

Instruction Servo5

Note, this manual is continuously updated. Check

www.mollehem.se/doc/instruktioner/instruktion_Servo5.pdf for the latest version.

Content

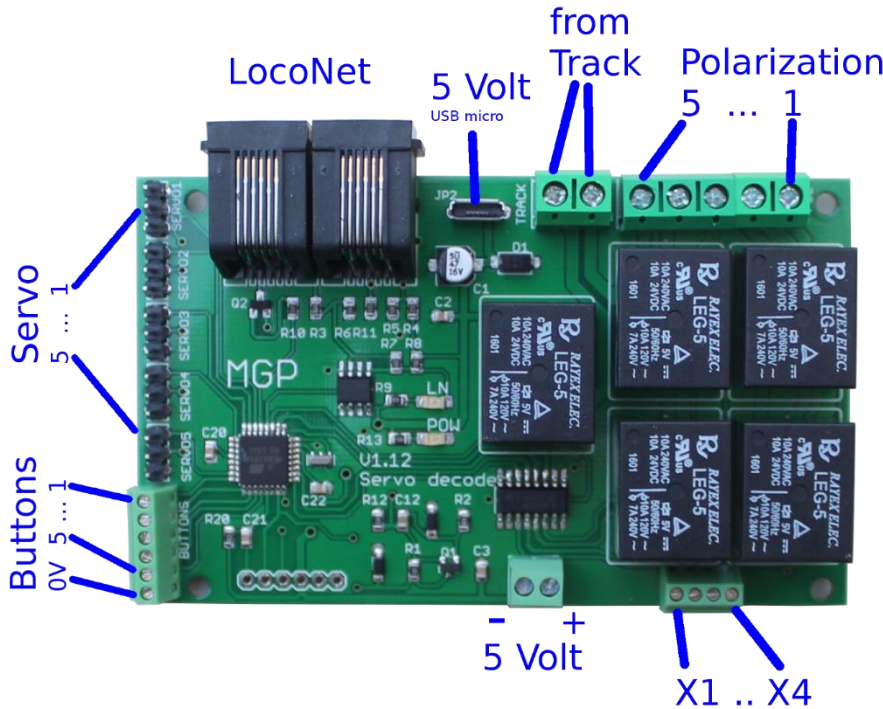
1	Servo decoder	2
1.1	Address to decoder and servos	3
1.2	Connecting buttons	3
1.3	LocoNet	4
1.4	Functions	5
1.4.1	Routes	5
1.4.2	Locking of turnouts	5
1.4.3	Secondary triggering	6
1.4.4	Occupancy detection	6
1.4.5	Direction dependant Occupancy detection	6
1.4.6	Rule control	7
1.4.7	X-connection	8
1.4.8	Track status from units of other brands	9
1.5	Mounting and adjusting servo/turnout	10
1.5.1	Posibility to turn servo motor off when servo is in position	11
1.5.2	Use the polarization relay separatly from servo	11
1.6	Support for basic configuration of turnout servos	11
1.7	System variables – SV, Servo decoder	13
Appendix A, example "Controlling a loopback with a servo decoder"		19
Appendix B, Hardware		21

1 SERVO DECODER

The servo decoder can handle up to 5 servos, including polarization of the frog/crossing and with push or switch buttons for control, see the picture for connections.

Standard servo cables are “brown, red, yellow” and is connected according to the picture. Common is also the colors “black, red, white” with black in the brown position.

Power is 5 Volt DC and the connection is of the type “USB micro”.



Push buttons are connected with one side to 0 Volt and the other side to the screw terminal 1 to 5. The servo will change position each time the push button is pressed.

If a switch button is used, the connection is the same, one side to 0 Volt and the other to the terminal 1 to 5. SV 30, “Type of Buttons” should also be changed from “Toggle” to “Switch”, see chapter about SV programming.

Note, if toggle switches are used and no LocoNet is connected to the decoder – then SV 44 “Send LN for Buttons” must be set to “No”!

Polarization controls the polarity of the middle part of a switch, the “frog”/“crossing”. For this to work, the power for each side of the track is connected to the terminal for “track in” and the crossing of each turnout is connected to polarization 1 to 5.

In the bottom of the picture there are 4 connections X1 – X4. They can be used for different purposes and their function is decided by SVs.

There are also an extra power connection that can be used to feed the decoder with power, or to connect other cards.

1.1 ADDRESS TO DECODER AND SERVOS

The decoder itself has one address stored in SV 21. The default address is **35**.

This address is used when the decoder is configured through its SV's. It is important that the address is unique on LocoNet, so it should be changed if more than one servo decoder is used.

The decoder address is changed in SV 21 through the programming app. Note that a changed address takes affect first at the next startup, so disconnect and reconnect power to the decoder after changing the address.

The connected servos are controlled through individual addresses. They are based on the main decoder address with servo 1 having the decoder address, servo 2 having "decoder address + 1" etc.

Important note,

there was a bug in the software of the panel decoder. Some address where "distorted" in the messages and could not be recognized by the servo decoder. So if some address is found to work when used from for example, the "App terminal", but not from a panel button, then it will be this bug.

This bug exist in panel decoders with version 4 and earlier. Decoders can be updated – contact MGP.

1.2 CONNECTING BUTTONS

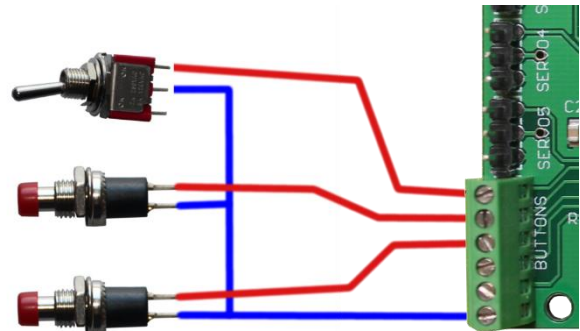
Buttons to control the servos are connected to the terminal for buttons.

Buttons can be push buttons or switches.

