

# LR5-LAN Socket Communication

## Sample Program

### (Windows VB.NET)

## content

LR5-LAN Socket Communication Sample Program (Windows VB.NET)	1
1. Overview	4
1.1. System Overview	4
2. Development Environment	4
3. Application Overview	5
3.1. Command Operation	5
3.1.1. Command list	5
3.1.2. Operation control command	6
3.1.3. Clear Command	6
3.1.4. Status Acquisition Command	6
3.2. Function Description	7
3.2.1. Function List	7
3.2.2. Connect to LR5-LAN	8
3.2.3. close socket	8
3.2.4. Send Command	9
3.2.5. PNS Command Operation Control Command Transmission	10
3.2.6. Send Clear Command For PNS Command	12
3.2.7. Send PNS Command Status Acquisition Command	13
3.3. Constant Description	14
3.3.1. Product Differentiation	14
3.3.2. PNS Command Identifier	14
3.3.3. PNS Command Response Data	14
3.3.4. LED unit pattern for operation control commands	14
3.3.5. Buzzer pattern for operation control commands	15
3.4. Structure Description	16
3.4.1. Motion control data structure	16
3.4.2. Operation control status data	16
4. Program Overview	17
4.1. Connect to LR5-LAN	17
4.2. close socket	18
4.3. Send Command	18
4.4. PNS Command Operation Control Command Transmission	19
4.5. Send Clear Command For PNS Command	20

4.6.	Send PNS Command Status Acquisition Command.....	21
------	--	----

## 1. Overview

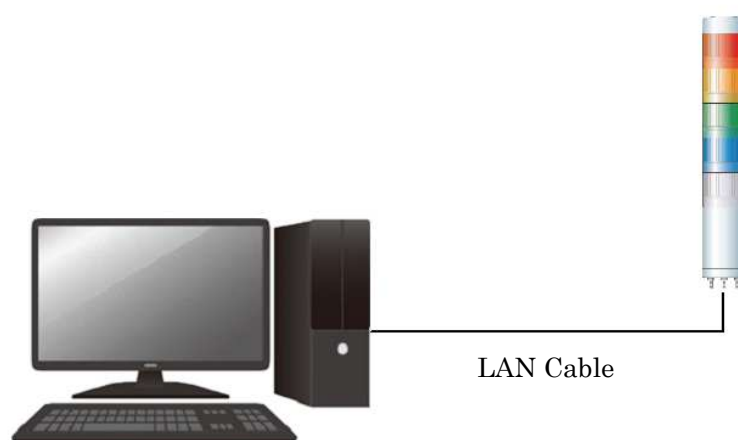
This is an outline of sample programming to control LR5-LAN via socket communication.

The programs are intended to control the unit using Microsoft Visual Basic .NET without using the DLLs provided by PATLITE.いる。

### 1.1. System Overview

The system configuration diagram of this program is as follows.

The sample program controls one LR5-LAN by socket communication.



