



## **Allen-Bradley Data Highway II Asynchronous-Device Interface (Cat. Nos. 1779-KFL, -KFLR)**

**When to Use This Document** Use this document as a supplement to the Data Highway II Asynchronous-Device Interface User's Manual, Publication 1779-6.5.1, December, 1986.

**What This Document Contains** This document provides the following updated information for the asynchronous-device interface:

- Product Enhancement to Add MMFS/1393
- Corrections to the User's Manual

The following sections show new and corrected information by chapter and page numbers within the chapters.

**Product Enhancement for Series A, Revision C** You can now use the 1779-KFL (Series A, Revision C) in MMFS/1393 mode. MMFS/1393 protocol allows you to communicate between Numerical Controllers (NCs) and robots over the Data Highway II network.

The following pages contain information to help you update your manual with MMFS/1393 information, as well as correct technical errors.

**Corrections to the User's Manual** The following sections provide corrected information for:

- Chapter 2, Introducing the 1779-KFL Interface
- Chapter 4, Installing the 1779-KFL Interface
- Chapter 6, The Full-Duplex Data-Link Layer
- Chapter 7, The Half-Duplex Data-Link Layer
- Chapter 8, Introduction to Message Sets
- Chapter 9, Data Highway Message Set
- Chapter 11, Remote Command Control Message Set
- Chapter 13, Station Management Message Set
- Message Formats for NATIVE and MMFS/1393 Modes

**Corrections to Chapter 2** — Page 2-15, Message Sets Available

*Add this item to the bulleted list:*

- MMFS/1393

**Corrections to Chapter 4**

**Page 4-6, Figure 4.4 — Host-Board Switches**

*Replace Figure 4.4 in the manual with the figure on the following page.*

**Page 4-7, Accept/Reject Memory Writes**

*Delete this section. These switches are now reserved for future use.*

**Page 4-7, Accept/Reject Third-Party Requests**

*Delete this section. These switches are now reserved for future use.*

**Page 4-7, Communication Mode**

*Replace the table in this section with the following table:*

To Select This Mode:	Set These Switches:		
	1	2	3
SUPERVISORY	OFF	ON	ON
MMFS 1393	ON	OFF	ON
PAD (XON/XOFF)	OFF	OFF	ON
NATIVE	ON	ON	OFF
KE/KF	OFF	ON	OFF
PAD	ON	OFF	OFF

