



ifm electronic



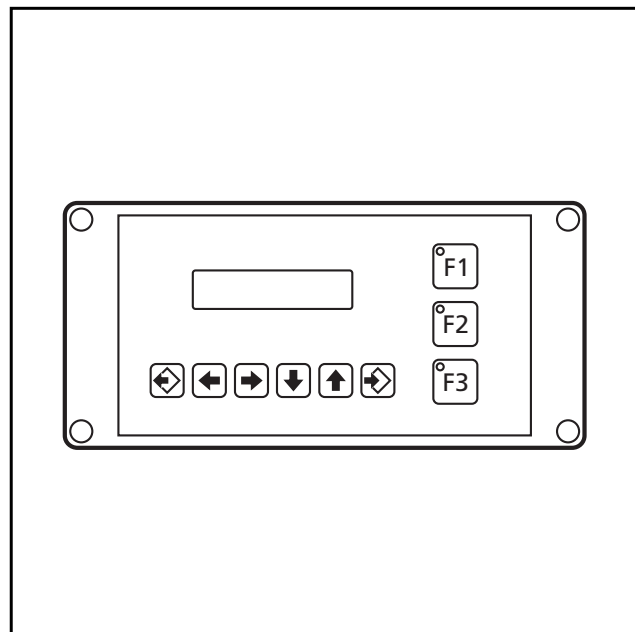
Bedienungsanleitung Installation Instructions

ecomat 100

Daten Dialog Modul
DDM 100
Data dialogue module
DDM 100

CS0014

Sachnr. 7390319/00 02/2004



DEUTSCH

ENGLISH

Inhaltsverzeichnis

1. Funktionen des DDM 100	4
2. Montage	5
2.1 Schaltschrankmontage	5
2.2 Wandmontage	6
3. Programmieren	7
3.1 Programmieren mit Datenübergabe	7
Beschreibung	7
Programmierbeispiel in FUP	9
3.2 Programmieren mit Adreßübergabe	10
Beschreibung	10
Programmierbeispiel in ST	11
3.3 Funktionstasten und LEDs programmieren	12
3.4 Kommunikation zwischen Steuerung und DDM freigeben	12
4. Bedienung	13
4.1 Normalbetrieb	13
4.2 Parametriermodus	15
In den Parametriermodus wechseln	15
Sprachauswahl	15
Meldezeit einstellen	16
Blinkzeit einstellen	16
CAN Identifizier einstellen	16
Bit Rate einstellen	16
4.3 Tastenübersicht	17
5. Erste Hilfe	18
6. Reinigung	18
7. Technische Daten	19
8. Konformitätserklärung	19

1. Funktionen des DDM 100

Das DatenDialogModul DDM 100 mit CAN-Schnittstelle ist ein Anzeige- und Eingabemodul für ifm-Steuerungssysteme mit CAN-Technik.

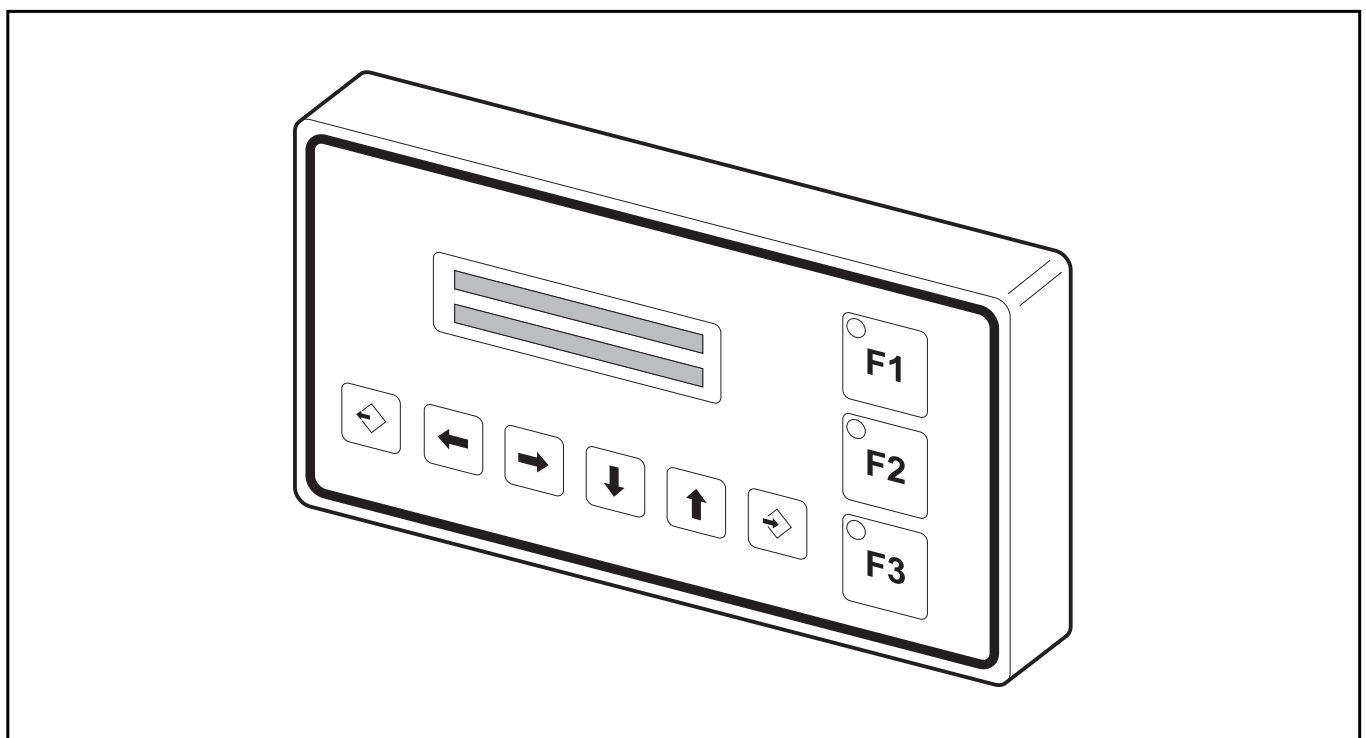
Das DDM hat folgende Eigenschaften und Funktionen:

Montage

- Schaltschrank- oder Wandmontage
- Dank der geringen Bauhöhe von nur 32 mm vielseitig und platzsparend montierbar
- Montage und Anschluß des DDM vor der Programmierung möglich, da das DDM über die Steuerung programmiert wird

Programmierung

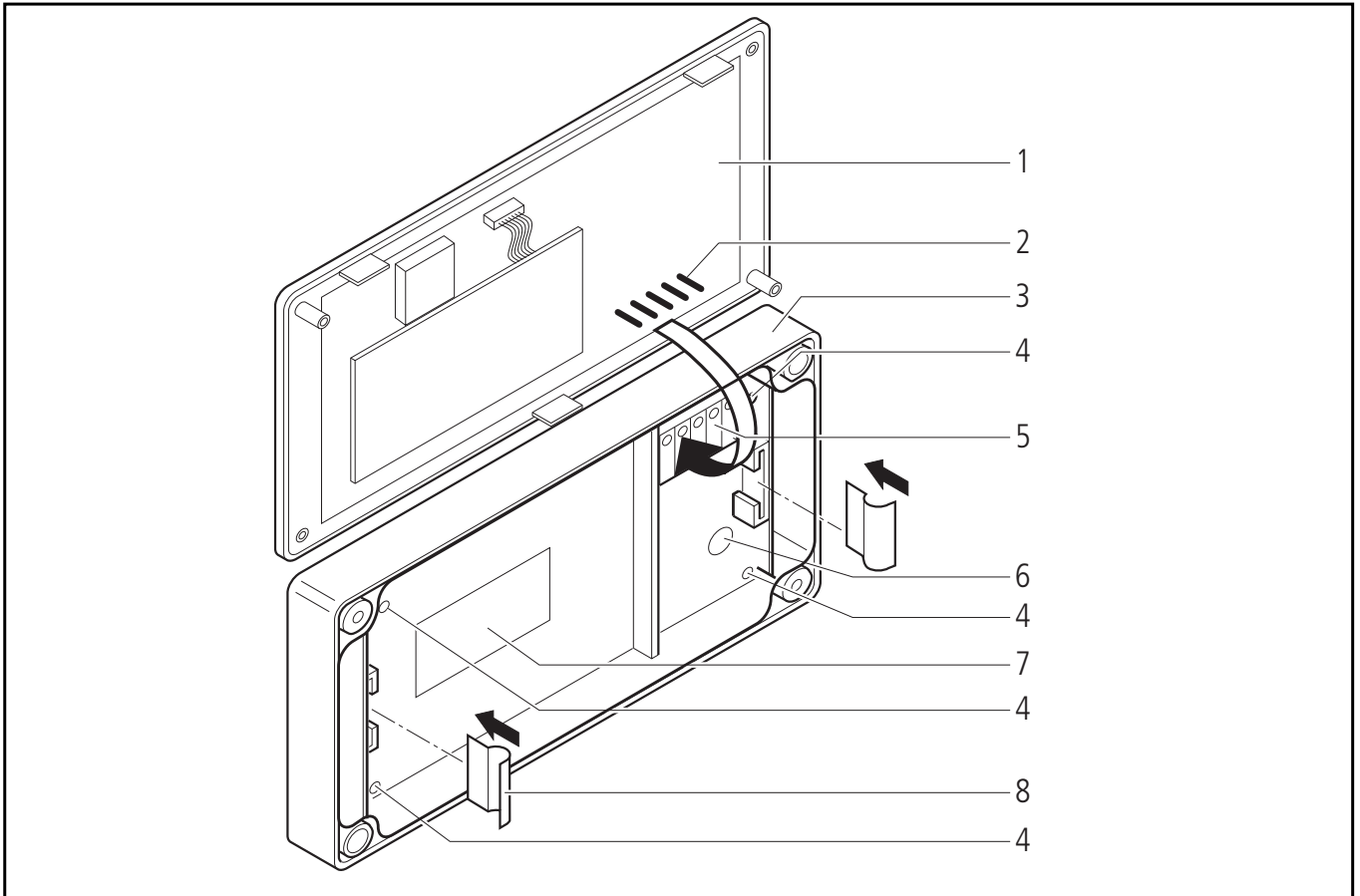
- Erstellen und Steuern der Anzeigetexte für das DDM innerhalb des Steuerungsprogramms
- Unbegrenzte Anzahl von programmierten Anzeigetexten
- 16 aktive zweizeilige Anzeigetexte mit 16 Zeichen pro Zeile für die hinterleuchtete Textanzeige
- Programmierbare Anzeigereihenfolge (Priorität) der bis zu 16 aktiven Texte
- Anzeigetexte mit gegebenenfalls eingebundenen Prozeßdaten
- Editierbarkeit der eingebundenen Prozeßdaten
- Freie Belegung der 3 Funktionstasten
- Freie Belegung der 3 LEDs



