

# ASTRA

Koyo Device Driver

[www.renuelectronics.com](http://www.renuelectronics.com)

## ⌘ Tables of Contents ⌘

<u>Preface</u>	3
<u>1. Introduction</u>	3
<u>2. Technical and Communication Details</u>	4
<u>3. Data Types and Addressing</u>	5
<u>4. Optimisations</u>	7
<u>5. Errors</u>	11

## ⌘ Preface ⌘

This document introduces you to the **KOYO** device driver document contains technical information about **KOYO** device driver. This document gives you a broad idea of how to use koyo device driver with Astra.

It broadly, tells you about the capabilities and technicalities of **KOYO** device driver and how to use the driver.

## ⌘ Introduction ⌘

The intent of this document is to assist users of the **KOYO K - Sequence** driver in conjunction with the MMI software package. A general knowledge of the **KOYO** PLC family is assumed. The addressing scheme is that of the **KOYO** programming software with some slight modifications which are explained. Description of the different data types and the addressing scheme should be understood before attempting to use the driver in a project.

The driver is intended to be used only on one PC serial port at a time and connected to one or more PLCs with valid multidrop IDs. The driver uses the **K - Sequence** protocol to communicate with the PLC. The driver will operate up to 19200 baud but 9600 baud is recommended. At 19200 baud, more errors are encountered in the communication and the end result is often a slower total throughput than running at 9600 baud.

The optimization features described in this document can improve performance, but they are not essential for use.

## ⌘ Technical & Communication Details ⌘

Device Make :	_____
Device Modles :	D230-512 D240-2560 TI425-4K D430-4K TI435-8K D440-8K D440-16K
Device Memory :	_____
Communication Protocol :	K - Sequence (Half Duplex)
Communication Parameters :	
<b>Baud Rate</b>	- 9600
<b>Parity</b>	- ODD
<b>Data Bits</b>	- 8
<b>Stop Bits</b>	- 1
Cable Connections :	The KOYO driver runs on the RS232 standard for serial communication.
Node ID :	Use the Node Id set in the PLC when defining the PLC Node in Astra.

## ⌘ Data Types and Addressing ⌘

### Data Types :

The following is a description of the way the KOYO driver interprets the information from the PLC as different data types. The PLC programmer is responsible for ensuring that referenced locations can logically be interpreted as correct type. This is particularly important for float point numbers as there are bit configurations that are incompatible with the IEEE floating point format, or BCD format. In case such a thing happens the driver will report the error.

All 16 bit word and 32 bit double words must start on 16 bit boundary. It is possible to overlap double words using this format. For example V2000 and V2001 both defined as data type long would share the 16 bit word at location V2001 as either their high word or low word respectively. Since this is probably not desirable behavior, care should be taken to avoid overlap situations.

The address usage is as follows :

Tag Type	Address Example	Actual Addresses fetched
Discrete	CT0	CT0
Unsigned Integer	V2000	V2000
Integer	V2000	V2000
Large Integer	V2000	V2000 - Low Word V2001 - High Word
Real	V2000	V2000 - Low Word V2001 - High Word

## ⌘ Data Types and Addressing ⌘

The Bit Interpretation is as follows :

ASTRA Tag Type	Size in Bits	Interpretation method in ASTRA	Example	
			Read Bits from PLC	Val.
Discrete	1	Discrete	1	1
Unsigned Integer	16	BCD	00000000 00010001	11
Integer	16	Decimal	00000000 00010001	17
Large Integer	32	BCD	00000000 00000000 00010001 00010001	1111
Real	32	IEEE	01000000 00000000 00000000 00000000	2
			11000000 00000000 00000000 00000000	-2
			01000000 10000000 00000000 00000000	4
			01000000 11000000 00000000 00000000	6
			01000000 00100000 00000000 00000000	2.5



## Optimizations



Use the following guidelines so that you can get an optimum performance from the driver PLC combination.

- ⦿ Whenever possible, use consecutive addresses, this reduces the overhead on the communication per requested data byte, word or double word.
- ⦿ When a same address is to be used for two different tags in Astra, make sure that the scan time is the same for both the tags, this ensures that the address is fetched only once for both the tags.
- ⦿ Use higher scan rates whenever the application allows to do so, this ensures that the critical tags with lower scan rates are fetched with minimum overhead.



## Errors



The entire time a Astra project is running, the Event Logger displays the status and any errors that the program generates. The driver utilizes the Event Logger to display error messages regarding the driver. Below are the error messages, the probable cause and most likely solution to all the errors the driver can generate.

