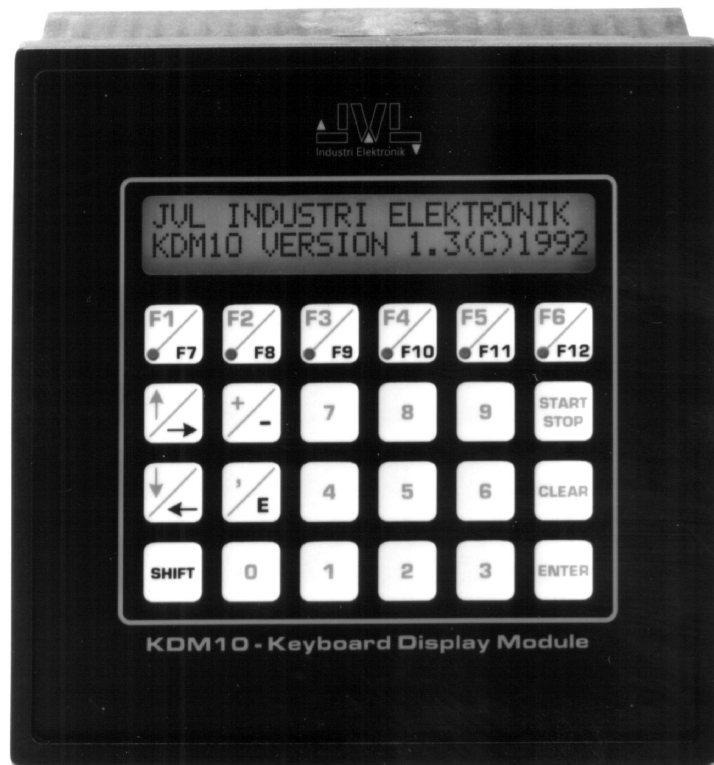


# **KDM10D/KDM10T**

## **Keyboard/Display Module**

### **User's Manual**



**JVL Industri Elektronik A/S August 1993**

# Contents

1.1	Introduction	3
1.2	Keyboard	4
1.3	Power Supply	5
1.4	Module Interface	6
1.5	External Loudspeaker	8
1.6	Start/stop Output	9

## **Software**

2.1	Command Overview	10
2.2	Overview of Registers and Flags	11
2.3	Print-out to the KDM10	12
2.4	Input from the KDM10	13
2.5	KDM10 Flags	15
2.6	KDM10 Registers	16
2.7	Error Messages	19

## **Appendix**

3.1	Electrical Specifications	20
3.2	Physical Dimensions	21

KDM10 is an intelligent keyboard/display module which can be connected to JVL Industri Elektronik's Step Motor Controllers that are equipped with registers and a module interface (Types SMC13, SMC14 and SMC30). The combination of a Keyboard/Display Module and Controller provides an easily operated user interface with a display and a simple keyboard for entering user instructions. The KDM10 Module thus enables motor operation parameters, register contents, etc., to be changed without editing the Controller program instructions. The Module's built-in loudspeaker can be used for generating audible alerts.

The KDM10 Module is available in two different versions: for mounting in a 19" rack or panel mounting.

Module KDM10 includes the following:

An LCD text display with 2 lines of 24 characters (48 characters in all). The Module Display is back-lit.

A 23-key keyboard with "Shift" key for secondary functions.

A built-in loudspeaker which can be used for emitting sounds of different frequency, duration and level.

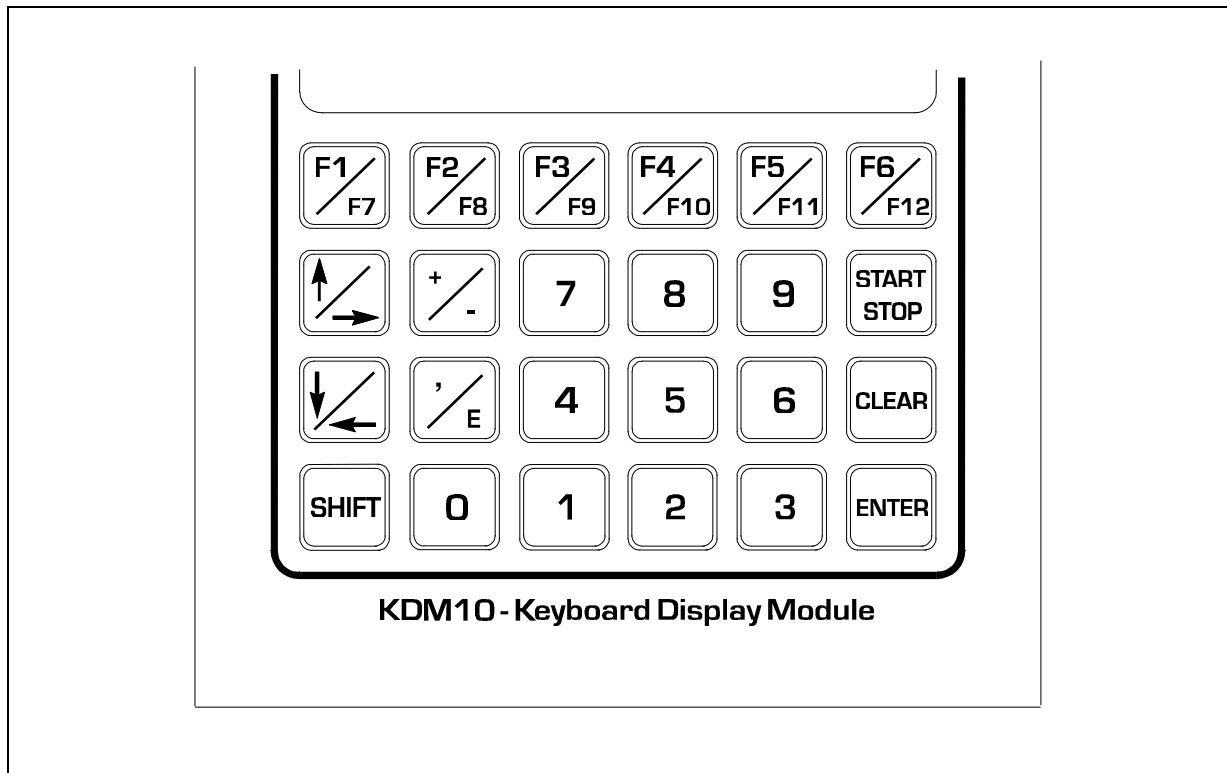
The KDM10 Module is controlled by the motion-system Controller, enabling the Controller program to write text and values to the Module display and read input from the Module keyboard. For example, text can be printed out from the Controller to the KDM10 display and thereafter the Module set to "*Input Mode*". The KDM10 will then wait for a user to key-in an instruction which is transmitted to the Controller once the *ENTER* key has been pressed.

