DSP004 Intelligent Character Display
User Manual

Pronto Electronic Displays LLC
www.prontodisplays.com

Doc.# 1031-DSP004-M REV.01

Contents

1.0 Theory of Operation 2
1.1 Establishing Communication 2
1.2 Operating Modes 2
1.3 On-Board EEPROM 2
1.4 Controlling Display Brightness 2
1.5 Board Power Requirements 3
1.6 Power Conditioning Circuit 3
1.7 Reset 3
2.0 Using the Menu 4
2.1 The Main Menu 4
2.2 Setting User Programmed Baud Rate 5
2.3 Setting the User Programmed RxIN EOL Terminator 5
2.4 The Brightness Mode Configuration Menu 6
2.4.1 Set to Automatic Mode 6
2.4.2 Adjust Automatic Mode Min & Max Settings 6
2.4.3 Set Display to Fixed Brightness Mode 7
2.4.4 Adjust Fixed Brightness Setting 7
2.4.5 Set to EXT PWM Mode 8
2.5 Enable/Disable User Programmed Display Scroll Mode 8
2.6 Change the User Programmed Scroll Speed 8
2.7 Create a New Custom Character 9
3.0 Using External PWM for Brightness Control 11
4.0 Programming the On-Board EEPROM 12
4.1 Programming Configuration Settings 12
4.2 User-Defined EEPROM Addresses 12
4.3 Addresses for Custom Characters 12
4.4 Code for EEPROM Programming 13
5.0 Writing Serial Data to the Display 13
6.0 EMI/RFI Considerations 14
7.0 Troubleshooting Table 15
Appendix A 17
A.1 Specification (Header) File for MC24AA024 Class Object 17
A.2 Implementation File (.cpp) for MC24AA024 Class Object using Arduino 18
A.3 Program code using MC24AA024 Class Object 19
A.4 Program Code for PWM signal generation using Arduino 20

1.0 Theory of Operation

1.1 Establishing Communication

Communication with the device is established using Serial +5V TTL/UART. For development purposes, VCP (Virtual COM Port) drivers are used to establish serial communications using USB, with Serial Terminal Emulators providing a user interface through which the user can interact with the device. Refer to document #1032-DSP004-QSG, Quick Start Guide on how to establish communications and configure the serial terminal emulators.

1.2 Operating Modes

The unit has two basic modes of operation, being determined by the position of the CONFIG switch. In the NORM position, the unit operates in a fixed factory default consisting of the following settings:

Baud rate= 38400

RXIN EOL Terminator = <CR>

TXOUT EOL Terminator = <CR><LF>

Brightness Mode = Automatic, with minimum brightness 10%, and maximum brightness 100%

Display set to Non-Scrolling Mode

Additionally, in the NORM position the UI menu can be accessed by pushing the Menu button; the MENU button is disabled when the CONFIG switch is in the PRGM position. The UI Menu purpose is to permit changing configuration settings, or writing custom characters. The user-programmed operating settings will take effect when the CONFIG switch is in the PRGM position. The UI Menu can be used to:

a) Change Baud rate settings.
b) Change EOL Terminator settings for RXIN.
c) Change Brightness mode/settings.
d) Change Scroll speed settings and scroll enable/disable.
e) Design up to 34 custom characters.

1.3 On-Board EEPROM

In addition to changing the settings with the UI, it is possible to directly write the settings, including custom characters, to the board, bypassing the UI and thereby allowing high-volume automated programming. This is accomplished using the SDA and SCL pins provided on the I/O connector.

1.4 Controlling Display Brightness

The unit has three different modes of display brightness controls:

1) Automatic mode with integrated ambient light sensor
2) Fixed brightness mode
3) External PWM, supplied through the PWM pin 1 on the I/O connector.

1.5 Board Power Requirements

The board requires +5VDC with a maximum current requirement of 1.0A. Device current draw changes substantially depending on the illumination brightness level and the total number of LED dots that are on at any given time. For example, in relatively low-light office conditions, and with all standard characters being used, current draw can be 20 mA or less. On the other hand, in the worst-case scenario, which consists of all LED dots being illuminated at maximum brightness, close to 1A of current will be drawn. Input voltage range is 4.75 - 5.3V.

1.6 Power Conditioning Circuit

The unit has an on-board power conditioning circuit, consisting of enhanced ESD protection, reverse voltage protection, and overvoltage protection. There is no undervoltage protection.

If reverse voltage is applied to the board, the Green PWR led will not be illuminated, as no power will be applied to the board.

If an overvoltage condition is applied to the board power, the Green PWR led will go off, and the red OVP led will be illuminated; it will stay illuminated until the OVP circuit is reset. To reset the OVP protection and resume normal operation, first turn off the overvoltage power. Then set the power to the correct voltage, and re-apply power.

1.7 Reset

The unit can undergo a reset by the following conditions:

1. Momentary Power Loss.
2. Command by means of User Interface.
3. Pushing the Reset Button.
4. Changing the position of the “CONFIG” switch.
5. Pulling the Reset pin 8 on the I/O connector to logic low.