*Add these two paragraphs following the above table:*

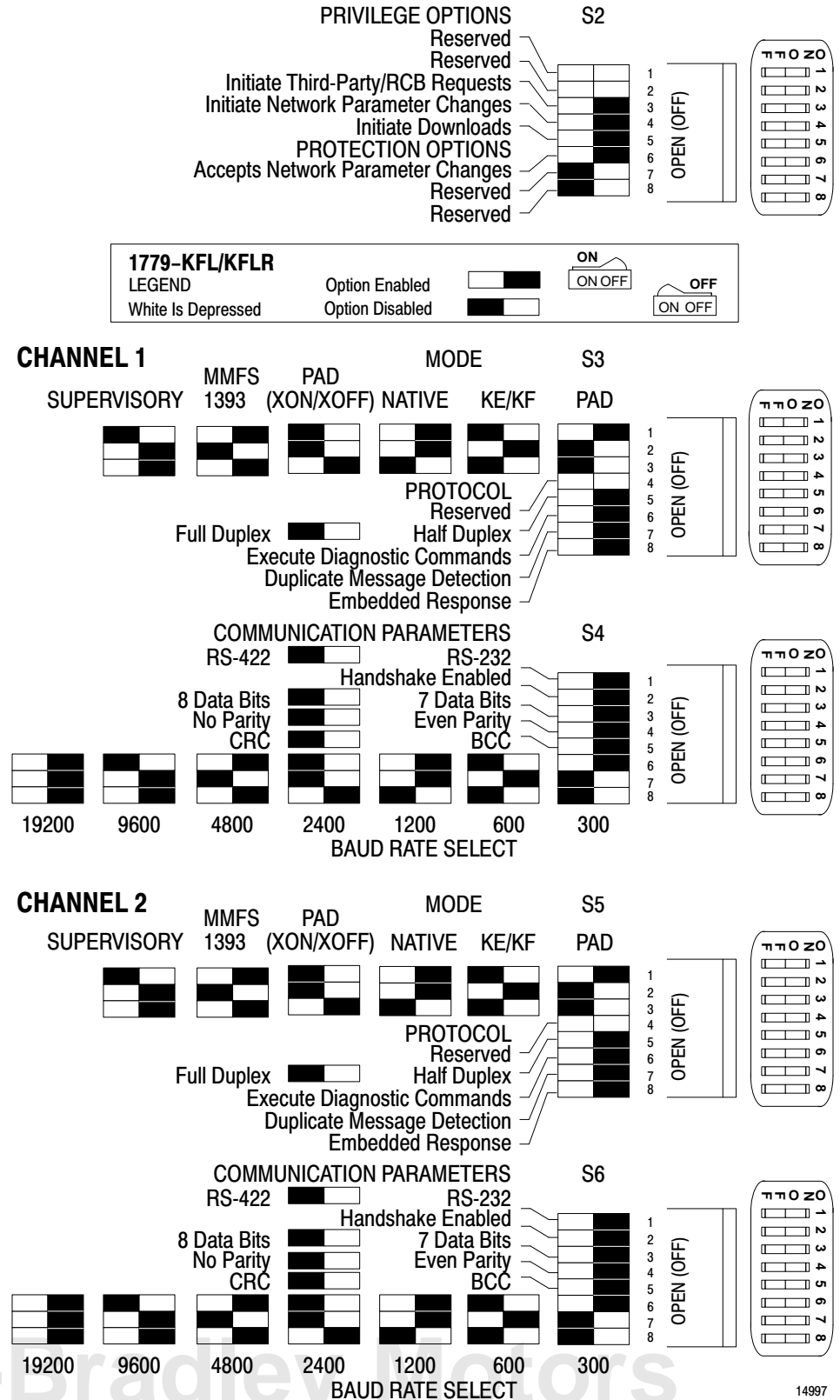
**Important:** Do not select SUPERVISORY mode. If you select SUPERVISORY mode, the 1779-KFL will not communicate with the host device. This mode is reserved for future use.

If you select MMFS/1393 mode, the 1779-KFL communicates with the host device in MMFS/1393 protocol. MMFS/1393 is selected through the SEL byte that is transmitted in each message header.

*In the last paragraph (If you select the PAD mode....), change:  
with:*

1779-KFM interface  
1779-KFL interface

**Figure 4.4**  
**Host-Board Switches**



## Corrections to Chapter 6

### Page 6-9, Message Characteristics

*Replace the second paragraph:* (The size...maximum.)  
*with the following paragraph:*

The size of a valid packet is dependent on the mode of the 1779-KFL. For NATIVE and MMFS/1393 modes, the maximum packet is 254 bytes. For PAD mode, the maximum packet size is 256 bytes. For KE/KF mode, the maximum packet size is 250 bytes. The minimum packet size for all modes is 6 bytes.

*Replace the last sentence in this section:* (In Native mode...header.)  
*with the following sentence:*

In NATIVE and MMFS/1393 modes, the SSQ, EXT, NODE, and user bytes make up the network packet header.

### Page 6-11, Figure 6.6 — Transmitter for Full-Duplex Protocol

*Replace:* BCC (in the Message Packet)  
*with:* BCC/CRC.