The keyed value may then be stored in a Controller register for use later in program execution.

In addition, it is possible to check whether a key has been pressed. The Controller can transmit a query to the KDM10 module to return a key code. If the returned code is not 255, a key has been pressed. The code itself indicates which key on the Module has been activated.

**Features:**

- Loudspeaker output for external loudspeaker.
- Splash-proof, membrane keyboard.
- Back-lit, supertwist LCD display.
- Flexible power supply (12-45V).
- Can be located up to 100m from the system controller.
- Optically isolated communication interface.
- Overload protected and protected against incorrect polarity connection.
- Several Controllers may be connected to the same Keyboard/Display Module.



#### Keyboard Functions:

##### *0-9:*

Keys 0-9 are used when numbers are keyed in for transmission to registers in the system Controller.

##### *CLEAR:*

The Clear key is used to erase input during keying, i.e. before the ENTER key has been pressed.

##### *ENTER:*

The ENTER key is used to confirm the keyed data shown on the display and transfer the data to the system Controller. The ENTER key is used in conjunction with the INPUT command.

##### *SHIFT:*

The SHIFT key is used to activate the keyboard keys' secondary functions (marked in black on the keyboard). The key codes indicated in the table in Section 2.4 are assigned a value equal to the table code plus 128. See Section 2.4 for details.

##### *START/STOP:*

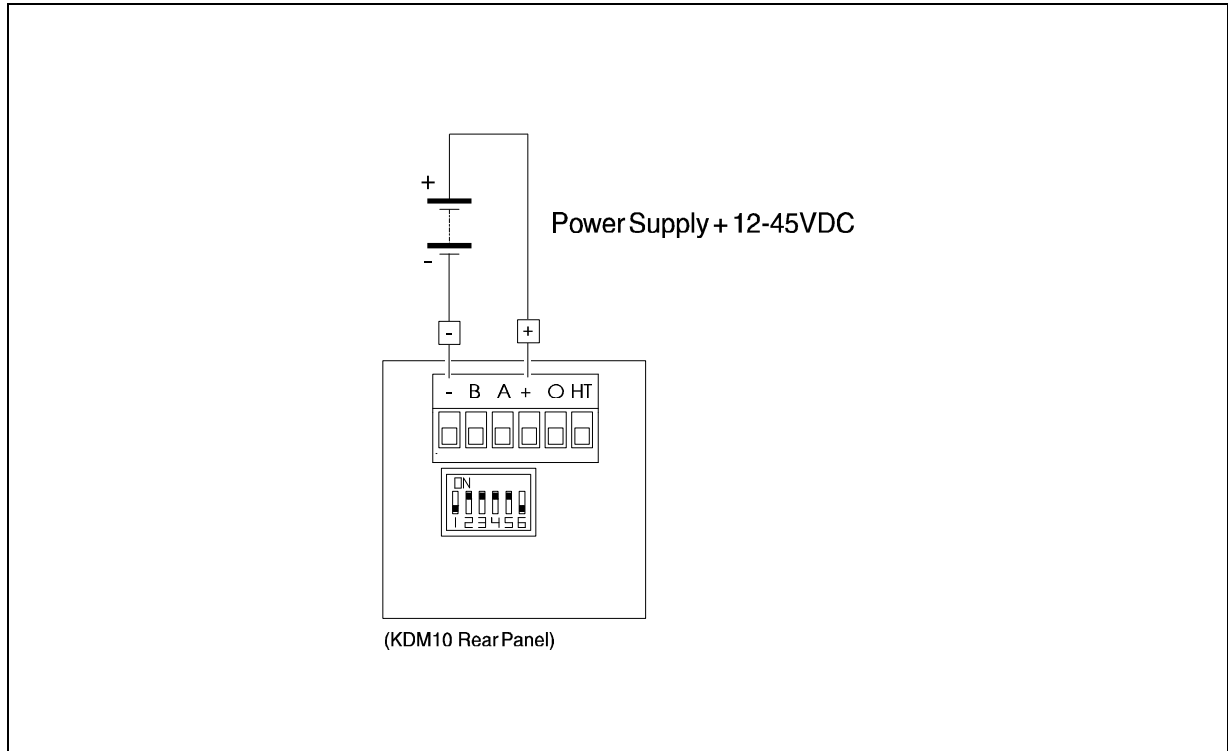
The START/STOP key can be used to start or stop motor operation. When the START/STOP key is pressed, the output "O1" is activated for 100 mS. The output can, for example, be connected to a Controller or PLC equipment to indicate that motor operation should start or stop. The key can also be read by software.

##### *F1-F12:*

The F1 to F12 keys are used as function keys. Their function depends on the current program loaded in the Controller. The 6 alternative functions, F7 to F12, are invoked using the SHIFT key in combination with the respective function key.

##### *Arrow Up/Down:*

These keys can be used if continuous motor operation is required when one of the keys is pressed and held. This function requires a special case of the Input command. See Section 2.4.

