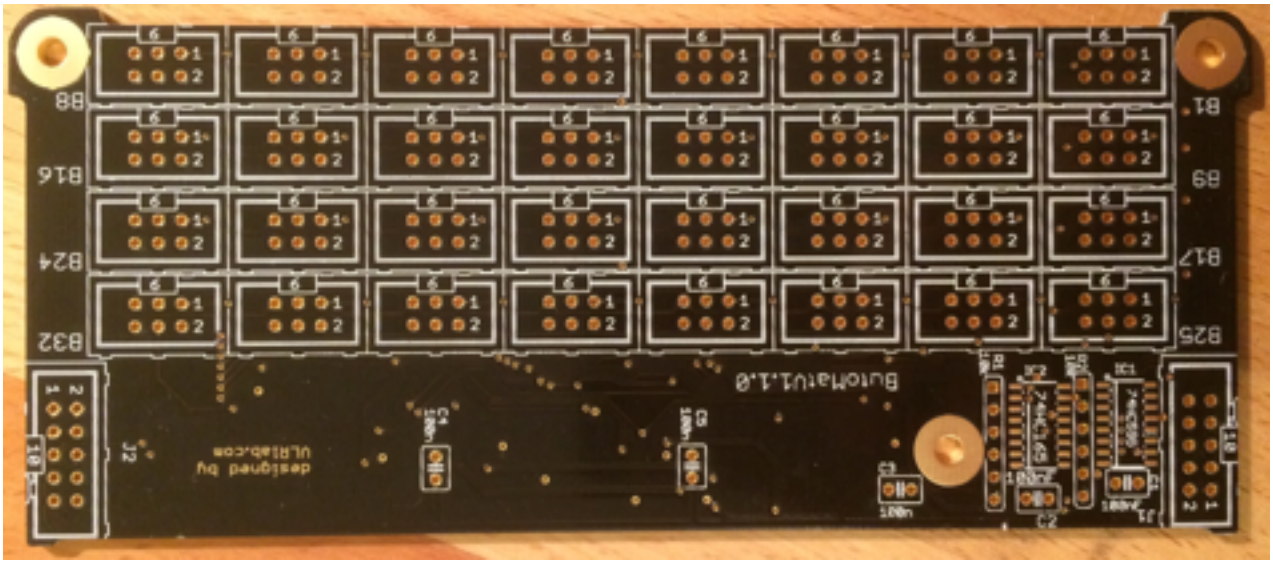


BUTOMAT — USERMANUAL

VERSION: 0.01

DISCLAIMER

Information contained in this manual is believed to be accurate and reliable. However, VLRlab assumes no responsibility for the use thereof nor for the rights of third parties, which may be affected in any way by the use thereof. Any representations in this document concerning performance of VLRlab products are for informational use only and are not warranties of future performance, either expressed or implied. As this product is for hobbyists and do it yourself builders there is no warranty upon the functionality of the Parts. The only warranty offered by VLRlab in relation to this product is the VLRlab standard limited warranty, stated in the sales contract or order confirmation form relaying to the sold part it self.
Although every attempt has been made to accurately describe the features, installation and operation of this product in this manual, no warranty is granted nor liability assumed in relation to any errors or omissions unless specifically undertaken in the VLRlab sales contract or order confirmation. Information contained in this manual is periodically updated and changes will be incorporated into subsequent editions. If you encounter an error, please notify VLRlab by e-mail. VLRlab reserves the right, without notice or liability, to make changes in equipment design or specifications. (contact@vlrlab.com)

LICENCE

Radical Sharing!

Even though we share all our designs freely with you, there are some ground rules you need to follow. Our software is GNU GPL and everything else unless otherwise noted is Creative Commons BY-SA licensed. This gives you both rights and obligations.