### Page 6-12, Table 6.A, Transmitter for Full-Duplex Protocol

*Replace Table 6.A with the revised Table 6.A on the following page.*

**Table 6.A**  
**Transmitter for Full-Duplex Protocol**

```

TRANSMITTER is defined as
  loop
    Message = GET-MESSAGE-TO-SEND
    Status = TRANSFER (Message)
    SIGNAL-RESULTS (Status)
  end
TRANSFER (Message) is defined as
  initialize NAK-limit and ENQ-limit
  SEND (Message)
  start timeout
  loop
    WAIT for response on Path 2 or timeout.

    if received DLE ACK then return SUCCESS
    else if received DLE NAK then
      begin
        if NAK-limit is exceeded then return FAILURE
        else
          begin
            count NAK retries;
            SEND-MESSAGE (Message);
            start timeout
          end
        end
      end
    else if timeout
      begin
        if ENQ-limit is exceeded then return FAILURE
        else
          begin
            count ENQ retries;
            send DLE ENQ on Path 1;
            start timeout
          end
        end
      end
    end loop
SEND (Message) is defined as
  begin
    BCC/CRC = 0
    send DLE STX on Path 1
    for every byte in the message do
      begin
        add the byte to the BCC/CRC;
        send the corresponding data code on Path 1
      end
    end
    if CRC
      add ETX to CRC
    end
    send DLE ETX BCC/CRC on Path 1
  end
GET-MESSAGE-TO-SEND
  This is an operating-system-dependent interface routine that waits and allows the rest
  of the system to run until the message source has supplied a message to be sent.
SIGNAL-RESULTS
  This is an implementation-dependent routine that tells the message source of the results
  of the attempted message transfer.
WAIT
  This is an operating-system-dependent routine that waits for any of several events to
  occur while allowing other parts of the system to run.

```

## Page 6-16, Table 6.B, Receiver for Full-Duplex Protocol

Replace Table 6.B with the revised Table 6.B below:

```
RECEIVER is defined as
variables
  LAST-HEADER is 4 bytes copied out of the last good message
  RESPONSE is the value of the last ACK or NAK sent
  BCC is an 8-bit block check accumulator

LAST-HEADER = Invalid
LAST RESPONSE = NAK

loop
  reset parity error flag
  GET-CODE
  if DLE STX then
    begin
      BCC/CRC = 0
      GET-CODE
      while it is a data code
        begin
          if buffer is not overflowed put data in buffer
          GET-CODE
        end
      if the control code is not DLE ETX then send DLE NAK
      else if error flag is set then send DLE NAK
      else if BCC/CRC is not correct then send DLE NAK
      else if message is too small then send DLE NAK
      else if message is too large then send DLE NAK
      else if header is the same as last message send a DLE ACK
      else if message sink is full send DLE NAK
      else
        begin
          send message to message sink
          send a DLE ACK
          save last header
        end
      end
    else if DLE ENQ then send LAST-RESPONSE
    else LAST-RESPONSE = NAK
  end
end
GET-CODE is defined as
loop
  variable
  GET-CHAR
  if char is not a DLE
    begin
      add char to BCC/CRC
      return the char and data flag
    end
  else
    begin
      GET-CHAR
      if char is a DLE
        begin
          add char to BCC/CRC
          return a DLE and a data flag
        end
      else if char is an ACK or NAK send it to the transmitter
      else if char is an ETX
        begin
          GET-CHAR
          if CRC add char to CRC BCC/CRC
          return ETX with a control flag
        end
      else return character with a control flag
    end
  end
end
GET-CHAR is defined as
an implementation-dependent function that returns one byte of data from the link interface hardware.
```

## Page 6-23, Embedded Response Option

Change: DLE ETX BCC Sequence  
to: DLE ETX BCC/CRC Sequence

## Corrections to Chapter 7

### Page 7-6, Block Check Character

*In the example in Paragraph 2  
of this section, replace the BCC value:*      A0  
*with:*      C0

*Also, the sum of the equation in Paragraph 3 (Page 7-7) should be:*

1100 0000    2's complement (C0 hex)

### Page 7-8, Message Characteristics

*Replace Paragraph 2:*      (The size...maximum.)  
*with the following paragraph:*

The size of a valid network packet is dependent on the mode of the 1779-KFL. For NATIVE and MMFS/1393 modes, the maximum packet size is 254 bytes. For PAD mode, the maximum packet size is 256 bytes. For KE/KF mode, the maximum packet size is 250 bytes. The minimum packet size for all modes is 6 bytes.