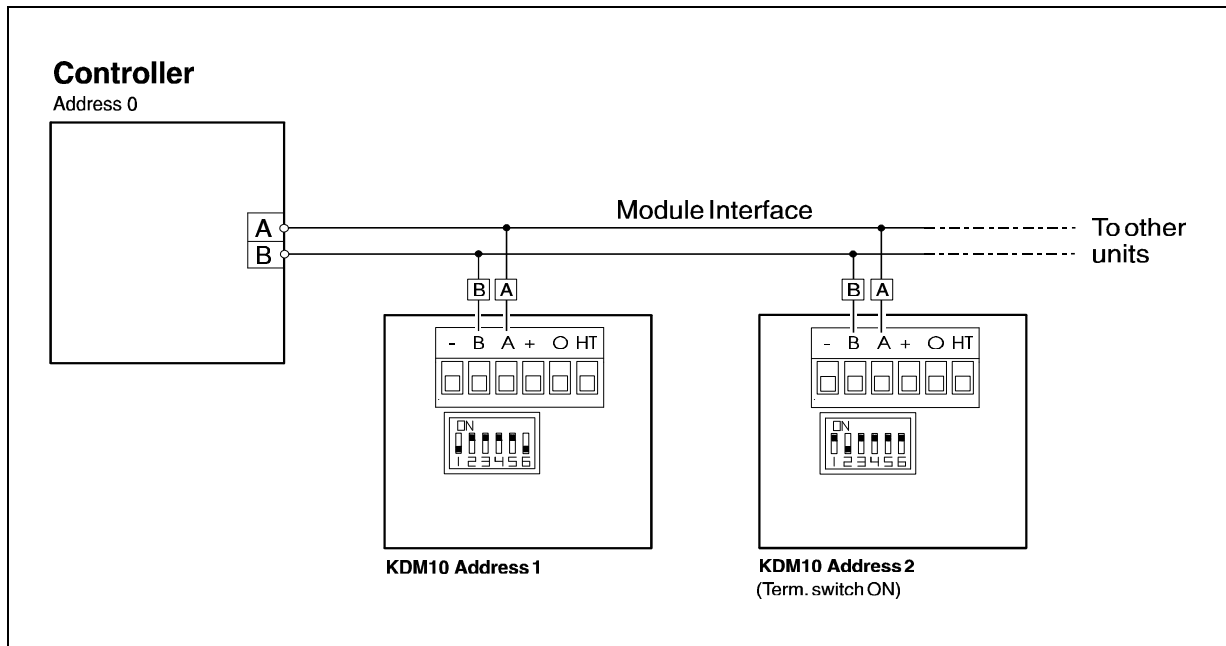


**Power Supply:**

The KDM10 Module is powered via the *P+* and *P-* terminals on the Module rear panel (see above illustration). The supply voltage must be in the range 12 to 45 VDC. To achieve maximum noise immunity, the *P+* and *P-* terminals are optically isolated from interface terminals *A* and *B*.

A 1.5 kW transient protection diode is used to protect against supply line transients. The Module's internal fuse will also be blown if the power supply is connected with incorrect polarity.

The supply voltage for the Module and interface circuitry is generated by a switch-mode power supply which ensures high efficiency and makes the external supply non-critical. If a voltage of more than 46 VDC is applied to the Module, the overload protection circuitry will be activated and the internal fuse will be blown.



#### RS485 Interface "A" and "B":

Interface terminals "A" and "B" are used to connect the Module to a system Controller. All of the Module's functions are controlled via the RS485 interface. Up to 31 modules and at least 1 Controller can be connected to the interface bus. The interface is protected against transients on the bus between the Module and Controller.

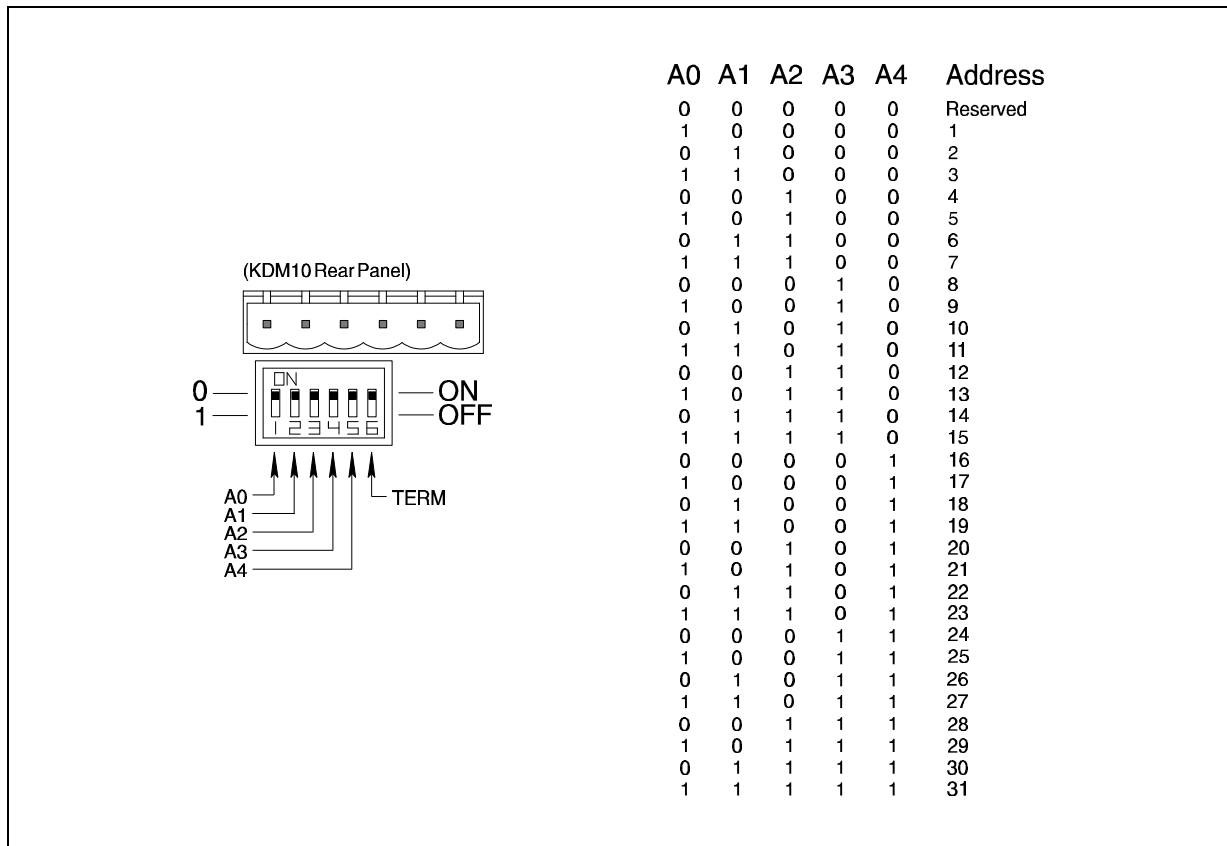
The RS485 Interface offers several advantages because of its balanced, low-impedance operation. These 2 factors enable trouble-free communication over long distances despite the presence of electrical noise.