Normally simple push buttons is recommended to change the state of the turnout.

Switches with two stable directions, toggle switches, can be used in case there is a need to show the state of the turnout with the position of the switch. The drawback with this is that the position will not show the correct state if the turnout has been changed by e.g. a LocoNet order.

Buttons are connected with one common side to 0 Volt, and the other connected to the terminal at the position for the controlled turnout.



Inputs where the buttons are connected can also be used for other purposes.

In the decoder it must be stated what is applied to the inputs. This is done in Sv180, "Input use" which should have the value "Switch buttons" or "Internal routes" when buttons are connected.

In the decoder stated what applied to the inputs. To do this in Sv180, "Input use" which must have the value "Switch buttons" or "Internal routes" when buttons are connected.

In decoder version 12, two more options is available: "Track status" and "Switch status".

The type of buttons used should also be specified. This is specified in Sv 36, "Type of buttons".

If a switch (as the top in the picture above) is used the value should be "Switch", and if push buttons (like the two lower in the picture above are used, the value should be "Toggle".

1.3 LOCONET

The servo decoder can be used without any LocoNet connection and is then controlled by connected buttons.

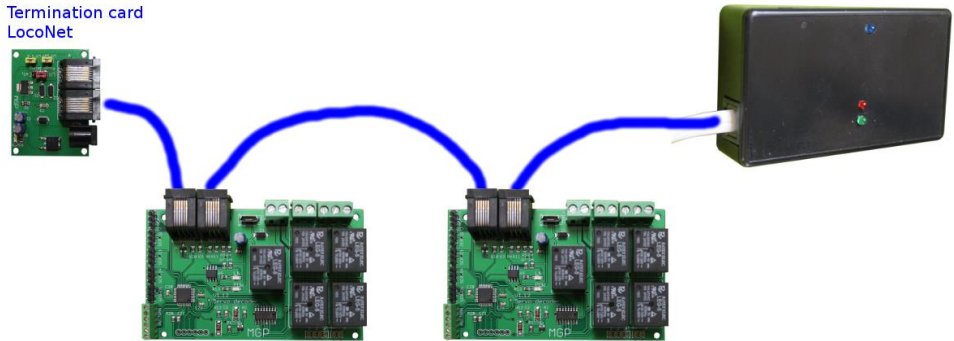
When the decoder is used in a larger layout or when the decoder is programmed through the programming app, LocoNet is used.

Note, when LocoNet is used there has to be current support for LocoNet, which either a command station or a termination card (e.g. MGPs termination card) supplies.

In the picture to the right, a termination card, two servo decoders and one

Bluetooth interface are connected to each other with LocoNet cables. LocoNet gets its power from the termination card, and that card can be replaced by a command station like Z21, Digitrax etc.

Termination card
LocoNet



1.4 FUNCTIONS

1.4.1 ROUTES

On a station there can be several turnouts that must be set to the correct state, e.g. to take the train from the west entry to platform 1. This can be done with one command if Routes are used.

When a route is triggered the turnouts defined for this route will be set to defined states.

A route can have an address of its own and be triggered through LocoNet from a remote control panel.

The decoder can handle 5 routes with up to 6 turnouts per route. A turnout in a route can be any turnout on the layout and not just local to the decoder. A turnout in a route can also be another route so several routes can be connected if needed.

The local buttons on the decoder can be configured to control routes.

Each route has a normal turnout address. The address is specified for each route in SV, "Route X, Address"

The address of the route acts as a normal turnout address and can be used where standard addresses are used, for example. from a button in a panel, in other routes etc.

The route is set when sending "Closed" to the address, however sending "Thrown" will have no effect.

The route sends feedback about its status, "Closed" when it is set, "Thrown" if any of the turnouts are in the wrong direction. This applies regardless of whether the turnout order is sent to the route's address or if each of the individual turnouts is controlled directly.

If a panel decoder like "MGP Panel" is used to build control panels, a LED can be used to show when a route is set in its whole. A defined route will return a normal "switch feedback", with "Closed", when the route is set, and "Thrown" otherwise.

If the turnouts must not be switched at the same time, e.g. due to power consumption, a delay can be defined.

In Sv170, "Delay in routes", the delay can be defined with the number of tenth seconds. If a delay of 1s is wanted, then define the number 10 (10*0.1s).

Tip – at the entry to a yard with a few parallel tracks, it could be easier to provide one button per track instead of one button per turnout. When a train arrives and should go to e.g. track 4, then press the button for track 4 instead of a few buttons to set the turnout correct.

This can easily be accomplished with routes.

1.4.2 LOCKING OF TURNOUTS

In a small yard there could be a demand to lock some of the turnouts.

When a train should use the yard, the turnouts could be unlocked either remotely from a distant control panel, or locally at the yard with a key.

In the decoder the turnouts that should be locked is defined in SV 47 and their state in SV 48.

If turnouts should be locked for only local change, then SV47 "Lock local" should be set. This will affect all turnouts on this decoder. Note, this was introduced from version 10 of the decoder firmware.

If a local key should be used, a key can be connected to input X3 or X4. This is done with a "key switch", with one side of the switch connected to 0 volt and the other to X3 or X4.

If locking should be controlled from a distant panel, an address is defined in SV 45 and locking is done with a switch command on that address.

1.4.3 SECONDARY TRIGGERING

For a turnout, another turnout can be defined to be triggered when the main turnout is triggered.

An example of usage is two turnouts connected as a crossover to enable the train to switch track on a double track. Here one button could be used to switch the first turnout, and the second turnout is switched secondary to the first.

1.4.4 OCCUPANCY DETECTION

The 5 inputs for buttons can be used for occupancy detection instead of control buttons.

To input 1 to 5, external electronic is connected that can e.g. sense the current consumed by a train a particular track. When the track becomes occupied or free, the decoder will report on LocoNet with standard status messages. This information can be used to light a LED at a control panel or set a signal by the track to Stop or Go.

Inputs where the occupancy detectors are connected, can also be used for buttons.