## 2. Development Environment

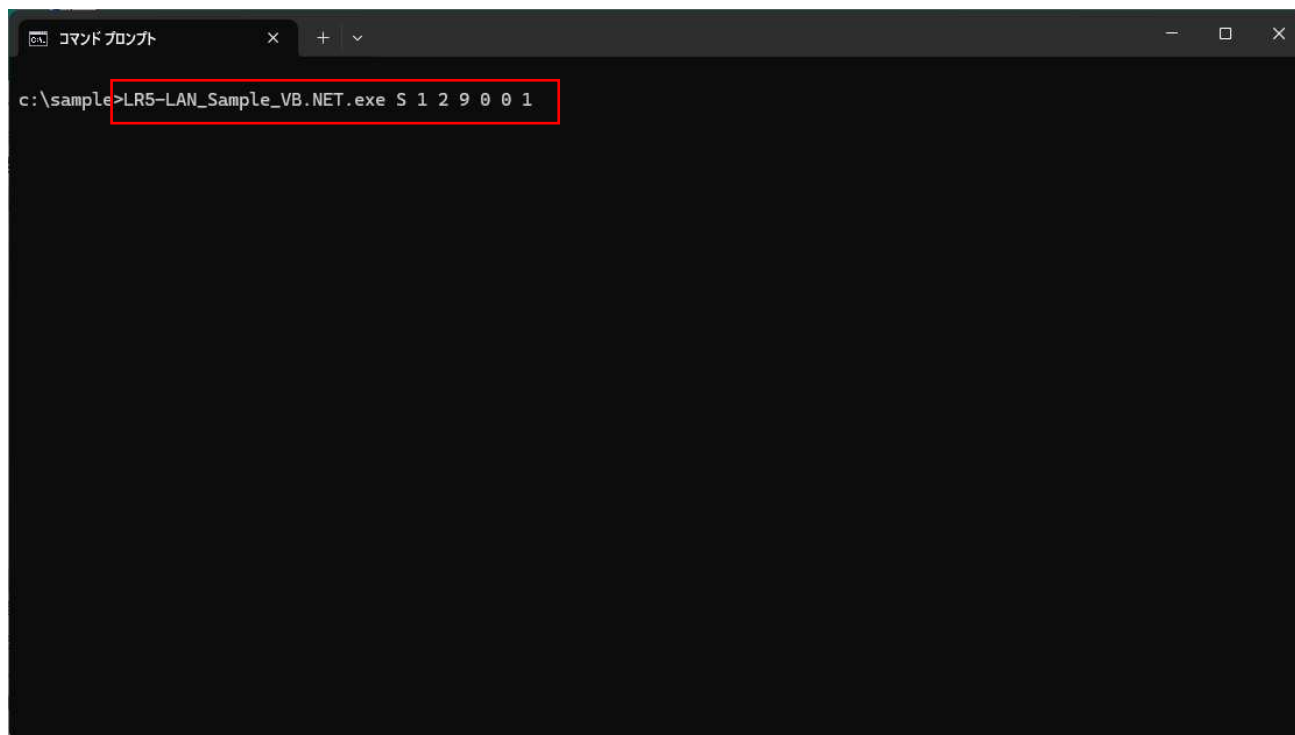
The development environment of the sample program is shown below.

Development Environment		Remarks
Development OS	Windows11 64bit	
Development Language	VB.NET	.Net Framework 4.5 Subsequent
Application	CUI APPLICATION	
Development tool	VisualStudio2022 Professional	

## 3. Application Overview

### 3.1. Command Operation

Open Command Prompt, navigate to where LR5-LAN\_Sample\_VB.NET.exe created during the build is located and specify the command line arguments to execute commands for each operation.



#### 3.1.1. Command list

command name	content
Operation control command	Control each color pattern and buzzer (On/Off) of the LED unit
Clear Command	Turn off the LED unit and turn off the buzzer
Status Acquisition Command	Used To Acquire Status Of Signal Lines And The Status Of The Led Unit And Alarm..

## 3.1.2. Operation control command

Execute command with the following command line arguments

No.	Command Line Argument	Value
1	Command ID	S
2	LED Unit Red	Off: 0
3	LED Unit Amber	On: 1
4	LED Unit Green	Flashing(slow): 2
5	LED Unit Blue	Flashing(medium): 3
6	LED Unit White	Flashing(fast): 4 Single flash: 5 Double flash: 6 Triple flash: 7 No change: 9
7	Alarm Pattern	Off: 0 On: 1 No change: 9

e.g.): LR5-LAN\_Sample\_VB.NET.exe S 1 2 9 0 0 1

## 3.1.3. Clear Command

Execute command with the following command line arguments

No.	Command Line Argument	Value
1	Command ID	C

e.g.): LR5-LAN\_Sample\_VB.NET.exe C

## 3.1.4. Status Acquisition Command

Execute command with the following command line arguments

No.	Command Line Argument	Value
1	Command ID	G

e.g.): LR5-LAN\_Sample\_VB.NET.exe G

## 3.2. Function Description

### 3.2.1. Function List

Function Name	Explanation
SocketOpen	Connect to LR5-LAN
SocketClose	close the socket
SendCommand	send command
PNS_RunControlCommand	Send PNS command operation control commands
PNS_ClearCommand	Send clear PNS command
PNS_GetDataCommand	Send PNS Command Status Acquisition Command

## 3.2.2. Connect to LR5-LAN

Function Name	Public Function SocketOpen(ByVal ip As String, ByVal port As Integer) As Integer	
Parameters	ByVal ip As String	LR5-LAN IP address
	ByVal port As Integer	LR5-LAN port number
Return Value	Integer	Success: 0, Failure: other than 0
Explanation	Connect to LR5-LAN with specified IP address and port number using socket communication	
How to use functions	<pre>' Definition of Socket class variables Private sock As Socket = Nothing  ' Main function Sub Main()     ' Connect to LR5-LAN     Dim ret As Integer     ret = SocketOpen("192.168.10.1", 10000)     If ret = -1 Then         Return     End If End Sub</pre>	
Remarks	Please refer to 「4.1Connect to LR5-LAN」For The Program Overview.	

