1. Communication Overview

To control Dynamixel, communication should be established according to the protocol of Dynamixel. Dynamixel is driven by receiving binary data. Examples of programs for the transmission of this kind of data are described in detail in the User's Manual of the Dynamixel-only controller or the USB2Dynamixel. Thus, this manual describes only the method and protocol of communication used in Dynamixel on the assumption that Main Controller can transfer binary data.

1. 1. Packet

Main Controller and Dynamixel communicate each other by sending and receiving data called Packet. Packet has two kinds: Instruction Packet, which Main Controller sends to control Dynamixel, and Status Packet, which Dynamixel responses to Main Controller.

1. 2. ID

ID is a specific number for distinction of each Dynamixel when several Dynamixels are linked to one bus. By giving IDs to Instruction and Status Packets, Main Controller can control only the Dynamixel that you want to control

1. 3. Protocol

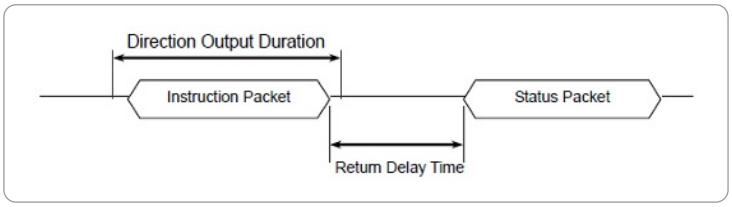
Dynamixel does the Asynchronous Serial Communication with 8 bit, 1 Stop bit, and None Parity.

If Dynamixel with the same ID is connected, packet will collide and network problem will occur. Thus, set ID as such that there is no Dynamixel with the same ID.

ID of Dynamixel is changeable. For this change, please refer to Changing IDs of Dynamixel. The factory default setting ID is 1.

1. 4. Half Duplex

Half duplex UART is a serial communication protocol where both TxD and RxD cannot be used at the same time. This method is generally used when many devices need to be connected to a single bus. Since more than one device are connected to the same bus, all the other devices need to be in input mode while one device is transmitting. The Main Controller that controllers the Dynamixel actuators sets the communication direction to input mode, and only when it is transmitting an Instruction Packet, it changes the direction to output mode.



1. 5. Tx, Rx Direction

For Half Duplex UART, the transmission ending timing is important to change the direction to receiving mode. The bit definitions within the register that indicates UART_STATUS are as the following

- **TXD_BUFFER_READY_BIT**: Indicates that the transmission DATA can be loaded into the Buffer. Note that this only means that the SERIAL TX BUFFER is empty, and does not necessarily mean that the all the data transmitted before has left the CPU.
- **TXD_SHIFT_REGISTER_EMPTY_BIT**: Set when all the Transmission Data has completed its transmission and left the CPU.

The **TXD_BUFFER_READY_BIT** is used when one byte is to be transmitted via the serial communication channel, and an example is shown below.

```
TxDByte(byte bData)
{
  while(!TXD_BUFFER_READY_BIT); //wait until data can be loaded.
  SerialTxDBuffer = bData; //data load to TxD buffer
}
```

When changing the direction, the **TXD_SHIFT_REGISTER_EMPTY_BIT** must be checked. The following is an example program that sends an Instruction Packet

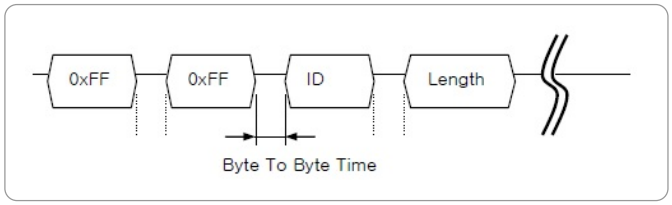
```
DIRECTION_PORT = TX_DIRECTION;
TxDByte(0xff);
TxDByte(0xff);
TxDByte(bID);
TxDByte(bLength);
TxDByte(bInstruction);
TxDByte(Parameter0); TxDByte(Parameter1); ...
DisableInterrupt(); // interrupt should be disable
TxDByte(Checksum); //last TxD
```

```
while(!TXD_SHIFT_REGISTER_EMPTY_BIT); //Wait till last data bit has been sent
DIRECTION_PORT = RX_DIRECTION; //Direction change to RXD
EnableInterrupt(); // enable interrupt again
```

NOTE : Please note the important lines between LINE 8 and LINE 12. Line 8 is necessary since an interrupt here may cause a delay longer than the return delay time and corruption to the front of the status packet may occur.

