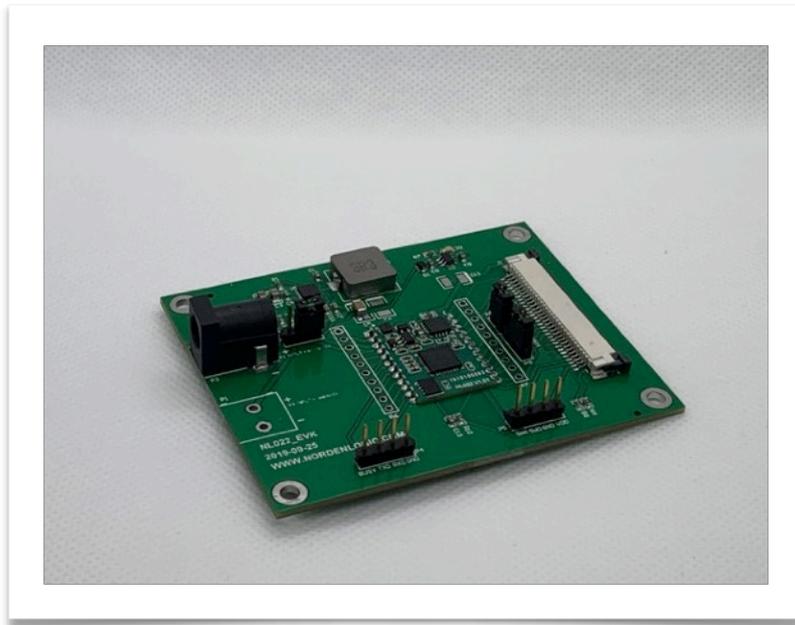


May 2021  
Doc: 1.0.2

This document describes how to use the Norden Logic evaluation board NL022\_EVK together with our Printerevk evaluation software and user supplied USB-to-TTL dongle.



Parts required to use the EVK board:

- Power supply between 7.2V and 12V supporting minimum of **3Amp** of current
- Serial Dongle PC: **USB-to-TTL** e.g. based on FTDI232 OR CP2102 **\*NOT RS232**
- Any of the following print heads:

**58mm:**

- Fujitsu FTP628MCL101
- Fujitsu FTP628MCL103
- Seiko LPTZ245B/D/...
- Seiko LTP01-245
- Samsung SMP685
- Samsung SMP695
- PRT PT486F
- ALPS PTMBL1B
- WinsPu TP2SX
- WinsPu TP2ZX

and compatible

## 80mm:

- Fujitsu FTP638MCL101
- Fujitsu FTP638MCL103
- PRT PT723F-B

and compatible

- **High quality** 58mm or 80mm thermal paper roll

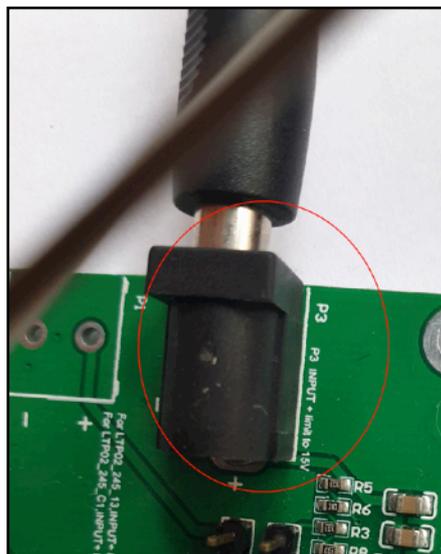
Our EVK board is designed to use 58mm and 80mm heads and a good stabilized power supply should be used. The type of head connected needs to be selected via the provided jumper bank and then set in flash in the controller via ESC/POS command or using our Printerevk Test software. **Care must be taken that the correct jumper setting is used!**

Power connectors:

There are 2 power input connectors on the EVK board. P3 and P1. P3 should be used with a regulated power supply in the range of 7.2V up to 12V with a minimum of 3Amp of current. The on-board IC manages the output voltage level for the print heads.