What are the license terms of the VLRlab designs?
Our software is released under GPL. Please read the license. http://www.gnu.org/licenses/gpl-3.0.de.html
Everything else (such as schematics, PCB designs, enclosure designs, spreadsheets etc. available for download) unless otherwise noticed is released under Creative Commons BY-SA. Please read http://creativecommons.org/licenses/by-sa/3.0/

How may I use your designs as open source?
Well, the licenses above defines this, but said shortly, you may copy, change and (re)distribute our work as long as you give us credit for the original creation and pass on the work under the same license terms. This effectively means (among other things) that you cannot adapt and redistribute our work without letting the recipient (for instance a customer) know that it's based on open source under the license terms given, which effectively means that there is no way to legally make a "closed product" out of any of it, including your own adaptations.

Can I use your software libraries in my own closed products?
No. A consequence of GPL is that if you use any of our libraries or code snippets from them, your software must be GPL as well. And GPL means that your client have to receive access to the source code for any compiled delivery such as a pre-programmed Arduino.

How should I attribute you in derivative works?
The CC-by-sa license says that you must attribute the work in a manner specified by the licensor (us), but in a way that doesn't suggest that we endorse your work. We specifically ask the following:
For enclosures, you must remove our logo from the case, and you must write on a visible place outside (on the back for instance) "Based on work from vlrlab.com under the CC-by-sa license"
For PCBs, you must remove „designed by VLRlab.com“ from the PCB and you must place "Based on work from VLRlab.com under the CC-by-sa license" on the silkscreen.
For documents like schematics, manuals, spreadsheets etc, you must include the text "Based on work from VLRlab.com under the CC-by-sa license" somewhere visibly in the document.

How should I attribute you if I use your designs inside my own designs?
This may sound like the previous question, but lets say you fully design your own enclosures for our hardware designs. Or if you develop your own hardware boards and combine with ours. In other words: if a lot of completely original work by you is mixed with works from us. In this case we suggest that you attribute us by stating "Includes work from vlrlab.com under the CC-by-sa and GPL licenses" on a visible place for the client. A visible place would be in manuals and on the back of an enclosure containing our designs.

But then, how can I make money?
Sell services. Sell your knowledge and customization services. Sell 24-7 support. Sell documentation. Sell warranty. We do some of that! It's not possible to charge money for open sourced intellectual property itself, because you are obliged to offer that part to your client for free.

Can I charge a client for an adaptation of your work?
Yes, that would be a service you do for that client. However, the adaptation itself (software, hardware design etc) would have to be licensed under the same terms as the original (GPL or CC-by-sa) and therefore the next client in line (and the public) should have it for free. And of course, in the spirit of sharing you would have made it public available already somehow.

How can I sneak around your license terms?
Hopefully you can't, but in reality there are probably many ways to "get away" with it. But we urge you to consider your deeper motives and attitudes. Although it can be very challenging, give the spirit of sharing a chance and play by the rules. Don't be like the Dead Sea that always accepts but never gives.

Repositories
All our freely available information is found at two locations:
- Various repositories on GitHub under the username "novski"
- The Manuals on vlrlab.com/support

