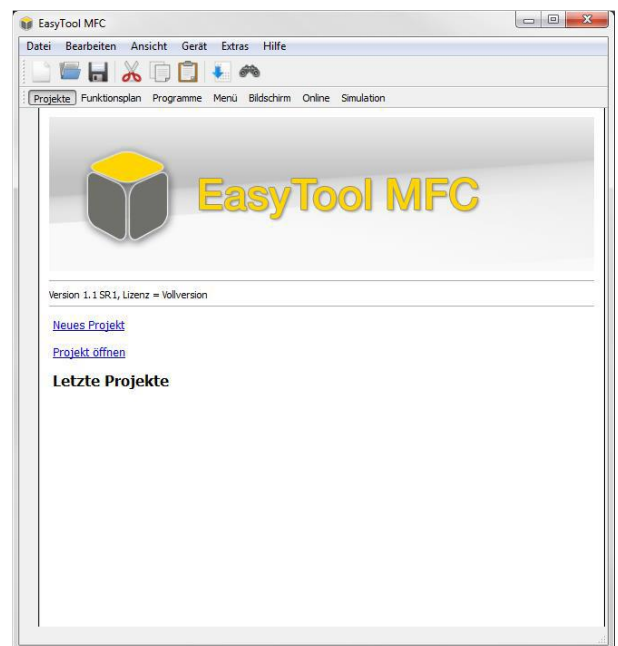


# HE 5697 MFC Multifunktionskontrolller

---



---

## Beschreibungen zur Software EasyTool MFC Variablen, Block- und Gerätefunktionen

## Inhaltsverzeichnis

<b>1</b>	<b>Blockfunktionen</b> .....	<b>4</b>
1.1	<b>STRUCT (Strukturblock)</b> .....	<b>4</b>
1.2	<b>AINPB (Analoger Blockeingang)</b> .....	<b>4</b>
1.2.1	Funktionsblockbeschreibung .....	4
1.3	<b>DINPB (Digitaler Blockeingang)</b> .....	<b>4</b>
1.3.1	Funktionsblockbeschreibung .....	4
1.4	<b>AOUTB (Analoger Blockausgang)</b> .....	<b>5</b>
1.4.1	Funktionsblockbeschreibung .....	5
1.5	<b>DOUTB (Digitaler Blockausgang)</b> .....	<b>5</b>
1.5.1	Funktionsblockbeschreibung .....	5
1.6	<b>COMMENT (Kommentar)</b> .....	<b>5</b>
<b>2</b>	<b>Gerätefunktionen</b> .....	<b>6</b>
2.1	<b>LED (LED-Anzeige)</b> .....	<b>6</b>
2.2	<b>KEY (Tasten)</b> .....	<b>6</b>
2.3	<b>CLOCK (Uhrzeit)</b> .....	<b>6</b>
2.4	<b>RELAIS (Relais)</b> .....	<b>7</b>
<b>3</b>	<b>HMI-Variablen</b> .....	<b>8</b>
3.1	<b>AINPH (Analoger HMI-Eingang)</b> .....	<b>8</b>
3.1.1	Funktionsblockbeschreibung .....	8
3.2	<b>DINPH (Digitaler HMI-Eingang)</b> .....	<b>8</b>
3.2.1	Funktionsblockbeschreibung .....	8
3.3	<b>AOUTH (Analoger HMI-Ausgang)</b> .....	<b>9</b>
3.3.1	Funktionsblockbeschreibung .....	9
3.3.2	Eingänge.....	9
3.3.3	Ausgänge.....	9
3.3.4	Parameter .....	10
3.3.5	Logik für Zweipunkt Bedienung .....	10
3.4	<b>DOUTH (Digitaler HMI-Ausgang)</b> .....	<b>10</b>
3.4.1	Funktionsblockbeschreibung .....	10
3.4.2	Eingänge.....	11
3.4.3	Ausgänge.....	11
3.4.4	Parameter .....	11
3.4.5	Logik für Zweipunkt Bedienung .....	12