P3 polarity is centre positive: 

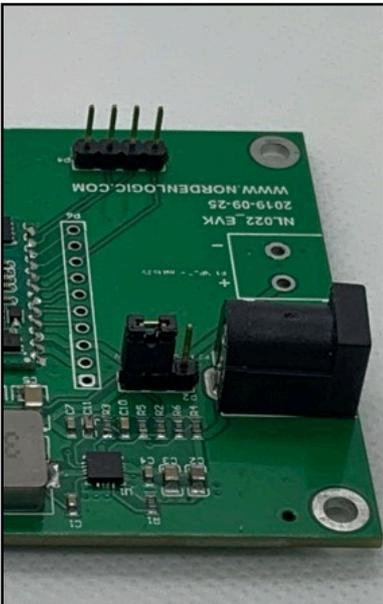
Power Port P3:



## Power Port P1:

Power port P1 allows for the developer to provide his own power circuitry and has not protection and is a direct power interface to the chip. Extreme care must be taken not to short circuit the board when using this power input port. In order to enable the P1 power port the jumper P2 must be placed onto the first 2 pins close to the P3 power connector. By default the board is configured to use P3 power port.

The image below shows the P2 jumper bank as configured for P3 power input (default):

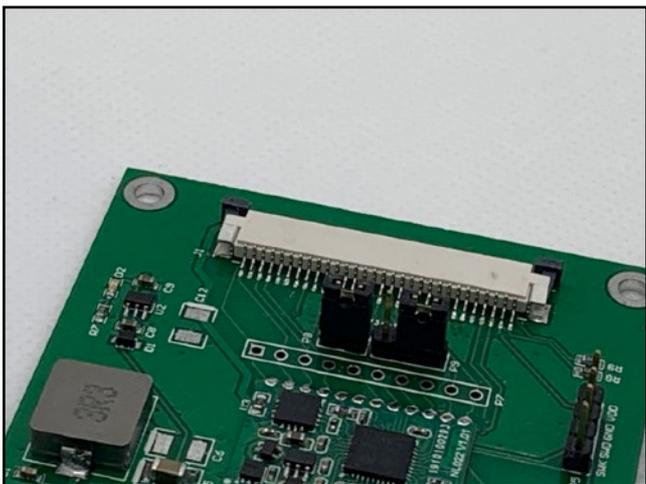


The image below shows the P1 power connection pads:

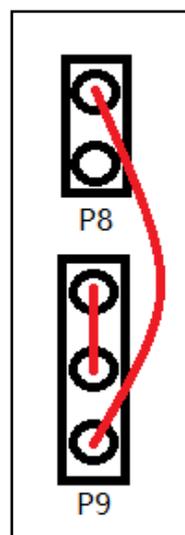
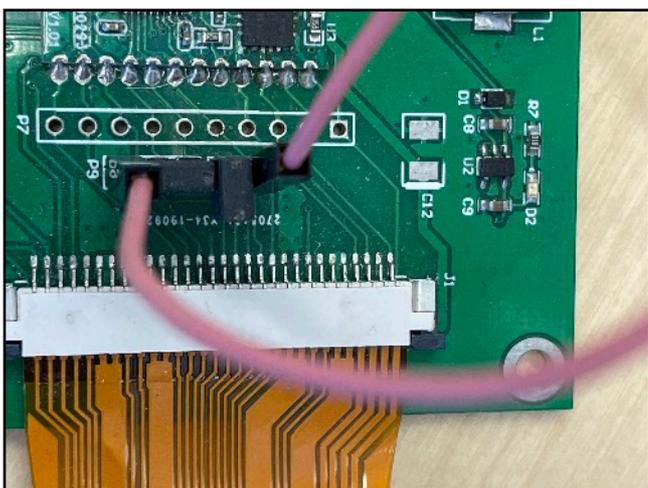


The NL022\_EVK board is by default configured to be used with 58mm print heads (see support print head types at the beginning of this document). In order to configure the EVK board for supported 80mm print head types the jumper P8 and P9 must be configured.