Read more at http://vlrlab.com/about/licenses/

INDEX

PART	VALUE	DEVICE	PACKAGE	LIBRARY	SHEET
B1-32		ML6	ML6	HARTING	1
C1	100nF	C-EU025-030X050	C025-030X050	CAPACITOR, European symbol	1
C2	100nF	C-EU025-030X050	C025-030X050	CAPACITOR, European symbol	1
C3	100n	C-EU025-030X050	C025-030X050	CAPACITOR, European symbol	1
C4	100n	C-EU025-030X050	C025-030X050	CAPACITOR, European symbol	1
C5	100n	C-EU025-030X050	C025-030X050	CAPACITOR, European symbol	1
D1	FMP1T148	FMP1T148	SC-74A		1
D2	FMP1T148	FMP1T148	SC-74A		1
D3	FMP1T148	FMP1T148	SC-74A		1
D4	FMP1T148	FMP1T148	SC-74A		1
D5	FMP1T148	FMP1T148	SC-74A		1
D6	FMP1T148	FMP1T148	SC-74A		1
D7	FMP1T148	FMP1T148	SC-74A		1
D8	FMP1T148	FMP1T148	SC-74A		1
IC1	74HC595	74HC595SO16D	SO16D		1
IC2	74HC165	74HC165NSO16D	SO16D		1
IC3	74HC595	74HC595SO16D	SO16D		1
IC4	74HC595	74HC595SO16D	SO16D		1
IC5	74HC595	74HC595SO16D	SO16D		1
J1	IN	ML10	ML10		1
J2	OUT	ML10	ML10		1
R1	10k	4306R	SIL6		1
R2	10k	4306R	SIL6		1
R3	220R	4816P-1SOIC16W	SOIC16W		1
R4	220R	4816P-1SOIC16W	SOIC16W		1
R5	220R	4816P-1SOIC16W	SOIC16W		1

1. 32 RGB Button Matrix

1.1. Connections

J1 - SPI connection to CORE
J2 - SPI connection to next Module
B1-32 Button Connectors

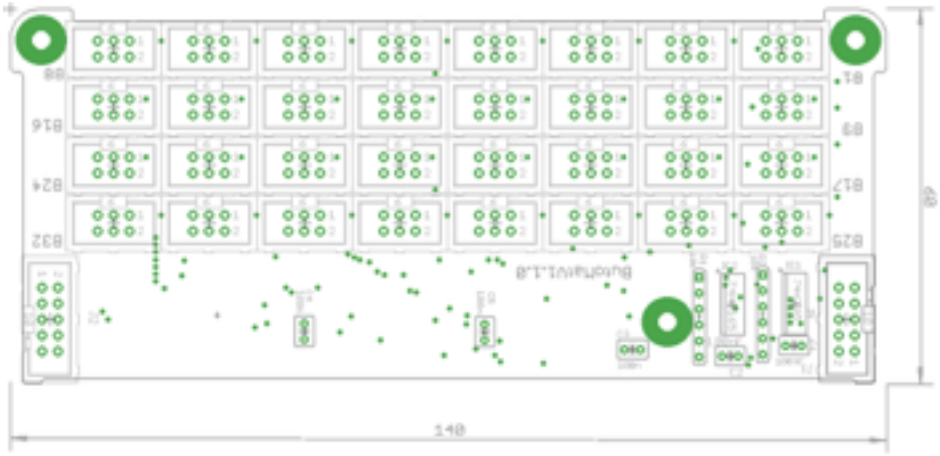
1.2. Electrical Specification

Supply Voltage: 5V
Power-consumption: ???

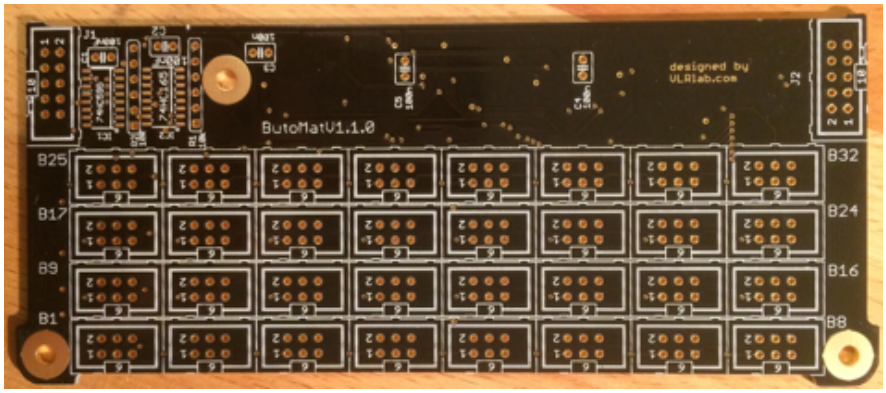
To simplify the soldering i engineered this PCB for SMD Diodes with common cathode.
The Diodes are not easy to solder so be sure you have enough skill to solder them.