The Module's 2 interface terminals are denoted *A* and *B* and should be connected to the corresponding *A* and *B* terminals on the Controller.

It is recommended that twisted pair cable is used to connect the KDM10 to other units on the interface.

In systems where the communication distance between 2 units exceeds 25 metres, the *TERM* DIP switch must be set to *ON* on those units which are more than 25m apart.

See the following page for the location of the DIP switch.

**Module Addresses:**

In communication systems where several modules are connected together, each module must be assigned a unique address for communication via the interface bus.

The address can be assigned in the range 1-31.

The above illustration shows the DIP switch settings to assign the Module address. The DIP switch is located on the rear panel of the KDM10.

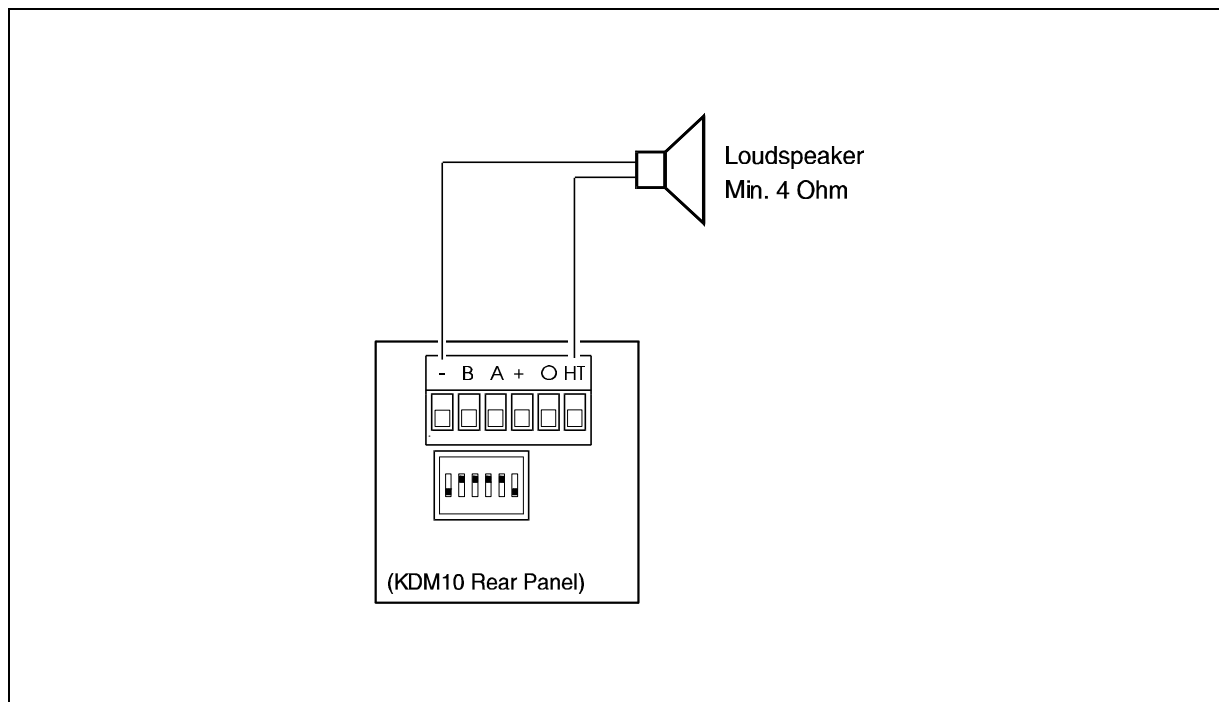
Note that each module address must be unique; no two modules on the interface may be assigned the same address. If the address is not unique, the Controller will stop program execution and indicate an error has occurred.

In some circumstances, the Modules in question may operate incorrectly.