In all cases when the board is reset, the input serial buffer is flushed. Any previous display data is erased, and the display will show the default reset characters “01234567”.

2.0 Using the Menu

2.1 The Main Menu

Upon successful connection, when the MENU button is pressed, the Main Menu will be displayed:

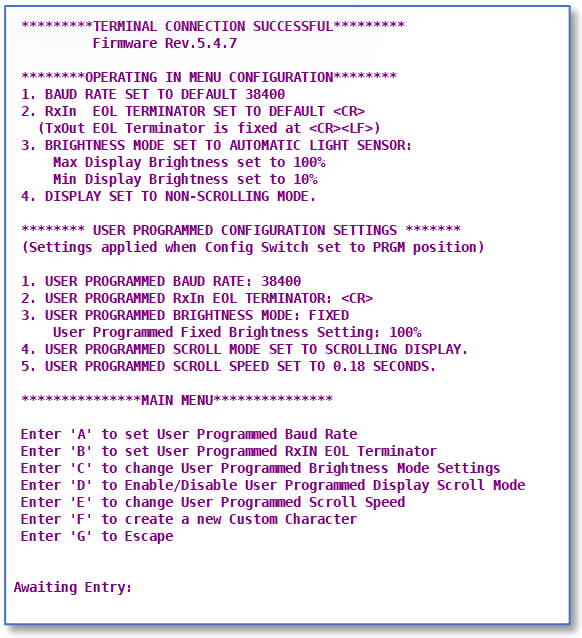


FIG.1, Headers and Main Menu

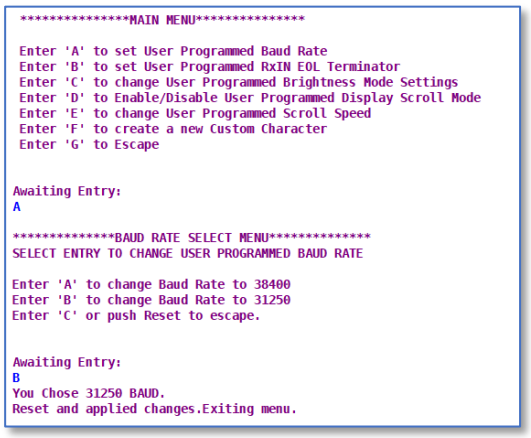
The “OPERATING IN MENU CONFIGURATION” header serves as a message to the user that the unit is currently operating in the default factory configuration. The settings for this configuration are listed in this header to serve as a reference for the user.

The “USER PROGRAMMED CONFIGURATION SETTINGS ” header displays the device configuration settings that the unit will operate under whenever the “CONFIG” switch is set to the “PRGM” position. The settings for this configuration are listed in this header to serve as a reference for the user. Note the unit comes with settings pre-programmed in the factory, which of course may be overwritten by the user.

The “MAIN MENU” header shows all the configuration settings that can be changed, as well as providing an entry allowing the creation of a custom character. The remainder of this section provides details for each entry made. Note in all cases upon placing an entry when the unit confirms the choice, it will then perform a reset, and will then display the statement “RESET AND APPLIED CHANGES. EXITING MENU”, at which time the unit then exits out of the Main Menu, and resumes normal operation. Push the MENU button again if it is desired to make more entries from the Main Menu. The newly applied settings will be shown in the “USER PROGRAMMED CONFIGURATION SETTINGS” header, which will appear the next time the MENU button is pushed again.

2.2 Setting User Programmed Baud Rate

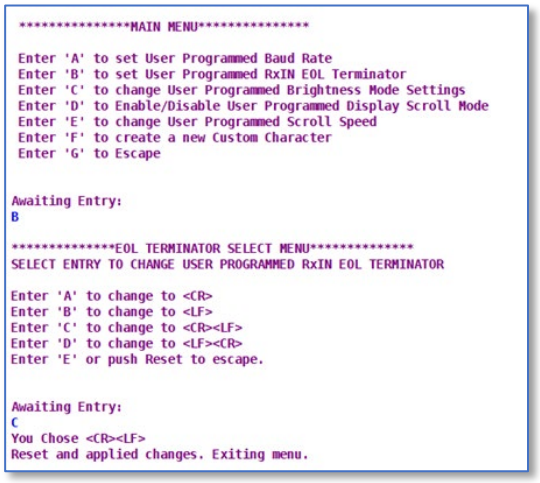
An “A” entry in the Main Menu results in a Baud rate selection menu being displayed:



A dialog will commence asking the user to place A,B or C.The unit will operate using the selected BAUD rate when the CONFIG switch is set to the “PRGM” position.Push the MENU button again to make more selections.

2.3 Setting the User Programmed RxIN EOL Terminator

A “B” entry in the Main Menu results in a Baud rate selection menu being displayed:



The EOL Terminator that is being selected is for the display RxIN line (Pin 3). (Note the TxOUT EOL terminator for the unit is always fixed at <CR><LF>). The unit will operate using the selected RxIN EOL terminator when the CONFIG switch is set to the “PRGM” position. Push the MENU button again to make more selections.