## 3.2.3. close socket

Function Name	Public Sub SocketClose()	
Parameters	None	
Return Value	None	
Explanation	Close the socket connected to LR5-LAN	
How to use functions	<pre>' Main function Sub Main()     ' Connect to LR5-LAN     Dim ret As Integer     ret = SocketOpen("192.168.10.1", 10000)     If ret = -1 Then         Return     End If      ' close socket     SocketClose() End Sub</pre>	
Remarks	Please refer to 「4.2close socket」For The Program Overview.	



## 3.2.4. Send Command

Function Name	Public Function SendCommand(ByVal sendData As Byte(), ByRef recvData As Byte()) As Integer	
Parameters	ByVal sendData As Byte()	Transmission Data
	ByRef recvData As Byte()	Received Data
Return Value	Integer	Success: 0, Failure: other than 0
Explanation	Send data to the connected LR5-LAN and return response data	
How to use functions	<pre> ' Main function Sub Main()     ' Connect to LR5-LAN     Dim ret As Integer     ret = SocketOpen("192.168.10.1", 10000)     If ret = -1 Then         Return     End If      ' Create transmission data     Dim sendData As Byte(7)     Dim recvData As Byte()     sendData(0) = &amp;H41     sendData(1) = &amp;H42     sendData(2) = &amp;H53     sendData(3) = &amp;H0     sendData(4) = &amp;H0     sendData(5) = &amp;H0     sendData(6) = &amp;H1      ' Send Command     ret = SendCommand(sendData, recvData)     If ret &lt;&gt; 0 Then         Debug.WriteLine("failed to send data")         Return -1     End If      ' close socket     SocketClose() End Sub </pre>	
Remarks	Please refer to 「4.3Send Command」For The Program Overview.	

## 3.2.5. PNS Command Operation Control Command Transmission

Function Name	Public Function PNS_RunControlCommand(ByVal runControlData As PNS_RUN_CONTROL_DATA) As Integer	
Parameters	ByVal runControlData As PNS_RUN_CONTROL_DATA	TRANSMISSION DATA THAT CONTROLS EACH COLOR PATTERN AND BUZZER OF THE LED UNIT For Details, See 「3.3.1Motion control data structure」For The Program Overview.
Return Value	Integer	Success: 0, Failure: other than 0
Explanation	Send PNS command operation control commands to control each color pattern and buzzer of the led unit	
How to use functions	<pre> ' Main function Sub Main()     ' Connect to LR5-LAN     Dim ret As Integer     ret = SocketOpen("192.168.10.1", 10000)     If ret = -1 Then         Return     End If      ' PNS Command Operation Control Command Transmission     ' Led pattern0: Off     ' Led pattern1: On     ' Led pattern2: Flashing(slow)     ' Led pattern3: Flashing(medium)     ' Led pattern4: Flashing(fast)     ' Led pattern5: Single flash     ' Led pattern6: Double flash     ' Led pattern7: Triple flash     ' Led pattern9: No change     ' Alarm Pattern0: Off     ' Alarm Pattern1: On     ' Alarm Pattern9: No change     Dim runControlData As PNS_RUN_CONTROL_DATA = New PNS_RUN_CONTROL_DATA With {         .ledRedPattern = PNS_RUN_CONTROL_LED_ON,         .ledAmberPattern = PNS_RUN_CONTROL_LED_BLINKING_SLOW,         .ledGreenPattern = PNS_RUN_CONTROL_LED_NO_CHANGE,         .ledBluePattern = PNS_RUN_CONTROL_LED_OFF,         .ledWhitePattern = PNS_RUN_CONTROL_LED_FLASHING_TRIPLE,         .buzzerPattern = PNS_RUN_CONTROL_BUZZER_RING     }     PNS_RunControlCommand(runControlData)      ' close socket     SocketClose() End Sub </pre>	
Remarks	Please refer to 「4.4PNS Command Operation Control Command Transmission」For	



## 3.2.6. Send Clear Command For PNS Command

Function Name	Public Function PNS_ClearCommand() As Integer	
Parameters	None	
Return Value	Integer	Success: 0, Failure: other than 0
Explanation	Send the PNS clear command to turn off the led unit and stop the buzzer	
How to use functions	<pre>' Main function Sub Main()     ' Connect to LR5-LAN     Dim ret As Integer     ret = SocketOpen("192.168.10.1", 10000)     If ret = -1 Then         Return     End If      ' Send Clear Command For PNS Command     PNS_ClearCommand()      ' close socket     SocketClose() End Sub</pre>	
Remarks	Please refer to 「4.5Send Clear Command For PNS Command」For The Program Overview.	