### Errors displayed as strings

1. NULL Pointer for Login Data
2. NULL Pointer for Project Path
3. NULL Pointer for Tag Table
4. NULL Handle for Data Manager

**Explanation :** Internal Fatal Error.

**Action :** Contact Astra support.

5. Insufficient Memory for Request Manager
6. Insufficient Memory for Transaction Manager
7. Insufficient Memory for Device Manager

**Explanation :** Internal Fatal Error.

**Action :** Try making more memory available for the project.

8. Cannot Pagelock Tag Table

**Explanation :** Internal Fatal Error.

**Action :** Contact Astra support.

9. Cannot Open File PLCTAG.DAT
10. Cannot Read File PLCTAG.DAT

**Explanation :** Internal Fatal Error. The input file PLCTAG.DAT does not exist or is corrupt.

**Action :** Open the project in the configuration mode and close it, this process recompiles the PLCTAG.DAT file.

11. Insufficient Memory for Tag
12. Insufficient Memory for Tag2
13. Insufficient Memory for Tag Container
14. Insufficient Memory for Node
15. Insufficient Memory for Node Container

**Explanation :** Internal Fatal Error.

**Action :** Try making more memory available for the project.

16. No Tags in the Project

**Explanation :** Internal Fatal Error. The driver detected no valid tags in the project.

**Action :** Recheck the project in the configuration mode. See if any tags are assigned to this particular device. See if the Node details are correct.



## Errors



### 17. No Valid Nodes in the Project

**Explanation :** Internal Fatal Error. The driver detected no valid nodes in the project.

**Action :** Recheck the project in the configuration mode. See if the Node details are correct.

### 18. Multidrop not Supported

**Explanation :** Internal Fatal Error. An attempt was made to attach two nodes on the same driver when Multidrop is not supported.

**Action :** Recheck the project in the configuration mode. See if the Node details are correct.

### 19. Multiple nodes with same ID

**Explanation :** Internal Fatal Error. An attempt was made to attach two nodes on the same driver with same Node IDs.

**Action :** Recheck the project in the configuration mode. See if the Node details are correct.

### 20. Insufficient Memory for Request

### 21. Insufficient Memory for Request2

### 22. Insufficient Memory for Request Container

### 23. Insufficient Memory for Dummy Request

### 24. Insufficient Memory for Action

### 25. Insufficient Memory for Action Container

### 26. Cannot Create Communication Window

**Explanation :** Internal Fatal Error.

**Action :** Try making more memory available for the project.

### 27. Cannot Open Communication Port

**Explanation :** Internal Fatal Error. Could not initialize the Communication port for the given settings.

**Action :** For the selected Communication port, check for -

- ⊙ If the port physically exists.
- ⊙ If the Communication hardware uses standard base addresses. COM1 uses hex 3F8 and COM2 uses hex 2F8.
- ⊙ If there is any IRQ contention at the hardware level. COM1 uses IRQ4 and COM2 uses IRQ3.
- ⊙ If any other program is already using the Communication port you have requested for
- ⊙ If any DOS level TSRs are running which are using the Communication port you have requested for.



## Errors



- ⊙ If a mouse driver is installed on the same Communication port you have requested for in Windows environment.
- ⊙ If a mouse driver is installed on the same Communication port you have requested for on DOS environment.
- ⊙ If you have directly manipulated the PROJECT.INI file section [COM1] or [COM2], check if the settings for Baud Rate, Data Bits, Stop Bits and the Parity are standard. Try using the Communication port setting utility provided with Astra in case you are in doubts about the standard settings.

### 28. Cannot Build Communication DCB

**Explanation :** Internal Fatal Error. Could not initialize the Communication port for the given settings.

**Action :** If you have directly manipulated the PROJECT.INI file section [COM1] or [COM2], check if the settings for Baud Rate, Data Bits, Stop Bits and the Parity are standard. Try using the Communication port setting utility provided with Astra in case you are in doubts about the standard settings.

### 29. Cannot Set Communication State

**Explanation :** Internal Fatal Error. Could not initialize the Communication port for the given settings.

**Action :** If you have directly manipulated the PROJECT.INI file section [COM1] or [COM2], check if the settings for Baud Rate, Data Bits, Stop Bits and the Parity are standard. Try using the Communication port setting utility provided with Astra in case you are in doubts about the standard settings.

