



7600RF RF Receiver

Manual

Revision 1.0B



Nico Technology Ltd.

6F, No. 93, Xinhua 1st Rd, Neihu Dist, Taipei City 114, Taiwan

Phone	+886-2-2790-6308
Fax	+886-2-2790-6603
Product Info	sales@nico-tech.com
Web site	http://www.nico-tech.com

This Manual

This manual provides any required information for installation, configuration and operation of the 7600RF RF Receiver.

It exclusively treats the handling of this device. It neither describes the LonWorks technology by Echelon nor the LonMark profile implemented in detail. More specific information concerning these subjects can be found in the documentation of Echelon (www.echelon.com) and the LonMark Interoperability Association (www.lonmark.org).

The first part of this manual provides a survey about the device and its installation in chapters 1 to 3. The 2nd part describes the implemented application for lighting control and its configuration possibilities. Chapter 4 contains a description of the firmware interface while chapter 5 describes the implemented LonMark Objects in detail providing an outlook of the individual objects, their tasks and their relevant configuration parameters.

Chapter 6 explains the basics required to connect the objects to each other.

This manual is relevant for all variants of the 7600RF RF Receiver where applications for lighting and switch control are implemented.

This documentation is subject to modification at any time without prior advice. Nico does not take over any responsibility for mistakes or inaccuracies in this documentation and eventually occurring consequences.

In any case Nico as well as its representatives and staff are not reliable for eventual defaults, damages caused indirectly or during use, occurring due to the use or non-usability of the software or the accompanying documentation.

Nico is a registered trademark of Nico Technology Ltd.

Echelon, LON, LonMark, LonWorks and Neuron are registered trademarks of Echelon Corporation. Other name may be registered trade marks of the respective companies.

Contents

Contents	3
2 Product Information	4
2.1 Functional Elements.....	4
3 Installation.....	4
3.1 Warnings	5
3.2 Mounting.....	6
3.3 Connections.....	6
3.4 Software Installation	7
4 Device Description.....	8
4.1 Hardware Survey	8
4.2 Operation and Display Elements	8
4.3 Connection Pin Assignment.....	8
4.4 EMC.....	10
4.5 Technical Specifications	10
4.6 Dimensions	11
5 Application Software for RF Remote Controller.....	12
5.2 Interoperable Interface	13
6 System Objects.....	14
6.1 Node Object	14
6.2 Remote Control Object	15
6.3 Nico Tech. NV Type Translator Object.....	16
6.4 Virtual KeyPad Object	17
6.5 Programming Examples.....	21
6.6 DIP Switch Setting.....	22
6.6.1 DIP Switch for ON 5508RF RF Remote Controller	22
6.6.2 Wireless network Channel ID Setting.....	23
6.6.3 Remote Controller ID Setting	23

2 Product Information

2.1 Functional Elements

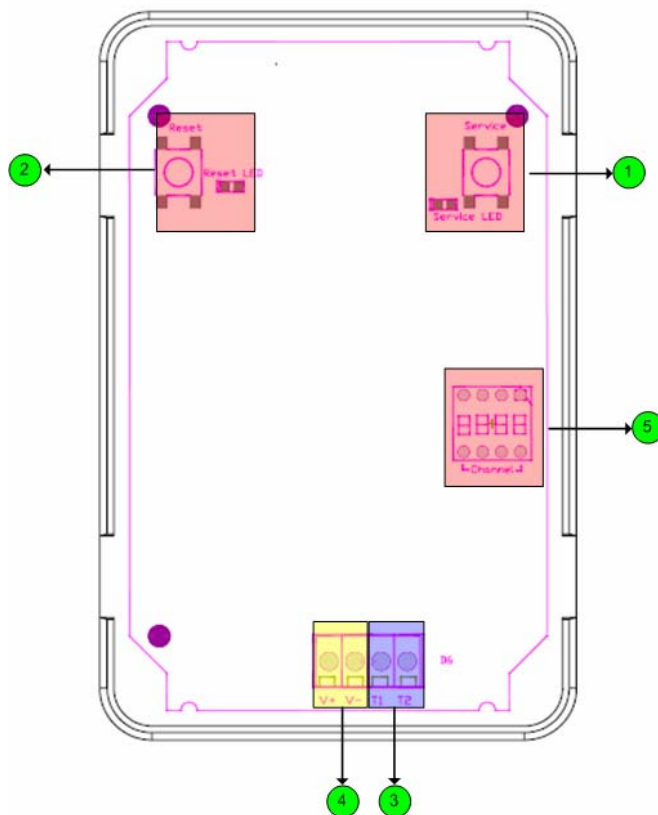


Figure 1.1 7600RF RF Receiver

No	Description
1	Service Pin button and service pin LED indicator
2	Reset button and reset LED indicator
3	LonWorks network interface
4	Power supply. 12~24VDC
5	DIP switch for wireless channel ID setting