1. 6. Byte to Byte Time

The delay time between bytes when sending an instruction packet. If the delay time is over 100ms, then the Dynamixel actuator recognizes this as a communication problem and waits for the next header (0xff 0xff) of a packet again.



2. Instruction Packet

Instruction Packet is the command data sent to the Device.

Header1	Header2	ID	Length	Instruction	Param 1	...	Param N	Checksum
0xFF	0xFF	ID	Length	Instruction	Param 1	...	Param N	CHKSUM

2. 1. Header

The field indicates the start of the Packet.

2. 2. Packet ID

The field that indicates the ID of the Device that should receive the Instruction Packet and process it

1. Range : 0 ~ 253 (0x00 ~ 0xFD), which is a total of 254 numbers that can be used.
2. Broadcast ID : 254 (0xFE), which makes all connected devices execute the Instruction Packet.

2. 3. Length

The length of the Packet(Instruction, Parameter, Checksum fields). Length = number of Parameters + 2

2. 4. Instruction

The field that defines the type of instruction.

Value	Instructions	Description
0x01	Ping	Instruction that checks whether the Packet has arrived to a device with the same ID as Packet ID
0x02	Read	Instruction to read data from the Device
0x03	Write	Instruction to write data on the Device
0x04	Reg Write	Instruction that registers the Instruction Packet to a standby status; Packet is later executed through the Action instruction
0x05	Action	Instruction that executes the Packet that was registered beforehand using Reg Write
0x06	Factory Reset	Instruction that resets the Control Table to its initial factory default settings
0x08	Reboot	Instruction that reboots Dynamixel(See applied products in the description)
0x83	Sync Write	For multiple devices, Instruction to write data on the same Address with the same length at once
0x92	Bulk Read	For multiple devices, Instruction to write data on different Addresses with different lengths at once This command can only be used with MX series.

2. 5. Parameters

Parameters are used when additional data is required for an instruction.

2. 6. Instruction Checksum

It is used to check if packet is damaged during communication. Instruction Checksum is calculated according to the following formula.

Instruction Checksum = ~(ID + Length + Instruction + Parameter1 + … Parameter N)

Where “~” is the Binary Ones Complement operator. When the calculation result of the parenthesis in the above formula is larger than 255 (0xFF), use only lower bytes.

For example, when you want to use below Instruction Packet,

ID=1(0x01), Length=5(0x05), Instruction=3(0x03), Parameter1=12(0x0C), Parameter2=100(0x64), Parameter3=170(0xAA)

Checksum = ~ (ID + Length + Instruction + Parameter1 + … Parameter 3) = ~ [0x01 + 0x05 + 0x03 +

0x0C + 0x64 + 0xAA] = ~ [0x123] // Only the lower byte 0x23 executes the Not operation. = 0xDC

Thus, Instruction Packet should be 0xFF, 0xFF, 0x01, 0x05, 0x03, 0x0C, 0x64, 0xAA, 0xDC.

3. Status Packet(Return Packet)

Header1	Header2	ID	Length	Error	Param 1	...	Param N	Checksum
0xFF	0xFF	ID	Length	Error	Param 1	...	Param N	CHKSUM

3. 1. Error

This field displays the error status occurred during the operation of Dynamixel.

Bit	Error	Description
Bit 7	0	-
Bit 6	Instruction Error	In case of sending an undefined instruction or delivering the action instruction without the reg_write instruction, it is set as 1
Bit 5	Overload Error	When the current load cannot be controlled by the set Torque, it is set as 1
Bit 4	Checksum Error	When the Checksum of the transmitted Instruction Packet is incorrect, it is set as 1
Bit 3	Range Error	When an instruction is out of the range for use, it is set as 1
Bit 2	Overheating Error	When internal temperature of Dynamixel is out of the range of operating temperature set in the Control table, it is set as 1
Bit 1	Angle Limit Error	When Goal Position is written out of the range from CW Angle Limit to CCW Angle Limit , it is set as 1
Bit 0	Input Voltage Error	When the applied voltage is out of the range of operating voltage set in the Control table, it is as 1

For example, when Status Packet is returned as below

0xFF 0xFF 0x01 0x02 0x24 0xD8 It means that the error of 0x24 occurs from Dynamixel whose ID is 01. Since 0x24 is 00100100 as binary, Bit5 and Bit2 become 1. In order words, Overload and Overheating Errors have occurred.

