**RMCQ HO LAYOUT PROGRAMMING CONVENTIONS**

PLAYING AROUND

Listed below are the standards and conventions that should be used for the naming and programming of the control panels on the Upper and Lower decks of the club permanent HO layout.

These standards and conventions are not set in stone and variations can be made if an individual situation dictate that the standards in this document cannot be achieved without major changes to the specific situation.

**Definitions and Conventions**

1. **Layout Decks:**
   1. The two layout decks will be nominated as Upper and Lower.
2. **Control Panels:**
   1. **Identification**: Control Panels will be identified using an alpha/numeric identification using the criteria in the table. Refer to the SCARM drawing for panel locations and further detail.

|  |  |  |
| --- | --- | --- |
| **Layout Area** | **Track** | **Panel Identification** |
| Lower Deck | 16.5mm | LHA, LHB etc. |
| Lower Deck | 12mm | LNA, LNB etc. |
| Upper Deck | 16.5mm | UHA, UHB etc. |
| Upper Deck | 12mm | UNA, UNB etc. |
| Upper Deck | 9mm | U9A, U9B etc. |

1. **Turnouts (points):**
   1. **Terminology:** Turnout will be used instead of ‘points’ as the whole unit is a turnout i.e. a change of track direction whereas the term point is a part of the turnout.
   2. **Identification:** on the Lower deck each turnout is allocated a unique three-digit number. For the Upper deck, each turnout is allocated a three-digit number and where the turnouts operate in tandem I.e. crossovers both turnouts will receive the same number with a letter suffix i.e. 221A 221B. This numbering is not necessarily sequence throughout the entire layout but is sequential within the parent Panel or panel area of control. This allows for expansion of the panel area of control if changes are incorporated.
   3. **Direction of travel:** is allocated a single alpha character of ‘C’ for CLOSED (turnout set straight) or ‘T’ for THROWN when in the NOT straight position. The direction is prefixed with the turnout number to give a unique turnout state. Expressed as ‘141C’, 141T’, etc.
   4. **Turnout Switches:** There are two switches mounted on the servo mounts**,** in most cases one switch is used for indication and the other for other functions as deemed necessary. The switch that is used for indication is the one that has pressure on the switch operating arm when the turnout is in the THROWN position. In this position, the COM and NO contacts in the switch will be made. There is no specific convention for the other switch on the turnout servo mount.
   5. **Turnout Control Connector:** is used to control the servo. Pin 1 is the control wire, generally yellow or orange or white depending on servo manufacturer. The number two pin (centre) is +5VDC and is generally red. Pin 3 is ground.



Figure 1 - Servo Standard Signal Configuration

* 1. **Turnout Indication Connector:** is used to connect the Servo Mounting Module switch to the bus. Pin 1 is connected to the micro switch common (‘C’). Pin 2 is connected to the micro switch Normally Closed (‘NC’). Pin 3 is connected to the micro switch Normally Open (‘NO’). These read when the switch lever is not activated (depressed).



Figure 2 - Servo Feedback Micro Switch Signal Configuration

1. **Servo’s**
   1. **Servo Mounting Modules:** are used to support the turnout control servo and two switches. These are mounted, in most case directly below the turnout change slider and connected to the turnout and the two switches. These are labelled ‘Turnout ‘followed by the turnout number. Expressed as ‘Turnout 100’, ‘Turnout 101’, etc.

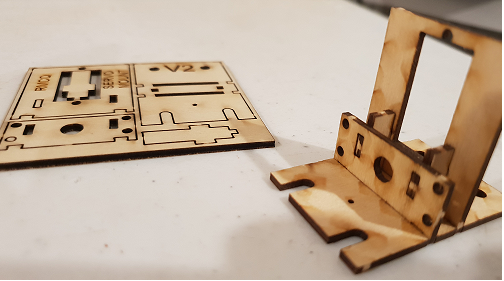


Figure 3 - RMCQ Servo Mounting Bracket Kit

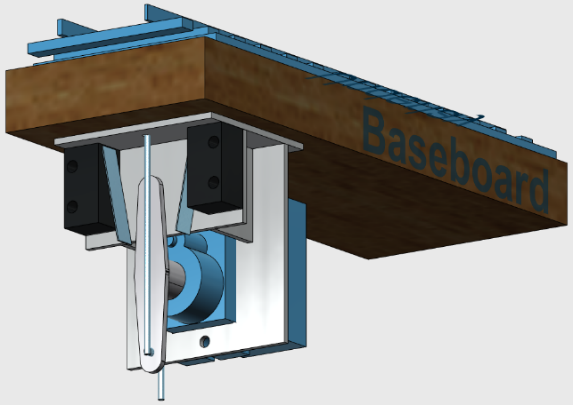
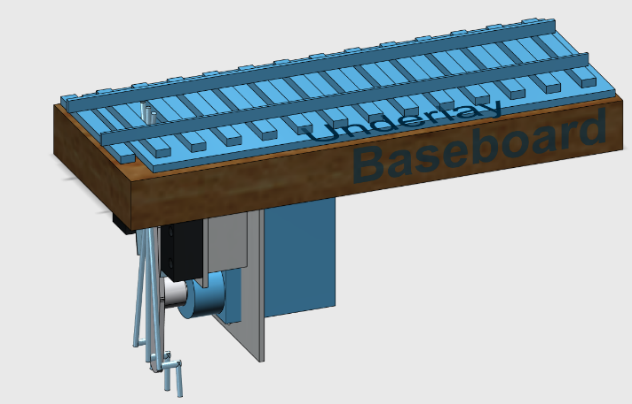
 ****