2. Montage



C-MOS-Bauelemente werden durch statische Aufladung beschädigt. Vor dem Montieren des Gehäuseoberteils geerdetes Teil anfassen, um die statische Aufladung abzubauen.



2.1 Schaltschrankmontage

1. Mit Hilfe der beiliegenden Schablone einen Ausschnitt mit 4 Bohrungen in der Fronttafel des Schaltschranks vorbereiten (maximale Dicke der Fronttafel 2,5mm).
2. Gehäuse-Unterteil (3) von der Schaltschranksinnenseite her in den Ausschnitt einsetzen.
3. Die beiliegenden Federklammern (8) von der Schaltschrankaußenseite her in die Halterungen rechts und links im Gehäuseunterteil schieben und Gehäuseunterteil im Ausschnitt befestigen.
4. Sollbruchstelle für Kabeldurchführung (6) ausbrechen.
5. Beiliegende PG16-Kabeldurchführung einsetzen.
6. Kabel durchführen.
7. Anschlüsse gemäß Aufkleber (7) im Gehäuseunterteil (3) herstellen. Das Gerät muß von Fachpersonal angeschlossen werden.
8. Gehäuseoberteil (1) so auf das Gehäuseunterteil (3) setzen, daß die Kontaktstifte (2) des Gehäuseoberteils in die Klemmen (5) im Gehäuseunterteil eingreifen.
9. Gehäuseoberteil auf dem Gehäuseunterteil festschrauben.

2.2 Wandmontage

Siehe Montageskizze auf Seite 5.

1. Sollbruchstelle für Kabeldurchführung (6) ausbrechen oder Kabeldurchführung in unterer Seitenwand innerhalb des Anschlußraumes bohren (siehe beiliegende Schablone) und entsprechendes Gewinde PG7 schneiden.
2. Gegebenenfalls beiliegende PG11-Kabeldurchführung einsetzen.
3. Kabel durchführen.
4. Gehäuseunterteil mit vier Schrauben o. ä. (4) an der Wand befestigen.
5. Anschlüsse gemäß Aufkleber (7) im Gehäuseunterteil (3) herstellen. Das Gerät muß von Fachpersonal angeschlossen werden.
6. Gehäuseoberteil (1) so auf das Gehäuseunterteil (3) setzen, daß die Kontaktstifte (2) des Gehäuseoberteils in die Klemmen (5) im Gehäuseunterteil eingreifen.
7. Gehäuseoberteil auf dem Gehäuseunterteil festschrauben.

3. Programmieren

Um das DDM 100 in eine Anlagensteuerung einzubinden, müssen Sie folgendes innerhalb des Applikationsprogramms der Steuerung programmieren:

- Freigabe der Kommunikation zwischen Steuerung und DDM mittels der Systemvariable TEXT_MODE
- Texte, die angezeigt werden sollen
- Anzeigepriorität der Texte
- Prozeßdaten, die gegebenenfalls in die Texte eingebunden werden
- gegebenenfalls Freigabe der eingebundenen Prozeßdaten zum Editieren
- Belegung der Funktionstasten
- Belegung der LEDs

Das Applikationsprogramm der Steuerung schreiben Sie in der Programmier- und Entwicklungsumgebung ecolog 100^{plus}. Diese ermöglicht das Programmieren mit Programmiersprachen entsprechend IEC 1131. Das Programmieren der Steuerung mit ecolog 100^{plus} ist in einem separaten Handbuch beschrieben.

Es gibt zwei Strategien, um die Texte für das DDM zu programmieren:

- Datenübergabe (siehe Kapitel 3.1)
- Adreßübergabe (siehe Kapitel 3.2; empfohlen bei laufzeitkritischen Anwendungen und bei Anwendungen mit vielen Texten)

3.1 Programmieren mit Datenübergabe

Beschreibung

In der gerätespezifischen Bibliothek der Steuerung finden Sie den Baustein "TEXTD". Für jeden Anzeigetext erstellen Sie eine Instanz des Bausteins "TEXTD".

Jeder Instanz weisen Sie die Textdaten für den zugehörigen Anzeigetext zu:

- ENABLE : BOOL; Textfreigabe
- PRIORITY : BYTE; Anzeigepriorität 1 ... 7, 7 = höchste
- LINE1 : STRING(20); Anzeigetext Zeile 1, ggf. mit Platzhalter "%" und Formatierungszeichen für einzubindende Prozeßdaten
- LINE1_VALUE1 : INT; der von links nach rechts erste eingebundene Wert in Zeile 1
- LINE1_MIN1 : INT; die untere Editiergrenze, Wert 1
- LINE1_MAX1 : INT; die obere Editiergrenze, Wert 1
- LINE1_EDIT1: BOOL; Editierfreigabe für Wert 1
- LINE1_VALUE2 : INT; der zweite Wert
- LINE1_MIN2 : INT; die untere Editiergrenze, Wert 2
- LINE1_MAX2 : INT; die obere Editiergrenze, Wert 2

- LINE1_EDIT2: BOOL; Editierfreigabe für Wert 2
- LINE2 : STRING(20); Anzeigetext Zeile 2, ggf. mit Platzhalter "%" und Formatierungszeichen für einzubindende Prozeßdaten
- LINE2_VALUE1 : INT; der von links nach rechts erste eingebundene Wert in Zeile 2
- LINE2_MIN1 : INT; die untere Editiergrenze, Wert 1
- LINE2_MAX1 : INT; die obere Editiergrenze, Wert 1
- LINE2_EDIT1: BOOL; Editierfreigabe für Wert 1
- LINE2_VALUE2 : INT; der zweite Wert
- LINE2_MIN2 : INT; die untere Editiergrenze, Wert 2
- LINE2_MAX2 : INT; die obere Editiergrenze, Wert 2
- LINE2_EDIT2: BOOL; Editierfreigabe für Wert 2

Die Textzeilen können blinkend dargestellt werden.

Die blinkende Darstellung wird aktiviert, indem als erstes Zeichen ein „\“ eingetragen wird (Backslash, ASCII 5Chex).

In jede Zeile des Anzeigetextes können maximal zwei Werte (Prozeßdaten / Variable) eingebunden werden.

Anzeigeformate für die eingebundenen Werte:

%nd	=	SIGNED INT	=	-32767 ... 0 ... 32767
%nu	=	UNSIGNED INT	=	0 ... 65535
%nx	=	HEX	=	0 ... ffff
%nX	=	HEX	=	0 ... FFFF

n = 0 ... 5; optional verwendbar:

Anzahl der dargestellten Stellen der Prozeßdaten, ohne n werden alle Stellen dargestellt (ohne führende Nullen)