*Replace the last sentence in this section:*      (In Native...header.)  
*with the following sentence:*

In NATIVE and MMFS/1393 modes, the SSQ, EXT, NODE, and USER bytes make up the network packet header.

### Page 7-11, Transceiver Actions

*In Paragraph 3 (When a slave...addressed to it.), replace:*

- BCC      *with:*      BCC/CRC
- or the Global Node      *with:*      in the future the Global  
  Number 377 (octal)      Node Number 377 (Octal)

### Page 7-14, Figure 7.4 — Normal Message Transfer

*In Figure 7.4 and the paragraph  
before it, change:*      BCC  
*to:*      BCC/CRC

*Add the following item to the bulleted list:*

- MMFS/1393

*At the end of this section, add the sentence:*

If you configure the KFL for MMFS/1393 mode, you can use a limited set of messages.

### **Page 8-2, Remote Command Control Message Set**

*In Paragraph 2 of this section:* (A remote...controllers.)  
*Add the following sentence after the second sentence (You can store...interface) in the paragraph:*

To determine the number of available storage locations for a remote station, refer to the user's guide of the remote station you are using.

### **Page 8-3, Station Management Message Set**

*Add the following new section after the Station Management Message Set:*

#### **MMFS/1393 Message Set**

You can send and receive MMFS commands between your host device and the KFL, only if the KFL is in either NATIVE or MMFS/1393 mode. MMFS lets numerical controllers (NCs) and robots talk over the Data Highway II network. If you are communicating between two MMFS type machines, the KFL (MMFS mode) will pass the commands through as they are. If you are using a Programmable Controller (PLC), the KFL will decode the MMFS commands Read, Write, and Bit Write to Data Highway II commands so that the PLC can understand them.

### **Page 8-5, Program and Message Types**

*In Paragraph 1, add the following sentence to the second bullet (Reply Messages):*

Not all message sets require a reply message (i.e., NATIVE, PAD, MMFS/1393).



*In Paragraph 2:* (You can...it receives.)  
*Delete the sentence in  
parenthesis:*

(A rare exception occurs when the data link delivers a message but receives no acknowledgment to verify delivery.)

## Corrections to Chapter 9

### Page 9-1, Selecting Data Highway Messages

*Add this note at the end of this section:*

**Important:** All Data Highway message fields are sent low byte then high byte.

### Page 9-7, Data Highway Fields

*The caution in this section should read:*



**CAUTION:** If a channel in the KFL is in KE/KF mode, the device on that channel can only send commands **and responses** to stations with node numbers below 200 octal.

---

### Page 9-8, Data Highway Fields (Cont'd)

*Delete Paragraph 2 (While a KFL channel...over 177 octal).*

### Page 9-15, Table 9.A — Status Information Received from a 1779-KP2 Interface

*Replace the table on Page 9-15 with the table on the following page.*

**Table 9.A**  
**Status Information (in Hex) Received from a 1779-KP2 Interface**

	Byte	Bit	Value	Meaning
<b>Diagnostic Counters</b>	8	0-7	--	Bytes 8 and 9 contain the starting byte address of diagnostic counters.
	9	0-7	--	
<b>Revision and Series Level of the 1779-KP2</b>	10	0-3	0 (Hex) 1 (Hex) 2 (Hex)	Revision A Revision B Revision C Etc.
		4-7	0 (Hex) 1 (Hex)	Series A Series B Etc.
--	11	--	--	Not Used
<b>Switch Settings</b>	12	0-1	0	Not Used
		2	0 1	Accepts Third-Party Requests Rejects Third-Party Requests
		3	0 1	Accepts Network Parameter Changes Rejects Network Parameter Changes
		4	0 1	Accepts Writes to Data Table Rejects Write to Data Table
		5	0 1	Accepts Writes to Program Areas Rejects Writes to Program Areas
		6-7	--	Not Used
<b>Direct/Indirect Communication</b>	13	0-2	--	Not Used
		3	0 1	Interface is connected directly to PLC-2 family processor. Interface is connected to PLC-2 family processor via some other device (i.e., 1771-KG, 1771-KA2, 1770-RG, 1770-T3, or another 1779-KP2).
		4-7	--	Not Used
--	14	--	--	Not Used

**Page 9-34 and 9-36, Message Format Drawings for Write Block, Read Block, Write File, and Read File Sections**

The **Packet Offset** and **Total Trans** packets in these message formats **should be two bytes** long instead of one byte.