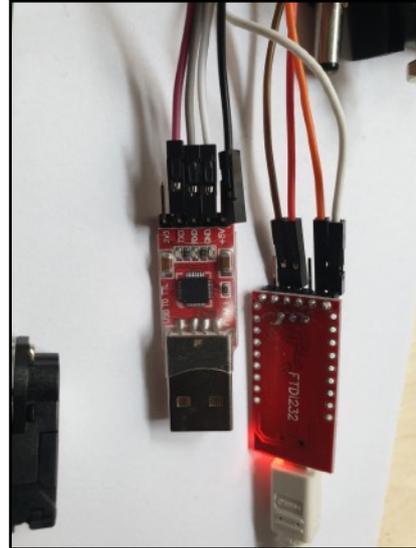
The image below shows the jumper settings for supported 58mm print heads. P8 is closed and P9 connects pins 2 and 3:



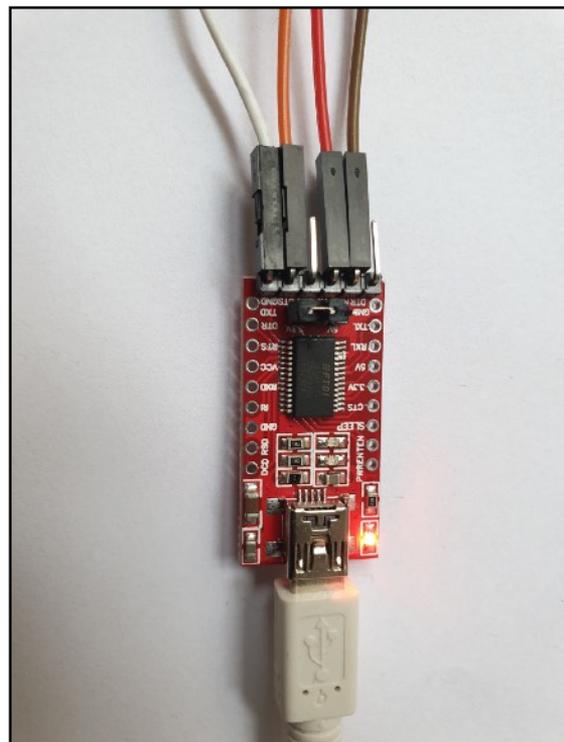
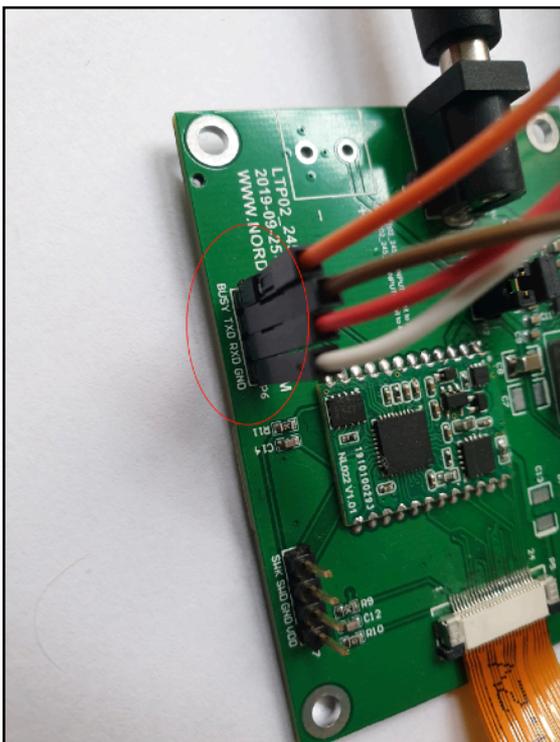
In order to configure the board for supported 80mm print heads P8 must be open and P9 connect pin 1 and 2 and a jumper wire must be used from P9 - pin 3 to P8 - pin 1 as in the image below:



In order to communicate with a PC a user supplied Serial USB-to-TTL dongle must be provided and wired accordingly to the EVK communication pins. Below are 2 examples of the most common USB-to-TTL dongles on the market. **Important: DO NOT USE RS232 DONGLES!**