2. BOM

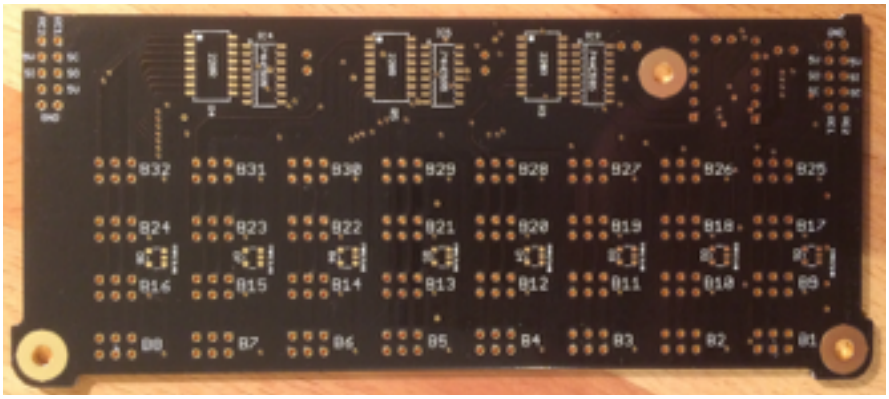
3. OUTLINES



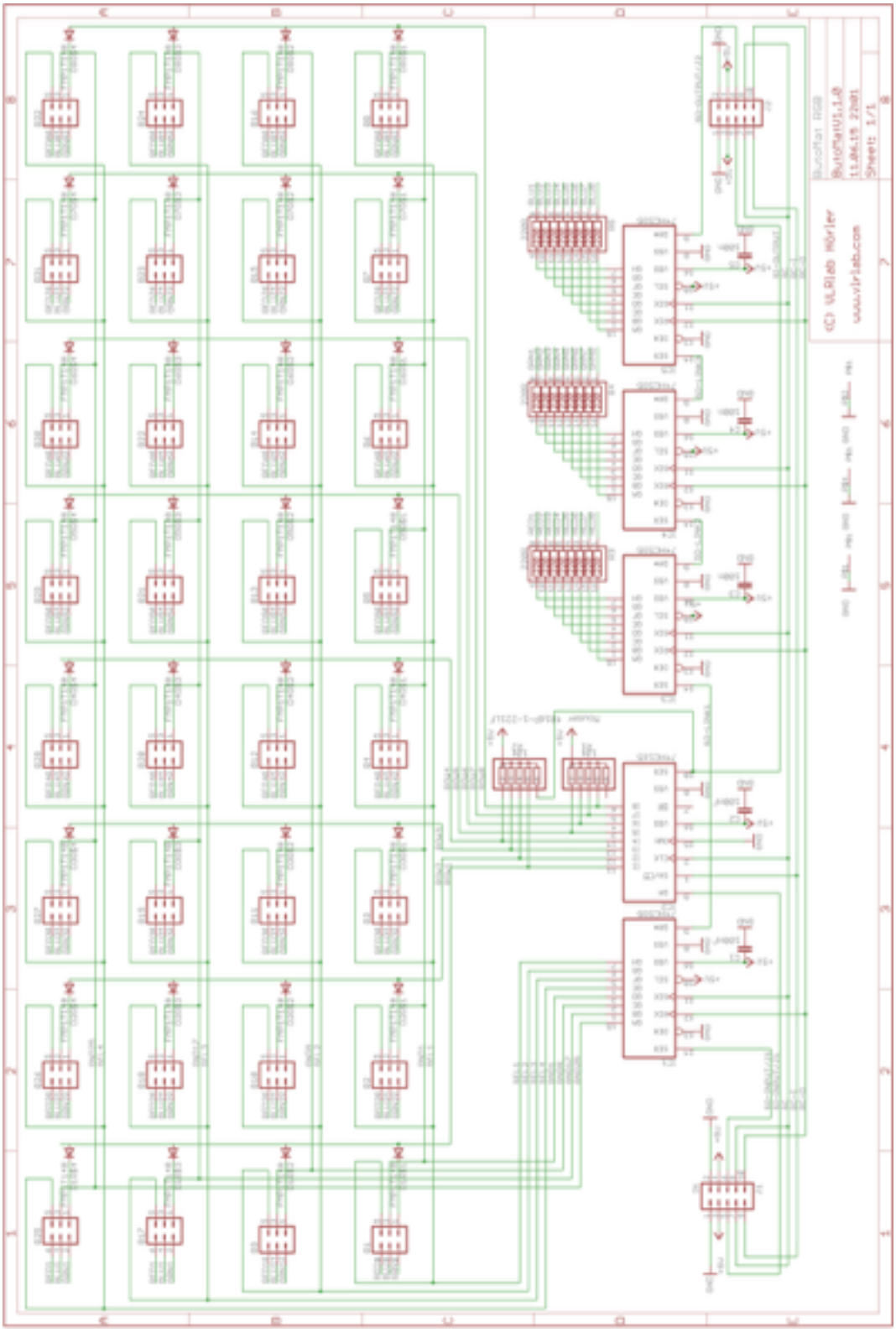
4. FRONTVIEW



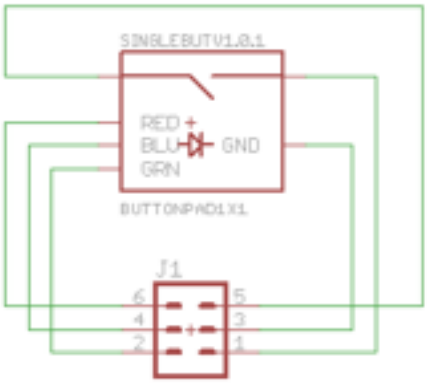
5. REARVIEW



6. Schematics



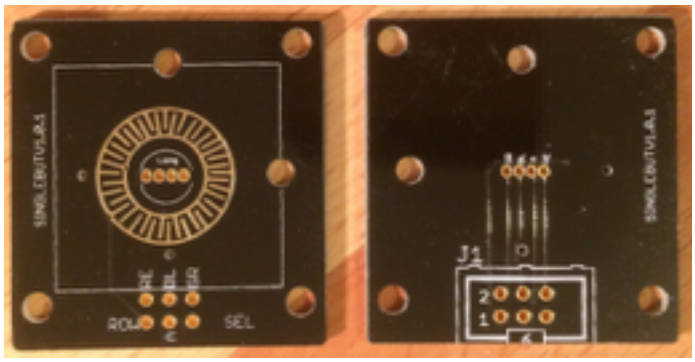
7. Button-Pinout

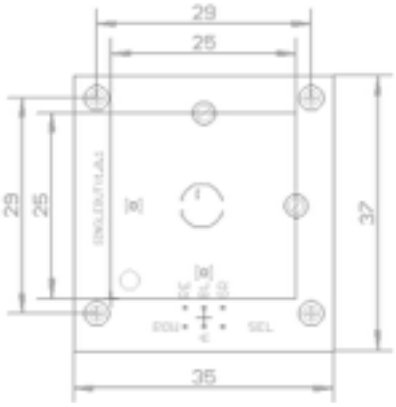


Its possible to connect any Button. The LED must be with a common Kathode.
Of corse i sell a variant. Its based on the cheapest version i cold find. The sparkfun Rubber button. Its Button and cap at the same time what makes it cheap and easy to get it straight trough your Frontpanel without any additional parts…

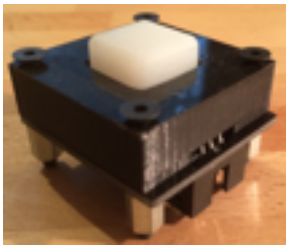
8. But

A PCB for a single Rubberbuton.