2.2 Variants and Identifications

2.3 Scope of Delivery

3 Installation

This chapter first describes the installation of the device; the installation of the configuration software is described in section 2.4.

3.1 Warnings

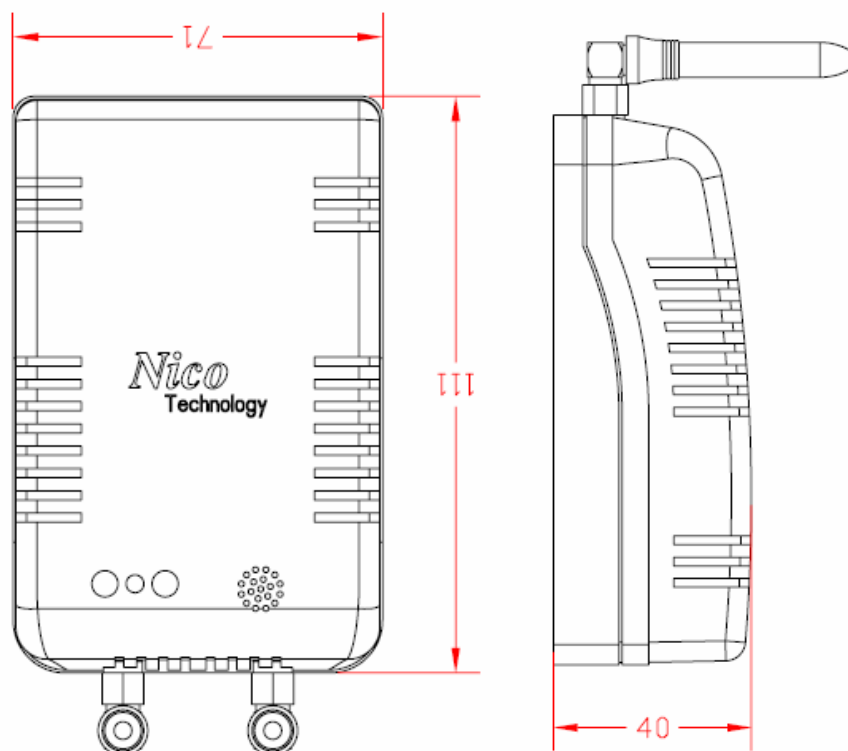
Attention

The device must be installed in compliance with the relevant DIN/VDE regulations or the relevant national standards. The connection to the supply voltage must be performed in accordance with VDE 0100 and VDE 0160 or the relevant national standards. Installation should perform by qualified and technical experienced personnel only.

CAUTION

At the connections of the output channels 1 – 4 (Fig. 1.1, terminals 7, 8, 9, A) and the power supply (terminals 1, 2) 12~24VDC main voltage with load guard band is accessible. The installation of the unit therefore has to be effected in a switch cabinet or behind a respective cover.

3.2 Mounting



3.3 Connections

The 7600RF has to be connected to a 12~24VDC power supply and to the LonWorks network. According to the respective application peripheral equipment has to be connected to outputs.

Attention

Before connecting peripheral equipment the power supply device has to be switched off.

The connection is effected by means of the included screw less terminals.

Clamping range of the screw terminals:

- Power supply connections (5.08mm grid, terminals **0** , **1**): 0.2 – 2.5mm

The pin assignment of the connections is described in chapter 3.1, also containing wiring details.

Voltage

The 12~24 Volt DC connections are through connected in order to achieve easy installation.

LonWorks Network

The connection to the LonWorks network is made by means of power line channels.

Attention

The 7600RF RF Receiver power supply recommend use 24VDC power supply.

3.4 Software Installation

The configuration software of the 7600RF RF Receiver has to be installed by starting the program Setup.exe on the data carrier provided. It runs under Windows 9x/2000 and NT.