In the decoder it should be defined that occupancy detection is used. This is done in Sv180, "Input use" which must have the value "Track status".

Each detected track will have its own address. In Sv 181, "Status starting address", the address of the first connected track is defined. The other tracks will have consecutive addresses.

To filter disturbing signals on the input from occupancy detection or buttons there are two filter parameters to use:

"Input filter pre" sets how long a signal must be to be accepted as a new signal. The value is the time in milliseconds. With this, temporary disturbance can be removed when e.g. a loco has bad contact with the track.

If the input is used for buttons this time can be set very low or to 0, to avoid noticeable delay when a button is pressed.

"Input filter succ" is also time in milliseconds. When a new signal has been registered, other immediate signals will be discarded. The time before a new signal will be accepted is this filter time.

This is mainly used for buttons to filter away contact bounces that always appear on push buttons etc.

1.4.5 DIRECTION DEPENDANT OCCUPANCY DETECTION

In some cases it can be useful with a occupancy indication that gives information about the direction of the trains movement.

A direction dependent detection is based on two normal detections that are placed close to each other. Logic will register when a train moves between those two detections and will send occupancy messages based on the direction of travel.

The direction dependent detection will have two addresses that will indicate direction, one for direction "A to B" and the other for "B to A".

The decoder has a total of four direction dependent detections, each with detections for both directions.

The addresses are based of one start address, defined in SV 450, "Direction dependent status"->"Start Address".

The first of the four detections will use the start address for direction "A to B" and the address+1 for direction "B to A".

The second detection will use "start address"+2 etc.

For each of the direction dependent detections, the two addresses of the normal detectors, that the direction dependency is based on, are defined.

Those normal detection addresses are defined in Sv "Direction dependent status"->"Direction dependent status X: Address A" and "Direction dependent status"->"Direction dependent status X: Address B".

A normal detection that is used in a direction dependent detection can still be used as a normal detection.

The direction dependent detection will signal "Occupied" when the train reaches the border between the two normal detectors, that is when both detectors signal "Occupied".

The direction dependent detection will signal "Free" when the train leaves the border between the two normal detectors, that is when one detector signal "Free".

A delay for the acceptance of the normal detector going from "Occupied" to "Free" can be set. This can be used to minimize the problem for example when train has bad current collection.

The delay time is set in Sv 452, "Direction dependent status"->"Delay free status (0.1s)". This time is given in the number of 1/10 of a second, so e.g. the value 10 will give a delay of 1 second.

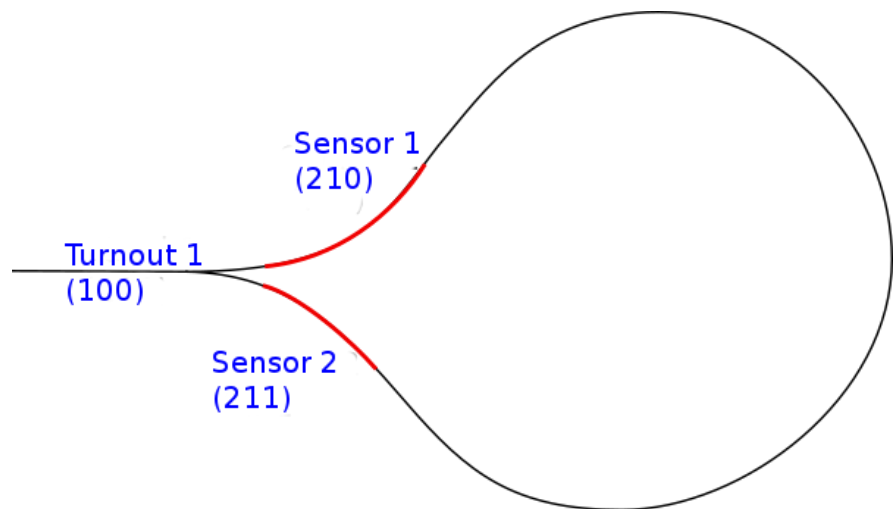
1.4.6 RULE CONTROL

Sometimes logic is needed to automate things controlled by servos.

An example could be to set a turnout when a track gets occupied, and this can be accomplished by a rule.

Example with a reverse loop:

When a train has entered the loop and is on the way out again, the turnout must be changed. This can be done with a sensor that see the train approaching the turnout. In the decoder we can have rules that tell the decoder to set the turnout correct.



We create a rule that sets the "turnout 1" to "closed" (i.e. up in the loop) and is triggered by an indication "occupied" from "sensor 1".

Another rule is created that sets "turnout 1" to "thrown" when "sensor 2" indicate "occupied".

A rule controls one of the turnout (1-5) to a set direction. To design the rule there are characteristics to decide how and when the turnout should be set:

Rule active	Controls if the rule should be used.
Switch number	The number of the turnout that should be set. The number is the internal number, 1-5, not the external address. Only turnouts local to the decoder can be controlled by rules.
Controlled state	The state of the turnout that should be controlled. The turnout is set to straight track (Closed) or diverging track (Thrown)
Triggering	Decides if the rule should actively trigger the turnout, or if it should be passive and e.g. prevent another order to set the turnout.
Mandatory	Decides if the rule must be true for the turnout to be set, or if it will be allowed to set the turnout with another order against the rule.
Scope	Controls if the rule will affect the state only when it becomes true, or also when it becomes false.

In the previous example with the loopback, the first rule could be created like this:

Turnout number is 1 (if the turnout is connected to servo 1),
 the state to control is Closed (straight forward),
 Triggering is "Triggers switch", we want the turnout to be triggered when the rule becomes true,
 Mandatory is "Not mandatory", we want to be able to switch the turnout with other orders e.g. from a control panel,
 "Scope" becomes "Active only at true state", we do not want to switch the change when the train leaves the sensor.

For each rule there are a set of logical conditions that decides if the rule is true or not – in the example that will be when the sensors are occupied.

Up to 5 conditions can be defined for each rule. Logic can be defined to require that all conditions must be true for the rule to be true, or just one, or a combination.

