

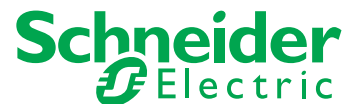
ATV IMC Controller

ATV IMC UserLib Library Guide

04/2012

E10000001052.00

www.schneider-electric.com



The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

© 2012 Schneider Electric. All rights reserved.

Table of Contents

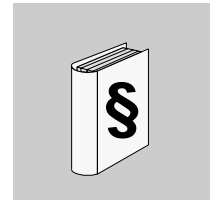


	Safety Information	7
	About the Book	9
Chapter 1	Application Fault and Alarms Management	13
	ApplicationAlarmReset	14
	ApplicationAlarmSet	15
	ApplicationAlarmTextSet	16
	ApplicationFaultHelpTextSet	17
	ApplicationFaultReset	18
	ApplicationFaultSet	19
	ApplicationFaultTextSet	20
Chapter 2	ATV IMC Setting	21
2.1	Counters	22
	Counter1 Activate	23
	Counter1 Get	25
	Counter1 Set	26
	Counter2 Activate	27
	Counter2 Get	29
	Counter2 Set	30
2.2	CycleTimeMinMaxGet	31
	CycleTimeMinMaxGet	31
2.3	CycleTimeSet	32
	CycleTimeSet	32
2.4	CycleTimeStatusGet	33
	CycleTimeStatusGet	33
2.5	LEDSet	34
	LEDSet	34
2.6	ReadSwitch	35
	ReadSwitch	35
2.7	RealTimeClockGet	36
	RealTimeClockGet	36
2.8	RealTimeClockSet	37
	RealTimeClockSet	37
2.9	Synchronized Task	38
	SyncTaskPeriodSet	38

2.10	WatchDogSet.	39
	WatchDogSet.	39
Chapter 3	Display Setting	41
3.1	Customizable List.	42
	DisplayListElementTextSet	43
	DisplayList_1_Set	44
	DisplayList_2_Set	46
	DisplayList_3_Set	47
	DisplayList_4_Set	48
3.2	DisplayBitFieldSet	49
	DisplayBitFieldSet	49
3.3	DisplayLanguageGet	50
	DisplayLanguageGet	50
3.4	DisplayMenuTextSet	51
	DisplayMenuTextSet	51
3.5	DisplayNumericSet	52
	DisplayNumericSet	52
3.6	DisplayTextSet.	54
	DisplayTextSet.	54
3.7	Predefined List.	55
	DisplayList_NO_YES_DONE_Set.	56
	DisplayList_NO_YES_Set	57
	DisplayList_ON_OFF_Set	58
	DisplayList_POS_NEG_Set.	59
	DisplayList_START_STOP_Set.	60
Chapter 4	Drive Control	61
	DriveFunctionC41*.	62
	DriveRunForward.	63
	DriveRunReverse.	64
	DriveStopFast	65
	DriveStopFreeWheel	66
	DriveStopInjdc	67
	DriveStopQuick	68
	DriveStopRamp	69
Chapter 5	Drive Functions	71
	DriveConsistencySet	72
	DriveFaultDatation	73
	DriveFaultExternalSet	74
	DriveFaultReset	75
Chapter 6	Drive Parameter	77
6.1	Aperiodic_request	78
	DriveParameterRead1	79
	DriveParameterWrite1	80

6.2	ApplicationName	81
	ApplicationName	81
6.3	Cyclic_configuration	82
	DriveParameterCyclicReadSet	83
	DriveParameterCyclicWriteSet	85
Chapter 7	Drive Status	87
	DrivecomStateGet	88
	DriveStatusGet	89
Chapter 8	Drive Velocity, Torque, Position, and PI	91
	DriveActualPositionGet	92
	DriveActualTorqueGet	93
	DriveActualVelocityGet	94
	DriveReferencePISet	95
	DriveTargetTorqueSet	96
	DriveTargetVelocitySet	97
	DriveVelocityResolutionSet	98
Chapter 9	MANDATORY_AT_EACH_CYCLE	99
	MANDATORY_AT_EACH_CYCLE	99

Safety Information



Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a Danger safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

 **CAUTION**

CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

NOTICE

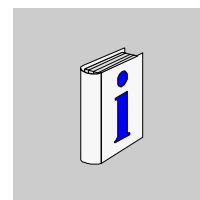
NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

About the Book



At a Glance

Document Scope

This document describes the functions of the ATV UserLib Library.

Validity Note

This document has been updated with the release of SoMachine V3.1.