4 Device Description

The 7600RF RF Receiver for LonWorks network in automation. Its peripheral scope has been specially designed for the use as switch control for device spreading control of applications such as lighting control or equipment control use in home automation control system.

4.1 Hardware Survey

The 7600RF RF Receiver disposes of one LonWorks circuit for each.

4.2 Operation and Display Elements

The 7600RF RF Receiver is fitted with a service button accessible via a small gap on the side (see Figure. 1.1, **8**). Activation of the buttons generates a service-pin message transmitted via the LonWorks network. The processor status as well as the service-pin status are displayed by the service LED (figure. 1.1, **A**), which is on while the service button is activated. By use the network management function Wink the service LED flashes.

Furthermore the 7600RF RF Receiver is fitted with a reset LED (figure. 1.1, **B**), displaying the availability of device occur reset. The LED is connection to an I/O pin of the Neuron chip processor.

4.3 Connection Pin Assignment

The following tables show the connector pin assignment of the individual connectors. Connections the **1** marking cf. Figure. 1.1 On previously page. In each clamp block pin 1 is situated on the left. For further wiring information see figure 3.2.

LonWorks Network Connection

The bus line can be connection either to **Net**. No polarity has to be considered by connecting the LonWorks network.

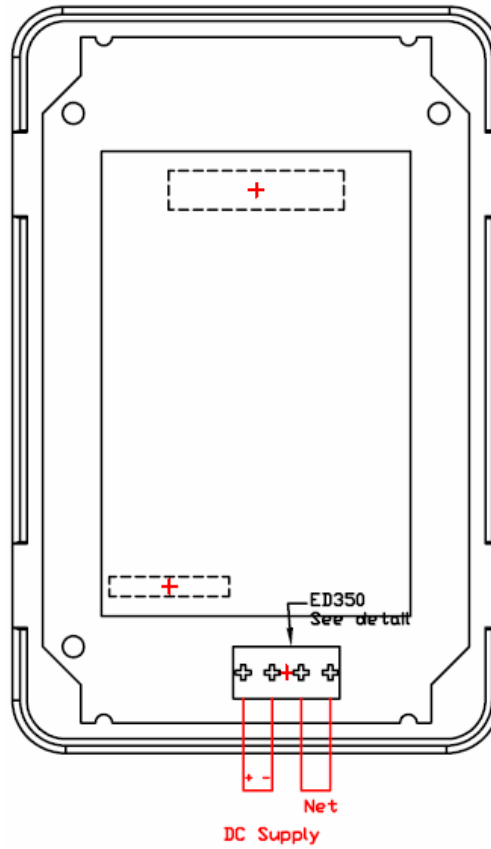


Figure 3.1 Connector pin assignment LonWorks network

4.4 EMC

The 7600RF RF Receiver is a CE certified device according to the regulation 89/336/EEC for electron magnetic compatibility, modified by 92/31/EEC". Concerning the emission it fulfills classification B (living area) according to EN 55022A/B, EN 55024 A/B, concerning the interference sensibility.

4.5 Technical Specifications

CPU	Echelon Neuron 3150,10MHz	
Memory	4Kbytes flash EPROM,4Kbytes RAM	
LonWorks Transceiver	FT-X1	
Power supply	12~24VDC(24VDC is recommend)	
Power consumption	1.5w	
Connection	M2.5 screw (Pitch:3.5mm)	
Temperature	Operation	0 ~ +50
	Storage	-20 ~ +70
Admitted relative humidity	10 ~ 90%, non condensing	
Dimensions	60 x 94 x 30 mm,	
Mounting	Wall mounting	
Display & Operation	Service-pin and Reset LED indicator and button	
Max. connections	8 wireless remote controller devices	
Max trans len	10 Meters	