Each condition is defined in SVs:

Logic	The conditions dependence with previous conditions. AND defines that both this condition AND the previous condition must be true OR defines that this condition must be true OR the previous (or both).
Status	Status of the device that controls the condition, eg. a turnout is closed or an occupancy detection is occupied.
Type	The type of the device, turnout or occupancy sensor etc.
Address	The external (LocoNet) address of the controlling device.

In the loopback example the condition for the first rule will be:

"Logic" – doesn't matter because this is the first condition,
 "Status" should be "Occupied", we will trigger when the sensor reports occupied,
 "Type" is "Sensor", i.e. occupancy detection,
 "Address" is 210 which is the address of the sensor according to the previous picture.
 (More about the loopback example can be found in appendix A.)

The relation between the statements is defined with "AND" and "OR".

"AND" binds harder than "OR".

Example:

for "A and B or C", the first and is evaluated "A and B", and then the result of this is used with "or C". With parentheses the expression would be: "(A and B) or C".

If more complicated rules, with more than 5 conditions, is needed, then one rule could be used in another.

1.4.7 X-CONNECTION

On the decoder there are extra connection named X1, X2, X3 and X4.

The function of these is controlled with SV 183 – SV 193.

For each X-connection there is one SV defining the function and one SV that defines the address to be used.

1.4.7.1 X1 AND X2

X1 and X2 can be used for input and they must have defined potentials of 0 resp. 5 Volt.

These connections can be used to switch servos or for occupancy detection.

1.4.7.2 X3 OCH X4

X3 and X4 can be used both as input and output.

As input they work in the same way as the button terminals i.e. they should connect to 0 Volt when active.

As input they can be used to control servos, for occupancy control and for locking selected servos.

As output terminals their value (0/5 Volt) can be set to follow the status of the servos.

Important – when using the output linked to a servo/relay, it is the "internal address" to be used, meaning 1-5 for servo 1-5!

As an example the output can be set to 1 when a servo is moving and then used for controlling a bell on a railroad crossing.

Output could also be used to follow an internal relay and could be used to control an external relay like the MGP relay card for changing track polarity in a loopback etc.

1.4.7.3 OPTO SENSORS ON X1/X3 OR X2/X4

Experimental support for optical sensors.

Optical sensor consists of a transmitter and a receiver (e.g. a led and a photodiode). If the light beam is broken between the transmitter and receiver, it indicates a train. It can also be used to indicate on reflections from a train.

The light is usually infrared (IR) to become independent of lighting conditions in the room. However, such as fluorescent lighting can still affect these detectors.

A transmitter can be connected to the X3 (X4) and the corresponding receiver is connected to X1 (X2).

The decoder will handle light detector in a manner that interference from, for example. fluorescent lighting is minimized.

Light emitting diode and photodiode should usually be connected to the X-connections with resistance, see instructions from sensor supplier.

The use of optical sensors is activated on the setting of the "Usage X1" or "Usage X2".

1.4.8 TRACK STATUS FROM UNITS OF OTHER BRANDS

Track sensors sends messages for busy/free tracks on the LocoNet.

For example, if a locomotive has poor connection with the track, it might send bursts of messages in vain. A track detection should have filtering capabilities to avoid these unnecessary messages but there are devices for track sense from other manufacturers that have no filter option

A rule that is based on free/busy tracks might give an unwanted behavior if unfiltered track indication is used, for example. a railroad crossing that will be opened when a track becomes available. Unnecessary message should be filtered at the source, but if that is not possible this decoder has a possibility to deal with this.

Decoders with software from version 11:

Up to 10 status addresses can be defined to be filtered. Filtering provides a delay before the change in status is accepted. If new status messages arrive during this time, the previous message is ignored and a new delay will be started for the new message.

The delay can be set to between 0 and 3 seconds.

Suitable delay for track indication can somewhere around 0.3 - 0.5 seconds.

The delay can be activated when the status is set to busy, free or in either case.

The normal use of this function is to filter the track indication. Poor track connection will result in false free messages and for this case it is appropriate to put delay active for signals that goes to 0/off.

Use this function as a last resort.

Try to avoid unnecessarily messages before they are sent!

1.5 MOUNTING AND ADJUSTING SERVO/TURNOUT

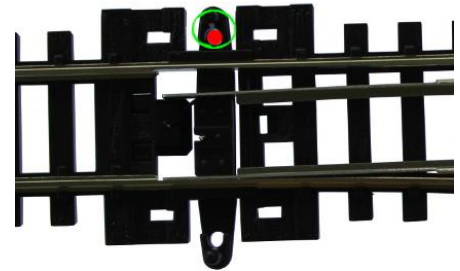
The following is a step-for-step description of how a servo could be mounted to control a turnout.

The servo is here mounted directly under the turnout and connects to the turnout with a short 1 mm piano wire.

In the base under the turnout a 6 mm hole is drilled.

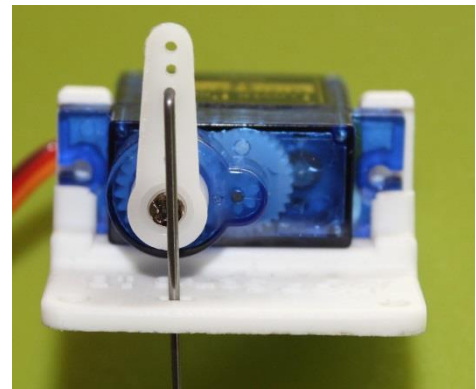
The hole should be directly under the “pull-rod”. The “pull-rod” is often prepared with a suitable hole (red dot in the picture), but if not a 1 mm hole is drilled.

The 6 mm hole through the base is drilled directly under the mounting hole in the turnout so that it is centered when the turnout is in between its states. In the picture the hole is marked in green.



Place the servo in the servo decoder and adjust SVs so that max and min angle is in the middle (around 95 and 90). Place the servo arm so it's standing straight up and mount it and the wire in the holder, see next picture.

