

Mimic panel – uses JMRI LogixNG to throw turnouts and set LEDs
turnouts connected via a different DCC++EX command station or CMRI, etc.

This was done using MCP23017 module.

5 momentary push buttons
6 Red / 6 Green LEDs

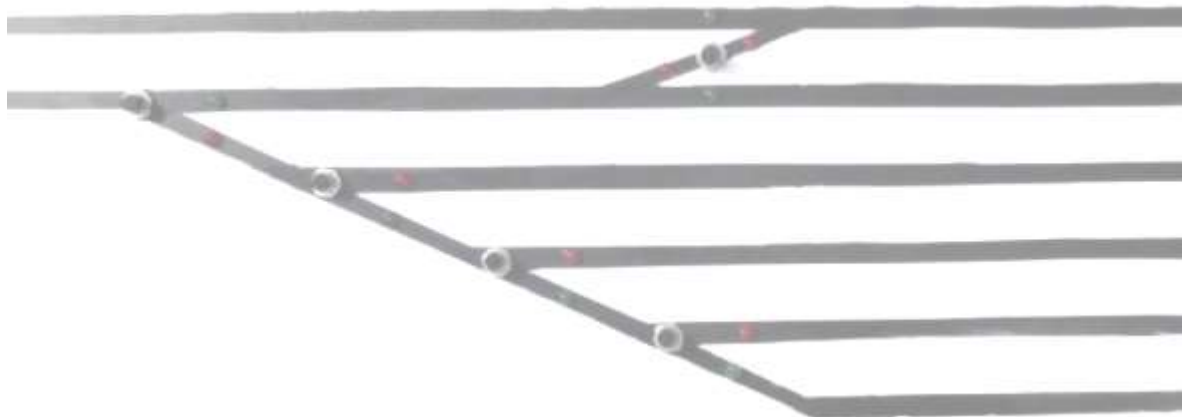
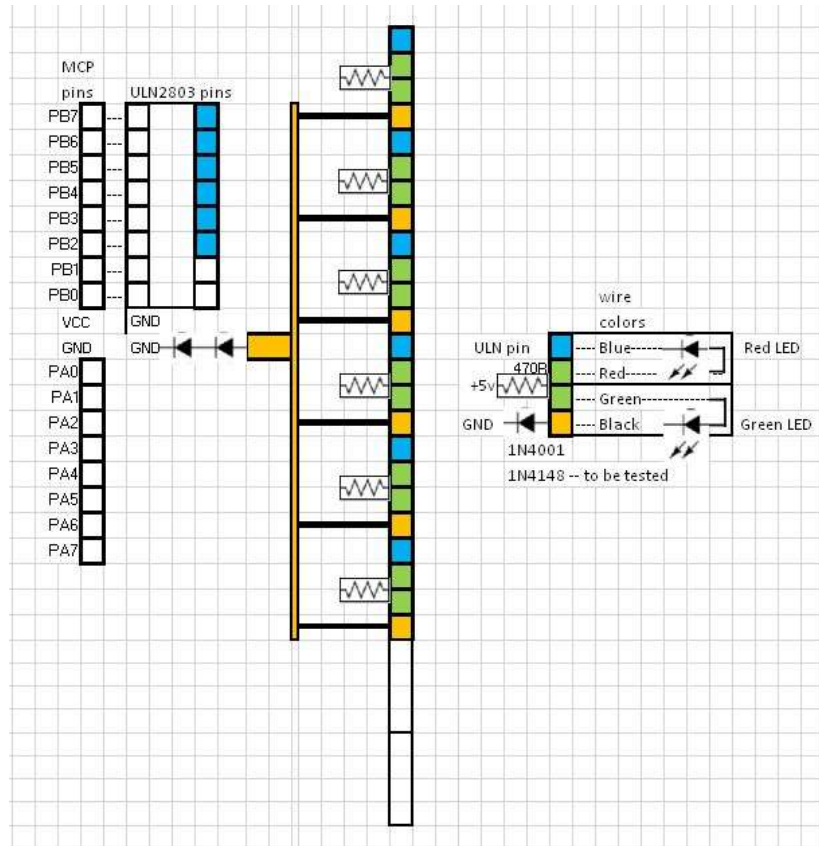
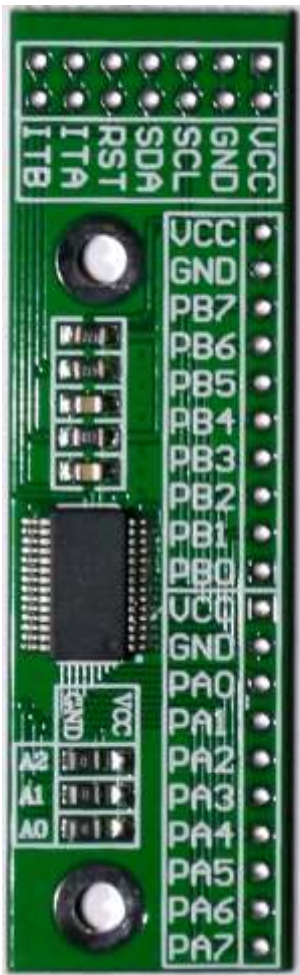
mySetup.h

```
// first MCP23017 module – pins 164-179
// 6 Red LEDs on pins PB2-PB7
SETUP("<Z 174 174 0>");
SETUP("<Z 175 175 0>");
SETUP("<Z 176 176 0>");
SETUP("<Z 177 177 0>");
SETUP("<Z 178 178 0>");
SETUP("<Z 179 179 0>");
// 5 buttons on PA3-PA7
SETUP("<S 167 167 1>");
SETUP("<S 168 168 1>");
SETUP("<S 169 169 1>");
SETUP("<S 170 170 1>");
SETUP("<S 171 171 1>");
```