## 1.5

## External Loudspeaker

---



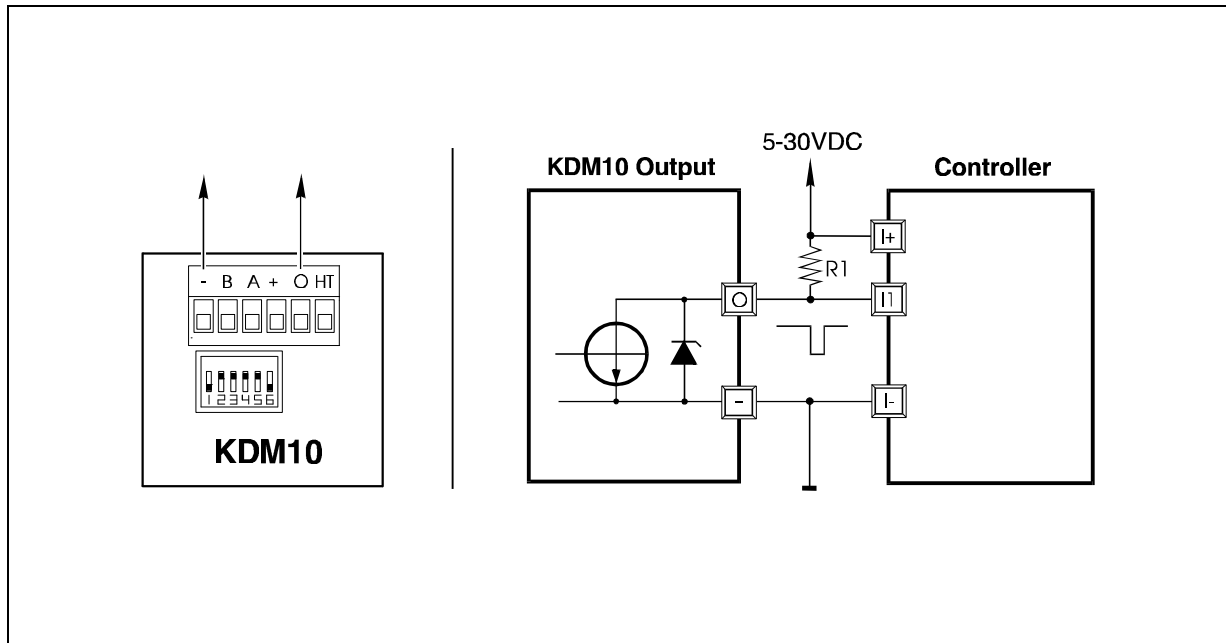
If the KDM10 Module is located in very noisy surroundings, it may not be possible to hear the Module's internal loudspeaker. The KDM10 is therefore equipped with an output which can be used for connecting an external loudspeaker.

This output is connected in parallel with the internal loudspeaker and can therefore be controlled in the same way as the internal loudspeaker.

Note that operation of the internal loudspeaker is unaffected by connection of an external loudspeaker.

The external loudspeaker output can supply approximately 3W RMS.





The Start/Stop Output may be used to start or stop motor operation at any time.

The Output should be used for this purpose, since during motor operation the Controller does not have sufficient time to communicate via the module interface and therefore cannot detect whether the Stop button has been activated.

When the Start/Stop key has been pressed, the output is pulled to ground for 100mS.

Note that the output is a NPN output and must only be loaded up to 0V. The load should be connected between the output and the P+ supply terminal.

This factor does however offer the advantage that the maximum pulse voltage (75V) can be externally controlled.

A "pull-up" resistor should be connected between the Controller's supply input and the pulse output (see above illustration).

If JVL connector board Type CON10 is used, no "pull-up" resistor is required, since this is built in to the CON 10 board.

The CON10 resistor can be used by setting the appropriate input jumper to position "1" - see CON10 documentation for further details.

The value of resistor R1 depends on the input used. It is recommended that R1 is approximately 2.7kOhm for an input voltage of 24V.

## 2.1

## Command Overview

---

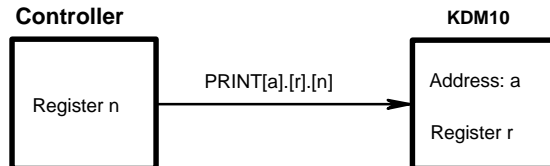
### Print Command :

PRINT[a],[r].[n]

(Print)

Print to register  $r$  in the Module with address  $a$ , the Controller value or register contents  $n$ .

*Example*: PRINT10.8.4



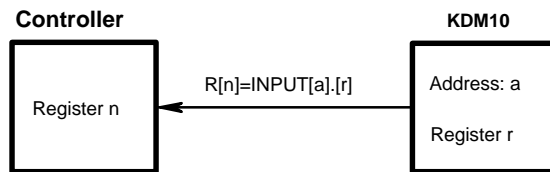
### Input Command :

R[n]=INPUT[a],[r]

(Input)

Read into Controller register  $n$  the contents of register  $r$  from the Module with address  $a$ .

*Example*: R4=INPUT10.8



### Activate Command :

AO[a],[f]

(Activate)

Activate flag  $f$  in Module with address  $a$ .

*Example*: AO10.2



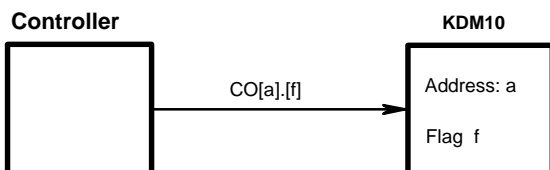
### Clear Command :

CO[a],[f]

(Clear)

Deactivate flag  $f$  in Module with address  $a$ .

*Example*: CO10.2



## 2.2

## Overview of Registers and Flags

Overview of KDM10 registers				Command	
Register	Function	Interval	Default	PRINT	INPUT
0	Print to last cursor position	Num/Text	Blank	Yes	No
1-24	Print to cursor position 1 to 24 (line 1)	Num/Text	Blank	Yes	No
41-64	Print to cursor position 41 to 64 (line 2)	Num/Text	Blank	Yes	No
200	Reserved for future use				
201	Post box 1	0-65535	0	Yes	Yes
202	Post box 2	0-65535	0	Yes	Yes
203	Post box 3	0-65535	0	Yes	Yes
204	Post box 4	0-65535	0	Yes	Yes
205-209	Reserved for future use				
210	Frequency Sound 1	0-10000	1000Hz	Yes	No
211	Duration Sound 1	1-65535	100mSec	Yes	No
212	Volume Sound 1	0-15	10	Yes	No
213	Frequency Sound 2	0-10000	1000Hz	Yes	No
214	Duration Sound 2	1-65535	100mSec	Yes	No
215	Volume Sound 2	0-15	10	Yes	No
216	Frequency Sound 3. Also used as key sound	0-10000	1000Hz	Yes	No
217	Duration Sound 3. Also used as key sound	1-65535	100mSec	Yes	No
218	Volume Sound 3. Also used as key sound	0-15	10	Yes	No
219	Country code register	0	0 (DK)	Yes	No
220	Contrast register	0-15	15	Yes	No
221	Keyboard buffer	0-255	255	No	Yes
222	Keyboard (direct)	0-255	255	No	Yes
254	Numeric register for last keyed value	0-255	0	No	Yes
255	Key register	0-255	-	No	Yes

