

6.4 AF101

6.4.1 Technische Daten



Bezeichnung	AF101
Allgemeines	
Bestellnummer	7AF101.7
Kurzbeschreibung	2003 Adaptermodul 4 Steckplätze für Anpassungsmodule
C-UL-US gelistet	JA
B&R ID-Code	\$C0
Anzahl	
CP430, EX270	2
CP470, CP770 CP474, CP476, CP774 EX470, EX770 EX477, EX777	4
Adapter-Interface	4 Steckplätze (Slots) für Anpassungsmodule
Potentialtrennung	NEIN
Status-LEDs	für jeden Kanal
Leistungsaufnahme	max. 0,3 W
Mechanische Eigenschaften	
Maße	B&R 2003 einfachbreit



Adaptermodule immer direkt rechts neben dem Controller betreiben!

Adaptermodule dürfen nur auf den ersten vier Moduladressen des Systems B&R 2003 betrieben werden.

6.4.2 Status-LEDs

Die grünen Status-LEDs des Adaptermoduls zeigen pro Slot folgende Betriebszustände an:

LED-Status	Bedeutung
Dunkel	Anpassungsmodul defekt oder nicht gesteckt
Langsam blinkend	Kommunikationsfehler mit Anpassungsmodul
Schnell blinkend	Anpassungsmodul neu oder gegen anderen Modultyp getauscht
Leuchtet	Anpassungsmodul ist betriebsbereit

6.4.3 Einschubstreifen

In die Modulfront kann von oben der im Lieferumfang enthaltene Einschubstreifen geschoben werden. Auf diesem können die Anpassungsmodule beschriftet werden.

6.4.4 Ansprechen der Anpassungsmodule

In jeder Modulbeschreibung ist ein Abschnitt "Variablen Deklaration" enthalten. In diesem Abschnitt wird auf die Variablen Deklaration über das PG2000 eingegangen.

In der Spalte VD-Kanal ist die Kanalnummer angegeben, mit der Daten- bzw. Konfigurationswörter angesprochen werden. Beim Zugriff auf Konfigurationswörter über I/O-Funktionsblöcke muß in Abhängigkeit vom Slot, auf dem das Anpassungsmodul auf der AF101 gesteckt ist, zum angegebenen VD-Kanal ein Offset dazugezählt werden.

Datenwort

Wenn der Datenzugriff auf ein Datenwort erfolgt, muß zum angegebenen VD-Kanal kein Offset dazugezählt werden, da der Slot in der Variablen Deklaration angegeben wird.

Konfigurationswort

Wenn der Datenzugriff über I/O-Funktionsblöcke auf ein Konfigurationswort erfolgt, muß zum angegebenen VD-Kanal in Abhängigkeit vom Slot ein Offset dazugezählt werden.

Der interne Offset, über den das gewünschte Konfigurationswort angesprochen wird, wird nach folgender Formel berechnet:

$$\text{interner Offset} = \text{VD-Kanal} + (\text{sl} - 1) \times 32$$

interner Offset Interner Offset über den das gewünschte Konfigurationswort angesprochen wird
VD-Kanal der in der Variablen Deklarationsbeschreibung angegebene Kanal
sl Slotnummer (1 - 4) des Anpassungsmoduls am AF101

Bitte beachten Sie auch die Beispiele im Kapitel 4 "Moduladressierung".

6.4.5 Befehle - AF101

Controller

Die Befehle sind für folgende Controller/Zugriffe verfügbar:

Controller / Zugriff	Befehle verfügbar
Zentraleinheit RPS 2003	JA
Remote Slaves	NEIN
CAN Slaves	NEIN
Zugriff über CAN-Identifizier	NEIN

Ausführen der Befehle

Zum Ausführen eines Befehls auf der AF101 werden I/O-Funktionsblöcke verwendet. Es muß ein Schaufel-auftrag zum Auslesen des Konfigurationswortes 14 (VD-Kanal 28) eines Anpassungsmoduls generiert werden.

Die Variable, in der die Befehlsantwort abgelegt wird, ist mit dem Befehlswort zu initialisieren. Nach erfolgreicher Ausführung des Schaufelauftrags wird die Befehlsantwort in dieser Variable gespeichert.

Die AF101-Befehle *Automatikmodus aus-/einschalten* sind erst ab der AF101 SW-Rev. 02.00 verfügbar. Die SW-Rev. des AF101 Moduls ist aus den letzten beiden Stellen der Rev. am Seriennummernaufkleber zu entnehmen (Rev. xx.11 -> SW-Rev. 01.10, Rev. xx.20 -> SW-Rev. 02.00).