NOTE : The error types on the table above are related to actuators, and the contents may vary depending on the type of Dynamixel.

3. 2. Status Checksum

It is used to check if packet is damaged during communication. Status Checksum is calculated according to the following formula.

Status Checksum = ~(ID + Length + Error + Parameter1 + … Parameter N)

4. Instruction Details

4. 1. Ping

This instruction requests the Status Packet from a specific ID. Even if Status Return Level(16) is 0, Dynamixel returns Status Packet all the time for Ping Instruction.

Length	Instruction	Parameter
0x02	0x01	-

4. 1. 1. Example

4. 1. 1. 1. Conditions

- ID 1(RX-64) is connected to the PC with an identical baudrate.

4. 1. 1. 2. Ping Instruction Packet

H1	H2	ID	LEN	INST	CKSM
0xFF	0xFF	0x01	0x02	0x01	0xFB

4. 1. 1. 3. ID 1 Status Packet

H1	H2	ID	LEN	ERR	CKSM
0xFF	0xFF	0x01	0x02	0x00	0xFC

4. 2. Read

This instruction is to read data in the Control Table of Dynamixel.

Length	Instruction	Param 1	Param 2
0x04	0x02	Starting Address of the Data	Length of Data to read

4. 2. 1. Example

4. 2. 1. 1. Conditions

- ID 1(RX-64) : Read Present Temperature, which is located at the address 43(0x2B)

4. 2. 1. 2. Read Instruction Packet

H1	H2	ID	LEN	INST	P1	P2	CKSM
0xFF	0xFF	0x01	0x04	0x02	0x2B	0x01	0xCC

4. 2. 1. 3. ID 1 Status Packet

H1	H2	ID	LEN	ERR	P1	CKSM
0xFF	0xFF	0x01	0x03	0x00	0x20	0xDB

4. 3. Write

This instruction is to write data to the Control Table of DYNAMIXEL

Length	Instruction	Param 1	Param 2	Param 3	Param N+1
N + 3	0x03	Starting Address of the Data	1st Byte	2nd Byte	Nth Byte

4. 3. 1. Example

4. 3. 1. 1. Conditions

- ID broadcast(RX-64) : Set the unknwon Dynamixel's ID as "1" by writing 1 to ID(3)

4. 3. 1. 2. Write Instruction Packet

H1	H2	ID	LEN	INST	P1	P2	CKSM
0xFF	0xFF	0xFE	0x04	0x03	0x03	0x01	0xF6

NOTE : Status Packet will not be returned if Broadcast ID(0xFE) is used.

4. 4. Reg Write

- Instruction that is similar to Write Instruction, but has an improved synchronization characteristic
- Write Instruction is executed immediately when an Instruction Packet is received.
- Reg Write Instruction registers the Instruction Packet to a standby status, and sets Control table Registered Instruction to '1'.
- When an Action Instruction is received, the registered Packet is executed, and sets Control Table Registered Instruction to '0'.

Length	Instruction	Param 1	Param 2	Param N+1
N+3	0x04	Starting Address of the Data	1st Byte	Nth Byte

4. 4. 1. Example

4. 4. 1. 1. Conditions

- ID 1(RX-64) : Reg Write 500(0x1F4) to Goal Position(30) and wait for Action instruction to move.

4. 4. 1. 2. Reg Write Instruction Packet

H1	H2	ID	LEN	INST	P1	P2	P3	CKSM
0xFF	0xFF	0x01	0x05	0x04	0x1E	0xF4	0x01	0xE2

4. 4. 1. 3. ID 1 Status Packet

H1	H2	ID	LEN	ERR	CKSM
0xFF	0xFF	0x01	0x02	0x00	0xFC

4. 5. Action

This instruction is to execute the registered Reg Write instruction. The Action instruction is useful when multiple Dynamixels are required to start moving at the same time. When several devices are controlled via communication, there is a minor time difference between enabling the first and last device. Dynamixel has resolved this problem by using Action instruction.