2.4 The Brightness Mode Configuration Menu

A “C” entry in the Main Menu results with the Brightness Mode Configuration Menu being displayed:

```
**************MAIN MENU**************

 Enter 'A' to set User Programmed Baud Rate
 Enter 'B' to set User Programmed RxIN EOL Terminator
 Enter 'C' to change User Programmed Brightness Mode Settings
 Enter 'D' to Enable/Disable User Programmed Display Scroll Mode
 Enter 'E' to change User Programmed Scroll Speed
 Enter 'F' to create a new Custom Character
 Enter 'G' to Escape


Awaiting Entry:
C

***************BRIGHTNESS MODE CONFIGURATION MENU***************
SELECT ENTRY TO CHANGE USER PROGRAMMED BRIGHTNESS MODE SETTINGS

Enter 'A': Set to Automatic Mode
Enter 'B': Adjust Automatic Mode Min & Max Settings
Enter 'C': Set Display to Fixed Brightness Mode
Enter 'D': Adjust Fixed Brightness Setting
Enter 'E': Set to EXT PWM Mode
Enter 'F' or push Reset to escape.


Awaiting Entry:
A
Light Sensor set to Automatic Mode.
Reset and applied changes.Exiting menu.
```

2.4.1 Set to Automatic Mode

An “A” entry in the “Brightness Mode Configuration Menu” will set the unit operating in the Automatic mode when the CONFIG switch is set to the “PRGM” position. In this mode, the device automatically modulates the display brightness depending upon ambient light conditions, utilizing the ambient light sensor which is located on the display side of the device directly alongside the leftmost character. Push the MENU button again to make more selections.

2.4.2 Adjust Automatic Mode Min & Max Settings

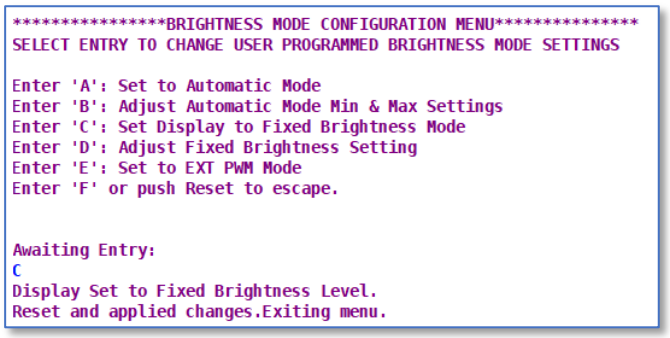
An “B” entry in the “Brightness Mode Configuration Menu” will bring up a dialog in which one may enter min and max brightness levels:

```
***************BRIGHTNESS MODE CONFIGURATION MENU***************
SELECT ENTRY TO CHANGE USER PROGRAMMED BRIGHTNESS MODE SETTINGS

Enter 'A': Set to Automatic Mode
Enter 'B': Adjust Automatic Mode Min & Max Settings
Enter 'C': Set Display to Fixed Brightness Mode
Enter 'D': Adjust Fixed Brightness Setting
Enter 'E': Set to EXT PWM Mode
Enter 'F' or push Reset to escape.


Awaiting Entry:
B
*******Automatic Mode Min and Max Settings Menu*******

******Set Max Brightness Level*******
Must enter an integer number from 2 to 100.
Input Max Brightness%:
100
Max Brightness programmed to 100 %

******Set Min Brightness Level*******
Must enter an integer number from 1 to something less than Max Brightness.
Input Min Brightness%:
5
Min Brightness programmed to 5 %

Reset and applied changes.Exiting menu.
```

These levels represent a linear percentage of min and max internally-generated PWM duty cycle. Note that the human perception of brightness is more sensitive to lower light levels than to brighter ones, as described by the Weber-Fechner laws; for example, the level of *difference* perceived for the lower PWM percentages of 10-30% , will not be perceived as being the same amount of difference at higher PWM levels of 60-90%.The brightness minimum value determines how dim the display will appear under the dimmest lighting conditions, and the brightness maximum value determines how bright the display will appear under the brightest ambient light conditions. After completing entry, push the MENU button again to make more selections.

2.4.3 Set Display to Fixed Brightness Mode

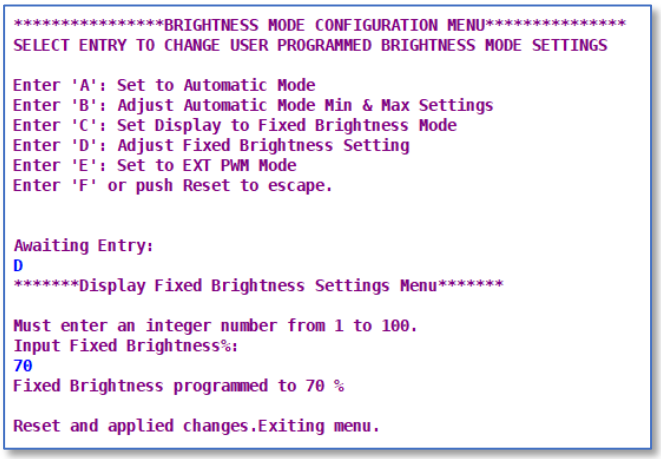
A “C” entry in the “Brightness Mode Configuration Menu” will cause the unit to operate at a display fixed brightness level when the CONFIG switch is in the PRGM position.