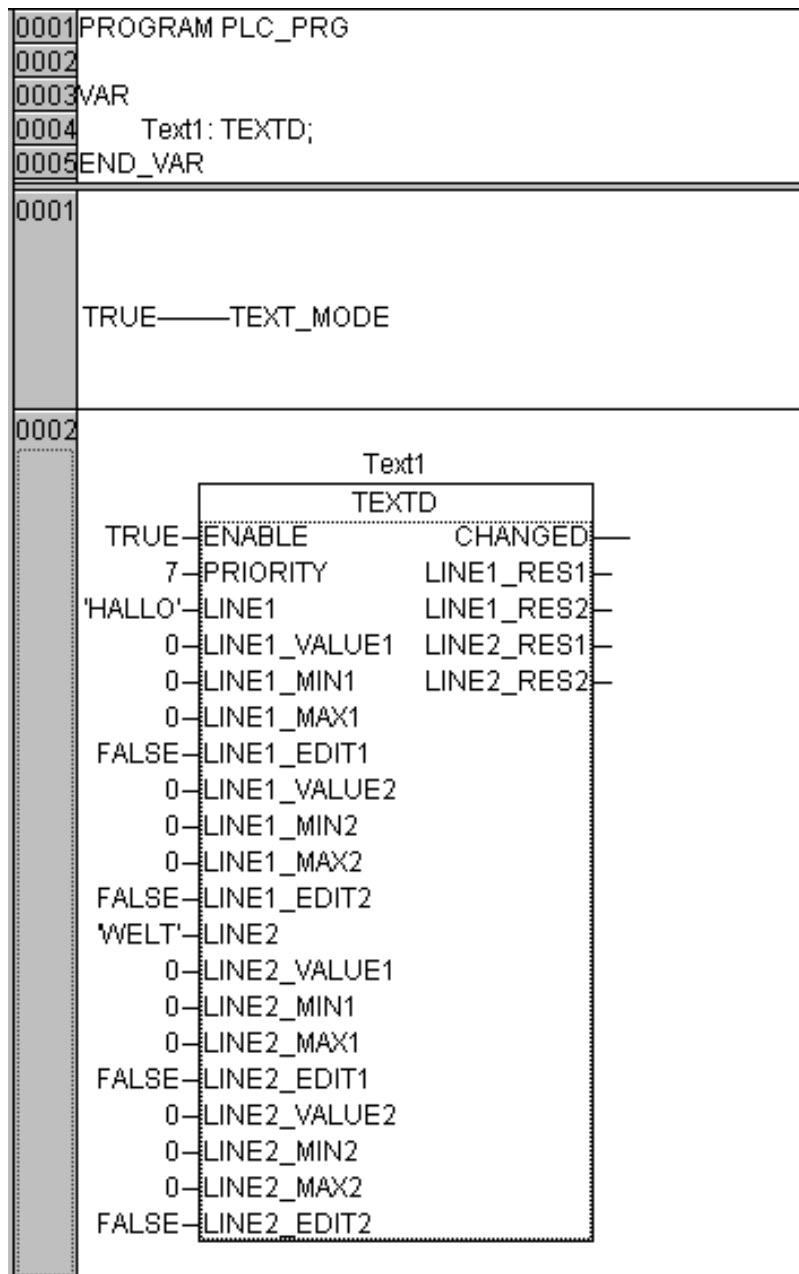
Jede Instanz liefert an den Ausgängen folgende Informationen:

- CHANGED : BOOL = TRUE, wenn einer der editierbaren Werte geändert wurde
- LINE1_RES1; der geänderte erste Wert von Zeile 1
- LINE1_RES2; der geänderte zweite Wert von Zeile 1
- LINE2_RES1; der geänderte erste Wert von Zeile 2
- LINE2_RES2; der geänderte zweite Wert von Zeile 2

Alle Instanzen des Bausteins "TEXTD" müssen zyklisch aufgerufen werden.

Programmierbeispiel in FUP

Der Programmauszug zeigt eine Instanz Text 1 des Bausteins "TEXTD" mit zugewiesenen Textdaten in der Programmiersprache Funktionsplan (FUP). Das Programm bringt den Text "HALLO WELT" in die Anzeige



3.2 Programmieren mit Adreßübergabe

Beschreibung

In der gerätespezifischen Bibliothek der Steuerung finden Sie die Datenstruktur "text_struct". Für jeden Anzeigetext definieren Sie eine Variable vom Typ "text_struct". Den Variablen vom Typ "text_struct" weisen Sie die Textdaten zu, indem Sie die Komponenten der Struktur mit den entsprechenden Daten belegen.

- .enable : BOOL; Textfreigabe
- .priority : BYTE; Anzeigepriorität 1 ... 7, 7 = höchste
- .line1 : STRING(20); Anzeigetext Zeile 1, ggf. mit Platzhalter "%" und Formatierungszeichen für einzubindende Prozeßdaten
- .line1_value1 : INT; der von links nach rechts erste eingebundene Wert in Zeile 1
- .line1_min1 : INT; die untere Editiergrenze, Wert 1
- .line1_max1 : INT; die obere Editiergrenze, Wert 1
- .line1_edit1 : BOOL; Editierfreigabe für Wert 1
- .line1_value2 : INT; der zweite Wert
- .line1_min2 : INT; die untere Editiergrenze, Wert 2
- .line1_max2 : INT; die obere Editiergrenze, Wert 2
- .line1_edit2 : BOOL; Editierfreigabe für Wert 2
- .line2 : STRING(20); Anzeigetext Zeile 2, ggf. mit Platzhalter "%" und Formatierungszeichen für einzubindende Prozeßdaten
- .line2_value1 : INT; der von links nach rechts erste eingebundene Wert in Zeile 2
- .line2_min1 : INT; die untere Editiergrenze, Wert 1
- .line2_max1 : INT; die obere Editiergrenze, Wert 1
- .line2_edit1 : BOOL; Editierfreigabe für Wert 1
- .line2_value2 : INT; der zweite Wert
- .line2_min2 : INT; die untere Editiergrenze, Wert 2
- .line2_max2 : INT; die obere Editiergrenze, Wert 2
- .line2_edit2 : BOOL; Editierfreigabe für Wert 2

Die Komponenten der Struktur sind mit 0 bzw. FALSE vorgelegt.

In jede Zeile des Anzeigetextes können maximal zwei Werte (Prozeßdaten / Variable) eingebunden werden.

Vorgesehene Anzeigeformate für die eingebundenen Werte:

%nd = SIGNED INT = -32767 ... 0 ... 32767

%nu = UNSIGNED INT = 0 ... 65535

%nx = HEX = 0 ... ffff

%nX = HEX = 0 ... FFFF

n = 0 ... 5; optional verwendbar:

Anzahl der dargestellten Stellen der Prozeßdaten, ohne n werden alle Stellen dargestellt (ohne führende Nullen)

In der gerätespezifischen Bibliothek der Steuerung finden Sie den Baustein "TEXTA". Von diesem Baustein erstellen Sie eine Instanz. Die Instanz schickt die Textdaten eines Anzeigetextes an das DDM 100. Die Instanz muß also zyklisch mit den Adressen aller definierten Variablen vom Typ "text_struct" aufgerufen werden.

Hierzu wird mit der Funktion "ADR()" die Adresse der Text-Variablen ermittelt.

Die Adresse vom Typ "DINT" wird mit der Funktion "DINT_TO_INT" in eine Variable vom Typ "INT" konvertiert und einer Instanz von TEXTA übergeben.

Programmierbeispiel in ST

Hinweis

Unten sehen Sie einen kommentierten Programmauszug mit Adreßübergabe in der Programmiersprache Strukturierter Text (ST).

Das Programm bringt den Text "HALLO WELT" in die Anzeige.

0001	PROGRAM PLC_PRG	
0002	VAR	
0003		
0004	Text1_dat : text_struct;	(* Definition von Text1_dat. Alle Komponenten *)
0005		(* von text_struct sind mit NULL / FALSE vorbelegt *)
0006		(* Define Text1_dat. All elements of text_struct *)
0007		(* have default setting NULL / FALSE *)
0008		
0009	Text1_adr : DINT;	(* Definition einer Variablen zur Aufnahme der *)
0010		(* Adresse von Text1_dat *)
0011		(* Definition of variable for the address of Text1_dat *)
0012		
0013	Text_Kom : TEXTA;	(* Instanz von TEXTA *)
0014		(* Instance of TEXTA *)
0015		
0016	END_VAR	
0001		
0002	TEXT_MODE := TRUE;	(* Schnittstelle zum DDM freigeben *)
0003		(* Enable DDM interface *)
0004		
0005	Text1_dat.enable := TRUE;	(* Text1 freigeben *)
0006		(* Enable Text1 *)
0007		
0008	Text1_dat.priority := 7;	(* Anzeigepriorität festlegen *)
0009		(* Define priority *)
0010		
0011	Text1_dat.line1 := 'Hello';	(* Anzeigetext der Zeile 1 *)
0012		(* Define text of line 1 *)
0013		
0014	Text1_dat.line2 := 'World';	(* Anzeigetext der Zeile 2 *)
0015		(* Define text of line 2 *)
0016		
0017	Text1_adr := ADR(Text1_dat);	(* Adresse von Text1 bilden und in Text1_adr ablegen *)
0018		(* Create address of Text1_dat and store in Text1_adr *)
0019		
0020	Text_Kom(A := Text1_adr);	(* Text_Kom mit der Adresse von Text1 aufrufen *)
0021		(* Call Text_Kom with the address of Text1 *)
0022		
0023		

3.3 Funktionstasten und LEDs programmieren

Die 3 Funktionstasten (F1, F2, F3) und die 3 LEDs (F1, F2, F3) können Sie frei belegen. Die Editiertasten sind fest belegt.