Place the servo/holder under the base so that the wire comes up to the turnout and through the hole in the “pull-rod”. Adjust the holder so that the turnout is between its two states and screw the holder in place.



Now adjust the movement of the servo:

Note, In the app there is new support to make the basic setting easily changed. This support is specific for this decoder, “Servo5”. See the chapter “Support for basic configuration of turnout servos” for more information.

The following text describes how to configure the parameters directly through Sv-support.

Connect the servo to the decoder with the max/min-angles set to 95 and 90.

Check the address of the servo. The decoder address is in SV 21 and that address is the same as for servo 1. Servo 2 has that address +1 etc.

Select “terminal” from the menu in the programming app and write the address to the servo. To the right of the address there is a button with the text “closed” or “thrown”.

Press on it until the servo moves and the button say “**closed**”. The servo is now in position “closed”. Go back to min/max-value and decrease the **min-value** until the turnout is correctly in one of its states.

Go back to “terminal” and change the turnout to “**thrown**”. Back to **max-angle** and adjust the value so that the turnout is firmly in its other position.

The turnout should now be in the “thrown” state, meaning that the turnout is set diverging. Check that the turnout indeed is in the diverging position and not straight (closed). If the position is wrong, change the SV for “Direction, LocoNet”.

Go back to the “terminal” and check that the turnout works nicely and that the text on the button (closed/thrown) is correct compared to the state of the actual turnout.

If the crossing of the turnout should be polarized from the decoder, connect the crossing to the same terminal number as the operating servo.

Check if there is a shortage if a loco passes the turnout. If there is a short, change the polarization with the SV “Direction, polarization”.

Note, that the relay does not change when the SV is changed. The turnout must be changed for the relay to become synchronized.

Lastly the speed of the servo can be changed with SV “Speed”.

1.5.1 POSIBILITY TO TURN SERVO MOTOR OFF WHEN SERVO IS IN POSITION

A servo is designed to be continuously active. This means that servos always receive control signals, even when they are in their desired position. This can be observed if you try to turn a connected servo by hand, and the servo will then try to prevent moving.

Some servo brands can be a bit bad at keeping its position and are not completely still when in position. In decoders with version 11 and up, there is the possibility to allow the decoder to stop sending control signals when servo come into position.

To activate this function change “Sv 414, Servo motor control” to “no servo control when in position”.

Normal mode for “Sv 414, Servo motor control”, which fits most servos, is “keep servo always active (STANDARD)”.

1.5.2 USE THE POLARIZATION RELAY SEPARATLY FROM SERVO

If there is no need for polarization, the relays can be used for other purposes.

An address can be defined for the relays and each relay can then be control with a switch order for this address.

The address is set in “Sv 418, Relay Address”. Relay 1 will get this address, relay 2 the address+1, etc.

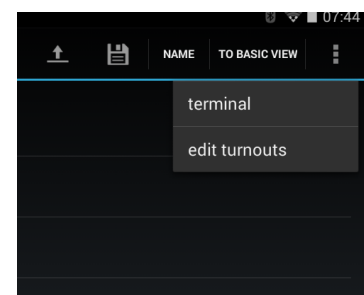
Important: the relay must be released from the servo to be used independently. This is done for each servo in “Sv49 Polarization switch for servo ..” which should be set to “Idle”.

1.6 SUPPORT FOR BASIC CONFIGURATION OF TURNOUT SERVOS

The app has direct support for the configuration of the basic parameters for servos in decoder “Servo5”.

In the app – start by opening the decoder that should be configured. Among the menus up to the right there is “edit turnouts” – select this.

The page that appears contains all the important parameters that need to be configured when mounting a servo to a turnout.

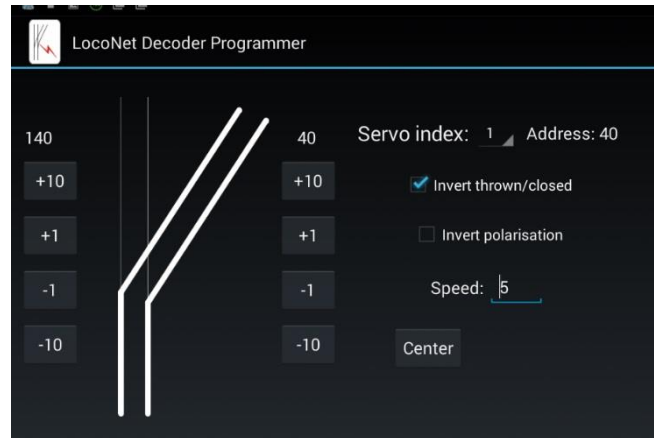


A symbol of the turnout shows the state of the turnout “closed” (straight through) or “thrown” (to the side). In the picture to the right the turnout is in position “thrown”.

Note that the symbol is a button, so it can be “clicked” to change the position of the turnout.

To the right in the picture, “Servo index” shows which servo is controlled at the moment. Here any of the servos (1-5), can be selected.

The address of the selected servo is also shown. The servo address can not be changed here as it depends on the basic address of the decoder. Change the basic decoder address (Sv21) to change the servo addresses.



“Invert thrown/closed” is used to align the position of the symbol with the actual position of the turnout. It is important that the turnout and the symbol shows the same positions.

“Invert polarisation” is used when the polarization of the turnout is wrong, meaning that there is a shortage when a train tries to pass the turnout.

“Speed” is the speed of the movement when the turnout changes position.

On both sides of the turnout symbol, there is buttons to change the angle of the servo.

To the left is the angle when the turnout is in the closed position.

To the right is the angle in thrown position.

The buttons with + and – is used to change the value of the angle.

Down to the right there is a button, “Center”. This can be used to put the servo in center position, which should be done before mounting the servo to the turnout.

1.7 SYSTEM VARIABLES – SV, SERVO DECODER

Note – some SVs is only visible if the programming app has been set to "advanced mode"!

LocoNet decoders has their configuration stored in System Variables, SVs. These can be changed through the programming app.