After finishing the entry, push the MENU button again to make more selections.

2.4.4 Adjust Fixed Brightness Setting

A “D” entry in the “Brightness Mode Configuration Menu” will bring up the “Display Fixed Brightness Settings Menu”:



In this menu the user is prompted for entering a percentage number from 1 to100. This number represents a linear percentage of the device’s internal PWM duty cycle. The display illumination will be at a level corresponding to this entry, if the unit is set to operate in the Fixed Brightness Mode. After completing entry, push the MENU button again to make more selections.

2.4.5 Set to EXT PWM Mode

An “E” entry in the “Brightness Mode Configuration Menu” will cause the unit to operate in EXT PWM mode when the CONFIG switch is set to the PRGM position:

```
***************BRIGHTNESS MODE CONFIGURATION MENU***************
SELECT ENTRY TO CHANGE USER PROGRAMMED BRIGHTNESS MODE SETTINGS

Enter 'A': Set to Automatic Mode
Enter 'B': Adjust Automatic Mode Min & Max Settings
Enter 'C': Set Display to Fixed Brightness Mode
Enter 'D': Adjust Fixed Brightness Setting
Enter 'E': Set to EXT PWM Mode
Enter 'F' or push Reset to escape.


Awaiting Entry:
E
Display Brightness selected to EXT PWM Mode.
Reset and applied changes.Exiting menu.
```

In this mode, the internal ambient light sensing is bypassed, and the display brightness can then be modulated by an external PWM signal supplied by the user on Pin 1. With the unit set to operate in this mode, the unit display brightness will be at maximum level, if no PWM signal is supplied to Pin 1. After completing entry, push the MENU button again to make more selections.

This concludes the section on the Brightness Mode Configuration menu. The next section begins again with Main Menu entries.

2.5 Enable/Disable User Programmed Display Scroll Mode

A “D” entry from the Main Menu will present a dialog to the user for enable/disable of the display scrolling feature:

```
***************MAIN MENU***************

Enter 'A' to set User Programmed Baud Rate
Enter 'B' to set User Programmed RxIN EOL Terminator
Enter 'C' to change User Programmed Brightness Mode Settings
Enter 'D' to Enable/Disable User Programmed Display Scroll Mode
Enter 'E' to change User Programmed Scroll Speed
Enter 'F' to create a new Custom Character
Enter 'G' to Escape


Awaiting Entry:
D
Enter A to enable Scrolling, or B for Non-scrolling mode:
A
User Programmed Display set to Scrolling mode.
Reset and applied changes.Exiting menu.
```

The user entry controls the display scroll mode the unit will operate in when the CONFIG switch is in the PRGM position.

2.6 Change the User Programmed Scroll Speed

An “E” entry from the Main Menu will present a menu for setting up the Scroll Speed the unit will operate with when it is running in Scrolling mode:

```
***************MAIN MENU***************

 Enter 'A' to set User Programmed Baud Rate
 Enter 'B' to set User Programmed RxIN EOL Terminator
 Enter 'C' to change User Programmed Brightness Mode Settings
 Enter 'D' to Enable/Disable User Programmed Display Scroll Mode
 Enter 'E' to change User Programmed Scroll Speed
 Enter 'F' to create a new Custom Character
 Enter 'G' to Escape


Awaiting Entry:
E
Enter scroll speed in seconds, from 0.10 to 1.00 seconds:
0.25
Scroll speed set to 0.25 seconds.
Reset and Applied Changes.Exiting menu.
```

The user may enter up to three significant digits with the range of 0.1 to 1.00 seconds. This time is the time each character remains illuminated; hence, a smaller number results in a faster scroll speed. After completing entry, push the MENU button again to make more selections.

2.7 Create a New Custom Character

An “F” entry in the Main Menu will bring up the “5x7 Character Writing Menu”:

```
***************MAIN MENU***************

 Enter 'A' to set User Programmed Baud Rate
 Enter 'B' to set User Programmed RxIN EOL Terminator
 Enter 'C' to change User Programmed Brightness Mode Settings
 Enter 'D' to Enable/Disable User Programmed Display Scroll Mode
 Enter 'E' to change User Programmed Scroll Speed
 Enter 'F' to create a new Custom Character
 Enter 'G' to Escape


Awaiting Entry:
F

*************** 5x7 CHARACTER WRITING PROGRAM ***************

34 Custom Characters are available, using ASCII numbers 128-161.

Enter the ASCII Number:
130
You entered: 130
Accept Entry? (must enter Y or N):
Y

ASCII Number 130 has been assigned to this character.

*** ROW DATA ENTRY SECTION ****

Entries must be made as a five-bit Binary number.
Enter Data for Row 1:
00000

You entered BINARY:00000
Accept Entry? (must enter Y or N):
Y
Enter Data for Row 2:
01010

You entered BINARY:01010
Accept Entry? (must enter Y or N):
Y
```