The technical characteristics of the device(s) described in this manual also appear online. To access this information online:

Step	Action
1	Go to the Schneider Electric home page www.schneider-electric.com .
2	In the Search box type the model number of a product or the name of a product range. <ul style="list-style-type: none">● Do not include blank spaces in the model number/product range.● To get information on a grouping similar modules, use asterisks (*).
3	If you entered a model number, go to the Product datasheets search results and click on the model number that interests you. If you entered the name of a product range, go to the Product Ranges search results and click on the product range that interests you.
4	If more than one model number appears in the Products search results, click on the model number that interests you.
5	Depending on the size of your screen, you may need to scroll down to see the data sheet.
6	To save or print a data sheet as a .pdf file, click Download XXX product datasheet .

The characteristics presented in this manual should be the same as those that appear online. In line with our policy of constant improvement we may revise content over time to improve clarity and accuracy. In the event that you see a difference between the manual and online information, use the online information as your reference.

Related Documents

Title of Documentation	Reference Number
ATV IMC Programming Guide	EIO0000000390 (ENG); EIO0000000391 (FRE); EIO0000000392 (GER); EIO0000000393 (SPA); EIO0000000394 (ITA); EIO0000000395 (CHS)
Altivar 61 - Installation Manual	1760643
Altivar 61 - Programming Manual	1760649
Altivar 61 Communication Manual	1760661
Altivar 71 - Installation Manual	1755843
Altivar 71 - Programming Manual	1755855
Altivar 71 Communication Manual	1755861

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

Product Related Information

Before Starting

Read and understand these instructions before performing any procedure with the drive.

WARNING

UNINTENDED EQUIPMENT OPERATION

Thoroughly read and understand the device manual for the characteristics and properties of the device before attempting to modify them.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Disconnect Drive Power

DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH

- Disconnect all power from all equipment including connected devices prior to removing any covers or doors, or installing or removing any accessories, hardware, cables, or wires except under the specific conditions specified in the appropriate hardware guide for this equipment.
- After disconnecting all power, wait for 15 minutes to allow the DC bus capacitors to discharge in the ATV drives.
- Always use a properly rated voltage sensing device to confirm the power is off where and when indicated.
- Replace and secure all covers, accessories, hardware, cables, and wires and confirm that a proper ground connection exists before applying power to the unit.
- Use only the specified voltage when operating this equipment and any associated products.

Failure to follow these instructions will result in death or serious injury.

NOTE: The DC bus voltage can exceed 1000 Vdc. Use a properly rated voltage-sensing device when performing this procedure. To measure the DC bus voltage.

Operating Environment

WARNING

DAMAGED DRIVE EQUIPMENT

Do not operate or install any drive or drive accessory that appears damaged.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical control functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop, power outage and restart.
- Separate or redundant control paths must be provided for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link.
- Observe all accident prevention regulations and local safety guidelines.¹
- Each implementation of this equipment must be individually and thoroughly tested for proper operation before being placed into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⁽¹⁾: For additional information, refer to NEMA ICS 1.1 (latest edition), Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techcomm@schneider-electric.com.

Application Fault and Alarms Management



Overview

This chapter describes the application fault and alarms management functions of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
ApplicationAlarmReset	14
ApplicationAlarmSet	15
ApplicationAlarmTextSet	16
ApplicationFaultHelpTextSet	17
ApplicationFaultReset	18
ApplicationFaultSet	19
ApplicationFaultTextSet	20

ApplicationAlarmReset

Functional Description

This program removes the alarm indication of the local drive. Alarm indication can be visible in the menu **Monitoring** -> **Alarms** of your Altivar graphic display.

ApplicationAlarmSet

Functional Description

This program sets an alarm indication of the local drive. Alarm indication can be visible in the menu **Monitoring** -> **Alarms** of your Altivar graphic display.

The `ApplicationAlarmTextSet` (*see page 16*) program can customize the name of the alarm.

ApplicationAlarmTextSet

Functional Description

This program gives a new description text to the application alarm. By default, the text is 'Option alarm'.

Input Parameter

Name	Data Type	Description
Text	STRING (15 characters max.)	Text to be displayed on the alarm fault.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function was not executed correctly.

ApplicationFaultHelpTextSet

Functional Description

When the `ApplicationFaultSet` (*see page 19*) program generates an application error, a help screen is displayed when you press the F1 or click the **Help** button on your Altivar graphic display.

Input Parameters

Name	Data Type	Description
TextLigne1	STRING (24 characters max.)	String to be displayed on line 1 of the help.
TextLigne2	STRING (24 characters max.)	String to be displayed on line 2 of the help.
TextLigne3	STRING (24 characters max.)	String to be displayed on line 3 of the help.
TextLigne4	STRING (24 characters max.)	String to be displayed on line 4 of the help.
TextLigne5	STRING (24 characters max.)	String to be displayed on line 5 of the help.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function was not executed correctly.

ApplicationFaultReset

Functional Description

This program resets the detected error in the local drive. The error indication appears on your Altivar graphic display.