The following system variables exists at the moment.

SV number	Name	Description	Decoder version
21	Address	Identifies the decoder during programming. This address should be unique on LocoNet. It is also the address to the servo 1. Servo 2 has the address +1, etc.	
23	Servo 1, Min angle	Controls the angle that the servo arm will move in the "low direction".	
24	Servo 1, Max angle	Controls the angle that the servo arm will move in the "upper direction".	
25	Servo 1, Speed	The speed of the servo arm. Low values are fast and high values are slow.	
26-37	Servo 2-5 in the same way as SV 23-25		
38	Type of Buttons	"Toggle" is for a momentary push button, "Switch" is for a switch with two stable positions.	
39	LocoNet direction	For LocoNet messages - defines which positions of the servo, that is regarded as "Closed" (straight forward) for the turnout. Bit 0 in the value controls servo 1, bit 1 controls servo 2 etc.	
40	Polarization direction	Defines which side of the track power that will be fed into the crossing of the turnout. If there is a shortage when a train goes into the turnout, this value should be changed. Bit 0 in the value controls servo 1, bit 1 controls servo 2 etc.	
41	Button direction	When local toggle switch buttons are used to control servos – defines the direction of the servo when the button is switched. Bit 0 in the value controls servo 1, bit 1 controls servo 2 etc.	
42	Feedback type	Feedback can be sent in two ways. Normally the two states "Closed"/"Thrown" is sent. The other choice is to also send "Unknown state". "Unknown state" is sent when the servo moves. Not all LocoNet devices is compatible with 3 states, but all MGP decoders can use it. 0 – use two states, "Closed" or "Thrown" 1 – use three states ("unknown state")	

43	Switch when button is...	Defines when change order should be sent – when the button is pressed or when it is released. Default is when it is pressed.	3
44, bit 0	Send LN for Buttons	Defines if a control order should be sent on LocoNet when local buttons are used.	3
44, bit 1	LocoNet connected	For better performance, set to "No" if no LocoNet is connected to the decoder	8
45	Lock Address	LocoNet address that control locked state (1-Closed) or unlocked (0-thrown)	3
47	Lock switches	Defines which servos that are affected by a lock order. Bit 0 in the value controls servo 1, bit 1 controls servo 2 etc.	3
47, bit 7	Lock local	Defines that the all servos should be locked from local change. Remote change is still possible.	10
48	Lock position	Defines the position of the servo when it is in the locked state. Bit 0 in the value controls servo 1, bit 1 controls servo 2 etc.	3
49	Relay usage	Defines if the relay should be used with the servo. Bit 0 in the value controls servo 1, bit 1 controls servo 2 etc.	3
50	Route 1, Address	LocoNet address for Route 1	
52	Route 1, Sw1, Address	LocoNet address to the turnout 1 in this Route	
54	Route 1, Sw1, Direction	The state of turnout 1 in this Route 0 – Closed 1 – Thrown	
55-68	Route 1, Sw2-Sw6	Turnout 2-6 in this Route in the same ways as SV 52-54. For an unused position use address 0.	
70-149	Route 2-5 in the same way as SV 50-68		
172 bit 0-3	Servo 1, Start delay, from THROWN	Delay, in seconds, between switch order and when the servo starts to move.	6
172 bit 4-7	Servo 1, Start delay, from CLOSED	Delay, in seconds, between switch order and when the servo starts to move.	6
173-176	Start delay servo 2-5	Same as 172 for the other servos	6
177	Lock relay 5 to	0 – use as normal, controlled with servo 5 1-4 – locked to servo 1 – 4	5
178	Input sensor, value for occupied state	0 – normal, occupied at low 1 – occupied at high	5
179	Input, unconnected state	0 – normal, always high 1 – undefined, floating	5

180	Input use	<p>Defines the use of input from the button terminals. They can be used for:</p> <ul style="list-style-type: none"> - "Buttons"(default), - "Routes" - "Occupancy detection". <p>In decoder version 12 two more options is available:</p> <ul style="list-style-type: none"> - "Track status" - "Switch status" 	3 12
181	Status – first address	<p>If button terminals are used for "Occupancy detection", this SV contains the address of the first input. The other inputs are the address+1, +2 etc.</p>	3
183	Usage X1	<p>Usage of X1:</p> <ul style="list-style-type: none"> 0 – Not used 1 - Switch Thrown 2 - Switch Closed 3 - Switch TOGGLE 6 – Track Status 15- Track status, opto X1/X3 16- Track status, opto X1/X3, reflection 	3-5 New in 6 New in 10
184	X1 address	Address used for X1	3
186	Usage X2	<p>Usage of X2:</p> <ul style="list-style-type: none"> 0 – Not used 1 - Switch Thrown 2 - Switch Closed 3 - Switch TOGGLE 6 – Track Status 15- Track status, opto X2/X4 16- Track status, opto X2/X4, reflection 	3-5 New in 6 New in 10
187	X2 address	Address used for X2	3
189	Usage X3	<p>Usage of X3:</p> <ul style="list-style-type: none"> 0= Not used 1= switch CLOSED 2= switch THROWN 4=Lock local 5=Lock as lock list 3 - Switch TOGGLE 6 – Track Status 20 – Output, Servo sync CLOSED 21 – Output, Servo sync THROWN 22 – Output, Servo sync Unknown 23 – Output, Relay sync 24 – Output, Relay sync opposite 	3-5 New in 6
190	X3 address	<p>Address used for X3</p> <ul style="list-style-type: none"> - When X3 is used as input, this is a LocoNet address in the interval 1-2054 (-4095) - When X3 is used as output linked to a servo/relay, then the number of the servo is used, 1-5. 	3
191, bit 7	Active Output X3	<p>Logic level for output that counts as active.</p> <ul style="list-style-type: none"> 0 = 0 1 = 1 	12