This menu creates a guided dialog in which the user can create a custom character without writing code. In the first part of the dialog, the user is prompted to enter an ASCII number for the character. In the second part of the dialog, the user is asked to enter 5 digits of either ones or zeros, for each character row, of which there are 7. To help understand the principles involved in the row data, the chart below provides an example of row data to be entered for a “heart” symbol:

	COL 1	COL 2	COL 3	COL 4	COL 5		
ROW 1	0	0	0	0	0		ROW 1 DATA: 00000
ROW 2	0	1	0	1	0		ROW 2 DATA: 01010
ROW 3	1	0	1	0	1		ROW 3 DATA: 10101
ROW 4	1	0	0	0	1		ROW 4 DATA: 10001
ROW 5	0	1	0	1	0		ROW 5 DATA: 01010
ROW 6	0	0	1	0	0		ROW 6 DATA: 00100
ROW 7	0	0	0	0	0		ROW 7 DATA: 00000

Data for Row 3 can be described as follows:

Row 3 COL1 COL2 COL3 COL4 COL5 Data: 10101

A “1” turns that pixel on, a “zero” turns it off. In all cases the “msb” for the entry represents the state of the Column 1 pixel, the “lsb” entry represent the state for the Column 5 pixel, with Column 5 being the rightmost pixel in the character. Row 1 is the top row, and Row 7 is the bottom row.

The menu exits after the last row entry is completed:

```
Enter Data for Row 3:
10101

You entered BINARY:10101
Accept Entry? (must enter Y or N):
Y
Enter Data for Row 4:
10001

You entered BINARY:10001
Accept Entry? (must enter Y or N):
Y
Enter Data for Row 5:
01010

You entered BINARY:01010
Accept Entry? (must enter Y or N):
Y
Enter Data for Row 6:
00100

You entered BINARY:00100
Accept Entry? (must enter Y or N):
Y
Enter Data for Row 7:
00000

You entered BINARY:00000
Accept Entry? (must enter Y or N):
Y
Row Data Entry Completed

Programming User EEPROM ASCII 130 completed.
Reset and applied changes.Exiting menu.
```

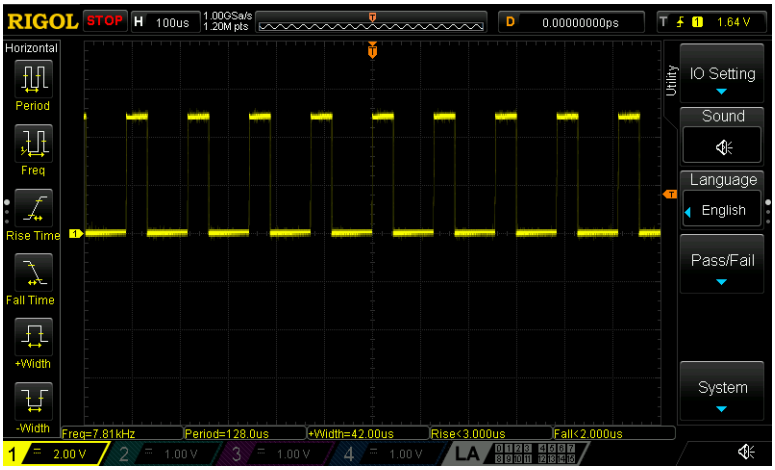
The custom character glyph is stored in the on-board EEPROM and is now available for use. Refer to Section 5 of this document for information on how to display the custom characters. After completing entry, push the MENU button again to make more selections.

3.0 Using External PWM for Brightness Control

The PWM pin is pulled to logic high by an internal 10k pullup resistor. Note if a PWM signal is supplied to this pin when the unit is not set to EXT PWM mode, the display may pulsate or go blank, due to interference with the user's PWM and the unit's internally-generated PWM signal. This is why it is important to ensure that no PWM signal is being applied to Pin 1 unless the unit is set for EXT PWM operation, which can be set by means of the user interface or by directly programming the on-board EEPROM. The PWM pin, if not used, should be disconnected. If it is connected to an MCU pin, then while not in use, the pins should either be set to a Hi-Z state, or set to Logic High voltage.

When the display is set to EXT PWM mode, the display will be at maximum brightness level when no external PWM signal is being applied.

The PWM signal applied should be TTL level with a frequency of 8 kHz +/- 10%. Below is an oscillogram of a signal applied to the PWM pin:



Note that frequencies other than 8kHz +/- 10% may be applied; however there is no guarantee that any such frequencies will not create interference artifacts due to the display multiplex characteristics.

Below are two lines of Arduino Uno code which may be used to generate an 7.812 kHz PWM signal suitable for use with the display unit. Note these settings are only applicable to Arduino Uno which uses the ATMEGA328 microcontroller:

```
10    TCCR2B = 0b00000010; / prescale,  divide by 8
20    TCCR2A = 0b00000011; / configure timer 2 to PWM Fast mode
```

Lines 10 and 20 will create PWM signal in Fast PWM mode for Timer 2 at a frequency of 7.812 kHz, which will be available on Arduino Uno pins D3 and D11.