## 3.2.7. Send PNS Command Status Acquisition Command

Function Name	Public Function PNS_GetDataCommand(ByRef statusData As PNS_STATUS_DATA) As Integer	
Parameters	ByRef statusData As PNS_STATUS_DATA	Status Acquisition Command の Received Data(LED UNIT AND BUZZER STATUS) For Details, See 「3.3.3Operation control status data」For The Program Overview.
Return Value	Integer	Success: 0, Failure: other than 0
Explanation	Send the status acquisition command of the PNS command to acquire the status of the led unit and buzzer.	
How to use functions	<pre>' Main function Sub Main()     ' Connect to LR5-LAN     Dim ret As Integer     ret = SocketOpen("192.168.10.1", 10000)     If ret = -1 Then         Return     End If      ' Send PNS Command Status Acquisition Command     Dim statusData As PNS_STATUS_DATA = New PNS_STATUS_DATA     PNS_GetDataCommand(statusData)      ' close socket     SocketClose() End Sub</pre>	
Remarks	Please refer to 「4.6Send PNS Command Status Acquisition Command」For The Program Overview.	

### 3.3. Constant Description

#### 3.3.1. Product Differentiation

Constant name	Value	Explanation
PNS_PRODUCT_ID	0x4142	LR5-LAN product classification

#### 3.3.2. PNS Command Identifier

PNS_RUN_CONTROL_COMMAND	0x53	Operation control command
PNS_CLEAR_COMMAND	0x43	Clear Command
PNS_GET_DATA_COMMAND	0x47	Status Acquisition Command

#### 3.3.3. PNS Command Response Data

Constant name	Value	Explanation
PNS_ACK	0x06	Normal Response
PNS_NAK	0x15	Abnormal Response

#### 3.3.4. LED unit pattern for operation control commands

Constant name	Value	Explanation
PNS_RUN_CONTROL_LED_ON	0x00	Off
PNS_RUN_CONTROL_LED_OFF	0x01	On
PNS_RUN_CONTROL_LED_BLINKING_SLOW	0x02	Flashing(slow)
PNS_RUN_CONTROL_LED_BLINKING_MEDIUM	0x03	Flashing(slow)
PNS_RUN_CONTROL_LED_BLINKING_HIGH	0x04	Flashing(slow)
PNS_RUN_CONTROL_LED_FLASHING_SINGLE	0x05	Single flash
PNS_RUN_CONTROL_LED_FLASHING_DOUBLE	0x06	Double flash
PNS_RUN_CONTROL_LED_FLASHING_TRIPLE	0x07	Triple flash
PNS_RUN_CONTROL_LED_NO_CHANGE	0x09	No change

## 3.3.5. Buzzer pattern for operation control commands

Constant name	Value	Explanation
PNS_RUN_CONTROL_BUZZER_STOP	0x00	Off
PNS_RUN_CONTROL_BUZZER_RING	0x01	On
PNS_RUN_CONTROL_BUZZER_NO_CHANGE	0x09	No change

## 3.4. Structure Description

### 3.4.1. Motion control data structure

名前	PNS_RUN_CONTROL_DATA
Definition	<pre> Public Class PNS_RUN_CONTROL_DATA     ' LED Unit Red pattern     Public ledRedPattern As Byte = 0     ' LED Unit Amber pattern     Public ledAmberPattern As Byte = 0     ' LED Unit Green pattern     Public ledGreenPattern As Byte = 0     ' LED Unit Blue pattern     Public ledBluePattern As Byte = 0     ' LED Unit White pattern     Public ledWhitePattern As Byte = 0     ' Buzzer status     Public buzzerMode As Byte = 0 End Class </pre>
Explanation	Structure of each color pattern and buzzer status of the LED unit in the data area sent by operation control command

### 3.4.2. Operation control status data

名前	PNS_STATUS_DATA
Definition	<pre> Public Class PNS_STATUS_DATA     ' Led pattern1~5     Public ledPattern As Byte() = New Byte(5) {}     ' Buzzer Mode     Public buzzer As Byte = 0 End Class </pre>
Explanation	Operation control Status Acquisition Command response data LED UNIT AND BUZZER STATUS structure



## 4. Program Overview

Describe only the main points of the program's operation.