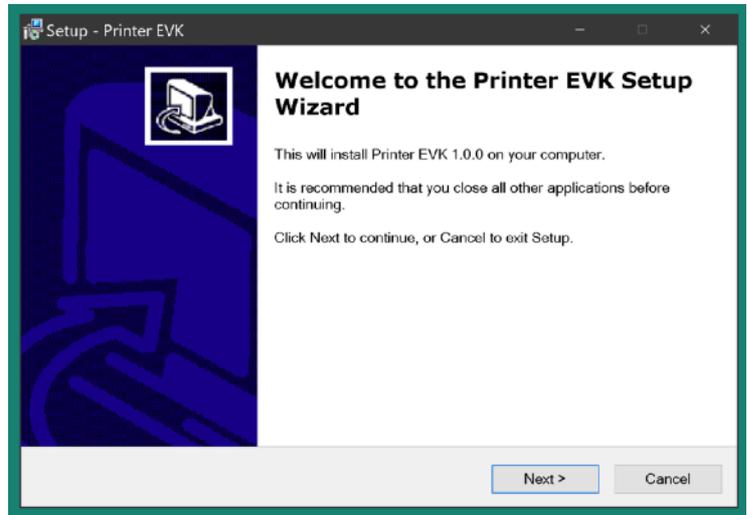


Please connect the TX, RX, GND and RTS wires from the dongle to the P6 pin bank. Important to note is that the dongle TX goes to the RX of the EVK board, the dongle RX goes to the TX of the EVK board. GND goes to the same on both. If your board supports RTS/CTS connect the EVK RTS line to the pin you will read the line to be set high or low for hardware flow control. In case your dongle has only TX, RX, GND that is no problem because NL022 has by default XON/XOFF software flow control enabled for preventing data loss when printing.