PWM can be generated from Arduino Uno pins D3 or D11 with the following commands:

analogWrite(11,pwm_factor) or analogWrite(3,pwm_factor)

Where : pwm_factor = an integer number from 0 to 255 with 0 being the lowest duty cycle (always off) and 255 being the highest duty cycle (always on).

When PWM is not being used and it is desired to allow the unit normal operation in other brightness modes such as Automatic or Fixed Brightness, set the PWM pin High by writing a PWM factor = 255. See the Appendix for Arduino Program code for generating a PWM signal.

4.0 Programming the On-Board EEPROM

The on-board EEPROM may be directly programmed by use of the SDA and SCL pins provided on the J1 I/O connector. Ensure the display unit is powered on when programming. The I2C frequency should be set to 100kHz. Programming can take place at any time during the device operation, with the exception of the device undergoing a RESET; ensure that the device is not RESET concurrently with EEPROM programming. The SDA and SCL pins are tied to board +5V power with 4.7k pullups. **Write operations to the on-board EEPROM should only be used for initial development / configuration of the device; EEPROM write operations must not be performed as part of the normal operating profile of the device due to the limited number of write cycles available for the on-board EEPROM.**

Below is a memory map for the on-board EEPROM:

ADDRESS#	FUNCTION	CONTENTS
0-237	CUSTOM CHARACTER FONT DATA	ROW DATA FOR 34 CUSTOM CHARACTERS, ASCII 128-161
238-244	USER DEFINED	USER DEFINED
245	BAUD RATE CODE NUMBER	01=38400 02=31250
246	EOL TERMINATOR CODE NUMBER	01=<CR> 02=<LF> 03=<CR><LF> 04=<LF><CR>
247	LIGHT SENSOR MODE CODE NUMBER	01=AUTO MODE 02=FIXED BRIGHTNESS MODE 03=EXTERNAL PWM MODE
248	FIXED BRIGHTNESS LEVEL	INTEGER NUMBER 1-100
249	MIN BRIGHTNESS LEVEL	INTEGER NUMBER 1-100
250	MAX BRIGHTNESS LEVEL	INTEGER NUMBER 1-100
251	SCROLL ENABLE	01=SCROLLING DISPLAY 02=NON-SCROLLING DISPLAY
252	SCROLL SPEED	INTEGER NUMBER 11 to 112
253-255	USER DEFINED	USER DEFINED

4.1 Programming Configuration Settings

Note addresses 245-252 are used for setting the operating configuration for the device when the CONFIG switch is placed in the PRGM position. Note also that only integer numbers need to be entered; no bit-masking or other complicated techniques are required, because each address represents only one functionality. There is no need to separately set certain bits within the same byte of data. The address contents above are shown in decimal values.

4.2 User-Defined EEPROM Addresses

Note that addresses 238 through 244 and 253 through 255 are available for user-defined applications. Some uses that may be programmed could include serial numbers, code configurations, etc.

4.3 Addresses for Custom Characters

The memory data location for custom ASCII characters is organized into individual addresses in memory based on what the ASCII number for that character is. For example, for ASCII 128, data for Row 1 will start at address zero, Row 2 data at address 1, Row 3 data at address 2, and so on, with Row 7 data occupying address 6. The chart below may be used for correctly loading custom character glyph data into memory if directly programming the on-board EEPROM:

CHARACTER ASCII #	ADDRESS NUMBER FOR ROW 1 DATA		CHARACTER ASCII #	ADDRESS NUMBER FOR ROW 1 DATA
128	0		145	119
129	7		146	126
130	14		147	133
131	21		148	140
132	28		149	147
133	35		150	154
134	42		151	161
135	49		152	168
136	56		153	175
137	63		154	182
138	70		155	189
139	77		156	196
140	84		157	203
141	91		158	210
142	98		159	217
143	105		160	224
144	112		161	231

4.4 Code for EEPROM Programming

The on-board EEPROM is a Microchip Technology P/N 24AA024-I/SN. Consult the manufacturer datasheet for I2C programming details particular for this chip. Arduino code for reading/writing to the on-board EEPROM is supplied in the appendix for user reference.

5.0 Writing Serial Data to the Display

This section offers some tips for sending data for display to the unit using Arduino. This section will show how to print serial data to the device using the various EOL terminator configurations available to the unit, as well as how to display custom characters. The brief descriptions of the Arduino print commands included here may be supplemented with more detailed information on the Arduino website.

Serial.println(“string”) : The text within quotes will be transmitted with <CR><LF> EOL terminators.

Serial.print(“string”) : The text within quotes will be transmitted without any EOL terminator.

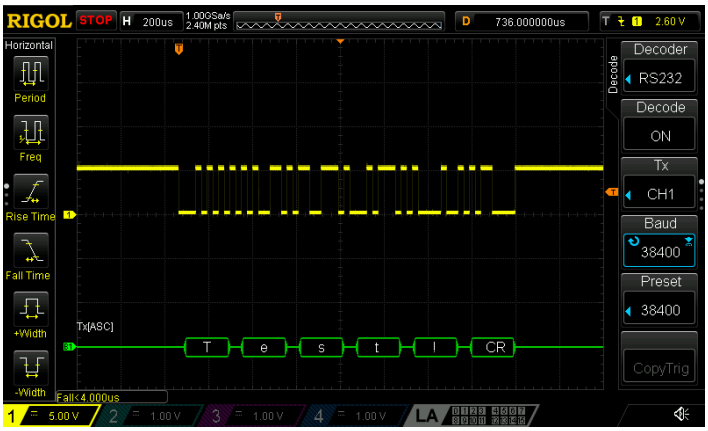
Serial.write(val) : The numeric value within parenthesis will be transmitted. This is useful for sending non-printable characters such as EOL terminators, or custom characters.