Table 3.1 Technical Specification

4.6 Dimensions

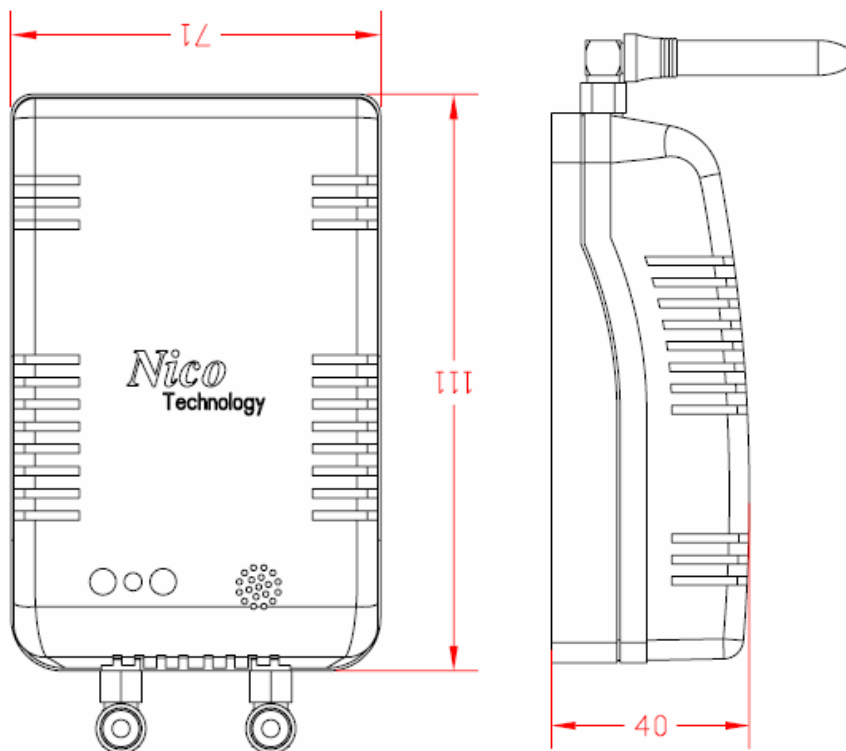


Figure 3.5 Device dimensions without plug-screw clamps

5 Application Software for RF Remote Controller

On the 7600RF RF Receiver an application for RF Remote Controller is implemented, making output functionality as well as switch control functions available.

Therefore the relevant LonMark profiles stated in Table 4.1 are implemented. The use of network variables (NV) complies with the LonMark standard, no customized network variables are used. SCPT's are used for parameterization by applying the read/write-memory method.

Title	Present Version	Identification
LonMark Application Layer Interoperability Guidelines	V3.1	078-0120-01D
The SNVT Master List and Programmer's Guide	V 8.0	
The SCPT Master List	V 8.0	
Virtual Functional Object		
RemoteControl		
VirtualKeypad		
NVTypeTranslator		

Table 4.1 Referring document about LonMark profiles

5.1 System Scope

The 7600RF RF Receiver is equipped between Wireless and LonWorks channels.

Each Wireless network allows then connection eight wireless RF Remote Controller. Virtual Functional Object according to LonMark Standard can be assigned to these output and configured.

Furthermore the 7600RF RF Receiver can act as constant RF remote controller.

The 7600RF RF Receiver is equipped with one LonWorks communication circuit to connect LonWorks network.

5.2 Interoperable Interface

The **LonMark profile is realized** in the 7600RF RF Receiver. As some customized NVs are used, the network interface remains standardized, clear and especially it is interoperable. That means, the 7600RF RF Receiver can be used in connection with network components by other manufactures. The following table contains a survey of the network variables defining the 7600RF RF Receiver network interface and their assignment.

NV Name	Type	Allocated Object
nvoRemoteState	SNVT_state	RemoteControl Object
nvoRemoteSwitch	SNVT_switch	RemoteControl Object
nviTranslator	Changeable Nv	NVTypeTranslator Object
nvoTranslator	Changeable Nv	NVTypeTranslator Object
SwInput	SNVT_switch	VirtualKeypad Object
SwitchCtrl	SNVT_switch	VirtualKeypad Object
SNVT_switch	SNVT_switch	VirtualKeypad Object

Table 4.4 Allocation of NVs and LonMark objects

Under the order code 7600RF a data carrier containing the interface describing files ***Nico_RemoteControl-5008RF_r1.XIF*** and the applications ***Nico_RemoteControl-5008RF_r1.APB*** is provided free of charge at simultaneous purchase of 7600RF. The XIF-file is necessary for integration with LonMaker for Windows or any other LonWorks network management tool.

6 System Objects

This chapter describes the LonMark objects implemented in the 7600RF RF Receiver. For each it states the network variable les used, special configuration properties, general object properties, response during modification of the configuration and after a reset, and, if available, further object properties.