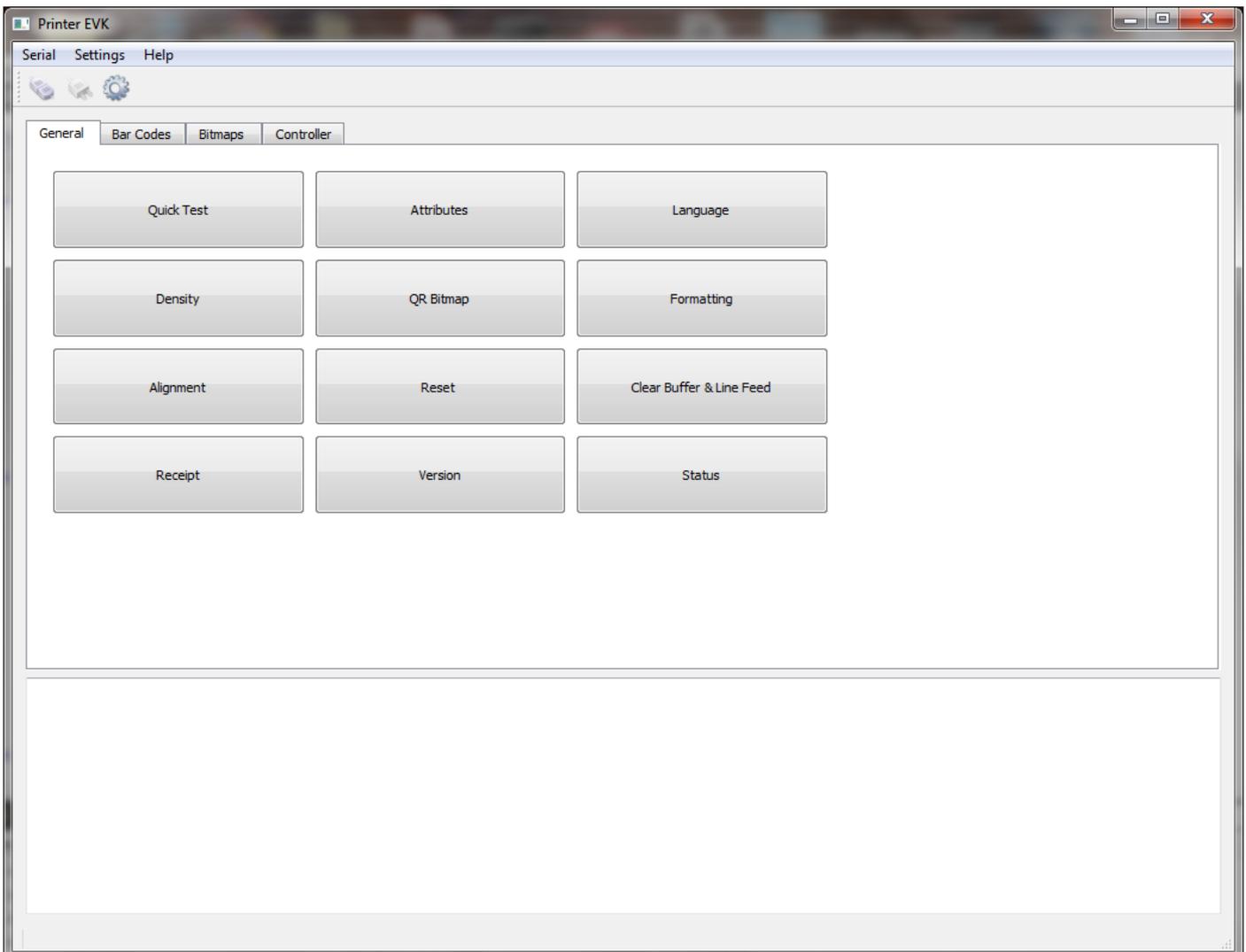


After everything is connected and set up on the hardware side - please still add paper to the print head, then the evaluation software Printerevk should be installed on a Windows PC (a Linux and Mac version available on request).

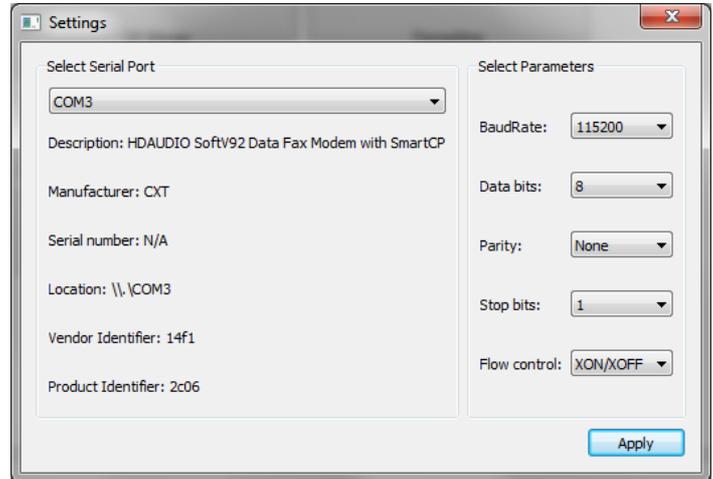
Printerevk comes with a Windows Setup program which installs all components needed for testing. Just follow the wizard for installation.