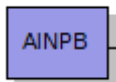
<b>4</b>	<b>Variablen.....</b>	<b>12</b>
<b>4.1</b>	<b>AOUTC (Analoge Variable (Quelle)).....</b>	<b>12</b>
4.1.1	Funktionsblockbeschreibung.....	12
<b>4.2</b>	<b>DOUTC (Digitale Variable (Quelle)).....</b>	<b>13</b>
4.2.1	Funktionsblockbeschreibung.....	13
<b>4.3</b>	<b>AINPC (Analoge Variable (Senke)).....</b>	<b>13</b>
4.3.1	Funktionsblockbeschreibung.....	13
<b>4.4</b>	<b>DINPC (Digitale Variable (Senke)).....</b>	<b>13</b>
4.4.1	Funktionsblockbeschreibung.....	13
<b>5</b>	<b>Notizen.....</b>	<b>14</b>

## 1 Blockfunktionen

### 1.1 STRUCT (Strukturblock)

Mit der Erstellung eines Strukturblockes im Funktionsplan, lassen sich die Abläufe eines Programmes übersichtlicher gestalten oder auch neue Funktionsbausteine generieren. Die analogen und digitalen Blockeingänge- sowie Ausgänge werden für die Integration der neu geschaffenen Struktur in eine Bestehende benötigt. Einbindung der Unterstrukturen in das Hauptprogramm.

### 1.2 AINPB (Analoger Blockeingang)



#### 1.2.1 Funktionsblockbeschreibung

Durch das Einbinden dieses Funktionsblocks in eine Struktur wird ein analoger Anschluss der Struktur (Signalquelle) hinzugefügt. Hierüber können dann Block- und Struktur-Ausgänge angebunden werden.

Das am Eingang  $x_1$  anliegende Signal wird am Struktur-Ausgang ausgegeben.

### 1.3 DINPB (Digitaler Blockeingang)

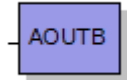


#### 1.3.1 Funktionsblockbeschreibung

Durch das Einbinden dieses Funktionsblocks in eine Struktur wird ein digitaler Anschluss der Struktur (Signalquelle) hinzugefügt. Hierüber können dann Block- und Struktur-Ausgänge angebunden werden.

Das am Eingang  $z_1$  anliegende Signal wird am Struktur-Ausgang ausgegeben.

## 1.4 AOUTB (Analoger Blockausgang)

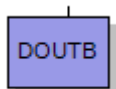


### 1.4.1 Funktionsblockbeschreibung

Analoger Blockausgang einer Struktur bzw. Anwenderblocks. Durch das Einbinden des Ausgangs in eine Struktur, wird ein analoger Anschluss der Struktur (Signalsenke) hinzugefügt. Über diesen können Block- und Struktur-Eingänge angebunden werden.

Innerhalb der Struktur wird der Eingang  $x_1$  mit einem analogen Ausgang verdrahtet, welches am Struktur-Ausgang verfügbar sein soll.

## 1.5 DOUTB (Digitaler Blockausgang)

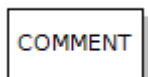


### 1.5.1 Funktionsblockbeschreibung

Durch das Einbinden des Ausgangs in eine Struktur, wird ein digitaler Anschluss der Struktur (Signalsenke) hinzugefügt. Über diesen können Block- und Struktur-Eingänge angebunden werden.

Innerhalb der Struktur wird der Eingang  $d_1$  mit einem digitalen Ausgang verdrahtet, welcher am Struktur-Ausgang verfügbar sein soll.

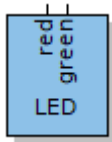
## 1.6 COMMENT (Kommentar)



Mit diesem Block können beschreibende Texte an beliebigen Stellen im Funktionsplan platziert werden. Die Texte können im Parameterdialog des Blocks bearbeitet werden. Die Größe des Kommentarblocks passt sich automatisch der Textmenge an.

## 2 Gerätefunktionen

### 2.1 LED (LED-Anzeige)



Dieser Funktionsblock ermöglicht das Setzen der roten und grünen Leuchtdiode, die sich rechts oberhalb des Bildschirms befinden.

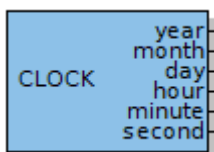
### 2.2 KEY (Tasten)



Der Funktionsblock dient dem Einlesen der Tasten unter dem Bildschirm. Die Ausgänge z1 bis z4 werden logisch 1, wenn die zugehörige Taste gedrückt ist (Zustand).