### 4.1. Connect to LR5-LAN

Program	Explanation
Main.vb  <pre>Private sock As Socket = Nothing</pre>	→Definition of socket member variables
Main.vb SocketOpen()  <pre>Public Function SocketOpen(ByVal ip As String, ByVal port   Try     ' Set the IP address and port     Dim ipAddress As IPAddress = IPAddress.Parse(ip)     Dim remoteEP As IPEndPoint = New IPEndPoint(ipAdc     sock = New Socket(ipAddress.AddressFamily, Socket      ' Create a socket     If sock Is Nothing Then       Console.WriteLine("failed to create socket")       Return -1     End If      ' Connect to LA-POE     sock.Connect(remoteEP)    Catch ex As Exception     Console.WriteLine(ex.Message)     SocketClose()     Return -1   End Try    Return 0 End Function</pre>	→Specify the device IP address and port number Default IP address: 192.168.10.1 Default port number: 10000 →Create a socket instance →Connect to the device using the socket Connect function

## 4.2. close socket

Program	Explanation
<pre> Main.vb SocketClose()  Public Sub SocketClose()     If sock IsNot Nothing Then         ' Close the socket.         sock.Shutdown(SocketShutdown.Both)         sock.Close()     End If End Sub </pre>	<p>→Shut down the socket and then call close</p>

## 4.3. Send Command

Create transmission data in the transmission data format for each command and send the command data to LR5-LAN  
Please refer to 「4.4PNS Command Operation Control Command Transmission」 and onwards for the transmission data format of each command.

Program	Explanation
<pre> Main.vb SendCommand()  If sock Is Nothing Then     Console.WriteLine("socket is not")     Return -1 End If  ' Send ret = sock.Send(sendData) If ret &lt; 0 Then     Console.WriteLine("failed to send")     Return -1 End If  ' Receive response data Dim bytes As Byte() = New Byte(1023) {} Dim recvSize As Integer = sock.Receive(bytes) If recvSize &lt; 0 Then     Console.WriteLine("failed to recv")     Return -1 End If  recvData = New Byte(recvSize - 1) {} Array.Copy(bytes, recvData, recvSize) </pre>	<p>→Send the created Transmission Data using the Send function</p> <p>→After sending, use the Receive function to get the response from the device.</p>

## 4.4. PNS Command Operation Control Command Transmission

Program	Explanation
<pre> Main.vb PNS_RunControlCommand() Dim sendData As Byte() = {}  ' 製品区分(AB) sendData = sendData.Concat(BitConverter.GetBytes(PNS_PROD))  ' コマンド識別子(S) sendData = sendData.Concat(New Byte() {PNS_RUN_CONTROL_CC})  ' 空き(0) sendData = sendData.Concat(New Byte() {0}).ToArray()  ' データサイズ、データエリア Dim data As Byte() = {     runControlData.ledRedPattern,      ' LEDユニット赤色の     runControlData.ledAmberPattern,    ' LEDユニット黄色     runControlData.ledGreenPattern,    ' LEDユニット緑色     runControlData.ledBluePattern,     ' LEDユニット青色     runControlData.ledWhitePattern,    ' LEDユニット白色     runControlData.buzzerMode          ' ブザーの状態 }  sendData = sendData.Concat(BitConverter.GetBytes(CUShort( sendData = sendData.Concat(data).ToArray()  ' PNSコマンドを送信 Dim recvData As Byte() = {} ret = SendCommand(sendData, recvData) If ret &lt;&gt; 0 Then     Console.WriteLine("failed to send data")     Return -1 End If  ' 応答データを確認 If recvData(0) = PNS_NAK Then     ' 異常応答を受信     Console.WriteLine("negative acknowledge")     Return -1 End If </pre>	<p>Create Transmission Data in the following order</p> <p>→ 1st byte : Product Differentiation (A : 0x41)</p> <p>→ 2nd byte : Product Differentiation (B : 0x42)</p> <p>→ 3rd byte : ID (S : 0x53)</p> <p>→ 4th byte : Unused (0x00)</p> <p>→ 5th byte : Data Size (0x00)</p> <p>→ 6th byte : Data Size (0x06)</p> <p>→ 7~11 : Data Area</p> <p>Data size is 6 bytes</p> <p>Data Set the value of "3.3.1 Motion control data structure" in the Data Area.</p> <p>→ Call "4.3 Send Command/Receive" and send data to the device</p> <p>→ Check response data after sending</p> <p>Normal Response : ACK (0x06)</p> <p>Abnormal Response : NAK (0x15)</p>