In der Steuerungskonfiguration finden Sie folgende Variablen, die die Tasten- und LED-Zustände beinhalten:

Funktionstasten	
Variable	Typ
TEXT_KEYS	WORD
TEXT_KEY_F1	BOOL (TEXT_KEYS, Bit 0)
TEXT_KEY_F2	BOOL (TEXT_KEYS, Bit 1)
TEXT_KEY_F3	BOOL (TEXT_KEYS, Bit 2)

Editiertasten	
Variable	Typ
TEXT_KEYS	WORD
TEXT_KEY_ESC	BOOL (TEXT_KEYS, Bit 3)
TEXT_KEY_LEFT	BOOL (TEXT_KEYS, Bit 4)
TEXT_KEY_RIGHT	BOOL (TEXT_KEYS, Bit 5)
TEXT_KEY_DOWN	BOOL (TEXT_KEYS, Bit 6)
TEXT_KEY_UP	BOOL (TEXT_KEYS, Bit 7)
TEXT_KEY_ENTER	BOOL (TEXT_KEYS, Bit 8)

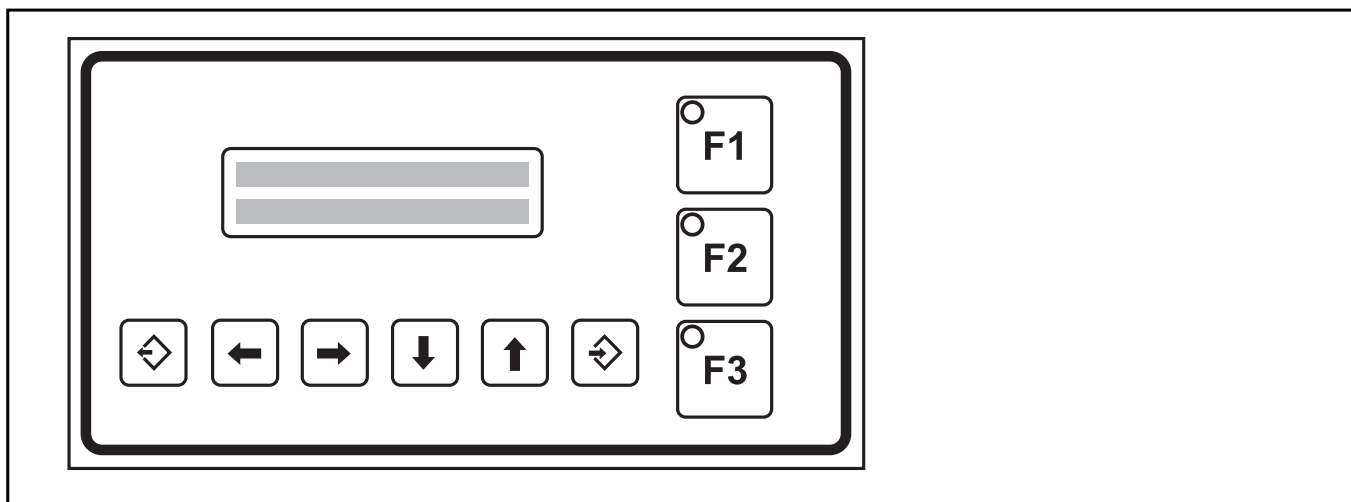
LEDs	
Variable	Typ
TEXT_LEDS	WORD
TEXT_LED_F1	BOOL (TEXT_LEDS, Bit 0)
TEXT_LED_F2	BOOL (TEXT_LEDS, Bit 1)
TEXT_LED_F3	BOOL (TEXT_LEDS, Bit 2)

3.4 Kommunikation zwischen Steuerung und DDM freigeben

In der gerätespezifischen Bibliothek der Steuerung ist für die Kommunikation zwischen Steuerung und DDM die globale Variable TEXT_MODE: BOOL definiert.

Kommunikation zwischen Steuerung und DDM freigeben:

- Im Steuerungsprogramm die System-Variable TEXT_MODE auf TRUE setzen.



4. Bedienung

4.1 Normalbetrieb

Nach dem Einschalten der Steuerung erscheint in der Anzeige des DDM:

ecomat 100
CS 0014

Das DDM ist betriebsbereit.

Anschließend erscheint der Text mit der höchsten Anzeigepriorität. Wenn kein Text aktiviert ist, erscheint

Keine
Meldungen

Wenn keine Verbindung mit der Steuerung zustande kommt, weil die Steuerung z. B. nicht eingeschaltet ist, erscheint in der Anzeige:

FEHLER
Schnittstelle

Sobald die Kommunikation zwischen DDM100 und Steuerung steht, erlischt die Fehlermeldung und es erscheint der Text mit der höchsten Anzeigepriorität.

Anderen aktiven Text abrufen:

→ Mit  oder  blättern



Die Texte erscheinen in der Reihenfolge der vergebenen Priorität.
Wenn keine Taste mehr gedrückt wird, bleibt der gewählte Text für die eingestellte Meldezeit in der Anzeige (Seite 15, Kapitel "Meldezeit der Texte einstellen"). Anschließend erscheint wieder der aktuell gültige Text mit der höchsten Priorität.



Vor Ablauf der Meldezeit wieder den Text mit der höchsten Priorität anzeigen:

→  drücken



Prozeßdaten in einem Text editieren:


→ Mit  oder  den Cursor in die Anzeige rufen

→ Mit  oder  den Cursor an die gewünschte Stelle des zu ändernden Wertes bewegen

→ Mit  oder  die Wertstelle über dem Cursor dekrementieren oder inkrementieren

Beim Überschreiten von 9 bzw. 0 findet ein Übertrag in die nächsthöhere Stelle des Wertes statt.

→ Wenn  oder  länger als 1 s gedrückt wird, wird automatisch dekrementiert bzw. inkrementiert

→ Editieren abbrechen mit 
Der Cursor verschwindet aus der Anzeige. Geänderte Werte werden nicht übernommen.

→ Editieren abschließen mit 
Der Cursor verschwindet aus der Anzeige. Die geänderten Werte werden übernommen und der Steuerung übergeben.

4.2 Parametriermodus

Im Parametriermodus können folgende Geräteeinstellungen vorgenommen werden:

- Sprache der fest hinterlegten Texte Englisch/Deutsch.
- Meldezeit eines Textes mit niedriger Priorität bis zum Rückfall auf den Text mit der höchsten Priorität wenn keine Taste mehr betätigt wird. Gleichzeitig Rückfallzeit aus dem Editiermodus wenn keine Taste mehr betätigt wird.
- Blinkzeit (wenn eine Textzeile blinkend dargestellt werden soll).
- CAN-Identifizier der Nachrichten die von der Steuerung an das DDM 100 gesendet werden. Die Kommunikation in der Richtung von der Steuerung zum DDM 100 umfasst 2 CAN-Nachrichten. Eingestellt wird der Identifizier der ersten Nachricht. Der Identifizier der zweiten Nachricht errechnet sich nach ID 1. Nachricht +1. Die Identifizier für die Kommunikation vom DDM 100 in Richtung Steuerung werden dem DDM 100 von der Steuerung übergeben.

Achtung: Eine Änderung der Identifizier der CAN-Nachrichten muss in der Steuerung und im DDM 100 durchgeführt werden. Abhängig vom Betriebssystem der Steuerung ist dies nicht immer möglich. In diesem Fall ist der hier einzustellende Identifizier 7F0 hex.

- Bitrate der CAN-Schnittstelle. Aus einer Liste von 8 fest hinterlegten Bitraten kann eine Auswahl getroffen werden. Die Bitrate des DDM 100 muß mit der Bitrate der Steuerung übereinstimmen.

Wechsel in den Parametriermodus

Nach dem Einschalten der Versorgungsspannung erscheint in der Anzeige der Text:



```
ecomat 100
CS0014
```

Während der Text erscheint, die Tasten  +  gleichzeitig einmal kurz betätigen.

Sprachauswahl

Mit den Tasten  oder  folgenden Text in die Anzeige rufen:

```
PARAMETER
Language English / Sprache Deutsch
```

Mit den Tasten  oder  die gewünschte Sprache einstellen.

Eingabe übernehmen mit Taste .


Eingabe verwerfen mit Taste .

Meldezeit einstellen

Mit den Tasten  oder  folgenden Text in die Anzeige rufen:

PARAMETER
display time xx s/ Meldezeit xx s

Mit den Tasten  oder  die Änderung des Wertes einleiten und den Cursor auf die gewünschte Stelle positionieren

Mit den Tasten  oder  den gewünschten Wert einstellen.



Eingabe übernehmen mit Taste . Eingabe verwerfen mit Taste .

Blinkzeit einstellen

Mit den Tasten  oder  folgenden Text in die Anzeige rufen:

PARAMETER
blink xxxx ms/ Blinker xxxx ms

Mit den Tasten  oder  die Änderung des Wertes einleiten und den Cursor auf die gewünschte Stelle positionieren



Mit den Tasten  oder  den gewünschten Wert einstellen.

Eingabe übernehmen mit Taste . Eingabe verwerfen mit Taste .

CAN Identifizier einstellen

Mit den Tasten  oder  folgenden Text in die Anzeige rufen:

PARAMETER
Rx Ident xxx / Rx Ident xxx

Mit den Tasten  oder  die Änderung des Wertes einleiten und den Cursor auf die gewünschte Stelle positionieren

Mit den Tasten  oder  den gewünschten Wert einstellen.


Eingabe übernehmen mit Taste . Eingabe verwerfen mit Taste .

Bit Rate einstellen


Mit den Tasten  oder  folgenden Text in die Anzeige rufen:

PARAMETER
BitRate xxx kBit/s / BitRate xxx kBit/s

Mit den Tasten  oder  durch die Liste der hinterlegten BitRate blättern.

Nach dem Ändern von CAN Identifizier oder Bit Rate mit der Taste  folgenden Text in die Anzeige rufen:

PARAMETER
load / übernehmen

Mit der Taste  die Übernahme der geänderten CAN-Parameter auslösen.