1) B&R ID-Code des Anpassungsmoduls lesen

Zum Auslesen des B&R ID-Codes ist die Variable mit 0 zu initialisieren. Je nach Slot den Kanaloffset nicht vergessen!

Befehlswort	\$0000	
Befehlsantwort	\$xxyy	xx.....B&R ID-Code yy.....reserviert

2) B&R ID-Code des AF101 Moduls lesen

Befehlswort	\$8000	
Befehlsantwort	\$C000	B&R ID-Code AF101

3) Automatikmodus

Ab AF101 SW-Rev. 02.00 werden die Datenwörter standardmäßig auf ein automatisch gebildetes I/O-Abbild gelegt (Automatikmodus). Lesebefehle werden durch Zugriff auf dieses I/O-Abbild ausgeführt.

Für Sonderbetriebsarten von Anpassungsmodulen ist manchmal das Ausschalten der I/O-Abbilderstellung erforderlich (z. B. wird bei der AI261 dadurch sichergestellt, daß der normierte Wert oder der kalibrierte Rohwert pro Datenausgabezyklus nur einmal gelesen wird).

Wenn die I/O-Abbilderstellung ausgeschaltet ist, werden die Befehle für die Bearbeitung der Datenwörter zu den Anpassungsmodulen durchgeschleift.

Bei AF101 Modulen mit einer SW-Rev. <02.00 ist der Automatikmodus noch nicht implementiert. Diese Module verhalten sich daher so, wie sich Module mit einer SW-Rev. ≥02.00 bei ausgeschaltetem Automatikmodus verhalten.

3.1) Automatikmodus ausschalten

Der Befehl funktioniert nur bei AF101 Modulen mit einer SW-Rev. ≥02.00 (AF101 Module mit einer SW-Rev. <02.00 verhalten sich so, wie sich Module mit einer SW-Rev. ≥02.00 bei ausgeschaltetem Automatikmodus verhalten).

Befehlswort	\$8400	
Befehlsantwort	\$F400	Befehl ausgeführt - AF101 SW-Rev. ≥02.00
	\$C000	AF101 SW-Rev. <02.00

3.2) Automatikmodus einschalten

Der Befehl funktioniert nur bei AF101 Modulen mit einer SW-Rev. ≥02.00. Bei AF101 Modulen mit einer SW-Rev. <02.00 ist der Automatikmodus nicht verfügbar.

Befehlswort	\$8500	
Befehlsantwort	\$F500	Befehl ausgeführt - AF101 SW-Rev. ≥02.00
	\$C000	AF101 SW-Rev. <02.00

6.4.6 Programmbeispiel

In diesem Programmbeispiel werden im INIT-UP die Schaufelaufträge für folgende Befehle definiert und ausgeführt:

- B&R ID-Code des Anpassungsmoduls lesen
- B&R ID-Code des AF101 Moduls lesen
- Automatikmodus ausschalten

Taskübersicht

Das Anwenderprogramm ist in folgende Tasks unterteilt:



I/O-Library

Bevor Sie die Funktionsblöcke zur Bedienung der Konfigurationswörter nutzen können, müssen Sie die I/O-Library (ab V 01.60) in die Projekt-Datenbank importieren. Dazu steht Ihnen die Funktion *Import Library* aus dem Pull-Down-Menü *Datei* zur Verfügung. Die I/O-Library muß als Systemmodul in die GDM eingetragen werden.

AF101

Dieser Task bearbeitet die Anpassungsmodule auf dem Adaptermodul AF101. Im INIT-UP werden die Schaufelaufträge zum Absetzen von Befehlen an das AF101 Modul definiert.