Overview of KDM10 flags				Command	
Flag	Function	Interval	Default	AO	CO
1	Reset display	0 / 1	0 (Off)	Yes	No
2	Cursor underscore	0 / 1	0 (Off)	Yes	Yes
3	Cursor-block define/undefine	0 / 1	0 (Off)	Yes	Yes
4	Activate Sound 1	0 / 1	0 (Off)	Yes	No
5	Activate Sound 2	0 / 1	0 (Off)	Yes	No
6	Activate Sound 3	0 / 1	0 (Off)	Yes	No

## 2.3

# Print-out to the KDM10

---

The KDM10 display consists of a back-lit LCD display with 2 lines of 24 characters.

The *PRINT* command enables text and register contents to be printed to the Module's LCD display. The command also enables the cursor position to be defined. In addition, the *AO* (Activate) and *CO* (Clear) commands enable the cursor appearance to be changed or the entire display to be cleared.

*Example :*

```
PRINT3.41."No. of sequences : "
```

The above example will print the text "No. of sequences : " on the Module display at cursor position 41 (1st character on line 2, see table below). In the example the Module's DIP switches are set to assign address 3 to the Module. Up to 24 characters may be printed using a single PRINT command, corresponding to the total display width. If cursor position 0 is specified, the text will be displayed at the current cursor position.

*Example :*

```
PRINT9.58.R1
```

The above example will print the contents of register 1 to cursor position 58 on the Module display. A value or other register (R,S,T etc.) can also be specified. If cursor position 0 is specified, the register contents will be displayed at the current cursor position.

The Module address in the example is set to 9.

Cursor Position Table :

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64

### Cursor Control:

In conjunction with input from the KDM10 Module it is often desirable to change the cursor's appearance. This can be achieved by activating or clearing certain flags in the Module. For example, by activating or clearing the appropriate flags, the cursor can be set to blink or a large cursor can be selected. The *AO* (Activate) and *CO* (Clear) commands are used to activate and clear Module flags.

*Example :*

```
AO10.3 ; Cursor area is activated and blinks.
```

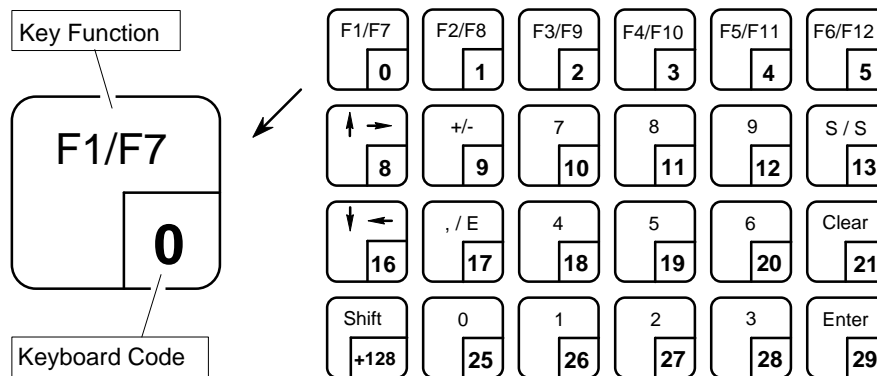
## 2.4

# Input from the KDM10

### Input from the Keyboard Buffer:

Each key on the KDM10 keyboard is designated by a key code. These codes can be read using the *INPUT* command which returns a code corresponding the key which has been pressed. The KDM10 keyboard buffer retains the key codes of the last 2 keys pressed.

Key Codes :



When the *INPUT* command is executed, the first ("oldest") key code in the keyboard buffer is returned. If no key has been pressed or the buffer is empty, a value of 255 is returned.

*Example :* R1=INPUT5.221

The above example reads the key code from the keyboard buffer (register 221) of the Module with address 5. The value returned is in the range 0-255. 255 corresponds to an empty keyboard buffer (no key has been pressed). If the returned value is not 255, a key has been pressed.

Note that the program does not wait for the user to press a key, but reads a value from the keyboard buffer.

By pressing the "Shift" key together with another key the returned keyboardcode is added 128.

The following example illustrates how key codes can be used to control program execution:

*Example :*

```

:READ_BUF      R1=INPUT3.221      ; Read key code from keyboard buffer
                IF R1<>5          ; If key code is not 5 (function key F6)
J:READ_BUF     ; Read keyboard again
J:EXEC_PRG     ; Key code is 5 and program execution continues
    
```

## 2.4

# Input from the KDM10

---

The *INPUT* command enables a user to key in values, parameters, etc., which are then transmitted to the system Controller. These can be transferred to registers and used during program execution.

*Example :*

```
R1=INPUT4.255
```

For example, a user may be required to key in a duration for motor operation. This is typically achieved by first using the *PRINT* command to display a prompt on the Module display such as "Key in Duration:". In the next program line, an *INPUT* command is used to read the keyed input once the Module *ENTER* key has been pressed. If an error has occurred during keying, the *CLEAR* key can be used to clear the Module display. When the *ENTER* key on the KDM10 is pressed, the keyed value is transmitted to a register in the system controller. Values keyed in via the keyboard must be in the range 0-65535.

Note that this form of the *INPUT* command pauses program execution until the *ENTER* key has been pressed. If no value has been entered before the *ENTER* key is pressed, a value of 0 is returned.

*Example :*

```
:READ_LEN      PRINT3.1."Key in Length"      ; Write prompt to LCD display
                R1=INPUT3.255        ; Wait for the user to key in length and press ENTER
                R50=R1                ; New length as specified by user
```

## 2.5

## KDM10 Flags

---

Module KDM10 provides several flags which can be set as required using Controller commands *AO* (Activate) and *CO* (Clear). These commands can be used for example, to change the appearance of the display cursor, activate the Module loudspeaker, etc.