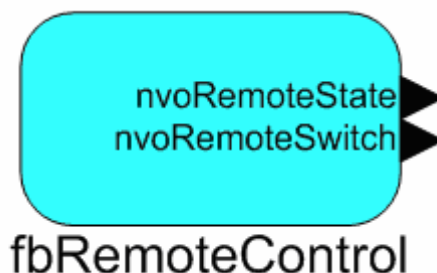
6.1 Node Object

The functionality of the node object is defined in the Application Layer Guidelines of LonMark Interoperability Association (www.lonmark.org).

Network Variables

NV Name	NV Type	Comment
nviRequest	SNVT_obj_request	Status request
nvoStatus	SNVT_obj_status	Status response
nvoAlarm	SNVT_alarm	Alarm generating
nvoFileDirectory	SNVT_address	Address of file for parameterization

6.2 Remote Control Object



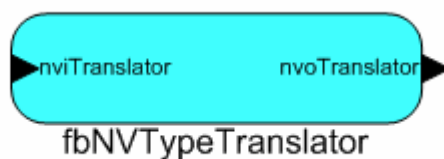
Network Variables

NV Name	NV Type	Comment
nvoRemoteState	SNVT_state	Single Remote controller's 1~ 8 button keys state.
nvoRemoteSwitch	SNVT_switch	Last RC button key value (pre-setting at UCPTswValTable)

Configuration Properties

CP Name	Comment
UCPTswValTable	Valid scope: swValue[0] ~ [7] swValue for Single Remote Control button key value mapping when received key 1~8 on Remote controller.

6.3 Nico Tech. NV Type Translator Object



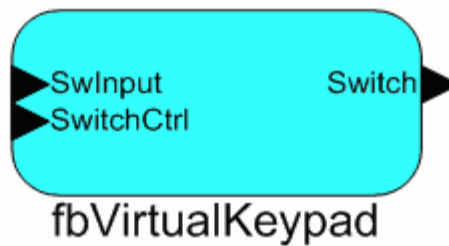
Network Variables

NV Name	NV Type	Comment
nviTranslator	SNVT_lev_percent(Changeable)	Type translator of Nv input value
nvoTranslator	SNVT_lev_percent(Changeable)	Type translator of Nv output value

Configuration Properties

CP Name	CP Type	CP Index	Comment
SCPTnvType	SCPTnvType	254	Nv Type nviTranslator
SCPTnvType	SCPTnvType	254	Nv Type of nvoTranslator
SCPTmaxNVlength	SCPTmaxNVlength	255	Read only; Maximum Nv type length of nviTranslator
SCPTmaxNVlength	SCPTmaxNVlength	255	Read only; Maximum Nv type length of nvoTranslator
SCPTobjMajVer	SCPTobjMajVer	167	Read only; Must read from device
SCPTobjMinVer	SCPTobjMinVer	168	Read only; Must read from device
SCPTtrnsTblX	SCPTtrnsTblX	28	Used in conjunction with Translation table X to scale and linearity a value
SCPTtrnsTblY	SCPTtrnsTblY	29	Used in conjunction with Translation table Y to scale and linearity a value

6.4 Virtual Keypad Object



Network Variables

NV Name	NV Type	Comment
SwInput	SNVT_switch	Reserve for simulate button key behavior
SwitchCtrl	SNVT_switch	SwitchCtrl is Switch Control abbreviation. Responsible for receive data (may from other devices) thru SwitchCtrl transfer into Virtual Keypad Object. Inside of Virtual Keypad Object has simple algorithm for process data. When data was receive from SwitchCtrl. All the behaviors of SwitchCtrl are following setting value of UCPTkeyPadCP and UCPTswVaTable.
Switch	SNVT_switch	Switch is Responsible for logical output. Switch logical output usually connects to physical device. For example. A Switch network variable connects to a Dimmer Controller or Power Controller. It's a lighting control functional.

Configuration Properties

CP Name	CP Type	Index	Comment
UCPTkeyPadCP	UCPTkeyPadCP	10	Responsible control algorithm reference's value. When SwitchCtrl receive a data change event, and Virtual Keypad Object will to execute setting value within UCPTkeyPadCP to output effect on the Switch Network Variable. All the behavior of Virtual Keypad object control apply to this property.