For example, in the below lines 10 and 20, the text “Test!” will be transmitted followed with <CR> for the EOL terminator:

10 Serial.print(“Test!”);

20 Serial.write(13);

The waveform below shows the serial signal being transmitted with the <CR> EOL terminator, using an oscilloscope with a protocol analyzer:



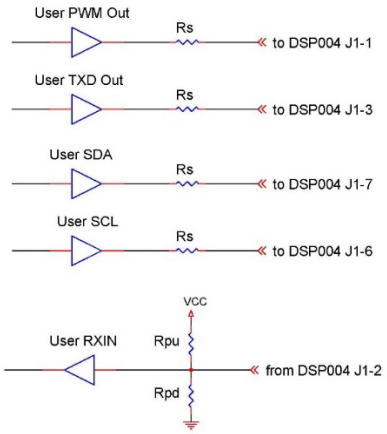
To display custom characters, use the Serial.write(val) command, where val= the ASCII number of the custom character. For example, the custom character ASCII 135 can be displayed with the command:

Serial.write(135);

Note that if serial data is transmitted with no EOL terminator, the unit will display the contents of the serial buffer after a 1 second timeout interval.

6.0 EMI/RFI Considerations

Emissions can be minimized with a source termination resistor placed near the user's signal outputs, and with a parallel termination scheme placed at the users RXIN input pin as shown below.



The values for Rs, Rpu and Rpd depend on the type of transmission medium and connectors used. Rs typically can have a value of 50 to 220 ohms.

Rpu and Rpd should be chosen such that their parallel resistance is equal to the impedance of the RXIN transmission line. Additionally, Rpd should be chosen such that (VOH/Rpd) is less than 25mA. For example, with a transmission line of impedance of 123 ohms, values could be Rpu=180 Ohms, Rpd= 390 Ohms. Output current with VOH=5.0V would be (5V/390 Ohms) = 13mA.

7.0 Troubleshooting Table

Symptom	Probable Cause	Corrective Action
1. Display is blank and unresponsive	a) Menu button has been pushed b) PWM Pin is at Logic Low	a) Push the Reset button b) When not used, PWM pin should be unconnected, or connected to a digital pin set to logic high or Hi-Z mode.
2. Unit stays at maximum brightness level	a) Unit operating in EXT PWM mode	a) Apply PWM signal to modulate brightness, or change operating modes.
3. Characters garbled	a) Incorrect Baud rate b) Incorrect EOL Terminator c) Out-of-range ASCII number being transmitted	a) Verify unit is operating with correct Baud rate setting b) Verify unit is operating with correct EOL Terminator setting c) Verify only printable ASCII numbers 32-126 are being transmitted. For custom characters, verify ASCII numbers are within range of 128-161 and that the glyphs used have been programmed to correct addresses.
4. Characters not displayed near end of message in scrolling mode	a) Too many characters have been transmitted	a) For 38400 Baud, max string length = 20 characters. For 31250 Baud, max string length =16 characters.
5. Display appears after a short 1 sec delay	a) Transmission is missing the EOL terminator	a) Check your transmission settings to ensure EOL terminator is being sent.
6. Display flashes and/or pulsates	a) PWM signal applied with unit not configured to operate in EXT PWM mode b) PWM signal frequency is too low	a) Ensure CONFIG switch is in PRGM position; also ensure PRGM mode operation is set to EXT PWM. b) PWM frequency should be 8kHz +/- 10%

7. OVP trips	a) Overvoltage applied to VIN pin b) Unit operating in a hot environment	a) Verify correct power input voltage: 4.75V - 5.3V b) Remove unit to a cooler location <40 deg.C., or lower the power input voltage
8. Menu does not appear	a) CONFIG switch in PRGM position b) Incorrect Baud rate or EOL terminator	a) Set CONFIG switch to NORM position b) Verify serial emulator is properly configured
9. Display exhibits inconsistent behavior	a) Intermittent connections	a) Check for worn or loose connectors on +5V, GND, Rx and Tx pins.
10. Display pulsates when testing brightness with light source	a) Testing Light source is pulsewidth modulated, creating stroboscopic interference	a) Switch to a non-dimmable LED or an incandescent light source for testing brightness function

For technical support, contact: support@prontodisplays.com

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

-Reorient or relocate the receiving antenna.

-Increase the separation between the equipment and receiver.

-Connect the equipment into an outlet on a circuit different from that to which the reciever is connected.

-Consult the dealer or an experienced radio/TV technician for help.

