MONITORX

USER MANUAL

1. Introduction

MONITORX is a serial monitor program designed to monitor and interact with serial communication. This user manual provides detailed instructions on how to use the program effectively.

2. Getting Started

2.1 Starting the Program

- Execute the program and choose one of the predefined configurations or opt for customization.
- Select the appropriate option based on your requirements.
- If customizing, provide the necessary information such as port name, baud rate, logging preferences, etc.

2.2 Customizing Serial Connection

- If you choose the customization option, you will be prompted to enter details such as port name, baud rate, logging preferences, timestamp preferences, and maximum display lines.
- Follow the on-screen instructions to set up the serial connection according to your needs.

3. Program Features

3.1 Serial Monitoring

- MONITORX continuously monitors the specified serial port for incoming data.
- Received data is displayed on the screen, including optional timestamps.

3.2 Logging

- You can enable or disable logging of received data.
- If logging is enabled, the program will create a log file with the specified filename.

3.3 Timestamps

- Choose to include timestamps with each received data entry.
- Timestamps are displayed on the screen and can be logged if logging is enabled.

3.4 Batching

- The program batches incoming data and clears the screen after a specified number of lines are received.
- This helps in organizing and managing the displayed information.

3.5 Custom Commands

- The program recognizes custom commands embedded in the incoming data.
- Example: Sending "clearscr" clears the screen and starts a new batch.

For Aurdino ==> Serial.println("clearscr");

3.6 Changing Serial Colors

- The program supports changing serial colors using specific commands.
- Example: Sending "Serialcolora" changes the color to option 'a'.

Serialcolor0 = Black	Serialcolor8 = Gray
Serialcolor1 = Blue	Serialcolor9 = Light Blue
Serialcolor2 = Green	Serialcolora = Light Green
Serialcolor3 = Aqua	Serialcolorb = Light Aqua
Serialcolor4 = Red	Serialcolorc = Light Red
Serialcolor5 = Purple	Serialcolord = Light Purple
Serialcolor6 = Yellow	Serialcolore = Light Yellow
Serialcolor7 = White	Serialcolorf = Bright White

For Aurdino ==> Serial.println(“Serialcolora”);

3.7 Clearing the Screen

- Send the "clearscr" command to clear the screen and start a new batch.

3.8 Customizing Load Screen

- During the customization process, you can set various parameters to tailor the program to your specific needs.

3.9 Calibration(Here for LDR Projects)

The program supports calibration for CLDC (Calli Detection) values.Calibration allows you to set higher and lower calli values for more accurate monitoring.(Calli=Callibration)

Setting Calibration Values

- To set the higher calli value, send the command "ldrcallih[value]" where [value] is the desired higher calli value. Example: Sending "ldrcallih500" sets the higher calli value to 500.

For Aurdino ==> Serial.println(“ldrcallih500”);

- To set the lower calli value, send the command "ldrcallil[value]" where [value] is the desired lower calli value. Example: Sending "ldrcallil100" sets the lower calli value to 100.

For Aurdino ==> Serial.println(“ldrcallil100”);

Enabling Calibration

Calibration will be explicitly enabled by setting the "ldrenable" parameter to 1.

When using Monitorx for LDR projects, we can enable LDR Callibration Mode on the program.

For Aurdino ==> Serial.println(“ldrcallih[value]”);

Here, on using this command program automatically enables LDR Callibration mode and add the higher calli and lower calli value in header.

- For Aurdino ==> Serial.println(“ldrenable”);

  This enables LDR Callibration mode.
- For Aurdino ==> Serial.println(“ldrenable”);

  This disables enabled LDR Callibration mode.

Note: Calibration is an advanced feature, and incorrect calibration values may affect the accuracy of calli detection. Ensure precise calibration values based on your specific requirements.

4. Troubleshooting

- If you encounter issues with the program, ensure that the serial port is correctly configured and available.
- Check the specified log file for any error messages.

Note: Ensure that you have the necessary permissions to access the specified serial port.

For any additional queries, refer to the program documentation or contact the developer.