			In <i>Virtual KeyPad Object</i> usually setting the entire control algorithm and working together with SwitchCtrl network variable value.
ctrlCMD.n.limitMin	Setting value of process button message id minimal id		
ctrlCMD.n.limitMax	Setting value of process button message id maximal id.		
ctrlCMD.n.ctrlType	Setting what kind of process will to do. When V_SwitchCtrl received a data from another LonWorks device(s). About ctrl_Type enumerate list as following shown.		
UCPTswValTable			UCPTswValTable is for all the Switch Value Table define. Responsible when UCPTkeyPadCP value assign to ctrl_CMD , and ctrl_Type of ctrl_CMD equal to Ctrl_PresetInex reference value. All of the value data are store in this property.

ctrl_Type enumerate list

Name of enumerate	Description
Ctrl_NOP	Nothing to do. When V_SwitchCtrl network variable received data value.
Ctrl_PresetValue	Direct pass value. When V_SwitchCtrl network variable received data value to V_Switch network variable. It's for output purpose. Usually, in 5006L programming model first Virtual KeyPad Object should setting on this.
Ctrl_PresetIndex	Get value from index table. The UCPTswValTable configure property is an index table for Ctrl_PresetIndex parameter. If selected Ctrl_PresetIndex and then you have to setting all the value in the UCPTswValTable configure property.
Ctrl_InactiveSw	Setting button as an Inactive (Off status) command. When button was release. (also call this for Off status).

	<p>Usually, Ctrl_InactiveSw working together with Ctrl_ActiveSw.</p> <p>If selected Ctrl_InactiveSw and then you have to setting a value in inactiveSw.value item and then value of inactiveSw.value will output to V_Switch network variable.</p>
Ctrl_ActiveSw	<p>Setting button as an Active (On status) command.</p> <p>When button was press. (also call this for On status)</p> <p>Usually, Ctrl_InactiveSw working together with Ctrl_InactiveSw.</p> <p>If selected Ctrl_InactiveSw and then you have to setting a value in activeSw.value item and then value of activeSw.value will output to V_Switch network variable.</p>
Ctrl_DecValue	<p>Setting button as a Decrease value command.</p> <p>When button was press, how many decrease value will output to V_Switch network variable.</p> <p>Usually, Ctrl_DecValue working together with Ctrl_IncValue item.</p> <p>If selected Ctrl_DecValue and then you have to setting a value in decSwVal item.</p>
Ctrl_IncValue	<p>Setting button as a Increase value command.</p> <p>When button was press, how many increase value will output to V_Switch network variable.</p> <p>Usually, Ctrl_IncValue working together with Ctrl_DecValue item.</p> <p>If selected Ctrl_IncValue and then you have to setting a value in incSwVal item.</p>

Collection all of above explain its can found the category of configure rule as following:

1. Single button to present an On/Off status. This is a Toggle functional button.

[Notion of configure]

Setting value of ctrl_CMDn.ctrl_ctrlType equal to Ctrl_ActiveSw or Ctrl_InactiveSw and then setting value of activeSw.value or inactiveSw.value item.

2. Single button to present a setting status. This is a Setting functional button.

[Notion of configure]

Setting value of ctrl_CMDn.ctrl_ctrlType equal to Ctrl_ActiveSw and then

setting value of activeSw.value item.

3. Single button to present an Increase value status. This is a Dimming Up functional button.

[Notion of configure]

Setting value of ctrl_CMDn.ctrl_ctrlType equal to Ctrl_IncValue and then setting value of incSwVal item.

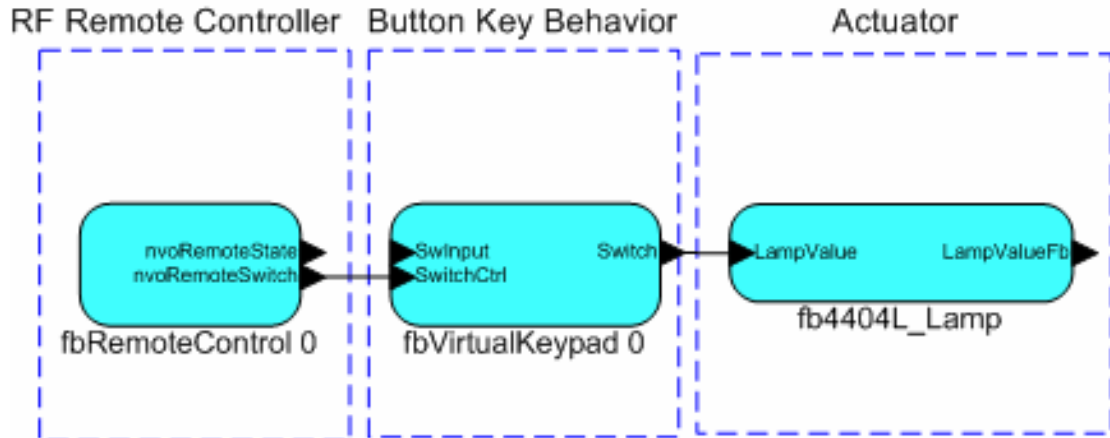
4. Single button to present a Decrease value status. This is a Dimming Down functional button.