The following flags are available:

- Flag 1:** When this flag is activated, the LCD display is cleared and the cursor located at position 1.
- Flag 2:** If Flag 2 is activated, only the lower 1/8 of the cursor area is activated. If the flag is cleared, the lower 1/8 of the cursor area is switched off. This flag is normally activated before an *INPUT* command where a user is required to key in data. Once the data has been entered, the flag is once more cleared so that the LCD display does not flicker during subsequent *PRINT* commands.
- Flag 3:** When this flag is activated, the upper 7/8 of the cursor area is activated. The cursor blinks. If the flag is cleared, the entire cursor area is switched off.

*Example :*

```
AO3.1           ; The entire LCD display is cleared and the cursor is set to position 1.
PRINT3.1."Key in Length:"
AO3.3           ; The cursor area is activated and blinks, indicating pause for user
                ; input.
R1=INPUT3.255   ; Read user-defined parameter (keyed Length).
CO3.3           ; Cursor area is cleared, position unchanged.
```

The following 3 flags are used to activate the internal loudspeaker in order to provide an audible signal to a user. Module KDM10 enables 3 different sounds to be used, each with different frequency, duration and volume. The sounds are activated by activating one of the 3 flags; the parameters of the generated sound are determined by the contents of corresponding registers (see following page).

- Flag 4:** Activation of Sound 1. Sound 1 parameters are determined by the values specified in registers 210, 211 and 212.
- Flag 5:** Activation of Sound 2. Sound 2 parameters are determined by the values specified in registers 213, 214 and 215.
- Flag 6:** Activation of Sound 3. Sound 3 parameters are determined by the values specified in registers 216, 217 and 218. In addition, the parameters set for Sound 3 also determine the "beep" sound emitted when a key is pressed.

*Example :*

```
PRINT3.210.500 ; Set Frequency of Sound 1 to 500 Hz.
AO3.4           ; Activate Sound 1 using values in registers 210, 211, and 212.
```

Module KDM10 contains a number of registers whose contents can be changed throughout program execution using the *PRINT* command. For example, the frequency, volume and duration of audible signals to the user can be changed. In addition, certain registers can be read using the *INPUT* command.

*Example :*

To change the frequency of Sound 3 to 2000 Hz, the following command is used:

**PRINT3.216.2000**

The following registers are available:

**Register 0:** If this register is specified in a *PRINT* command, the text or value to be printed is displayed at the current cursor position of the display.

*Example :*

PRINT4.0."Number = " ; Print text to be followed by a value.

PRINT4.0.15000 ; Print 15000 to the display at the current cursor position.

**Reg. 201-204:** Post box registers. These registers have no specific function in Module KDM10 but are typically used in systems with more than 1 Controller.

Since 2 Controllers cannot communicate directly with each other, these module registers enable data to be exchanged between Controllers, for example when Controller 1 prints to a Module post box register whose contents are then read by Controller 2.

**Reg. 205-209:** Reserved for future use.

**Register 210:** Loudspeaker frequency for Sound 1.  
The frequency can be specified in the range 1-10000 Hz.

**Register 211:** Duration of Sound 1. Values in the range 1-65535 can be specified, corresponding to a duration in milliseconds: a value of 1 activates Sound 1 for 1 ms; a value of 65535 corresponds to a duration of approximately 65 seconds.

**Register 212:** Volume of Sound 1. Values in the range 0-15 can be specified, with 15 corresponding to maximum volume.



**Register 213:** Frequency of Sound 2. Functions are the same as for register 210.

**Register 214:** Duration of Sound 2. Functions are the same as for register 211.

**Register 215:** Volume of Sound 2. Functions are the same as for register 212.

Note: The following 3 registers determine the frequency, duration and volume of the sound emitted by the Module loudspeaker when a key is pressed. Pressing a key on the Module keyboard corresponds to activating Flag 6 and the contents of registers 216-218 determine the sound emitted.

**Register 216:** Frequency of Sound 3. Functions are the same as for register 210. Sound 3 is also used when a key is pressed.

**Register 217:** Duration of Sound 3. Functions are the same as for register 211. Sound 3 is also used when a key is pressed.

**Register 218:** Volume of Sound 3. Functions are the same as for register 212. Sound 3 is also used when a key is pressed.

**Register 219:** Note!. The function of this register will be introduced in a later version of Module KDM10. Country code register. This register can be used to select the character set used in the Module to enable various national characters to be printed on the LCD display. For example, the special Danish characters Æ, Ø, Å, æ, ø, å are obtained by setting the register to 0. (0 is default).

**Register 220:** Contrast Adjustment. The contrast of the Module's LCD display can be changed by changing the value of register 220. It may be advantageous for example to change the contrast if the display is viewed at an angle. The contrast can be set to values of 0 to 15, with 15 corresponding to highest contrast (default value is 15).

**Register 221:** This register is used to read the contents of the Module keyboard buffer using the INPUT command. If the buffer is empty, a value of 255 is returned. When a key is activated, its key code is transferred to the key buffer. The key code is only transferred once, even if the key is held pressed. The keyboard buffer will therefore not overflow and each key-press is only detected once.

*Example :*

R1=INPUT4.221 ; Read first ("oldest") key code from buffer.

**Register 222:** When the contents of this register are read, a key code is read directly (i.e. not from the keyboard buffer). The key code of the last key pressed is read. In contrast to register 221, when the contents of register 222 are read, the key code is read as long as the key is held pressed. Normally the key code will only be read once. For example, this facility enables an arrow key to be pressed and held. Motor operation will then continue until the key is released (see example below).