Figure 4 - Servo Mounting Bracket and General Arrangement

Note that the servo is mounted so that the actuating pin is closest to the turnout’s direction of entry. This configuration gives maximum spacing to mounted assemblies when two turnouts are facing each other.

* 1. **Servos** should be named with the turnout number that they are fitted too. Expressed as ‘servo100’, ‘servo101’, etc.

1. **Modules:**
   1. **Control Modules:** are fitted at various locations around the layout near the associated turnouts and are used to house the microprocessing unit (Arduino Pro Mini) and supporting turnout control and indication circuitry. These are labelled ‘Control Module’ followed by the identification label for the panel in which they address. Expressed as ‘Control Module LHA’ for the unit controlled by Panel A on the lower deck for the 16.5mm track. In the case where there is more than one Control Module associated with a panel area of control, these will be given a sequential number following the identifier. Expressed as ‘Control Module LHA1’, ‘Control Module LHA2’, etc.
2. **Signal busses:**
   1. **Signal Buses:** shall be used wherever possible using ribbon cable, both internally and externally to the Control Panel. Buses run externally from the rear panel connector (standard connector D37 male), and the Control Modules, Servo Mounting Modules generally via pin header type connectors and are shall be labelled, further details to follow. There are four types of buses:
      1. **Route Command**;
      2. **Turnout Indication**;
      3. **Route Indication**;
      4. **Turnout Control**.
   2. **Route Command:** buses will carry the pushbutton signals to the Control Modules. These are labelled ‘Route Command’ followed by the Control Module number. Expressed as ‘Route Command LHA1’ or ‘RCLHA1’, Route Command LHA2’ or ‘RCLHA2’.
   3. **Turnout Indication:** buses will carry the switched signal from the Servo Mounting Module micro switch to the ‘Control module’. These are labelled ‘Turnout Indication’ followed by the ‘Control Module’ number it is connected too. Expressed as ‘Turnout Indication LHA1’ or ‘TILHA1’, ‘Turnout Indication LHA2’ or ‘TILHA2’, etc. This bus is split into several connectors. One for each micro switch that is contained within the bus. Each connector is labelled ‘Turnout Indication’ followed by the Servo Mounting Module number. Expressed as ‘Turnout Indication 100’ or ‘TI100’, ‘Turnout Indication 101’ or ‘TI101’, etc.
   4. **Route Indication:** buses will carry the processed indication signal from the ‘Control Module’ to the ‘Control Panel’ for route/turnout position indication via the panel rear connector to the indication LEDs. These are labelled ‘Route Indication’ followed by the ‘Control Module’ number. Expressed as ‘Route Indication LHA1’ or ‘RIA1’, ‘Route Indication LHA2’ or ‘RILHA2’.
   5. **Turnout Control:** buses will carry the Servo Control, +5VDC and GND signals from the Control Module to the Servo Mounting Module’s Servo. These are labelled ‘Turnout Control’ followed by the Control Module number it is connected too. Expressed as ‘Turnout Control LHA1’ or ‘TCLHA1’, ‘Turnout Control LHA2’ or ‘TCLHA2’, etc. This bus is split into several connectors. One for each Servo Mounting Module that is contained within the bus. Each connector is labelled ‘Turnout Control’ followed by the Servo Mounting Module number. Expressed as ‘Turnout Control 100’ or ‘TC100’, ‘Turnout Control 101’ or ‘TC101’, etc.
3. **Programming**
   1. **Control module pin Convention**
      1. **HIGH**, module pin will have 5V set on the pin, either from an external signal (pushbutton/switch) or internal commands (PULLUP)
      2. **LOW**, module pin will be at a GND state, either from an external signal (pushbutton/switch) or internal command (LOW)
   2. **Control module Programming:** The ‘Control Modules’ require four sets of commands
      1. **Route Command**;
      2. **Turnout Indication**;
      3. **Route Indication**;
      4. **Turnout Control**.
   3. **Route Command:**  is the input from the control panel pushbuttons. This will be wired such that when a pushbutton is pressed, the applicable pin will be set ‘LOW’. i.e. GND. The input pin will normally be in the 'HIGH" state with no pushbutton input due to the microprocessor pinMode configuration INTERNALL\_PULLUP.
   4. **Turnout Indication:** is the input from each turnout switch position.This will be wired such that when the turnout is in the THROWN position, the Arduino input pin will be drawn to a LOW state. (GND).
   5. **Route Indication:** is the output from the Control Module to the Control Panel/s. This will be programmed such that the relative LED pin output is set ‘HIGH’ when required to be illuminated. As a consequence, the LED will be wired to GND with a current limiting resistor.
   6. **Turnout Control:** is the output from the Control Module to operate the programmed turnouts. This will be programmed such that the relative microprocessor pins are set to control the servo position of the required turnout operation.
   7. **Pushbutton Inputs:** names should be labelled as ‘input’ followed by their route names. These names shall only be unique within the Control Module’s program, here is no need to make these unique across the layout. Expressed as ‘inputYardOut’, ‘inputMainOuter’, etc.
   8. **Sketch Identification:** with potentially so many control modules that can be used around the layout there needs to be away that a sketch that is loaded to a control unit be identifiable. The nature of Arduino programming is that a user cannot download a sketch of the board to see what is loaded. RMCQ will add some standard identification variables and print these variable to the Serial Port during the Arduino setup routine. The items are identified here;
      1. **progName** – a variable (#define progName “XXXXX”) that holds the name of the control/indication program. Example “RMCQ Control and Indication Module LHA1”,
      2. **version** – a variable (#define version “1.0”) that holds the version of the control/indication program. Example “1.0”,
      3. **date** – a variable (#define date “22 Oct 2018”) that holds the date at which the sketch was created or updated. Example “22 Oct 2018”,
      4. **author** - a variable (#define author “DLowe”) that holds the name of the last programmer to make any updates or changes to sketch. Example “DLowe”,