[Notion of configure]

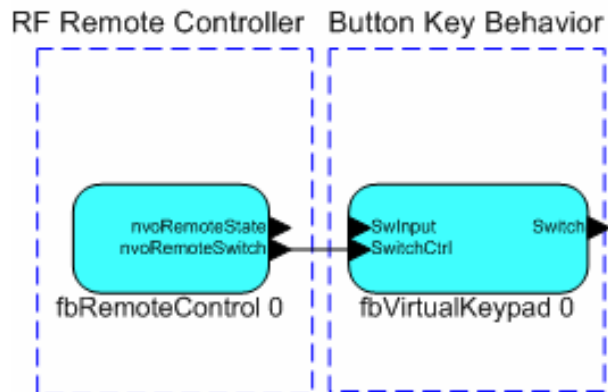
Setting value of ctrl_CMDn.ctrl_ctrlType equal to Ctrl_DecValue and then setting value of decSwVal item.

6.5 Programming Examples

Remote Controller binding with 4404L or other LonWorks device or object.



Remote Controller binding with 5005/6L Touch Keypad's Virtual Functional Block



6.6 DIP Switch Setting

DIP Switch overview

There are two kind of DIP Switch for 7600RF Receiver and 5008RF Remote Controller. On 7600RF Receiver have 5 DIP switch for configure setting Wireless network ID.

On 5508RF Remote Controller have 8 DIP switch for configure setting Wireless network ID (NET ID) and Remote Controller ID(RC ID).

In order to wireless devices can working correct, you have to configure correct NET ID on 5508RF and 7600RF devices.

6.6.1 DIP Switch for ON 5508RF RF Remote Controller

The DIP Switch is setting by binary and each 7600RF RF Receiver must configure as unique "Network ID". The valid channel id was between 0 and 31. DIP switch 4 ~ 8 are for setup Network ID (NET ID). 1 ~ 3 are for setting Remote Controller ID (RC ID).