Das Gerät verhält sich anschließend wie nach dem Einschalten der Versorgungsspannung und zeigt folgenden Text in der Anzeige:

ecomat 100
CS0014

4.3 Tastenübersicht



	Funktionstaste F1 und LED, Funktion nach Programm
	Funktionstaste F2 und LED, Funktion nach Programm
	Funktionstaste F3 und LED, Funktion nach Programm
	Escape-Taste, Editieren abbrechen oder Text mit höchster Priorität anzeigen
	Editiertaste, Cursor nach links bewegen
	Editiertaste, Cursor nach rechts bewegen
	Editiertaste, Wert an Cursorposition verringern oder Text mit niedrigerer Priorität anzeigen
	Editiertaste, Wert an Cursorposition erhöhen oder Text mit höherer Priorität anzeigen
	Enter-Taste, editierten Wert übernehmen

5. Erste Hilfe

Problem	mögliche Abhilfe
Das DDM reagiert nicht	Verkabelung und Versorgung prüfen
Eine Funktionstaste reagiert nicht	Programmablauf für das DDM im Steuerungsprogramm prüfen. Gehäuseoberteil einschicken
Eine Editiertaste reagiert nicht	Gehäuseoberteil einschicken
Die Fehlermeldung "Keine Meldungen" erscheint?	Programmablauf für das DDM im Steuerungsprogramm Texte aktivieren
Die Fehlermeldung "FEHLER Schnittstelle" erscheint?	Im Steuerungsprogramm prüfen, ob die Kommunikation zwischen Steuerung und DDM freigegeben ist. Falls nötig die Variable TEXT_MODE auf TRUE setzen. Verkabelung bzw. Steuerung prüfen.

6. Reinigung



Die Folientastatur kann durch unsachgemäße Reinigung verkratzen und unbrauchbar werden.

Reinigen Sie die Folientastatur des DDM mit einem Reinigungsmittel, das Polyester nicht angreift. Verwenden Sie einen weichen Lappen.

7. Technische Daten

Einsatzbereich	Anzeige- und Eingabemodul für Steuerungen
Anzeige	Hinterleuchtete LCD-Anzeige; 2 Zeilen zu je 16 Zeichen, frei programmierbar
Betriebsspannung [V]	10 ... 30 DC (SELV)
Stromaufnahme [mA]	100 (Die Stromaufnahme im Fehlerfall muß durch geeignete Maßnahmen auf max. 1 A begrenzt werden)
Schutzart	IP64 / III
Umgebungstemperatur [°C]	0 ... +50
Gehäuseabmessungen [mm]	190 x 90 (Länge x Breite) Bauhöhe bei Wandmontage = 32mm Bauhöhe bei Schaltschrankmontage = 7mm Einbautiefe bei Schaltschrankmontage = 25mm Maximale Dicke der Fronttafel = 2,5mm
Bedientasten	insgesamt 9, davon 3 Funktionstasten frei belegbar, 6 Tasten fest belegt
Funktionstasten mit LEDs	über Anwenderprogramm frei programmierbar
LED	3 x gelb, frei programmierbar
Schnittstelle	CAN: Spezifikation 2.0 A/B
Baudrate:	125 kBaud, High speed nach ISO/DIS 11898
Identifizier:	7F0 HEX ... 7F1 HEX (DDM empfängt Textliste/Textdaten) 7F8 HEX ... 7FA HEX (DDM sendet Textaufruf, editierte Prozeßdaten, Tastenzustände)
Anschlüsse	5 schraubenlose WAGO-Klemmen
Anschlußbelegung	1: CAN_H 2: CAN_L 3: GND 4: GND 5: +24 V

8. Konformitätserklärung

Das CE-Zeichen wird angebracht auf der Basis der EMV-Richtlinie EMV 89/336/EWG, realisiert in den Normen EN 500 81-1 und EN 61000-6-2, ferner auf der Basis der Niederspannungsrichtlinien, realisiert in den Normen EN 61010 und EN 61010 / A1:1995.

Contents

1. Functions of the DDM 100	21
2. Mounting	22
2.1 Control cabinet mounting	22
2.2 Wall mounting	23
3. Programming	24
3.1 Programming with data transfer	24
Description	24
Programming example in FBD	26
3.2 Programming with address transfer	27
Description	27
Programming example in ST	28
3.3 Program function keys and LEDs	29
3.4 Enable the communication between the controller and the DDM	29
4. Operation	30
4.1 Normal operation	30
4.2 Parameter setting mode	32
Activate parameter setting mode	32
Language selection	32
Set display time	33
Set flashing time	33
Set CAN identifier	33
Set bit rate	33
4.3 Presentation of the keys	34
5. Trouble shooting	35
6. Cleaning	34
7. Technical data	36
8. Declaration of conformity	37

1. Functions of the DDM 100

The data dialogue module DDM 100 with CAN interface is a display and input module for ifm control systems in CAN technology.

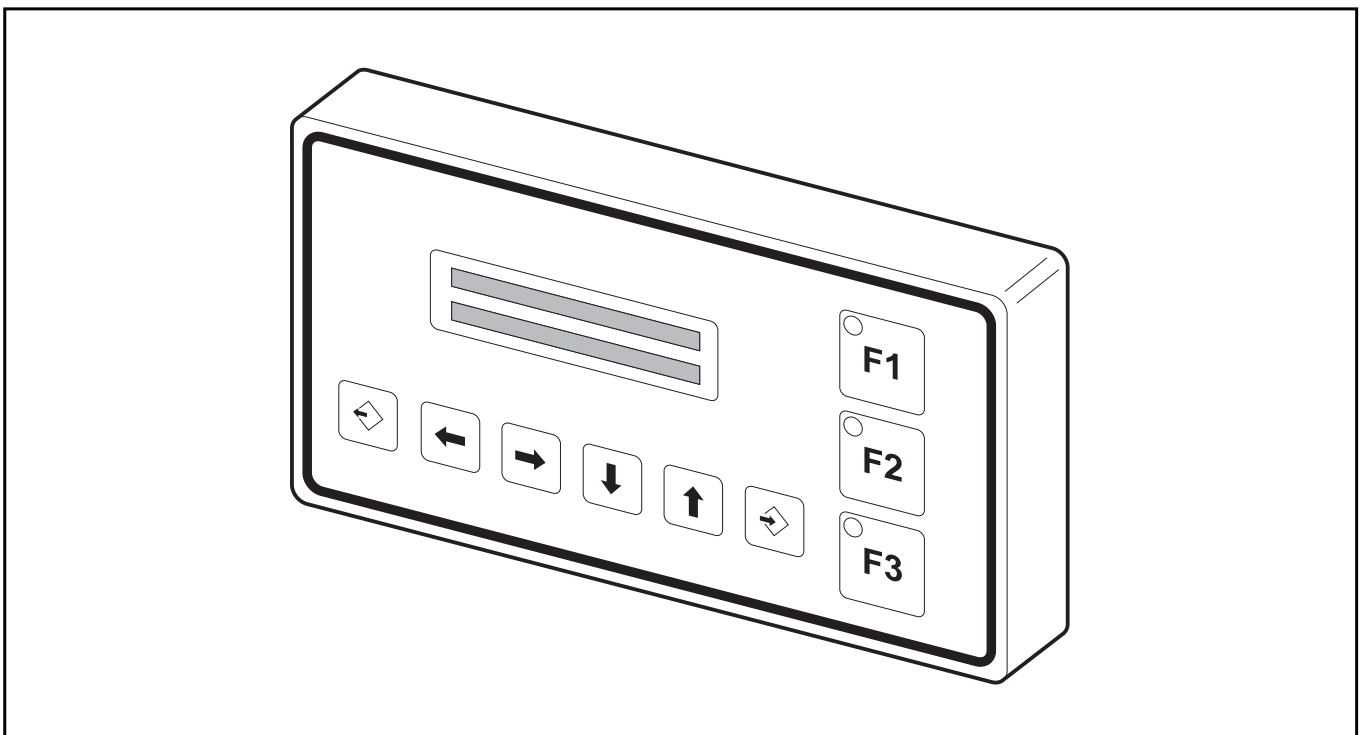
The DDM has the following features and functions:

Mounting

- Control cabinet or wall mounting
- Small compact housing with a height of only 32mm
- Mounting of the unit can be carried out before programming