The **Set Mask** and **Reset Mask** packets on Page 9-35, under Write Bit, **should also be two bytes** long.

**Page 9-33 to 9-36, Write Block, Read Block, Write File, Read File**

*In each of the sections listed above, this statement:*

Use the total-transaction field to specify the number of bytes to write in the total transaction.

*Should read:*

Use the total-transaction field to specify the number of words to write in the total transaction.

**Pages 9-38 and 9-39, Message Formats for Physical Write, Physical Read**

*Under each set of message formats add the following note:*

The address field is as follows:

Byte 1 - Bits 17-24

Byte 2 - Bits 25-32

Byte 3 - Bits 1-8

Byte 4 - Bits 9-16

**Corrections to Chapter 11**

**Page 11-11, Remote Command Fields (Application Layer)**

*In the message format illustration on this page, change the source address (S ADDR) and destination address (D ADDR) fields **from 2-35 bytes** in size to **2-51 bytes**.*

**Important:** The remote module must understand the message set selected. Some modules do not support all message sets.

*Under the description for the command (CMD) field (also Page 11-11), add the following item to the bulleted list:*

- 16 — Bit Write

*Also, the first sentence (Write block...command.) in the following paragraph should read:*

Bit write and write block are protected commands.

**Page 11-12, Description of S ADDR (Source Address) Field**

*The second sentence in the paragraph should read:*

The size of this field is variable (2 to **51** bytes).

*The last sentence in this paragraph should read:*

Refer to Appendix **B** for an explanation of logical addressing.

**Page 11-12/13, Description of D ADDR (Destination Address) Field**

*The second sentence in the paragraph should read:*

The size of this field is variable (2 to **51** bytes).

*The last sentence in this paragraph should read:*

Refer to Appendix **B** for an explanation of logical addressing.

**Corrections to Chapter 13**

**Page 13-7, Description of the DATA field**

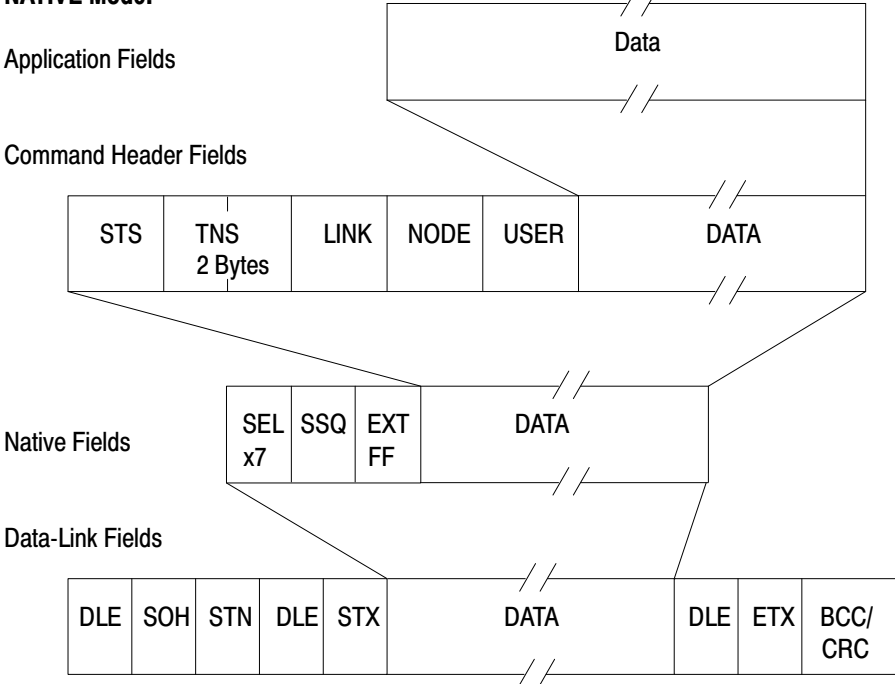
*Add this sentence at the end of the first paragraph:*

All data is sent low byte then high byte.