Ausgang	Taste
z1	Taste "Schraubenschlüssel" (links)
z2	Taste "Auf"
z3	Taste "Ab"
z4	Taste "Übernehmen" (rechts)

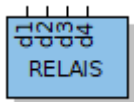
### 2.3 CLOCK (Uhrzeit)



Dieser Funktionsblock liefert die aktuelle Uhrzeit und das aktuelle Datum in Einzelwerten.

Ausgang	Beschreibung
<i>year</i>	Datum: Jahr (2014..)
<i>month</i>	Datum: Monat (1..12)
<i>day</i>	Datum: Tag (1..31)
<i>hour</i>	Uhrzeit: Stunde (0..23)
<i>minute</i>	Uhrzeit: Minute (0..59)
<i>second</i>	Uhrzeit: Sekunde (0..59)

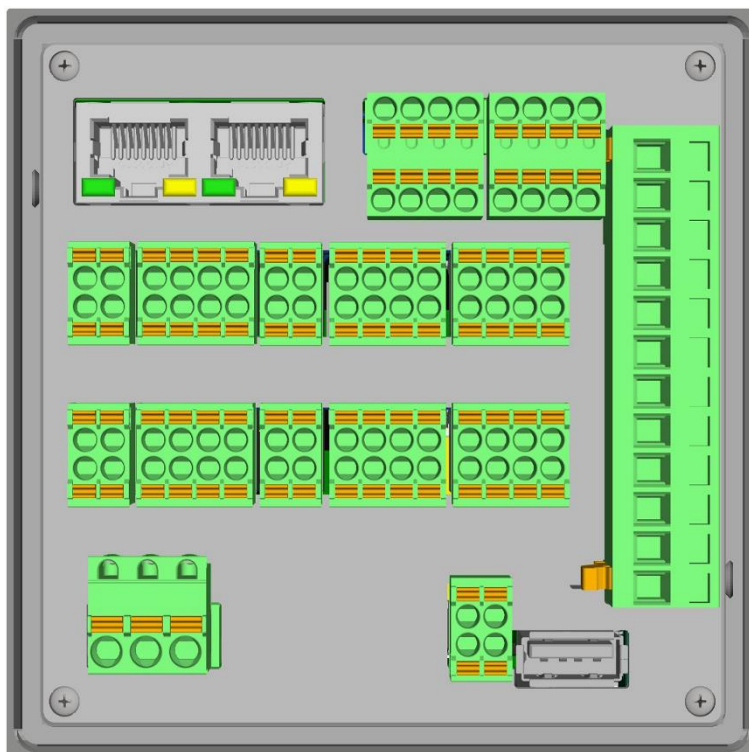
## 2.4 RELAIS (Relais)



Dieser Funktionsblock steuert die 4 Relaisausgänge.

### Anschluss

Geräterückseite: Relais

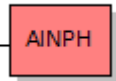


Belegung:

Nr.	Signal
1	Relais 4 - Arbeitskontakt
2	Relais 4 - Schließerkontakt
3	Relais 4 - Öffnerkontakt
4	Relais 3 - Arbeitskontakt
5	Relais 3 - Schließerkontakt
6	Relais 3 - Öffnerkontakt
7	Relais 2 - Arbeitskontakt
8	Relais 2 - Schließerkontakt
9	Relais 2 - Öffnerkontakt
10	Relais 1 - Arbeitskontakt
11	Relais 1 - Schließerkontakt
12	Relais 1 - Öffnerkontakt

## 3 HMI-Variablen

### 3.1 AINPH (Analoger HMI-Eingang)

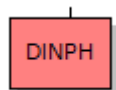


#### 3.1.1 Funktionsblockbeschreibung

Mit dem Anschluss des Blockeingangs wird eine analoge Variable für das HMI definiert. Ein Klick mit der rechten Maustaste öffnet das Dialogfeld („Umbenennen“ und „Querverweise“). Mit "Umbenennen" kann den Variablen eine individuelle Bezeichnung zugewiesen werden. Das logische Signal steht nun zur Verwendung im Bildschirm-Editor als Quelle bereit.

