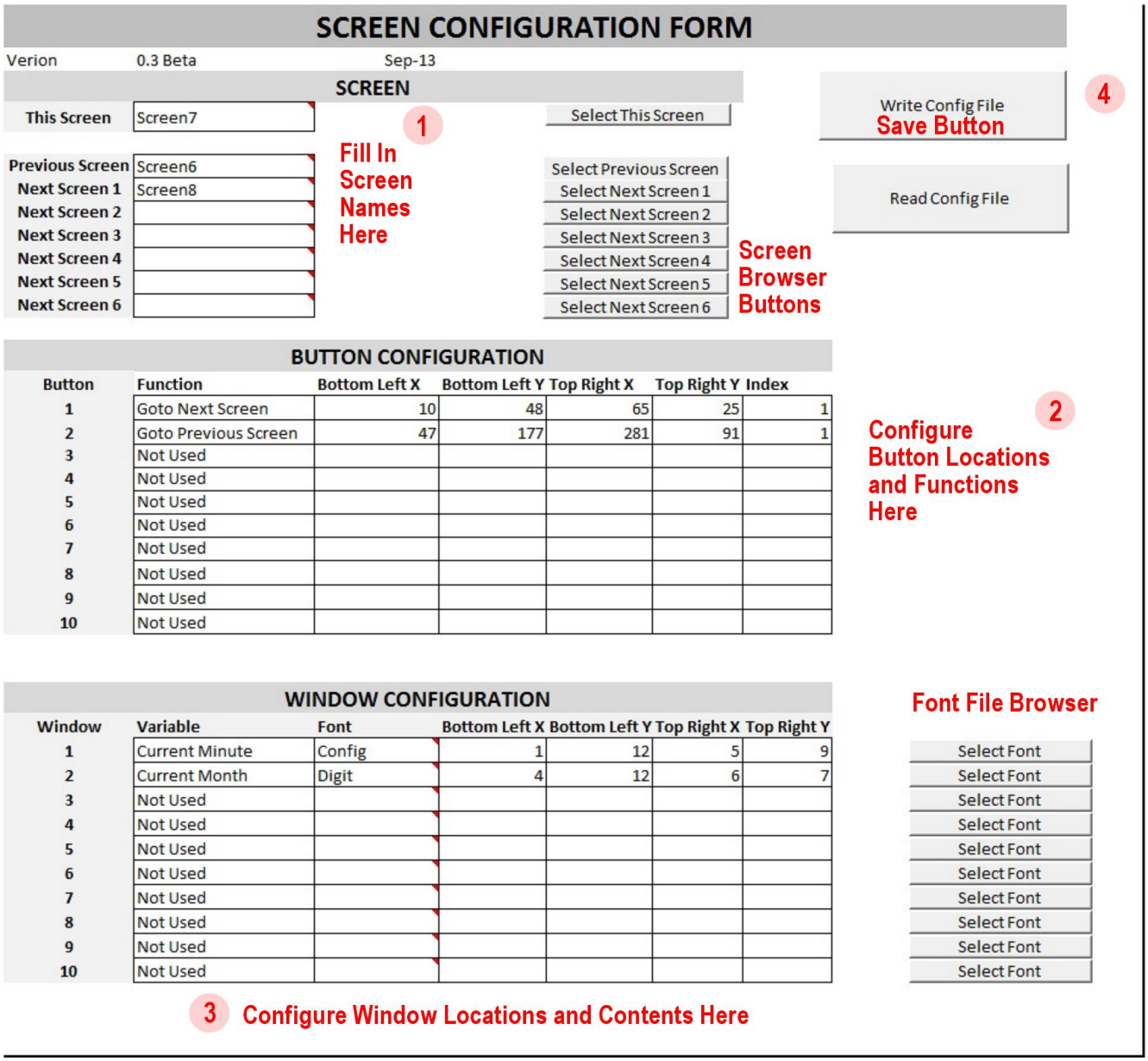


Figure 4.2 Example Screen Configuration Form

4.2.4 Window Creation and Programming – Windows are treated much the same as buttons. A window image is drawn on a screen image and its X/Y coordinates are entered into the **Window Configuration** table on the **Screen Configuration Form.** The "Variable" column in the same table has a pull-down that is used to select the base unit variable that will be displayed in the selected window. There

are additional steps required at this point to create the characters or images to be displayed in the window.

4.2.4.1 Font Files – Variables are displayed as pictures within a screen window. As an example, assume a three digit number is to be displayed in a window. To accomplish this, a set of font files will need to be created to represent the digits 0 through 9. Again, using a graphics program, a set of pictures with the digits 0 through 9 needs to be created and saved in .bmp format. The file name for each file should begin with the same characters and then end with characters unique to each file (e.g. digit0.bmp for the picture containing the "0", digit1.bmp for the picture containing the "1", etc.).

Note: Font file names must be seven characters or less, not including the ".bmp" extension.

For the example above, the word "digit" would be entered into the "Font" column in the **Window Configuration** table. When the display controller receives a requested variable from the base unit, it will associate each character in the variable (each digit in the example) and fill the window with the resulting characters (three numeric digits in the example). There are "Select Font" buttons located to the right of the Window Configuration table that can be used to browse for font files if the filename is not already known. The buttons will open up a Windows dialog box to enable file searching.

4.2.4.2 Window Sizing – It is important to size the screen window and the font files to match up. For the three digit display example noted above, if each font file digit were 10x10 pixels, then the display window should be sized at 30x10 pixels.

4.2.5 Saving Files – Once a form for a given screen is complete, it is saved using the "**Write Config File**" button on the form. The resulting saved file will have a .txt extension. Note: Config file names must be seven characters or less, not including the ".txt" extension. This entire process will be repeated for each display screen until all screens have been created. The result will be .txt files for each display screen form results, .bmp files for each screen image and .bmp files for all the font images. These files should then be loaded into the root directory of an SD card, with the card being 2GB or smaller in size.