INIT-UP

```
Projekt: af_auto                                Datei: InitUP : af101

Err_Ptest = 0
AF101_slot = 1
AT664_slot = 1

#####
;## Multipler physikalischer Schaufelauftrag zum Lesen des      ##
;## Parameterwortes 14 der AT664 auf AF101, Slot 1             ##
;#####
IO_struct.io_type=5                ;2003 IO
IO_struct.master_no=1              ;Master immer 1
IO_struct.slave_no=0               ;Slavenummer immer 0
IO_struct.module_adr=AF101_slot    ;Modulsteckplatz
IO_struct.intern_off= 32*(AT664_slot -1)+ 14 *2 ;Registernr. 14 auf ANP = AT664_slot
IO_struct.mode=%00100000           ;Lesend, Normal
IO_struct.data_len=2               ;2*Byte
IO_struct.reserve=0                ;Not used
IO_struct.data_adr=adr(AT_reg14)   ;Adresse des Datenbereichs,in dem das
                                   ;Parameterwort 14 abgebildet ist
AT_reg14 = $0000                    ;0 nötig für lesen des Modultyps

;Schaufelauftrag #0 deklarieren
IO_mphydef(1,adr(IO_struct),adr(af_sc_buf),0,status_mpl,af_ident)

if status_mpl=0 then
  loop          ;Schaufelauftrag zyklisch ausführen...
    IO_data(1,af_ident,status_dat)
    exitif status_dat <> 5559 ;...bis Schaufeln nicht mehr aktiv
  endloop
endif

if status_dat=0 then
  AT664_typ = AT_reg14
endif

#####
;## Multipler physikalischer Schaufelauftrag zum Lesen des      ##
;## Parameterwortes 14 (=Kennung) des AF101-Moduls             ##
;#####
IO_struct.io_type=5                ;2003 IO
IO_struct.master_no=1              ;Master immer 1
IO_struct.slave_no=0               ;Slavenummer immer 0
IO_struct.module_adr=AF101_slot    ;Modulsteckplatz
IO_struct.intern_off= 14 *2        ;Registernr. 14 auf AF101
IO_struct.mode=%00100000           ;Lesend, Normal
IO_struct.data_len=2               ;2*Byte
IO_struct.reserve=0                ;Not used
IO_struct.data_adr=adr(AF_reg14)   ;Adresse des Datenbereichs,in dem das
                                   ;Parameterwort 14 abgebildet ist
AF_reg14 = $8000                    ;Lesen der Kennung der AF101

IO_mphydef(1,adr(IO_struct),adr(af_sc_buf),0,status_mp2,af_ident)

if (status_mp2=0) then
  loop          ;Schaufelauftrag zyklisch ausführen...
    IO_data(1,af_ident,status_dat)
    exitif status_dat <> 5559 ;...bis Schaufeln nicht mehr aktiv
  endloop
endif
```

```

if status_dat=0 then
    AF_kennung = AF_reg14
endif

#####
;##          AF101 Automatikmodus ausschalten          ##
#####

;Schaufelauftrag kann weiterverwendet werden
AF_reg14 = $8400          ;Befehl Automatikmodus ausschalten
AUTO = 1          ;Automatikmodus noch aktiv

if (status_mp2=0) then
    loop          ;Schaufelauftrag zyklisch ausführen...
        IO_data(1,af_ident,status_dat)
        exitif status_dat <> 5559          ;...bis Schaufeln nicht mehr aktiv
    endloop

    if (status_dat=0) and (AF_reg14=$F400) then
        ;Automatikmodus erfolgreich ausgeschaltet
        AUTO = 0          ;Automatikmodus inaktiv
    else if (status_dat=0) and (AF_reg14=$C000) then
        ;AF101 SW Rev. < 2.00 -> kein Automatikmodus
        AUTO = 0          ;Automatikmodus inaktiv
    endif
endif
endif

```

Variablendeklaration

Name	gültig	Datentyp	Länge	I/O-Typ	Init-Wert	Langname
AF101_slot	lokal	BYTE	1	MERKER	* remanent	
AF_kennung	lokal	WORD	1	MERKER	* remanent	
AF_reg14	lokal	WORD	1	MERKER	* remanent	
AT664_slot	lokal	BYTE	1	MERKER	* remanent	
AT664_typ	lokal	WORD	1	MERKER	* remanent	
AT_reg14	lokal	WORD	1	MERKER	* remanent	
AUTO	lokal	BIT	1	MERKER	* remanent	
Err_Ptest	lokal	BYTE	1	MERKER	* remanent	
IO_struct	lokal	IO_struct	1	MERKER	-----	
af_ident	lokal	LONG	1	MERKER	* remanent	
af_sc_buf	lokal	BYTE	36	MERKER	* remanent	
status_dat	lokal	WORD	1	MERKER	* remanent	
status_mp1	lokal	WORD	1	MERKER	* remanent	
status_mp2	lokal	WORD	1	MERKER	* remanent	

Datentyp

Für die Variable "IO_struct" wird mit dem Datentyp *typedef* eine Struktur definiert. Nach Eingabe des Strukturnamens *IO_struct* können in einer Dialogbox die Elemente der Strukturvariable eingegeben werden.

Komponente	Datentyp	Länge
io_type	BYTE	1
master_no	BYTE	1
slave_no	BYTE	1
module_adr	BYTE	1
intern_off	BYTE	1
mode	BYTE	1
data_len	BYTE	1
reserve	BYTE	1
data_adr	LONG	1