JMRI obtains accessory definitions at startup

```
<* PARSING:T *>
<X>
<* PARSING:S *>
<Q 171 171 1>
<Q 170 170 1>
<Q 169 169 1>
<Q 168 168 1>
<Q 167 167 1>
<* PARSING:Z *>
<Y 174 174 0 0>
<Y 175 175 0 0>
<Y 176 176 0 0>
<Y 177 177 0 0>
<Y 178 178 0 0>
<Y 179 179 0 0>
```

Two diodes in series with negative side of Green LED
 – Green LED is on until Red LED is switched on
 ULN2803 provides switching to GND



Connection D: DCC++EX - Command Station connected via USB

Connection D2: DCC++EX connected via WiFi as Accessory Controller only

Sensor D2S167 is a momentary push button; D2T174 is the output LED

PanelPro 4.25.10ish+mstev+20220110T0109Z+R12f72b4a82, part of the JMRI® project
<http://jmri.org/PanelPro>

Active Profile: test4

DCC++: using DCC++ Serial Port on COM4

DCC++2: using DCC++ Ethernet on 192.168.1.42:2560

Java version 11.0.13 (en)

The screenshot shows the JMRI ConditionalNG editor interface. At the top, the system name is 'IQ:ALTO:0001' and the user name is 'Station 1'. Below this, a table lists the ConditionalNGs in order of calculation:

System Name	User Name	Thread	
IQ:ALTO:0001		LogixNG thread	Edit
IQ:ALTO:0002		LogixNG thread	Edit

Two windows are open showing the logic for these ConditionalNGs:

- Edit ConditionalNG IQ:ALTO:0001:** Logic: **If** Sensor D2S167 is Active **Then** Set turnout DT34 to state Toggle **Else** (empty).
- Edit ConditionalNG IQ:ALTO:0002:** Logic: **If** Turnout DT34 is Closed (-) **Then** Set turnout D2T174 to state Closed (-) **Else** Set turnout D2T174 to state Thrown (+).

The screenshot shows the DCC++ Traffic Monitor interface. At the top, there are tabs for 'All', 'DCC++', 'DCC++2', and 'Internal'. Below the tabs is a table showing the state of various turnout components:

System Name	User Name	State	Comment
DT34		Thrown	
DT35		Thrown	
DT180		Thrown	
D2T174		Thrown	
D2T175		Thrown	
D2T176		Thrown	
D2T177		Thrown	
D2T178		Thrown	
D2T179		Thrown	

Below the table are two windows showing traffic logs:

- DCC++ Traffic Monitor (D):** Shows a log of DCC++ traffic, including status commands and output commands for various turnout components.
- DCC++ Traffic Monitor (D2):** Shows a log of DCC++ traffic, including status commands and sensor replies for various turnout components.