CAN ICES-3(B) / NMB-3(B)

Appendix A

A.1 Specification (Header) File for MC24AA024 Class Object

```
1
2    /*
3    Specification File for 24AA024
4    Copywrite 2025 Pronto Electronic Displays LLC
5
6    I2C Address Writing Info Note: RW bit is not included in the "data word": the Wire
     library supplies that bit during compilation.
7    Note address must consist of control bits 1010 with A2 A1 A0 suffix added.Below table
     shows what to use for the address,depending on how A2,A1 and A0 are wired either high or
     low.
8
9
10              A2   A1   A0          0b         Hex
11              0    0    0     0b1010000       0x50
12              0    0    1     0b1010001       0x51
13              0    1    0     0b1010010       0x52
14              0    1    1     0b1010011       0x53
15              1    0    0     0b1010100       0x54
16              1    0    1     0b1010101       0x55
17              1    1    0     0b1010110       0x56
18              1    1    1     0b1010111       0x57
19
20   */
21
22   #ifndef M24AA024_h
23   #define M24AA024_h
24
25   class MC24AA024
26   {
27        private:
28        int device_addr;
29        uint8_t data_addr;
30        uint8_t databyte;
31
32        public:
33
34        MC24AA024(uint8_t addrl)//read address into constructor
35        {
36             device_addr = addrl;
37        }
38
39        void writeDataByte(uint8_t,uint8_t);// (address,data)
40
41        uint8_t readDataByte(uint8_t);//  // return readbyte from (address)
42
43   };
44
45   #endif
46
47
```

A.2 Implementation File (.cpp) for MC24AA024 Class Object using Arduino

```
1
2   /*
3   Implementation File (.cpp) for MC24AA024
4   Copywrite 2025 Pronto Electronic Displays LLC
5   */
6
7
8   #include <Wire.h>
9   #include<MC24AA024.h>
10  #include<arduino.h>
11
12                  void MC24AA024 :: writeDataByte(uint8_t addr,uint8_t data)
13  {
14        data_addr = addr;
15        databyte = data;
16        Wire.beginTransmission(device_addr);
17        Wire.write(data_addr);
18        Wire.write(databyte);
19        Wire.endTransmission();
20        delay(5);
21  }
22
23  uint8_t MC24AA024 :: readDataByte( uint8_t addr)
24  {
25        data_addr = addr;
26        uint8_t readbyte=0;
27        int i = 0;
28        Wire.beginTransmission(device_addr);
29        Wire.write(data_addr);
30        Wire.endTransmission(false);
31        Wire.requestFrom(device_addr,1);
32
33        if(Wire.available())
34              {
35                          readbyte = Wire.read();// Read the data in the buffer
36              }
37
38        return readbyte;
39  }
40
```

A.3 Program code using MC24AA024 Class Object

The following Arduino program code may be used for testing read/write to MC24AA024 using the MC24AA024 Class Object.

```
1
2  /* MC24AA024 Test Program
3  Copyright 2025 Pronto Electronic Displays LLC
4  This program checks EEPROM functionality by allowing writes/reads to specific
   Addresses.
5  Serial port is used to verify read/write operation.
6  Uses default pins for SDA,SCL.
7  */
8
9  #include <Wire.h>
10 #include <MC24AA024.h>
11
12 // Instantiate MC24AA024 Class Object. Pass the address into the constructor:
13
14 MC24AA024 U1(0x57);// Address used depends on how A0,A1,A2 pins are wired
15
16 void setup()
17 {
18
19 Serial.begin(9600);
20 Wire.begin();
21 Wire.setClock(100000);
22
23 }
24
25 void loop()
26 {
27
28 uint8_t data_address = 163;
29 uint8_t written_data = 47;
30 U1.writeDataByte(data_address,written_data);// (address, data)
31 delay(5);
32 uint8_t read_data;
33 read_data = U1.readDataByte(data_address);
34 Serial.print("Address: ");
35 Serial.println(data_address);
36 Serial.print("Written data: ");
37 Serial.println(written_data);
38 Serial.print ("Read data: ");
39 Serial.println(read_data);
40 Serial.println();
41 delay (5000);
42
43 }
44
45
```

A.4 Program Code for PWM signal generation using Arduino

The following code may be used to generate PWM signal to the DSP004 unit using the Arduino Uno. Code is only for use with ATMEGA328 series microcontrollers.

```
/*
Copyright 2025 Pronto Electronic Displays LLC
PWM Program for dimming the DSP004 Display
This program will dim and brighten display in repeated cycles
*/

//************* BEGIN SETUP SECTION ***************

void setup()
{
   // initialize pins

   pinMode(11,OUTPUT);// Pin D11 used for PWM Dimming

// Set PWM parameters, 8kHz
 TCCR2B=0b00000010; // divide by 8 Prescaler
 TCCR2A=0b00000011; // Fast PWM mode for Timer 2

}
//############### END SETUP###############

void loop()
{

        // Begin PWM Dimming Loop

        for (int i = 255; i>0; i--)
           {
              analogWrite(11,i);
              delay(20);
           }
        for (int i = 0; i<256; i++)
           {
              analogWrite(11,i);
              delay(20);
           }

}
```