Length	Instruction	Parameter
0x02	0x05	-

4. 5. 1. Example

4. 5. 1. 1. Conditions

- All Dynamixels have received Reg Write instructions.

4. 5. 1. 2. Action Instruction Packet

H1	H2	ID	LEN	INST	CKSM
0xFF	0xFF	0xFE	0x02	0x05	0xFA

NOTE : Status Packet will not be returned if Broadcast ID(0xFE) is used.

4. 6. Factory Reset

This instruction is to reset the Control Table of Dynamixel to the factory default values.

CAUTION : Please be careful as Reset instruction will overwrite factory reset values in the EEPROM.

CAUTION : Broadcast ID(0xFE) cannot be used for Reset instruction.
Applied Products : MX-12W(V41), MX-28(V40), MX-64(V40), MX-106(V40), X-series(except XL-320), MX series with Protocol 2.0

Length	Instruction	Parameter
0x02	0x06	-

4. 6. 1. Example

4. 6. 1. 1. Conditions

- ID 0(RX-64) : Factory Reset the Dynamixel

4. 6. 1. 2. Factory Reset Instruction Packet

H1	H2	ID	LEN	INST	CKSM
0xFF	0xFF	0x00	0x02	0x06	0xF7

4. 6. 1. 3. ID 0 Status Packet

H1	H2	ID	LEN	ERR	CKSM
0xFF	0xFF	0x00	0x02	0x00	0xFD

4. 7. Reboot

This instruction restarts Dynamixel.

- Applied Products : MX-12W(V41), MX-28(V40), MX-64(V40), MX-106(V40), X-Series(except XL-320), MX series with Protocol 2.0

4. 7. 1. Example

4. 7. 1. 1. Conditions

- ID 1(XM430-W210) : Reboot ID 1 Dynamixel

4. 7. 1. 2. Reboot Instruction Packet

H1	H2	ID	LEN	INST	CKSM
0xFF	0xFF	0x01	0x02	0x08	0xF4

4. 7. 1. 3. ID 1 Status Packet

H1	H2	ID	LEN	ERR	CKSM
0xFF	0xFF	0x01	0x02	0x00	0xFC

4. 8. Sync Write

This instruction is used to control multiple Dynamixels simultaneously with a single Instruction Packet transmission. When this instruction is used, several instructions can be transmitted at once, so that the communication time is reduced when multiple Dynamixels are connected in a single channel. However, the SYNC WRITE instruction can only be used to a single address with an identical length of data over connected Dynamixels. ID should be transmitted as Broadcasting ID.

Item	Description
Instruction	0x83
Length	((L + 1) * N) + 4, L:Data Length, N:Number of Dynamixel
Parameter 1	Starting address

Parameter 2	Length of Data to write
Parameter 3	[1st Device] ID
Parameter 4	[1st Device] 1st Byte
Parameter 5	[1st Device] 2nd Byte
...	...
Parameter L+3	[1st Device] L-th Byte
Parameter L+4	[2nd Device] ID
Parameter L+5	[2nd Device] 1st Byte
Parameter L+6	[2nd Device] 2nd Byte
...	...
Parameter 2L+4	[2nd Device] L-th Byte

4. 8. 1. Example

4. 8. 1. 1. Conditions

- ID 0(MX-64) : Write 0x010 to Goal Position(30, 0x1E) and write 0x150 to Moving Speed(32, 0x20)
- ID 1(MX-64) : Write 0x220 to Goal Position(30, 0x1E) and write 0x360 to Moving Speed(32, 0x20)

4. 8. 1. 2. Sync Write Instruction Packet

H1	H2	ID	LEN	INST	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	CKSM
0xFF	0xFF	0xFE	0x0E	0x83	0x1E	0x04	0x00	0x10	0x00	0x50	0x01	0x01	0x20	0x02	0x60	0x03	0x67

NOTE : Status Packet will not be returned if Broadcast ID(0xFE) is used.

4. 9. Bulk Read

This instruction is used for reading values of multiple `MX series` DYNAMIXELs simultaneously by sending a single Instruction Packet. The packet length is shortened compared to sending multiple READ commands, and the idle time between the status packets being returned is also shortened to save communication time. However, this cannot be used to read a single module. If an identical ID is designated multiple times, only the first designated parameter will be processed.