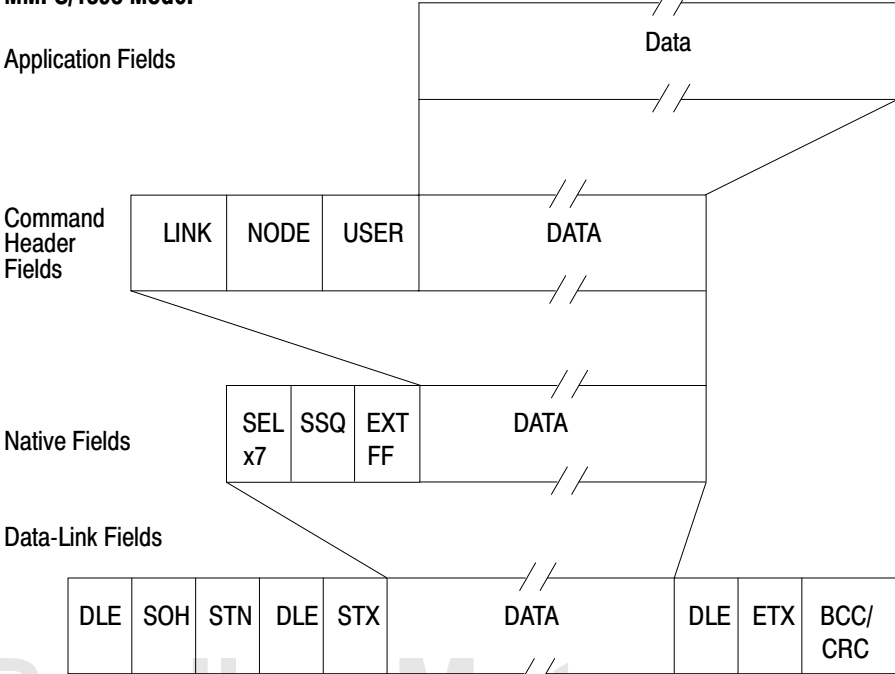
**Additional Message Formats**

The following illustrations show the message format for NATIVE MODE and MMFS/1393 MODE:

**NATIVE Mode:**



**MMFS/1393 Mode:**





**ALLEN-BRADLEY**  
A ROCKWELL INTERNATIONAL COMPANY

With offices in major cities worldwide

---

**WORLD  
HEADQUARTERS**  
1201 South Second Street  
Milwaukee, WI 53204 USA  
Tel: (414)382-2000  
Telex: 43 11 016  
FAX: (414)382-4444

**EUROPE/INDIA/  
MIDDLE EAST/AFRICA  
HEADQUARTERS**  
Allen-Bradley Europa B.V.  
Amsterdamseweg 15  
1422 AC Uithoorn  
The Netherlands  
Tel: (31)2975/68811  
Telex: 884 18042  
FAX: (44)2975/60222

**ASIA/PACIFIC  
HEADQUARTERS**  
Allen-Bradley (Hong Kong)  
Limited  
2901 Great Eagle Center  
23 Harbour Road  
G.P.O. Box 9797  
Wanchai, Hong Kong  
Tel: (852)5-739391  
Telex: 780 64347  
FAX: (852)5-745326

**CANADA  
HEADQUARTERS**  
Allen-Bradley Canada Limited  
135 Dundas Street  
Cambridge, Ontario N1R 5X1  
Canada  
Tel: (519)623-1810  
Telex: 069 59317  
FAX: (519)623-8930

**LATIN AMERICA  
HEADQUARTERS**  
1201 South Second Street  
Milwaukee, WI 53204 USA  
Tel: (414)382-2000  
Telex: 43 11 016  
FAX: (414)382-2400

PRINTED IN USA