To make the Rubberbutton not slip away and get the right distance between your Frontpanel and the Button I decided to create a downloadable 3d Design of a bezel that can be printed with almost any 3d printer.
My prototype lookes like that:



9. Config

1. MIOS

```
RESET_HW

LCD "%C"
LCD "@(1:1:1)UP-and-OK!"

LCD „@(1:1:1) RGB Demo"

DIN_MATRIX  n=1  rows=4  sr_dout_sel1=1  sr_din1=1                  button_emu_id_offset=1001
DOUT_MATRIX n=1  rows=4  sr_dout_r1=2    sr_dout_g1=3 sr_dout_b1=4     led_emu_id_offset=1025

# These button functions forward their value also to LEDs
# In this demo we configure individual brightness levels for the LEDs from EVENT_BUTTON events by rising the
red:green:blue=15:0:0 value 0-15

# This is Row0 - rising brightness from button 1001-1008
EVENT_BUTTON  id=1001  fwd_id=LED:1025  type=NoteOn key=36 chn=1  rgb=15:0:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1002  fwd_id=LED:1026  type=NoteOn key=37 chn=1  rgb=3:0:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1003  fwd_id=LED:1027  type=NoteOn key=38 chn=1  rgb=5:0:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1004  fwd_id=LED:1028  type=NoteOn key=39 chn=1  rgb=7:0:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1005  fwd_id=LED:1029  type=NoteOn key=40 chn=1  rgb=9:0:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1006  fwd_id=LED:1030  type=NoteOn key=41 chn=1  rgb=11:0:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1007  fwd_id=LED:1031  type=NoteOn key=42 chn=1  rgb=13:0:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1008  fwd_id=LED:1032  type=NoteOn key=43 chn=1  rgb=15:0:0 lcd_pos=1:1:2 label="^std_btn"
# This is Row1
EVENT_BUTTON  id=1009  fwd_id=LED:1033  type=NoteOn key=52 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1010  fwd_id=LED:1034  type=NoteOn key=53 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1011  fwd_id=LED:1035  type=NoteOn key=54 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1012  fwd_id=LED:1036  type=NoteOn key=55 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1013  fwd_id=LED:1037  type=NoteOn key=56 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1014  fwd_id=LED:1038  type=NoteOn key=57 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1015  fwd_id=LED:1039  type=NoteOn key=58 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1016  fwd_id=LED:1040  type=NoteOn key=59 chn=1  rgb=0:15:0 lcd_pos=1:1:2 label="^std_btn"
# This is Row2
EVENT_BUTTON  id=1017  fwd_id=LED:1041  type=NoteOn key=68 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1018  fwd_id=LED:1042  type=NoteOn key=69 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1019  fwd_id=LED:1043  type=NoteOn key=70 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1020  fwd_id=LED:1044  type=NoteOn key=71 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1021  fwd_id=LED:1045  type=NoteOn key=72 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1022  fwd_id=LED:1046  type=NoteOn key=73 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
```

```
EVENT_BUTTON  id=1023  fwd_id=LED:1047  type=NoteOn key=74 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1024  fwd_id=LED:1048  type=NoteOn key=75 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
# This is Row3
EVENT_BUTTON  id=1025  fwd_id=LED:1049  type=NoteOn key=68 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1026  fwd_id=LED:1050  type=NoteOn key=69 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1027  fwd_id=LED:1051  type=NoteOn key=70 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1028  fwd_id=LED:1052  type=NoteOn key=71 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1029  fwd_id=LED:1053  type=NoteOn key=72 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1030  fwd_id=LED:1054  type=NoteOn key=73 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1031  fwd_id=LED:1055  type=NoteOn key=74 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
EVENT_BUTTON  id=1032  fwd_id=LED:1056  type=NoteOn key=75 chn=1  rgb=0:0:15 lcd_pos=1:1:2 label="^std_btn"
```