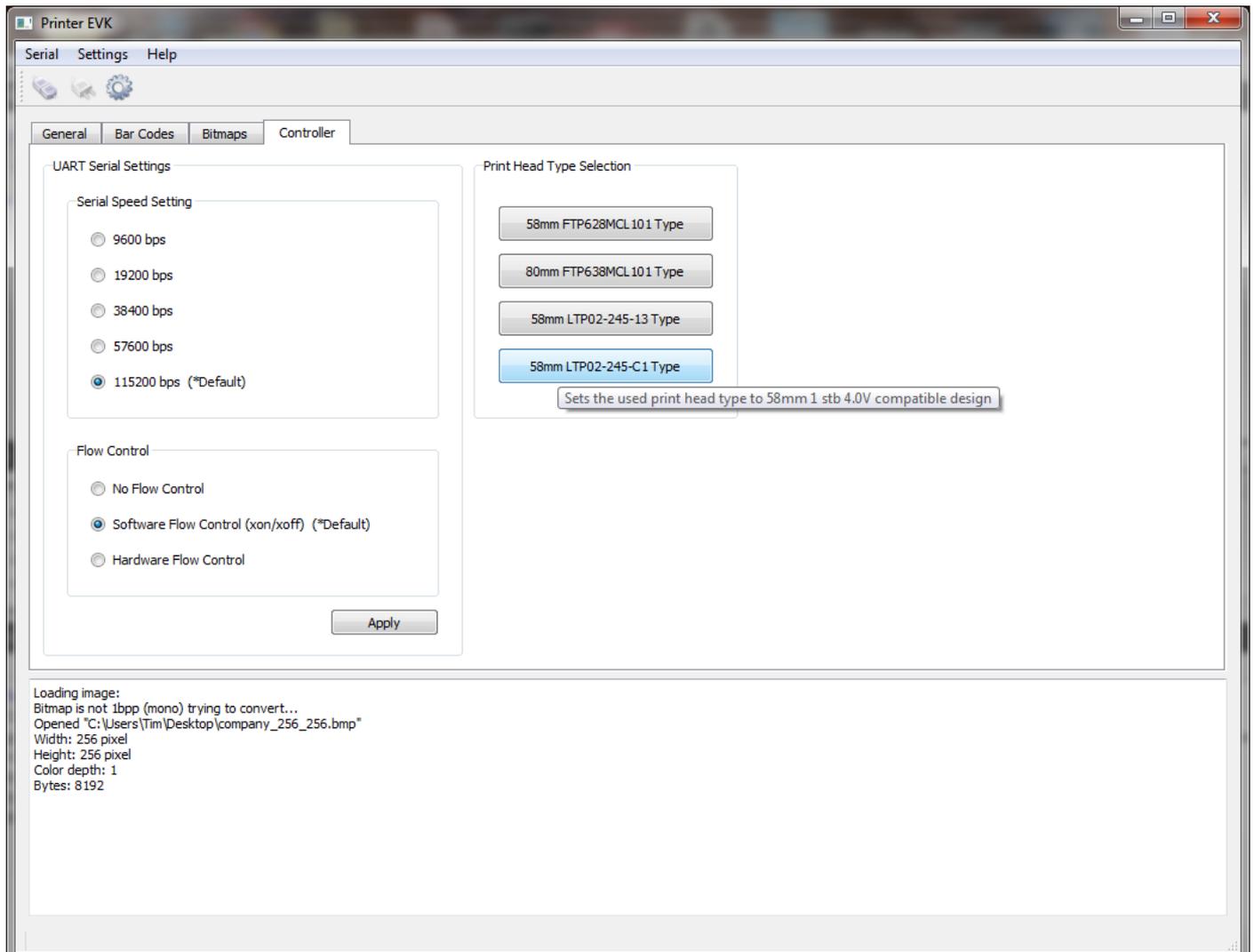
Make sure you have installed the drivers for your Serial USB-to-TTL dongle before starting to test the EVK board and you are confident the dongle is operational.



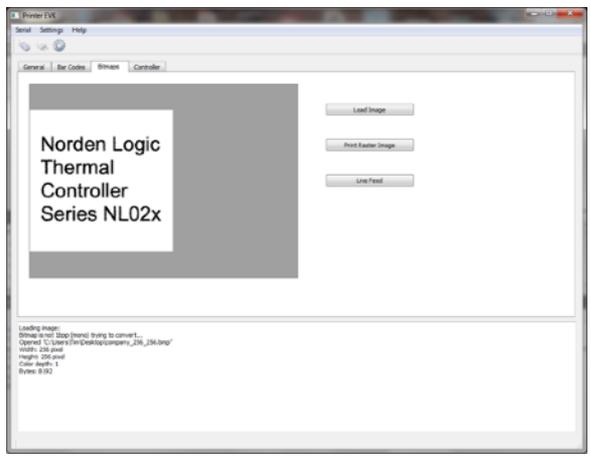
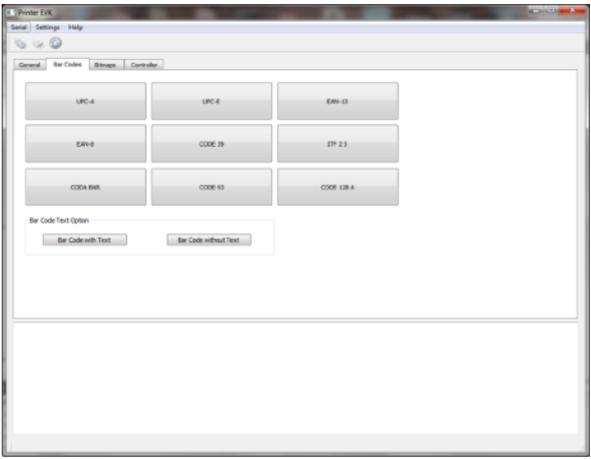
Printerevk comes with UART serial settings preset as a default, 115200bps and xon/xoff enabled. You can review these settings but all you need to do is click apply and go and connect to the EVK board using the connect option in the menu or toolbar.