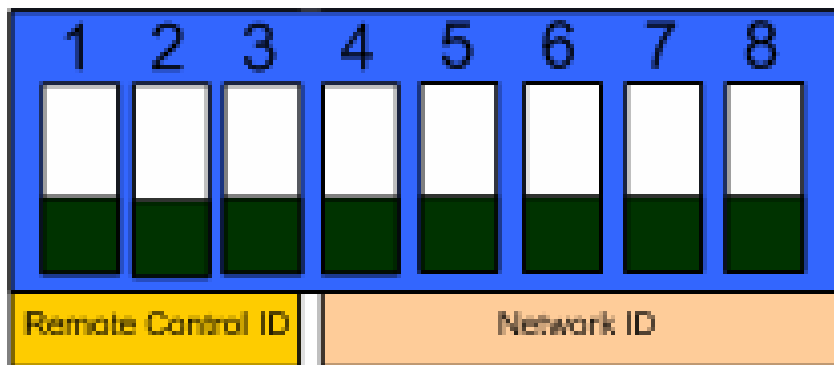


Figure 6.1 Identify DIP Switch for Remote Controller ID and Wireless network ID

6.6.2 Wireless network Channel ID Setting

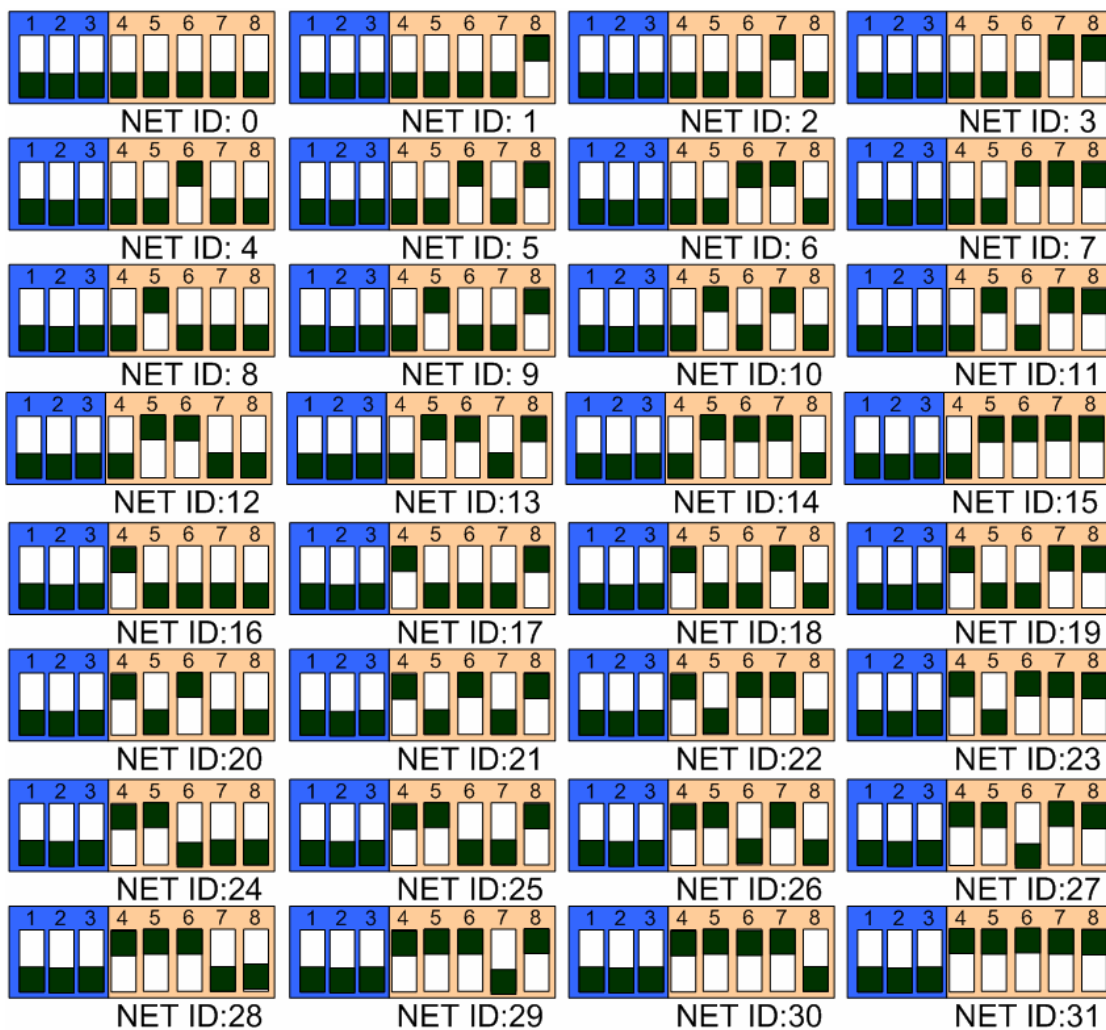


Figure 6.2 DIP Switch for Wireless network ID

6.6.3 Remote Controller ID Setting

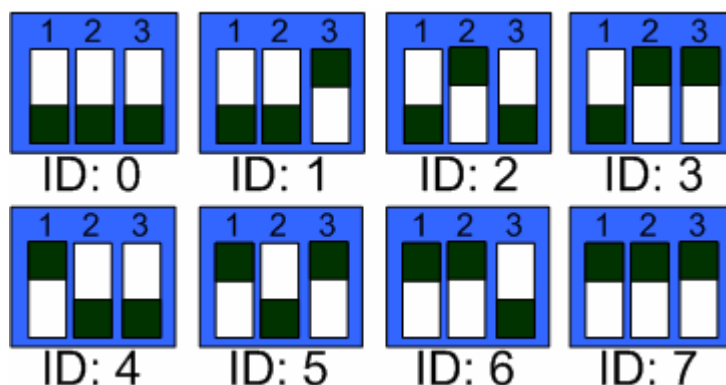


Figure 6.3 DIP Switch for Remote Controller ID