### 30.NULL Pointer for Model Names

### 31.Read Queue Full

**Explanation :** Internal Fatal Error.

**Action :** Contact Astra support.

### 32. Device Time Out

**Explanation :** The Device did not respond and the Device driver timed out. The Driver will retry the request to Device for a specified number of times and if the Device still does not respond the driver will HALT its transactions with the Device.

**Action :** If this happens during **initialization**, check –

- ⊙ Whether the Device power is on.
- ⊙ Whether the cable connections to the device are proper.



## Errors



- ⊙ Whether the Node ID settings are proper in case the Device supports it.
- ⊙ Whether the Device model is the same as configured in the Node Configuration.
- ⊙ Whether the Communication hardware is proper and works.
- ⊙ Whether strong EMI or RFI fields are existent which cause noise on the Communication line.
- ⊙ Whether some turnaround delay is required, try changing the entries in the DRIVERS.INI file. This may be typically required for faster PCs on which Astra runs.

If this happens during the **Run**, check –

- ⊙ Whether other applications block the Windows, in such a case the retry mechanism will normally re-establish the Communication.
- ⊙ Whether the cable connections have been disturbed.
- ⊙ Whether the Device has malfunctioned.
- ⊙ Whether the Communication hardware is proper and works.

### 33. Invalid IEEE Format

**Explanation** : The 32 bits read from the Device contained bit values such that it could not be interpreted as a valid IEEE format.

**Action** : Use OEM software and initialize floating type tags in the plc.

### 34. Write Queue Full

**Explanation** : The write request sent by the Astra is queued for faster execution, the current limit for the queue size is 300. If the queue is full this message will be prompted and the latest request will be ignored.

**Action** : Try configuring the Project such that at a time less than 150 write requests are raised. Also make sure that the Device gets enough time to serve these write requests.

### 35.No Valid Tags in the Project

**Explanation** : Internal Fatal Error. The driver detected no valid tags in the project.

**Action** : Recheck the project in the configuration mode. See if any tags are assigned to this particular device. See if the Node details are correct.



## Errors



36. Insufficient Memory for Register

37. Insufficient Memory for Tag Container2

38. Insufficient Memory for Register Container

**Explanation** : Internal Fatal Error.

**Action** : Try making more memory available for the project.

39. Tag Address Invalid

40. Tag Address Invalid2.

**Explanation** : The address entered for a Tag is invalid.

**Action** : Reconfigure the project and check.

41. Driver Scan Halted

42. Driver Scan Halted2

**Explanation** : The driver has stopped communicating with the device. This may happen in two situations –

⊙ When the initial scan is complete - in this case this is just a status information.

⊙ When time-out has occurred and retry for establishing communication has failed.

**Action** : In the second case check –

⊙ If the cable connections have been disturbed.

⊙ If the Device has malfunctioned.

⊙ If the Communication hardware is proper and works.

43. Cannot Find INI File Entry, Setting Default Port

**Explanation** : The [PROTOCOL] section in PROJECT.INI does not have the driver name against the COM1 or the COM2 entry. In such a case default COM1 is selected as the Communication port.

**Action** : Run the Communication port setting utility provided with Astra and set all the parameters properly.

44. Cannot Initialise Driver Twice

**Explanation** : Due to some abnormal termination in a previous run the Device Driver has not unloaded itself and hence could not reinitialise itself.

**Action** : Restart the project. ]

45. Cannot Run Without Initialisation

46. Cannot Run Without Initialization2

47. Cannot Write Without Initialisation

48. Cannot Build Frames Without Initialisation

**Explanation** : Due to some abnormal termination in a previous run, the Device Driver has not unloaded itself and hence could not reinitialise itself.

**Action** : Restart the project.



## Errors



### 49. NULL Pointer for Queue

**Explanation** : Internal Fatal Error.

**Action** : Contact ASTRA support.

### 50. Invalid IEEE Format2

**Explanation** : The 32 bits read from the Device contained bit values such that it could not be interpreted as a valid IEEE format.

**Action** : Use OEM software and initialise floating type tags in the device.

### 51. Cannot Pagelock Buffer

**Explanation** : Internal Fatal Error.

**Action** : Contact ASTRA support.

### 52. Device Response Delay

**Explanation** : The Device did not respond and the Device driver timed out. The Driver will retry the request to Device for a specified number of times and if the Device still does not respond the driver will HALT its transactions with the Device.