192	Usage X4	Usage of X4: 0= Not used 1= switch CLOSED 2= switch THROWN 4=Lock local 5=Lock as lock list 3 - Switch TOGGLE 6 – Track Status 20 – Output, Servo sync CLOSED 21 – Output, Servo sync THROWN 22 – Output, Servo sync Unknown 23 – Output, Relay sync 24 – Output, Relay sync opposite	3-5 New in 6
193	X4 address	Address used for X4 - When X4 is used as input, this is a LocoNet address in the interval 1-2054 (-4095) - When X4 is used as output linked to a servo/relay, then the number of the servo is used, 1-5.	3
194, bit 7	Active Output X4	Logic level for output that counts as active. 0 = 0 1 = 1	12
195, bit 0-6	Auto return time, servo 1	0 if not used 1 -127 is time in seconds before the servo returns to the other direction Version 6 has values 1 – 63	4, 6
195, bit 6	Auto return time factor	Can be set to 10 which means that the value for “autoreturn time” will be 10 times longer, 10 - 630 seconds	6
195, bit 7	State for Auto return, servo 1	0 – Thrown 1 – Closed	4
196-199	Auto return for servo 2-5	Auto return for servo 2-5, same as 195	4 6
200, bit 0-2	Secondary trigger servo 1, type	Defines if something else should be triggered when servo 1 is changed. 0 – Not used 1 – send when servo 1 goes to “Closed” 2 – send when servo 1 goes to “Thrown” 3 – send at each change of servo 1	
200, bit 3	Secondary trigger servo 1, when to send	Defines when the secondary command shall be sent: 0 – before the movement starts for servo 1 1 – when servo 1 has moved	6
201	Secondary trigger servo 1, command	Defines what secondary order that should be sent 0 – the same order 1 – the opposite order 2 – send “Closed” 3 – send “Thrown”	
202	Secondary trigger servo 1, Address	Address to which the secondary trigger should be sent	
204-219	Sec trigger for servo 2-5, same as 200-203		

220 - 299	Rules	Rules to control the movement of servos based on the status of other devices	5
220, bit 0	Rule 1, Active	0 – inactive, 1 – active	5
220, bit 1-3	Rule 1, Switch number	The local number of the servo that is controlled by the rule, 1 till 5. 0 – no switch 1-5 – switch number	5
220, bit 4	Rule 1, Controlled state	The state that will be set to the servo if the Rule is true. 0 – thrown 1 – closed	5
220, bit 5	Rule 1, Triggering	Defines if the rule should actively trigger 0 – Rule will not trigger when active 1 – Rule will trigger switch when status change	5
220, bit 6	Rule 1, Mandatory	Defines if the rule is mandatory for the servo, or if a control order can be given to the servo against the rule. 0 – not mandatory 1 – mandatory	
220, bit 7	Rule 1, Scope	Defines if the rule should trigger the servo when it becomes TRUE or when both states occur 0 – Rule active only at TRUE 1 – Rule active at both TRUE and FALSE	
221, bit 0	Rule 1, condition 1, Logic	0 – OR 1 – AND	5
221, bit 1	Rule 1, Condition 1, Status	0 – Thrown/0 1 – Closed/Occupied/1	5
221, bit 2-4	Rule 1, Condition 1, Type	0 – not used 2 – Switch Status 3 – Sensor 4 – Other rule	5
222,223	Rule 1, Condition 1, Address	Address of switch/Occupancy sensor	5
224-226	Rule 1, Condition 2	Same as 221-223	5
227-229	Rule 1, Condition 3	Same as 221-223	5
230-232	Rule 1, Condition 4	Same as 221-223	5
233-235	Rule 1, Condition 5	Same as 221-223	5
236-251	Rule 2	Same as 220-235	5
252-267	Rule 3	Same as 220-235	5
268-283	Rule 4	Same as 220-235	5
284-299	Rule 5	Same as 220-235	5
400-401	Input filter pre (ms)	Defines the minimum time that a signal must be stable on the input to be detected	6
402-403	Input filter succ (ms)	Defines the time that a signal will be discarded directly after a previous signal has been detected	6

405	Polarisation dependency servo 1	The polarization for this servo will also be dependent of the polarization of the state of another servo 0 – not used 1..5 – dependent on this servo	12
406--409	Polarisation dependency servo 2-5	The same as for servo 1 in 404	12
410	Optosensor high diff	Experimental! Used to trim the opto sensor for different light circumstances	10
411	Optosensor low diff	Experimental! Used to trim the opto sensor for different light circumstances	10
412	Optosensor delay	Experimental! Used to trim the opto sensor for different light circumstances	10
414, bit 1-5	Servo update, Servo 1-5	Bitwise, 1 – always update servo with it's position (recommended) 0 – stop updates to servo when servo in position	11
418	Relay address	If relays should be used separately with there own addresses, set an address here: First relay will get this address, next relay will get address+1, etc. Note: the relay must be disconnected from the servo in Sv49!	11
420 – 440	Delay of selected status messages		11
420	Incoming Status Delay Type	When the message should be delayed, either when is signals 0, or 1, or both.	11
421	Delay time	Delay for incoming status messages in 1/10 seconds, value 0 – 30 meaning 0.0 – 3.1 seconds	11
422,423	Delay address 1	Address of message to be delayed, 1 - 4095	11
424-440	Delay address 2-10	Same as for 422	11
450 - 468	Direction Dependency Detection		12
450	Start Address	First Address of messages for Direction dependent occupancy. The rest of will be address+1, address+2 etc.	12
452	Delay Free Status	Delay of the acceptance for transision between "Occupied" and "free". Value is number of 0.1 seconds.	12
453	Direction dependant status 1: Address A	The address of the first normal detection that this direction dependency will be based on.	12
455	Direction dependant status 1: Address B	The address of the second normal detection that this direction dependency will be based on.	12

457-467		Address A and B for direction dependent detectors 2-4, in the same way as 453 and 455.	12
---------	--	--	----

Some SV has complex values on a bit level.