Programming

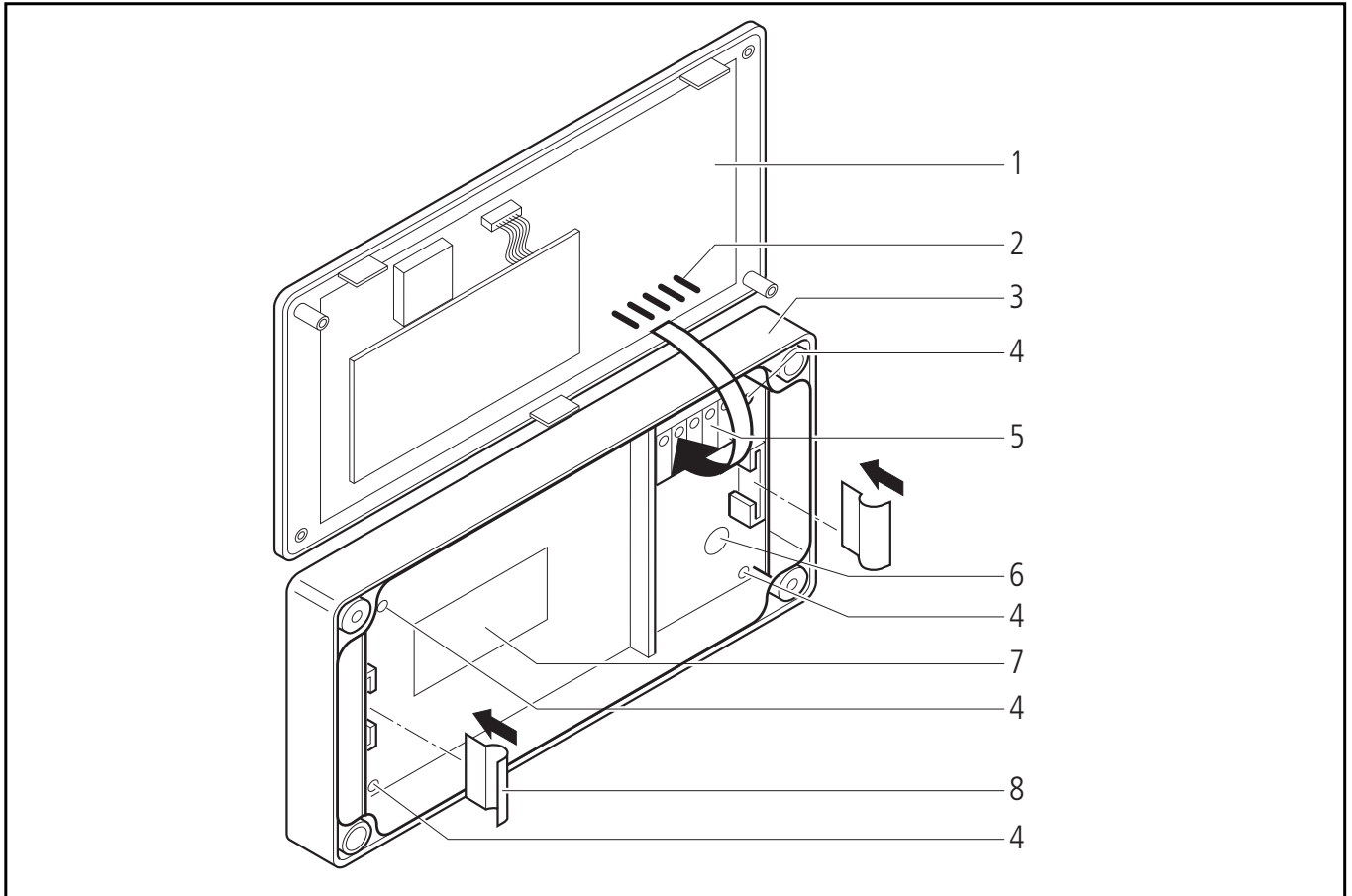
- Create and control the display of the text screens for the DDM within the controller program
- Unlimited number of programmed display text screens
- 16 text screens active at any one time with 16 characters per line (max. 2 lines) for the backlit text display
- Prioritising of up to 16 active text screens
- Display texts with integrated process data, if necessary
- Integrated process data can be edited
- Free assignment of the 3 function keys
- Free assignment of the 3 LEDs



2. Mounting



C-MOS components are damaged by electrostatic discharge. Before mounting the front panel, touch the grounded part to reduce the electrostatic charging.



2.1 Control cabinet mounting

1. By means of the supplied template prepare a cut-out and drill 4 holes for fixing (maximum thickness of the front panel = 2.5mm).
2. Insert the housing lower part (3) into the cut-out from the inside of the control cabinet.
3. Slip the supplied spring clamps (8) into the holders to the right and left in the housing lower part from the outside of the control cabinet and fasten the housing lower part in the cut-out.
4. Knock out available for PG 16 cable gland (6).
5. Insert the supplied PG16 cable gland.
6. Insert the cable.
7. Make the connections as per the label (7) in the housing lower part (3). The unit must be connected by qualified personnel.
8. Place the housing upper part (1) onto the housing lower part (3) in such a way that the contact pins (2) of the housing upper part engage with the terminals (5) in the housing lower part.
9. Screw the housing upper part with the housing lower part.

2.2 Wall mounting

See the mounting sketch on page 22.

1. Knock out available for PG 16 cable gland (6), or make a cable gland hole in the lower side wall within the connection chamber (see supplied template) and cut a suitable PG7 thread.
2. Insert the supplied PG16 cable gland, if necessary.
3. Insert the cable.
4. Fasten the housing lower part with four screws or similar elements (4) to the wall.
5. Make the connections as per the label (7) in the housing lower part (3). The unit must be connected by qualified personnel.
6. Place the housing upper part (1) onto the housing lower part (3) in such a way that the contact pins (2) of the housing upper part engage with the terminals (5) in the housing lower part.
7. Screw the housing upper part with the housing lower part.

3. Programming

To integrate the DDM 100 in an ifm controller you have to program the following in the user program of the controller:

- Enable the communication between the controller and the DDM by means of the system variable TEXT_MODE.
- Text screens to be displayed
- Display priority of the text screens
- Process data to be integrated into the texts, if necessary
- If necessary, enable the integrated process data for editing
- Assignment of the function keys
- Assignment of the LEDs

Write the user program of the controller in the ecolog 100^{plus}. You can use the programming languages to IEC 1131. For the description of how to program the controller please refer to the ecolog 100^{plus} manual.

There are two strategies to program the texts for the DDM:

- Data transfer (see chapter 3.1)
- Address transfer (see chapter 3.2, recommended for run time critical applications and for applications with many texts)

3.1 Programming with data transfer

Description

In the device-specific library of the controller you find the function block "TEXTD". Create an instance of the function block "TEXTD" for each display text. Assign the text data for the corresponding display text screen to each instance.

- ENABLE : BOOL; enable text
- PRIORITY : BYTE; display priority 1 ... 7, 7 = highest
- LINE1 : STRING(20); display text line 1, if necessary with wildcard "%" and formatting character for process data to be integrated
- LINE1_VALUE1 : INT1; from left to right the first integrated value in line 1
- LINE1_MIN1 : INT; the lower edit limit, value 1
- LINE1_MAX1 : INT; the upper edit limit, value 1
- LINE_EDIT1 : BOOL; edit enable for value 1
- LINE1_VALUE2 : INT; the second value
- LINE1_MIN2 : INT; the lower edit limit, value 2
- LINE1_MAX2 : INT, the upper edit limit, value 2

- LINE1_EDIT2 : BOOL; edit enable for value 2
- LINE2 : STRING(20); display text line 2, if necessary with wildcard "%" and formatting character for process data to be integrated
- LINE2_VALUE1 : INT; from left to right the first integrated value in line 2
- LINE2_MIN1 : INT; the lower edit limit, value 1
- LINE2_MAX1 : INT; the upper edit limit, value 1
- LINE2_EDIT1 : BOOL; edit enable for value 1
- LINE2_VALUE2 : INT; the second value
- LINE2_MIN2 : INT; the lower edit limit, value 2
- LINE2_MAX2 : INT; the upper edit limit, value 2
- LINE2_EDIT2 : BOOL; edit enable for value 2

Flashing indication of text lines is possible.

Flashing indication is activated by entering "\" as first character (backslash, ASCII 5Chex).

In each line of the display text maximum two values (process data / variable) can be integrated.

Allowed display formats for the integrated values:

%nd	=	SIGNED INT	=	-32767 ... 0 ... 32767
%nu	=	UNSIGNED INT	=	0 ... 65535
%nx	=	HEX	=	0 ... ffff
%nX	=	HEX	=	0 ... FFFF

n = 0 ... 5; option:

Number of the represented places of the process data, without n all places are represented (without leading zeros).

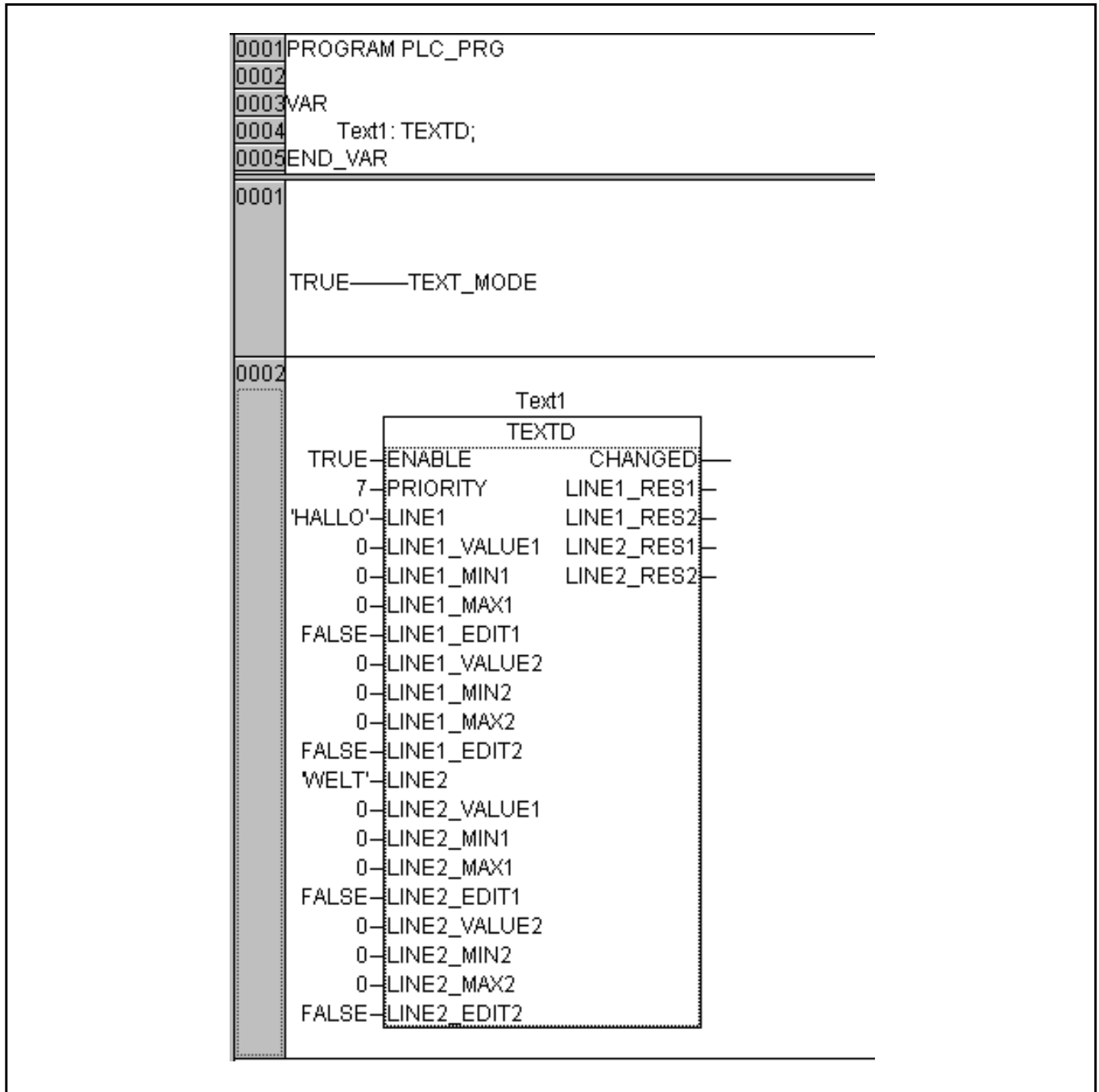
Each instance provides its outputs with the following information:

- CHANGED : BOOL = TRUE, if one of the values for editing was changed
- LINE1_RES1; the changed first value of line 1
- LINE1_RES2; the changed second value of line 1
- LINE2_RES1; the changed first value of line 2
- LINE2_RES2; the changed second value of line 2

All instances of the function block "TEXTD" must be called cyclically.

Programming example in FBD

The program extract shows an instance Text1 of the function block "TEXTD" with assigned text data in the programming language Function Block Diagram (FBD). The program displays the text "HALLO WELT".



3.2 Programming with address transfer

Description

In the device-specific library of the controller you find the data structure "text_struct". For each display text screen define a variable type "text_struct". Assign the text data to the variables type "text_struct" by initialising the components of the structure with the corresponding data.

- .enable : BOOL; text enable
- .priority : BYTE; display priority 1 ... 7, 7 = highest
- .line1 : STRING(20); display text line 1, if necessary with wildcard "%" and formatting character for process data to be integrated
- .line1_value1 : INT, from left to right the first integrated value in line 1
- .line1_min1 : INT; the lower edit limit, value 1
- .line1_max1 : INT; the upper edit limit, value 1
- .line1_edit1 : BOOL; edit enable for value 1
- .line1_value2 : INT; the second value
- .line1_min2 : INT; the lower edit limit, value 2
- .line1_max2 : INT; the upper edit limit, value 2
- .line1_edit2 : BOOL; edit enable for value 2
- .line2 : STRING(20); display text line 2, if necessary with wildcard "%" and formatting character for process data to be integrated
- .line2_value1 : INT; from left to right the first integrated value in line 2
- .line2_min1 : INT; the lower edit limit, value 1
- .line2_max1 : INT; the upper edit limit, value 1
- .line2_edit1 : BOOL; edit enable for value 1
- .line2_value2 : INT; the second value
- .line2_min2 : INT; the lower edit limit, value 2
- .line2_max2 : INT; the upper edit limit, value 2
- .line2_edit2 : BOOL; edit enable for value 2

The default of the components of the structure is 0 or FALSE.

In each line of the text display screen a maximum of two values (process data / variable) can be integrated.

Allowed display formats for the integrated values:

%nd	=	SIGNED INT	=	-32767 ... 0 ... 32767
%nu	=	UNSIGNED INT	=	0 ... 65535
%nx	=	HEX	=	0 ... ffff
%nX	=	HEX	=	0 ... FFFF

n = 0 ... 5; option:

Number of the represented places of the process data, without n all places are represented (without leading zeros).

In the device-specific library of the controller you find the function block "TEXTA". Create an instance of this function block. The instance sends the text data of a display text to the DDM 100. So the instance must be called cyclically with the addresses of all defined variables type "text_struct". To do so, the address of the text variable is determined with the function "ADR()". The address type "DINT" is converted with the function "DINT_TO_INT" into a variable type "INT" and transferred to an instance of TEXTA.

Programming example in ST

Note: Below you find a commented program extract with address transfer in the programming language Structured Text (ST).

The program displays the text "HALLO WELT".

```

0001 PROGRAM PLC_PRG
0002 VAR
0003
0004     Text1_dat : text_struct;          (* Definition von Text1_dat. Alle Komponenten *)
0005                                     (* von text_struct sind mit NULL / FALSE vorbelegt *)
0006                                     (* Define Text1_dat. All elements of text_struct *)
0007                                     (* have default setting NULL / FALSE *)
0008
0009     Text1_adr : DINT;                (* Definition einer Variablen zur Aufnahme der *)
0010                                     (* Adresse von Text1_dat *)
0011                                     (* Definition of variable for the address of Text1_dat *)
0012
0013     Text_Kom : TEXTA;                (* Instanz von TEXTA *)
0014                                     (* Instance of TEXTA *)
0015
0016 END_VAR
-----
0001 TEXT_MODE := TRUE;                (* Schnittstelle zum DDM freigeben *)
0002                                     (* Enable DDM interface *)
0003
0004 Text1_dat.enable := TRUE;           (* Text1 freigeben *)
0005                                     (* Enable Text1 *)
0006
0007 Text1_dat.priority := 7;            (* Anzeigepriorität festlegen *)
0008                                     (* Define priority *)
0009
0010 Text1_dat.line1 := 'Hello';         (* Anzeigetext der Zeile 1 *)
0011                                     (* Define text of line 1 *)
0012
0013 Text1_dat.line2 := 'World';        (* Anzeigetext der Zeile 2 *)
0014                                     (* Define text of line 2 *)
0015
0016
0017 Text1_adr := ADR(Text1_dat);        (* Adresse von Text1 bilden und in Text1_adr ablegen *)
0018                                     (* Create address of Text1_dat and store in Text1_adr *)
0019
0020 Text_Kom(A := Text1_adr);          (* Text_Kom mit der Adresse von Text1 aufrufen *)
0021                                     (* Call Text_Kom with the address of Text1 *)
0022
0023

```

3.3 Program function keys and LEDs

The 3 function keys (F1, F2, F3) and the 3 LEDs (F1, F2, F3) can be freely assigned. The assignment of the edit keys is fixed.

In the controller configuration you find the following variables which include the key and LED status:

Function keys	
Variable	Type
TEXT_KEYS	WORD
TEXT_KEY_F1	BOOL (TEXT_KEYS, Bit 0)
TEXT_KEY_F2	BOOL (TEXT_KEYS, Bit 1)
TEXT_KEY_F3	BOOL (TEXT_KEYS, Bit 2)

Edit keys	
Variable	Type
TEXT_KEYS	WORD
TEXT_KEY_ESC	BOOL (TEXT_KEYS, Bit 3)
TEXT_KEY_LEFT	BOOL (TEXT_KEYS, Bit 4)
TEXT_KEY_RIGHT	BOOL (TEXT_KEYS, Bit 5)
TEXT_KEY_DOWN	BOOL (TEXT_KEYS, Bit 6)
TEXT_KEY_UP	BOOL (TEXT_KEYS, Bit 7)
TEXT_KEY_ENTER	BOOL (TEXT_KEYS, Bit 8)

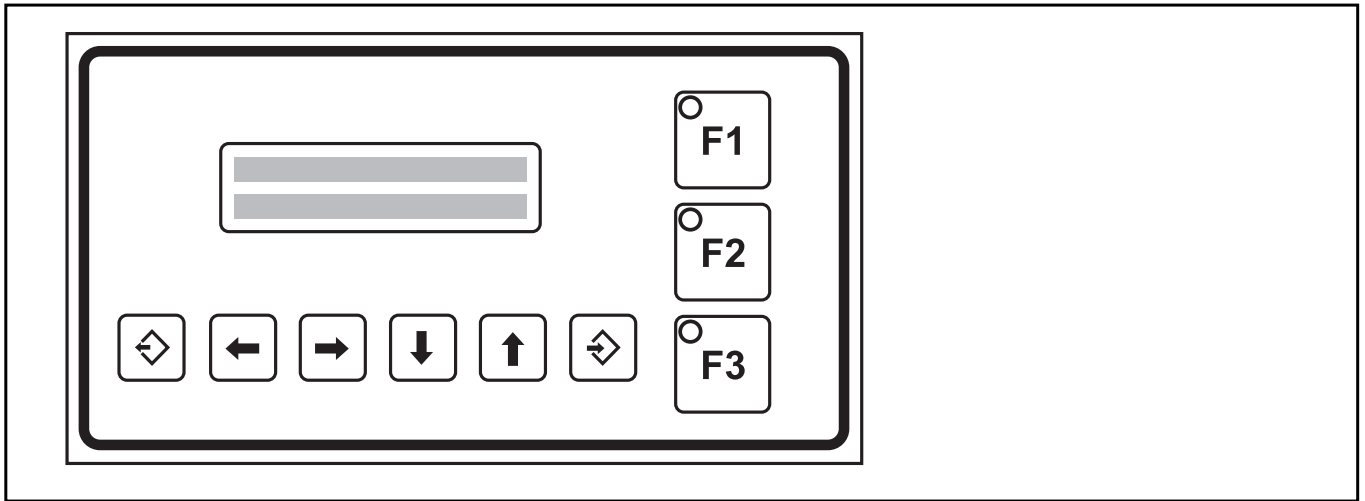
LEDs	
Variable	Type
TEXT_LEDS	WORD
TEXT_LED_F1	BOOL (TEXT_LEDS, Bit 0)
TEXT_LED_F2	BOOL (TEXT_LEDS, Bit 1)
TEXT_LED_F3	BOOL (TEXT_LEDS, Bit 2)

3.4 Enable the communication between the controller and the DDM

In the device-specific library of the controller the global variable TEXT_MODE : BOOL is defined for the communication between the controller and the DDM.