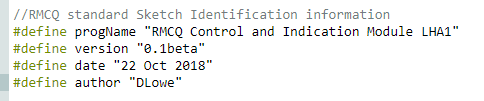


Figure 5 - RMCQ Standard Sketch Identification

* + 1. In order for the items to be displayed in the serial monitor. Doing this will allow a programmer to inspect the module to determine the program loaded to the Arduino. Insert the following lines of code in the setup() routine, this is using the default Serial port for the device at 9600 board rate,

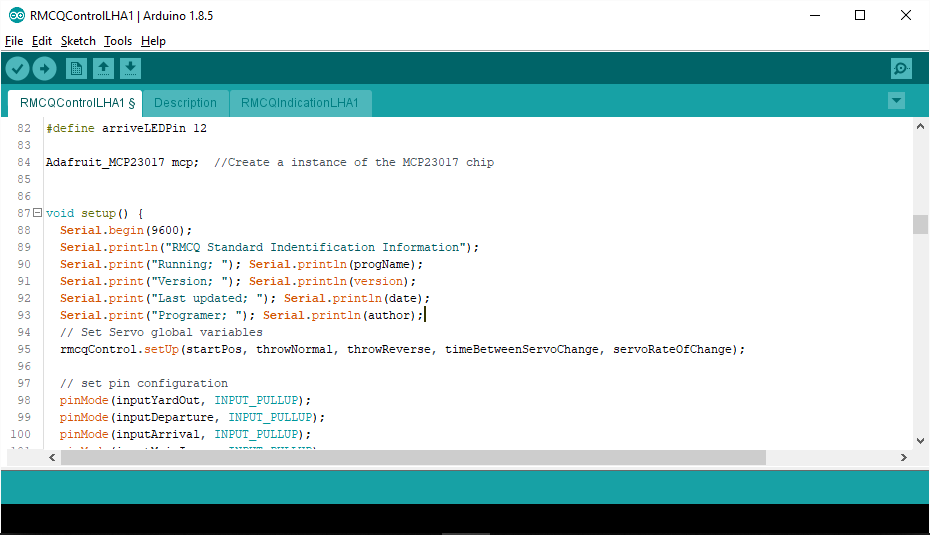
**

Figure 6 - Standard Code for Identification display in Serial Monitor

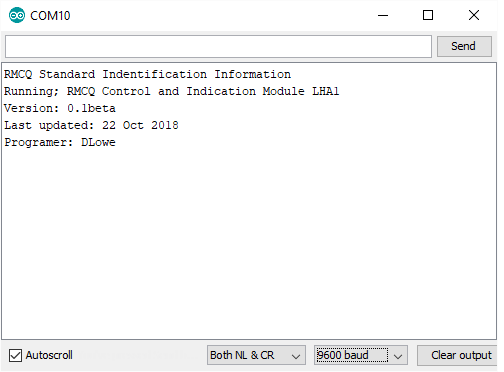


Figure 7 - Displayed Information on the Serial Monitor