Once you are connected the very first thing you should do regardless if the board has been configured before is just to set the EVK again to the print head you have connected. You find this option in the Controller tab.



After that you are ready to run all the tests available within the software. Below examples of the screens:



Helpful together with the software tool is also having a look in the ESC/Pos documentation and/or the provided ready to use and integrated C source library files:

Norden Logic

## 2. Command Reference

### 2.1 ESC/POS Command Overview

ASCII	Hex format	Explanation
LF	0A	Print and feed paper
CR	0D	Carriage return
ESC J	1B 4A n	Print and feed paper n dots
ESC d	1B 64 n	Print and feed paper n lines
ESC 3	1B 33 n	Set line spacing to n dots
ESC 2	1B 32	Set line spacing to default values
ESC l	1B 6C n	Set left margin
ESC Q	1B 51 n	Set right margin
ESC \$	1B 24 nL nH	Set absolute print position
ESC !	1B 21 n	Set character printing mode
ESC a	1B 61 n	Set print alignment
ESC m	1B 6D n	Set font grayscale
FS &	1C 26	Select double-byte character mode
FS .	1C 2E	Cancel double-byte character mode
International 8-bit and 16-bit Unicode		Printing International character sets
DoubleByte Character Sets		Printing DoubleByte character sets

```

11 //
12 //
13 #include <stdio.h>
14 #include <stdlib.h>
15 #include <string.h>
16 #include <stdint.h>
17 #include <fcntl.h>
18 #include <unistd.h>
19 //
20 #include "nl02x.h"
21 //
22 //
23 //
24 // *****
25 // Write wrapper
26 // *****
27 static void Write(const uint8_t * buf, size_t buflen)
28 {
29     //Implement here your own function body for writing to
30     //your serial port - UART
31 }
32 //
33 // *****
34 // Set serial communications parameters
35 #define 9600bps n=1 19200bps n=2 38400bps n=3 57600bps n=4 115200bps
36 #define No Flow Control n=1 Data n=2 Software Flow Control (xon/xoff)
37 #define Hardware Flow Control RTS/CTS n=1 Host Stop Send Data RTS/busy=0 Host Send Data
38 //
39 // Default is n=4 115200 n=1 xon/xoff
40 // *****
41 void setComMode(unsigned char ucComSet, unsigned char ucFlowSet)
42 {
43     uint8_t buf[4] = {0x1D, 0x28, 0x40, 0x00}; //UGS (n m)
44     buf[2] = ucComSet; // 0, 1, 2, 3, 4
45     buf[3] = ucFlowSet; // 0, 1, 2
46     Write(buf, 4);
47 }
48 //
49 //
50 //
51 // Set corrected Print Head. Option is viewed into flash
52 #define MMSB_FTTP68MCL101 58mm 6 pin 7.2V compatible designs *default value
53 #define MMSB_FTTP68MCL101 60mm 5pin 7.2V compatible designs
54 #define MMSB_LTP02_245_13 58mm 1pin line 7.2V designs
55 #define MMSB_LTP02_245_C1 58mm 1pin line 4.0V designs
56 //
57 void setPrinterHead(unsigned char ucHeadIndex)
58 {
59     unsigned char buf[3] = {0x1B, 0x26, 0x00}; //ESC Z n
60     buf[2] = ucHeadIndex;
61     Write(buf, 3);
62 }
63 //
64 //
65 // *****
66 // Printer ESC/POS init command
67 // *****

```