Item	Description
Instruction	0x92
Length	3N + 3
Parameter 1	0x00
Parameter 2	[1st Device] Length of Data to read
Parameter 3	[1st Device] ID
Parameter 4	[1st Device] Starting address
...	...
Parameter 3N+2	[Nth Device] Length of Data to read
Parameter 3N+3	[Nth Device] ID
Parameter 3N+4	[Nth Device] Starting address

4. 9. 1. Example

4. 9. 1. 1. Conditions

- ID 1(RX-64) : Read the 2-byte Goal Position value(30, 0x1E).
- ID 2(RX-64) : Read the 2-byte Present Position value(36, 0x24).

4. 9. 1. 2. Bulk Read Instruction Packet

H1	H2	ID	LEN	INST	P1	P2	P3	P4	P5	P6	P7	CKSM
0xFF	0xFF	0xFE	0x09	0x92	0x00	0x02	0x01	0x1E	0x02	0x02	0x24	0x1D

When Bulk Read instruction is received, Dynamixel with ID 2 monitors the status packet being sent from ID 1 of the data bus (the preceeding device ID), and when device ID 1's status packet transmission is completed, ID 2 sends its own status packet.

4. 9. 1. 3. ID 1 Status Packet

H1	H2	ID	LEN	ERR	P1	P2	CKSM
0xFF	0xFF	0x01	0x04	0x00	0x00	0x80	0x7A

4. 9. 1. 4. ID 2 Status Packet

H1	H2	ID	LEN	ERR	P1	P2	CKSM
0xFF	0xFF	0x02	0x04	0x00	0x00	0x80	0x79

5. More Packet Examples

Example 6	Reads the Model Number and Firmware Version.
Hint	Instruction = READ_DATA, Address = 0x00, Length = 0x04
Communication	Instruction Packet : FF FF 01 04 02 00 03 F5 Status Packet : FF FF 01 05 00 40 00 08 B1
Status Packet Result	Model Number = 64 (0x40) Firmware Version = 0x08

Example 7	Changes the ID of RX-64 from 1 to 0.
Hint	Instruction = WRITE_DATA, Address = 0x03, DATA = 0x00
Communication	Instruction Packet : FF FF 01 04 03 03 00 F4 Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 8	Changes the Baud Rate to 1M bps.
Hint	Instruction = WRITE_DATA, Address = 0x04, DATA = 0x01
Communication	Instruction Packet : FF FF 01 04 03 04 01 F2 Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 9	Resets Return Delay Time as 4usec.
Hint	Instruction = WRITE_DATA, Address = 0x05, DATA = 0x02
Communication	Instruction Packet : FF FF 01 04 03 05 02 F0 Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 10	Restricts the movement angle from 0 to 150°.
Hint	Since CCW Angle Limit 0x3FF means 300°, 150°corresponds to 0x200. Instruction = WRITE_DATA, Address = 0x08, DATA = 0x00, 0x02
Communication	Instruction Packet : FF FF 01 05 03 08 00 02 EC Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 11	Resets the highest limit of operating temperature as 80°.
Hint	Instruction = WRITE_DATA, Address = 0x0B, DATA = 0x50
Communication	Instruction Packet : FF FF 01 04 03 0B 50 9C Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 12	Sets the operating voltage as 10 to 17V.
Hint	Data of 10V is 100 (0x64) while 17V is 170 (0xAA). Instruction = WRITE_DATA, Address = 0x0C, DATA = 0x64, 0xAA
Communication	Instruction Packet : FF FF 01 05 03 0C 64 AA DC Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 13	Only generates 50% of the maximum torque.
Hint	Sets the value of MAX Torque located in the EEPROM area as 0x1FF, which is 50% of the maximum value 0x3FF. Instruction = WRITE_DATA, Address = 0x0E, DATA = 0xff, 0x01
Communication	Instruction Packet: FF FF 01 05 03 0E FF 01 E8 Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR
The change of Max Torque can be checked by turning the power off and then on.	