Enable the communication between the controller and the DDM:

- Set the system variable TEXT_MODE to TRUE in the controller program.



4. Operation

4.1 Normal operation

After switch-on of the controller the DDM displays

ecomat 100
CS 0014

The DDM is ready for operation.

Then the text with the highest display priority is indicated. If no text is active,

no messages

is indicated.



If there is no communication with the controller, for example because the controller is not switched on,

ERROR
communication

is displayed

As soon as the DDM100 and the controller communicate, the error message disappears and the text with the highest display priority is indicated.

Call another active text:

→ Scroll with  or 



The texts are displayed in the order of priority.



If no key is pressed, the selected text is displayed for the set display time (see the section "Set the display time" on page 32). Then the current text with the highest priority is displayed again.

Before the display time has elapsed, pressing of the  will display the highest priority screen.



Edit process data in a text:


→ Get the cursor displayed with  or 

→ Move the cursor to the requested position of the value to be changed with  or 

→ Decrement or increment the place above the cursor with  or 

When 9 is exceeded or in the case of 0 a carry is made to the next decade.

→ If  or  is pressed for more than 1s, decrementing or incrementing is performed automatically

→ Cancel editing with 

→ End edit with 
The cursor disappears from the display. Changed values, if any, are not saved.

4.2 Parameter setting mode

In the parameter setting mode the following unit settings can be made:

- language of the stored texts in English/German.
- display time of a text with low priority until the text with the highest priority is displayed again when no button is pressed. It is also the time until the edit mode is left when no button is pressed.
- flashing time (for a flashing indication of a text line).
- CAN identifier of the messages the controller transfers to the DDM 100. The communication from the controller to the DDM 100 comprises 2 CAN messages. The identifier of the first message is set.

The identifier of the second message is calculated as follows: ID1 message + 1. The identifiers for the communication from DDM 100 to the controller are transferred to the DDM100 by the controller.

Caution: The identifier of the CAN messages must be changed in the controller and in the DDM 100. Depending on the operating system of the controller this is not always possible. In this case the identifier to be set is 7F0 hex.

- bit rate of the CAN interface. You can select from a list of 8 stored bit rates. The bit rate of the DDM 100 must be the same as the bit rate of the controller.

Activate parameter setting mode

After application of the supply voltage the following text is displayed:

ecomat 100
CS0014


Press the  +  buttons simultaneously once while this text is displayed.


Language selection

When the  or  button is pressed, the following text is displayed:

PARAMETER
Language English / Sprache Deutsch

Set the desired language by pressing the  or  button.



Confirm setting by pressing the  button.

Cancel setting by pressing the  button.



Set display time

When the  or  button is pressed, the following text is displayed:

PARAMETER
display time xx s/ Meldezeit xx s

By pressing the  or  button start changing the value and position the cursor to the desired position.



By pressing the  or  button set the desired value.



Confirm setting by pressing the  button. Cancel setting by pressing the  button.



Set flashing time

When the  or  button is pressed, the following text is displayed:

PARAMETER
blink xxxx ms/ Blinker xxxx ms

By pressing the  or  button start changing the value and position the cursor to the desired position.


By pressing the  or  button set the desired value.

Confirm setting by pressing the  button. Cancel setting by pressing the  button.



Set CAN identifier

When the  or  button is pressed, the following text is displayed:

PARAMETER
Rx Ident xxx / Rx Ident xxx

By pressing the  or  button start changing the value and position the cursor to the desired position.



By pressing the  or  button set the desired value.

Confirm setting by pressing the  button. Cancel setting by pressing the  button.

Set bit rate

When the  or  button is pressed, the following text is displayed:

PARAMETER
BitRate xxx kBit/s / BitRate xxx kBit/s

By pressing the  or  button scroll the list of stored bit rates.

When the CAN identifier or the bit rate has been changed, press the x button for indication of the following text:

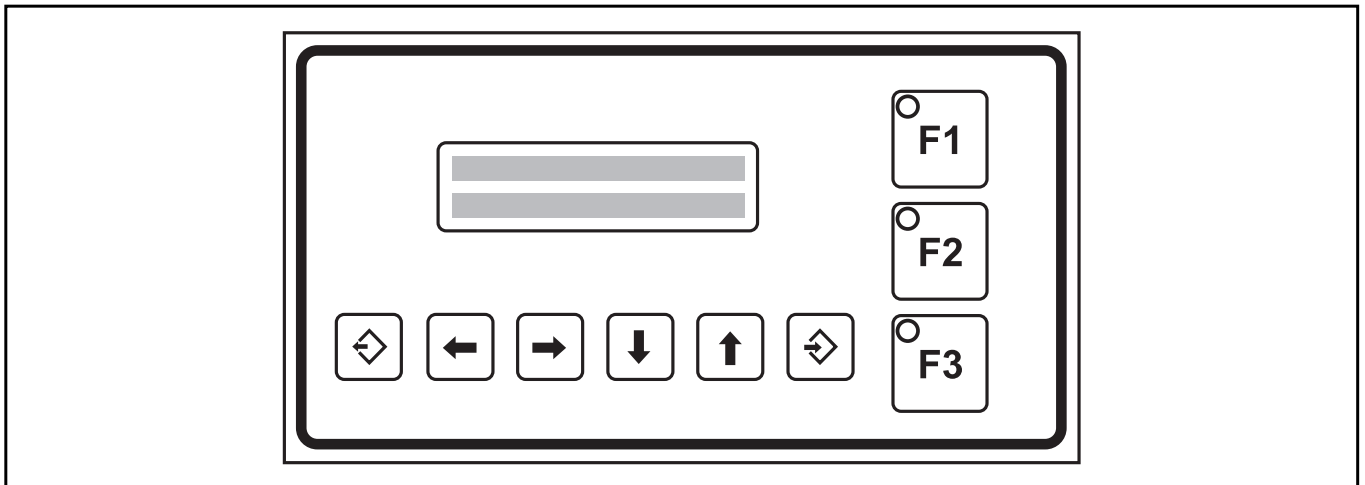
PARAMETER
load / übernehmen










The changed CAN parameters are activated by pressing the  button.

As after application of the supply voltage the unit then displays the following text:

ecomat 100
CS0014

4.3 Presentation of keys



	Function key F1 and LED, function according to the program
	Function key F2 and LED, function according to the program
	Function key F3 and LED, function according to the program
	Escape key, cancels editing or display text screen with the highest priority
	Edit key, moves the cursor to the left
	Edit key, moves the cursor to the right
	Edit key, decreases the value at the cursor position or display text screen with lower priority
	Edit key, increases the value at the cursor position or display text screen with higher priority
	Enter key, confirms the edited value

5. Trouble-shooting

Problem	Possible solutions
The DDM does not react	Check wiring and the supply
A function key does not react	Check the program code that has been written in the controller for this type of operation. Return the front panel to the manufacturer for inspection.
An Edit key does not react	Return the front panel to the manufacturer for inspection
The message "no messages" is indicated	Enable the text display screens in the controller program
The message "ERROR communication" is indicated	Check in the controller program whether the communication between the controller and DDM has been enabled. If necessary, set the variable TEXT_MODE to TRUE. Check wiring and the controller.

6. Cleaning



The membrane keys can be scratched by improper cleaning thus becoming unusable.

Clean the membrane keys of the DDM with a cleaning agent which does not attack polyester. Use a soft cloth.

7. Technical data

Application

Display and input module for controllers

Display	backlit LCD display 2 lines with 16 characters each, freely programmable
Operating voltage [V]	10 ... 30 DC (SELV)
Current consumption [mA]	100 (In case of a fault the current consumption must be limited to max. 1 A by an appropriate fuse)
Protection rating	IP64 / III
Operating temperature [°C]	0 ... +50
Housing dimensions [mm]	190 x 90 (length x width) height for wall mounting = 32mm height for control cabinet mounting = 7mm installation depth for control cabinet mounting = 25mm maximum thickness of the front panel = 2.5mm
Operating keys	a total of 9, of which, 3 function keys, can be freely assigned, 6 keys with defined function
Function keys with LEDs	freely programmable via the application program
LEDs	3 x yellow, freely programmable
Interface	CAN: specification 2.0 A/B
Baud rate:	125 kBaud, High speed to ISO/DIS 11898
Identifiers:	7F0 HEX ... 7F1 HEX (DDM receives text list/text data) 7F8 HEX ... 7FA HEX (DDM sends text call, edited process data, key status)
Connections	5 WAGO cage-clamp terminals
Wiring	1: CAN_H 2: CAN_L 3: GND 4: GND 5: +24 V

8. Declaration of conformity

The CE mark is granted on the basis of the EMC guideline EMC 89/336/EEC which has been considered in the standards EN 50081-1 and EN 61000-6-2 and also on the basis of the low voltage guideline considered in the standards EN 61010 and EN 61010 / A1:1995.