*Example :*

```

:START      PRINT6.1."Press Arrow Key for Operation"
:READ      R1=INPUT6.222      ; Get key code.
           IF R1=8            ; If key code = 8 (ARROW up)
           J:RUN+1           ; Jump to routine, move 1 step forward.
           IF R1=16         ; If key code = 16 (ARROW down)
           J:RUN-1           ; Jump to routine, move 1 step back.
           IF R1=29         ; If ENTER key activated, jump
           J:CONTINUE        ; to CONTINUE (continue program)
           J:READ            ; No key has been pressed, read again.
:RUN+1     +1                ; Move 1 step forward.
           J:READ            ; 1 step executed, read key code again.
:RUN-1     -1                ; Move 1 step back
           J:READ            ; 1 step executed, read key code again.
:CONTINUE  A1                ; Continue program execution, activate output 1.
           .
           .

```

**Register 254:** When this register is read using the INPUT command, the last keyed number will be transferred to the Controller. The value in the register is not changed or cleared when it is read by the INPUT command.

**Register 255:** When this register is read, for example using the command R1=INPUT3.255, Module KDM10 is set to receive keyboard input. The user must then key in a value and press *ENTER*. The value will be transferred to one of the Controller's registers. Normally an *INPUT* command of this type will be preceded by a *PRINT* command to prompt the user.

*Example :*

```

:NUMBER    PRINT3.1."Key in Number"      ; Print prompt to display.
           R2=INPUT3.255                 ; Pause for user key input and
                                           ; store in register.
           IF R2<100                     ; If keyed value < 100
           J:LENGTH                       ; Jump to read Length.
           J:NUMBER                       ; Else keyed value too large, read again.
:LENGTH    PRINT3.0."Key in Step"        ; Print prompt to display.
           .
           .

```

If the Module receives a command which cannot be interpreted or for other reasons cannot be executed, an error message is transmitted to the Controller. This error message will stop execution of the Controller's program.

Once program execution has stopped, the Controller "F" (Feedback) command can be used to query the error condition. One of the following error messages will then be returned.

Error Message Syntax:

**EE[e].[a].[n]** EE indicates that a error has occurred in communication with one of the external modules. *e* indicates the type of error that has occurred, *a* indicates the address of the module in which the error has occurred. *n* indicates the program line number at which the error occurred.

**Error Types (e):**

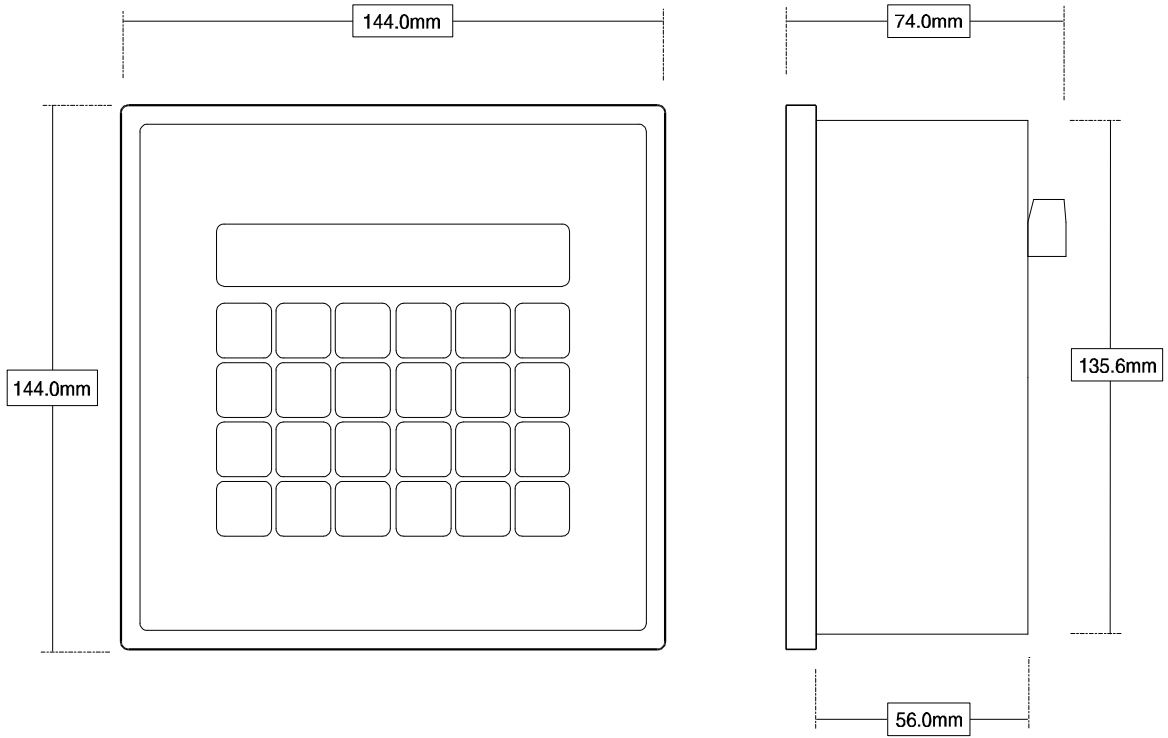
- 1 A Checksum Error occurred during communication.
- 2 Communication Error. Successive errors occurred in communication (start/stop bit). This may be caused by excessive noise transients in the communication channel (A and B). Use 2-core screened or twisted-pair cable. If screened cable is used, the screen must be earthed and both A and B must be screened.
- 3 A command has been used that is not recognised by the Module or that the Module is unable to comply with. This error may arise for example if an attempt is made to print to a KDM10 register which does not exist.

### 3.1

## Electrical Specifications

	Min.	Typ.	Max.	Units
<b>Power Supply :</b>				
Supply Voltage	12		45	VDC
Power Consumption		1.6		W
<b>Module interface (RS485) :</b>				
Communication Rate		50		kbit/sec.
Communication Distance			100	Metre
<b>Loudspeaker Output :</b>				
Output			3	W
Loudspeaker Impedance	4		8	Ohm
<b>Start/stop Output ("O") :</b>				
Voltage Range	0		75	VDC
Output Current	0		-100	mA
<b>Diverse :</b>				
Operating Temperature KDM10T	0		45	°C
Operating Temperature KDM10D	0		50	°C

### KDM10T



### KDM10D