## 4.5. Send Clear Command For PNS Command

Program	Explanation
<pre> Main.vb PNS_ClearCommand()  Dim sendData As Byte() = {}  ' Product Category (AB) sendData = sendData.Concat(BitConverter.GetBytes(PNS_PROD))  ' Command identifier (C) sendData = sendData.Concat(New Byte() {PNS_CLEAR_COMMAND})  ' Empty (0) sendData = sendData.Concat(New Byte() {0}).ToArray()  ' Data size sendData = sendData.Concat(BitConverter.GetBytes(CUShort(0)))  ' Send PNS command Dim recvData As Byte() = {} ret = SendCommand(sendData, recvData) If ret &lt;&gt; 0 Then     Console.WriteLine("failed to send data")     Return -1 End If  ' check the response data If recvData(0) = PNS_NAK Then     ' receive abnormal response     Console.WriteLine("negative acknowledge")     Return -1 End If </pre>	<p>Create Transmission Data in the following order</p> <ul style="list-style-type: none"> <li>→1st byte:Product Differentiation(A:0x41)</li> <li>→:Product Differentiation(B:0x42)</li> <li>→3rd byte:ID(C:0x43)</li> <li>→4th byte:Unused(0x00)</li> <li>→5th byte:Data Size(0x00)</li> <li>→6th byte:Data Size(0x00)</li> </ul> <p>Data size is 0 bytes</p> <p>No data area</p> <p>→Call “4.3 Send Command/Receive” and send data to the device</p> <p>→Check response data after sending</p> <p>Normal Response:ACK(0x06)</p> <p>Abnormal Response:NAK(0x15)</p>

## 4.6. Send PNS Command Status Acquisition Command

Program	Explanation
<pre> Main.vb PNS_GetDataCommand() Dim sendData As Byte() = {} ' Product Category (AB) sendData = sendData.Concat(BitConverter.GetBytes(PNS_PROD ' Command identifier (G) sendData = sendData.Concat(New Byte() {PNS_GET_DATA_COMMA ' Empty (0) sendData = sendData.Concat(New Byte() {0}).ToArray() ' Data size sendData = sendData.Concat(BitConverter.GetBytes(CShort(0 ' Send PNS command Dim recvData As Byte() = {} ret = SendCommand(sendData, recvData) If ret &lt;&gt; 0 Then     Console.WriteLine("failed to send data")     Return -1 End If ' check the response data If recvData(0) = PNS_NAK Then     ' receive abnormal response     Console.WriteLine("negative acknowledge")     Return -1 End If ' LED Pattern 1 to 5 statusData.ledPattern = New Byte(5) {} Array.Copy(recvData, statusData.ledPattern, statusData.le ' Buzzer Mode statusData.buzzer = recvData(5) </pre>	<p>Create Transmission Data in the following order</p> <ul style="list-style-type: none"> <li>→1st byte:Product Differentiation(A:0x41)</li> <li>→:Product Differentiation(B:0x42)</li> <li>→3rd byte:ID(G:0x47)</li> <li>→4th byte:Unused(0x00)</li> <li>→5th byte:Data Size(0x00)</li> <li>→6th byte:Data Size(0x00)</li> </ul> <p>Data size is 0 bytes</p> <p>No data area</p> <p>→Call “4.3 Send Command/Receive” and send data to the device</p> <p>→Check response data after sending</p> <p>Normal Response: The response data in “3.3.3 Operation control status data” is obtained.</p> <p>Abnormal Response:NAK(0x15)</p> <p>Acquire each data of response data using the following process.</p> <ul style="list-style-type: none"> <li>→1st to 5th byte:LED UNIT STATUS</li> <li>•1st byte:LED Unit Redstatus</li> <li>•:LED Unit Amberstatus</li> <li>•3rd byte:LED Unit Greenstatus</li> <li>•4th byte:LED Unit Bluestatus</li> <li>•5th byte:LED Unit Whitestatus</li> <li>•6th byte:Buzzer status</li> </ul>