Die Variablen werden fortlaufend als  $hmi_1 \dots hmi_n$ , chronologisch ihrer Verwendung im Engineering benannt. Dabei ist es unerheblich, ob es sich um eine digitale oder analoge Variable handelt.

### 3.2 DINPH (Digitaler HMI-Eingang)



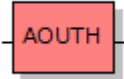
#### 3.2.1 Funktionsblockbeschreibung

Mit dem Anschluss des Blockeingangs  $d_1$  wird eine digitale Variable für das HMI definiert. Ein Klick mit der rechten Maustaste öffnet das Dialogfeld ("Umbenennen", „Querverweis“, und „Parameter“). Mit "Umbenennen" kann den Variablen eine individuelle Bezeichnung zugewiesen werden. Das logische Signal steht nun zur Verwendung im Bildschirm-Editor als Quelle bereit. Wählt man im Dialogfeld den Punkt „Parameter“ kann das Übertragungsverhalten des digitalen Eingangs bestimmt werden, direkt oder invertiert.

Die Variablen werden fortlaufend als  $hmi_1 \dots hmi_n$ , chronologisch ihrer Verwendung im Engineering benannt. Dabei ist es unerheblich, ob es sich um eine digitale oder analoge Variable handelt.



## 3.3 AOUTH (Analoger HMI-Ausgang)



### 3.3.1 Funktionsblockbeschreibung

Der Funktionsblock stellt eine beliebige, zuvor definierte analoge HMI-Variable im Engineering zur Verfügung.

Per Doppelklick wird die Senke mit einer HMI-Variablen verknüpft, die anschließend am Blockausgang  $y_1$  bereitsteht.

Der Ausgabewert kann ebenfalls mit dem analogen Eingang  $x$  verändert werden. Damit werden Bedienschritte des Anwenders, die aus dem Engineering heraus erfolgen, auch im HMI aktualisiert. Es lässt sich somit eine „Zweipunkt Bedienung“ realisieren. (Siehe unten: Kap. „Logik für Zweipunkt Bedienung“)

Über den Blockparameter *store* lässt sich der Ausgabewert  $y_1$  dauerhaft speichern. Der letzte aktive Wert steht bei Netzwiederkehr im Engineering wieder zur Verfügung.

### 3.3.2 Eingänge

Analoge Eingänge	
x	Analoge Eingangsgröße: Vorgabewert des Anwenders (aus dem Engineering heraus erzeugt)

### 3.3.3 Ausgänge

Analoge Ausgänge	
y	Aktueller / wirksamer Wert

### 3.3.4 Parameter

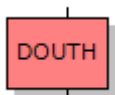
Parameter	
value	<p>Initialisierungswert: Nach dem Einschalten der Versorgungsspannung wird dieser Wert am Ausgang wirksam gemacht.</p>
store	<p>Dauerhaft speichern: 0: Änderungen werden nicht gespeichert 1: Änderungen werden gespeichert. Beim nächsten Einschalten der Versorgungsspannung wird der letzte aktive Wert wieder wirksam.</p>

### 3.3.5 Logik für Zweipunkt Bedienung

Der am Eingang anliegende Wert wird vorrangig behandelt. Hat der Eingang einen anderen Wert als vom HMI vorgegeben, wird dieser übernommen und automatisch in das HMI übertragen.

Wird der Wert vom HMI aktiv verändert, wird dieser sofort am Ausgang wirksam. Die Verarbeitung des am Eingang anliegenden Wertes wird für eine Sekunde gesperrt. Soll eine vom Eingang abweichende Anforderung des HMI dauerhaft wirksam bleiben, muss der analoge Eingang durch eine Logik im Engineering, während der Sperre von einer Sekunde, entsprechend nachgeführt werden.

## 3.4 DOUTH (Digitaler HMI-Ausgang)



### 3.4.1 Funktionsblockbeschreibung

Der Funktionsblock stellt eine beliebige, zuvor definierte digitale HMI-Variable im Engineering zur Verfügung.

Per Doppelklick wird die Senke mit einer HMI-Variablen verknüpft, die anschließend am

Blockausgang  $z_1$  bereitsteht.