**Action** : If this happens during **Initialisation** check –

- ⊙ If the Device is powered on.
  - ⊙ If the cable connections to the device are proper.
  - ⊙ If the Node ID settings are proper in case the Device supports it (Applicable to KOYO).
  - ⊙ If the Device model is the same as configured in the Node Configuration.
  - ⊙ If the Communication hardware is proper and working.
  - ⊙ If strong EMI or RFI fields are existent which cause noise on the Communication line.
- If this happens during the **Run** check –
- ⊙ If in case other applications block the Windows, in such a case the retry mechanism will normally re-establish the Communication.



## Errors



- ⊙ If the cable connections have been disturbed.
- ⊙ If the Device has malfunctioned.
- ⊙ If the Communication hardware is proper and working.

### 53. Response Check Sum Error

**Explanation :** The Device did respond but the bytes received were corrupt. The Driver will retry the request to Device.

**Action**

: If this happens during **Initialisation** check –

- ⊙ If the Communication hardware is proper and working.
- ⊙ If strong EMI or RFI fields are existent which cause noise on the Communication line.
- ⊙ If the Communication port settings are proper.

If this happens during the **Run** check –

- ⊙ If the cable connections have been disturbed.
- ⊙ If the Device has malfunctioned.
- ⊙ If the Communication hardware is proper and working.

### 54. Data Over Flow

**Explanation :** Unexpected data in large volume was received on the Communication port.

**Action**

: Check –

- ⊙ If the cable connections have been disturbed.
- ⊙ If the Device has malfunctioned.
- ⊙ If the Communication hardware is proper and working.

### 55. Model Name Invalid

**Explanation :** Internal Fatal Error. The model name associated with a particular Node was invalid.

**Action**

: Open the project in the configuration mode. Check the model in the Node Configuration and close it.

### 56. Cannot Open File PLCTAG.DAT 2

### 57. Cannot Read File PLCTAG.DAT 2

### 58. Cannot Read File PLCTAG.DAT 3

**Explanation :** Internal Fatal Error. The input file PLCTAG.DAT does not exist or is corrupt.

**Action**

: Open the project in the configuration mode and close it, this process recompiles the PLCTAG.DAT file.



## Errors



### 59. Cannot Open File FRAME.OUT

**Explanation** : Internal Fatal Error. This file is used for diagnostic dumping and is prepared only when [TUNING] Section in the PROJECT.INI file has "Dump = 1".

**Action** : The size of this file totally depends on the number of tags used hence free some disk space and try out again.

### 60. Cannot Open File DUMP.OUT

**Explanation** : Internal Fatal Error. This file is used for diagnostic dumping and is prepared only when [TUNING] Section in the PROJECT.INI file has "Dump = 2".

**Action:** The size of this file totally depends on the number of tags used hence free some disk space and try out again.

### 61. Invalid BCD Format for a WORD

**Explanation** : The 16 bits read from the Device contained bit values such that it could not be interpreted as a valid BCD format.

**Action** : None.

### 62. Invalid BCD Format for a DWORD

**Explanation** : The 32 bits read from the Device contained bit values such that it could not be interpreted as a valid BCD format.

**Action** : Use OEM software and initialise respective tags in the device.

### 63. Invalid number for conversion to BCD for WORD

**Explanation** : The 16 bits given for write from ASTRA to the Device contained bit values such that it could not be interpreted as a valid BCD format. Write will not be done in these cases.

**Action** : Avoid such values.

### 64. Invalid number for conversion to BCD for DWORD

**Explanation** : The 32 bits given for write from ASTRA to the Device contained bit values such that it could not be interpreted as a valid BCD format. Write will not be done in these cases.

**Action** : Avoid such values.



## Errors



### 65. Error Composing Write Request

**Explanation :** Write request could not be composed. This may happen in two cases –

⊙ Invalid number for write.

⊙ Write Queue full.

**Action :** Avoid non interpretable values, Avoid writing too fast. Increase WriteQueueSize in project.ini file section [QueueSize]. Lower limit for WriteQueueSize is 300 and the upper limit is 5000.

### 66. Error Composing Read Request After Write

**Explanation :** A read request immediately following a write request could not be composed.

This may happen in two cases –

⊙ Invalid number for write.

⊙ Write Queue full.

**Action :** Avoid non interpretable values, Avoid writing too fast.



Renu Electronics Pvt Ltd.

S.No. 2/6, Baner Road,

Pune 411045, India.

Tel: + 91 20 2729 2840,

Fax: + 91 20 2729 2839

Email: [info@renuelectronics.com](mailto:info@renuelectronics.com)

Website: [www.renuelectronics.com](http://www.renuelectronics.com)