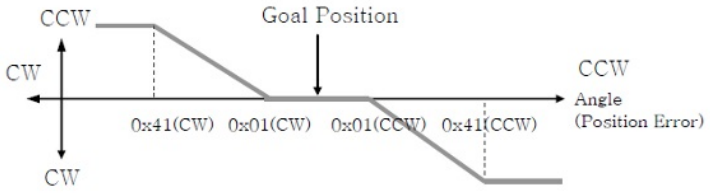
Example 15	Sets the Alarm as such that LED flickers and shutdown (torque off) when the operating temperature is higher than the limit temperature.
Hint	Since Overheating Error is Bit 2, set up Alarm value as 0x04. (0x04=00000100) Instruction = WRITE_DATA, Address = 0x11, DATA = 0x04, 0x04
Communication	Instruction Packet: FF FF 01 05 03 11 04 04 DD Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 16	Turns on the LED and enables Torque.
Hint	Instruction = WRITE_DATA, Address = 0x18, DATA = 0x01, 0x01
Communication	Instruction Packet: FF FF 01 05 03 18 01 01 DC Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR
You can check the Torque Enable state by touching the axis of Dynamixel you're your hand.	

Example 17	Locates at the Position 512 with about the 30% speed.
Hint	Instruction = WRITE_DATA, Starting Address = 0x1E Goal Position(Address 0x1E) = 512(0x200) Moving Speed(Address 0x20) = 300(0x12C) DATA = 0x00, 0x02, 0x2C, 0x01
*Writing multiple data on Sequential Addresses requires Starting Address only	
Communication	Instruction Packet: FF FF 01 07 03 1E 00 02 2C 01 A7 Status Packet : FF FF 01 02 00 FC
Status Packet Result	NO ERROR

Example 18 Sets Compliance Margin=1 and Compliance Slope=0x40.

Hint The suggested condition can be depicted in a graph as below.



A: CCW Compliance Slope (Address 29 (0x1D)) = 0x40 (about 18.8°)
B: CCW Compliance Margin (Address 27 (0x1B)) = 0x01 (about 0.3°)
C: CW Compliance Margin (Address 26 (0x1A)) = 0x01 (about 0.3°)
D: CW Compliance Slope (Address 28 (0x1C)) = 0x40 (about 18.8°)

Instruction = WRITE_DATA, Address = 0x1A,
DATA = 0x01, 0x01, 0x40, 0x40

Communication Instruction Packet: FF FF 01 07 03 1A 01 01 40 40 58
Status Packet : FF FF 01 02 00 FC

Status Packet Result NO ERROR

Example 19 Sets the minimum output Torque (Punch) as 0x40.

Hint Instruction = WRITE_DATA, Address = 0x30,
DATA = 0x40, 0x00

Communication Instruction Packet : FF FF 01 05 03 30 40 00 86
Status Packet : FF FF 01 02 00 FC

Status Packet Result NO ERROR

Example 20 Locates RX-64 with ID 0 at Position 0° and RX-64 with ID 1 at Position 300°. Start only two RX-64s at the same point.

Hint When the WRITE_DATA command is used, two RX-64s cannot be started at the same point.
Thus, REG_WRITE and ACTION are used.
ID=0, Instruction = REG_WRITE, Address = 0x1E,
DATA = 0x00, 0x00
ID=1, Instruction = REG_WRITE, Address = 0x1E,
DATA = 0xff, 0x03
ID=0xfe(Broadcasting ID), Instruction = ACTION,

Communication Instruction Packet: FF FF 00 05 04 1E 00 00 D8
Status Packet : FF FF 00 02 00 FD
Instruction Packet: FF FF 01 05 04 1E FF 03 D5
Status Packet : FF FF 01 02 00 FC
Instruction Packet: FF FF FE 02 05 FA (LEN:006)
Status Packet //No return packet

Status Packet Result NO ERROR

Example 21	Unable to change values except Address 24 to Address 35.
Hint	Sest Lock (Address 47 (0x2F)) as 1. Instruction = WRITE_DATA, Address = 0x2F, DATA = 0x01
Communication	Instruction Packet : FF FF 01 04 03 2F 01 C7 Status Packet : FF FF 01 02 00 FC
Status Packet Result	Status Packet Result NO ERROR
Once locked, It is impossible to unlock unless the power is off.	
When other data is accessed while locked, an error is returned.	