Der Ausgabewert kann ebenfalls mit dem analogen Eingang  $x$  verändert werden. Damit werden Bedienschritte des Anwenders, die aus dem Engineering heraus erfolgen auch im HMI aktualisiert. Es lässt sich somit eine „Zweipunkt Bedienung“ realisieren. (Siehe unten: Kap. „Logik für Zweipunkt Bedienung“)

Über den Blockparameter *store* lässt sich der Ausgabewert  $z_1$  dauerhaft speichern. Der letzte aktive Wert steht bei Netzwiederkehr im Engineering wieder zur Verfügung.

### 3.4.2 Eingänge

Analoge Eingänge	
$x$	Analoge Eingangsgröße: Vorgabewert des Anwenders (aus dem Engineering heraus erzeugt)

### 3.4.3 Ausgänge

Analoge Ausgänge	
$y$	Aktueller / wirksamer Wert

### 3.4.4 Parameter

Parameter	
value	Initialisierungswert: Nach dem Einschalten der Versorgungsspannung wird dieser Wert am Ausgang wirksam gemacht.

inv	Übertragungsverhalten: 0: Direkt 1: Invers
store	Dauerhaft speichern: 0: Änderungen werden nicht gespeichert 1: Änderungen werden gespeichert. Beim nächsten Einschalten der Versorgungsspannung wird der letzte aktive Wert wieder wirksam.

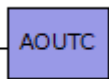
### 3.4.5 Logik für Zweipunkt Bedienung

Der am Eingang anliegende Wert wird vorrangig behandelt. Hat der Eingang einen anderen Wert als vom HMI vorgegeben, wird dieser übernommen und automatisch in das HMI übertragen.

Wird der Wert vom HMI aktiv verändert, wird dieser sofort am Ausgang wirksam. Die Verarbeitung des am Eingang anliegenden Wertes wird für eine Sekunde gesperrt. Soll eine vom Eingang abweichende Anforderung des HMI dauerhaft wirksam bleiben, muss der analoge Eingang durch eine Logik im Engineering während, der Sperre von einer Sekunde, entsprechend nachgeführt werden.

## 4 Variablen

### 4.1 AOUTC (Analoge Variable (Quelle))

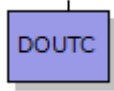


#### 4.1.1 Funktionsblockbeschreibung

Mit Anschluss des Blockeingangs  $x_1$  wird eine analoge Variable im Engineering definiert. Ein Klick mit der rechten Maustaste öffnet das Dialogfeld ("Umbenennen" und „Querverweis“). Mit „Umbenennen“ kann den Variablen eine individuelle Bezeichnung zugewiesen werden.

Die Variablen werden fortlaufend als  $var_1 \dots var_n$ , chronologisch ihrer Verwendung im Engineering benannt. Dabei ist es unerheblich, ob es sich um eine digitale oder analoge Variable handelt.

## 4.2 DOUTC (Digitale Variable (Quelle))

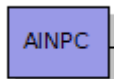


### 4.2.1 Funktionsblockbeschreibung

Mit dem Anschluss des Blockeingangs  $d_1$  wird eine digitale Variable im Engineering definiert. Ein Klick mit der rechten Maustaste öffnet das Dialogfeld ("Umbenennen" und "Querverweis"). Mit „Umbenennen“ kann den Variablen eine individuelle Bezeichnung zugewiesen werden.

Die Variablen werden fortlaufend als  $var_1 \dots var_n$ , chronologisch ihrer Verwendung im Engineering benannt. Dabei ist es unerheblich, ob es sich um eine digitale oder analoge Variable handelt.

## 4.3 AINPC (Analoge Variable (Senke))



### 4.3.1 Funktionsblockbeschreibung

Der Funktionsblock stellt eine beliebige, zuvor definierte analoge Variable im Engineering zur Verfügung.

Per Doppelklick wird die Senke mit einer Variablen verknüpft, die anschließend am Blockausgang bereitsteht.

## 4.4 DINPC (Digitale Variable (Senke))



### 4.4.1 Funktionsblockbeschreibung

Der Funktionsblock stellt eine beliebige, zuvor definierte digitale Variable im Engineering zur Verfügung.

Per Doppelklick wird die Senke mit einer Variablen verknüpft, die anschließend am Blockausgang bereitsteht.

## 5 Notizen