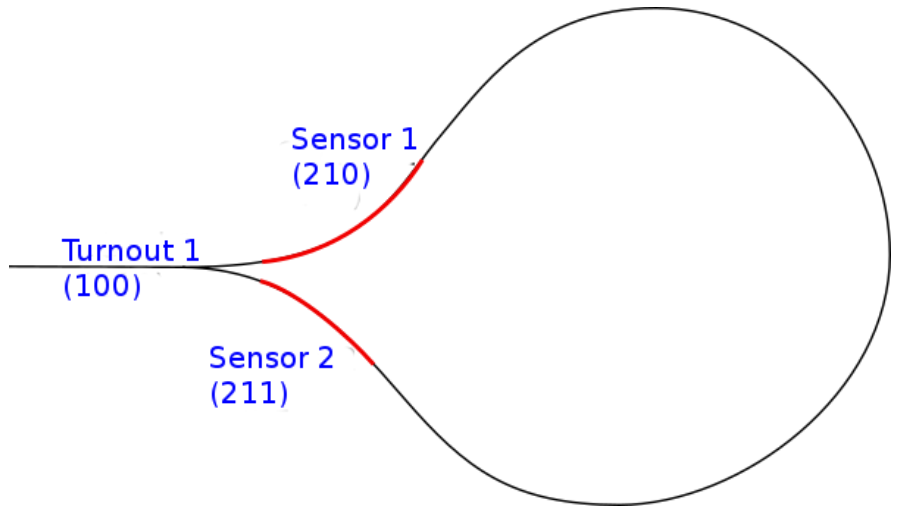
Use the programming app to avoid manipulation on bit level!

APPENDIX A, EXAMPLE "CONTROLLING A LOOPBACK WITH A SERVO DECODER"

In the chapter about Rules, there is an example of building an automated loopback.

Here comes a complete description of how this application can be built.

The description result in a fully automatic loopback control where the trains themselves trigger the change of the turnout and where polarization in the loopback is controlled to avoid shortage.



Program version 5 or better is required for this example – contact MGP if an update of the decoder is required.

1.7.1.1 TURNOUT CONTROL AND OCCUPANCY DETECTION

To control the loopback, a servo decoder and occupancy detection for two tracks. The occupancy detectors are connected to button terminal 1 and 2 of the servo decoder.

In picture we have given the turnout **address 100 (SV 21 in the decoder)**.

We use sensors for occupancy detection connected to the button terminals 1 and 2. For this to work we need to set **SV 180 to "Track status"** and the address in **SV 181 to 210**.

1.7.1.2 POLARIZATION OF THE LOOPBACK

A loopback the left rail with its left rail polarity will come back as the right rail and a shortage will occur. To avoid that, the track inside the loopback is isolated from the turnout and a relay is used to switch sides of the track polarity when needed.

Polarity for the crossing in the turnout is controlled by the polarity relay for servo 1. The inner rail of the loopback track should always have the same polarity as the crossing in the turnout, so the **inner rail can be connected to the same wire as the crossing of the turnout**.

The outer rail should always have the opposite polarity to the inner rail. This can be accomplished by using one of the other relays on the decoder and make that relay move in the opposite direction as the first relay. **The outer rail is connected to the polarization output for servo 5**.

Now we must force the polarization relay of servo 5 to follow the relay of servo 1, but in opposite direction. This can be done in several ways, but the easiest is to use SV 177 that controls exactly this. Set **SV 177 to "Locked to servo 1"**.

An alternative way to control polarization is to connect an external relay. The relay card from MGP is well suited for this and can switch sides on both rails on a track.

The external relay can be connected to X3 or X4 and defined to follow the movement of the turnout servo.

If the servo decoder is used to control a double track turnout, the easiest way to accomplish polarization is to use two external relay cards.

1.7.1.3 CREATING RULES

When a train is in the loopback and approaches the turnout to exit, the turnout has to change. This should happen when the sensor at the exit indicates occupied.

Two rules is needed, one for each sensor.

"rule 1" triggers "turnout 1" (address 100) to "closed" (i.e. up in the loopback) when "sensor 1" (address 210) indicates "occupied".

"rule 2" triggers "turnout 1" to "thrown" when "sensor 2" (address 211) indicates "occupied".

Rule 1:

Turnout, with address 100, is set to "closed" when Sensor 210 indicates "occupied".

SV 220 and the following SVs: "Switch number" – "Switch 1", "Controlled status" – " Switch closed ", "Triggering" – "Trigger switch", "Mandatory" – "Not mandatory", "Scope" – "Active only at true state"	Rule 1, condition 1: "Logic" – "And" (AND / OR does not matter in this case as it is the first condition) "Status" – "Occupied" "Type" – "Sensor" "Address" – " 210 "
--	---

Rule 2:

Turnout, with address 100, is set to "thrown" when Sensor 211 indicates "occupied".

SV 236 and the following SVs: "Switch number" – "Switch 1", "Controlled status" – " Switch thrown ", "Triggering" – "Trigger switch", "Mandatory" – "Not mandatory", "Scope" – "Active only at true state"	Rule 2, condition 1: "Logic" – "And" "Status" – "Occupied" "Type" – "Sensor" "Address" – " 211 "
--	---

When the rules are defined they can be set active in SV 220 and 236.

APPENDIX B, HARDWARE

The MGP decoders are compatible with the Arduino computer card family.

They can be reprogrammed with the Arduino IDE and in that case be treated as “Arduino Pro mini”.

The six unpopulated connector holes that sits at the border of each decoder, is the same interface that is found on a “Pro mini”. Looking from outside of the board, “GND” is the hole to the right.

To use them with LocoNet, the LocoNet libraries from “Model Railroading with Arduino” can be used. The first versions of the decoders used them, but due to available memory, the current versions use more compact and less generic code. To use this LocoNet library, use D8 as incoming and D9 as outgoing pins.

To update the decoders with new version of the firmware, the available images can be loaded into the decoder. But - these images are not suitable to load if the decoder has been used with other code before. If a decoder needs to be reloaded with the normal firmware – please contact MGP.