ApplicationFaultSet

Functional Description

This program forces the local drive into an error state. The error is displayed on your Altivar graphic display. This drive error will be added to the history in the menu **Diagnostics -> Fault History** of your Altivar graphic display.

It is possible to:

- customize the text displayed with the `ApplicationFaultTextSet` (*see page 20*) program.
- this text can be customize before or after the drive error generation.
- add a help screen to give more information about the drive error using `ApplicationFaultHelpTextSet` (*see page 17*).
- specify a sub-code. This sub-code can be visualized in the **More Fault Info** menu of your Altivar graphic display.

Input Parameter

Name	Data Type	Description
subcode	BYTE	Sub-code error

ApplicationFaultTextSet

Functional Description

This program customizes the text displayed when a drive error has been set.

Input Parameter

Name	Data Type	Description
Text	STRING (10 characters max.)	Text to be displayed on the drive error.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function was not executed correctly.

ATV IMC Setting

2

Overview

This chapter describes the ATV IMC settings functions and function blocks of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
2.1	Counters	22
2.2	CycleTimeMinMaxGet	31
2.3	CycleTimeSet	32
2.4	CycleTimeStatusGet	33
2.5	LEDSet	34
2.6	ReadSwitch	35
2.7	RealTimeClockGet	36
2.8	RealTimeClockSet	37
2.9	Synchronized Task	38
2.10	WatchDogSet	39

2.1

Counters

What's in this Section?

This section contains the following topics:

Topic	Page
Counter1Activate	23
Counter1Get	25
Counter1Set	26
Counter2Activate	27
Counter2Get	29
Counter2Set	30

Counter1Activate

Functional Description

This program enables the alternative use of the ATV IMCs digital input LI_51 as a counter. The input appears in the process image, but status changes at the input can be counted and read by the program `Counter1Get`.

Another use of this input is to combine with digital input LI_52 as a single-ended incremental encoder input (A/B line). The program `Counter1Get` can also read the value of the incremental encoder.

The ATV IMC drive controller only supports 2 counters simultaneously. You can choose these 2 counters through 4 possibilities:

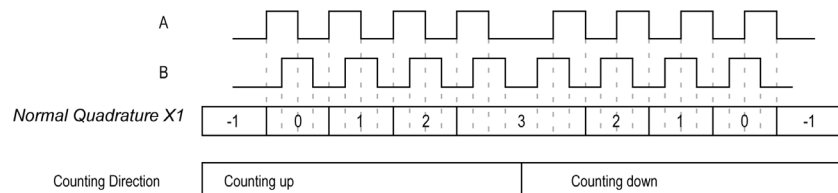
- `Counter1Activate` program
- `Counter2Activate` program
- HSC0, available through SoMachine configuration, and therefore configured before the program runs.
- HSC1, available through SoMachine configuration, and therefore configured before the program runs.

An error appears if you try to configure more than 2 counters simultaneously. If the LI_51 or LI_52 are used by an HSC configuration (HSC0 or HSC1), the `Counter1Activate` returns an error.

Input Parameters

Name	Data Type	Description
<code>bExecute</code>	BOOL	TRUE = The parameter at <code>ucMode</code> is activated.
<code>ucMode</code>	BYTE	0 = LI_51 is digital I/O. 1 = LI_51 works as a counter. 2 = LI_51 and LI_52 work as an encoder interface (in a Normal Quadrature x1 mode). NOTE: To change mode between 1 and 2, you must pass by mode 0.

Normal Quadrature x 1 mode diagram:



Output Parameter

Name	Data Type	Description
bDone	BOOL	TRUE = The parameter at ucMode is successfully activated.

Counter1Get

Functional Description

This program reads the number of rising edges on LI_51, or the encoder counts since the function `Counter1Activate` has been executed.

Input Parameter

Name	Data Type	Description
<code>bExecute</code>	BOOL	TRUE = Executes the function. The output <code>liValue</code> of the function block is updated with the current counter or encoder value.

Output Parameters

Name	Data Type	Description
<code>liValue</code>	DINT	Counter or incremental encoder value read at the last execution of the function.
<code>bDone</code>	BOOL	TRUE = The output <code>liValue</code> of the function block was updated. FALSE = The output <code>liValue</code> of the function block was not updated.

Counter1Set

Functional Description

This program sets the counter1 value of the ATV IMC. This function block can be used irrespective of the mode of Counter1Activate.

Input Parameters

Name	Data Type	Description
bExecute	BOOL	TRUE = Executes the function.
liValue	DINT	The value written to the counter.

Output Parameter

Name	Data Type	Description
bDone	BOOL	TRUE = The liValue is successfully written to the counter.

Counter2Activate

Functional Description

This program enables the alternative use of the ATV IMCs digital input LI_59 as a counter. The input appears in the process image, but status changes at the input can be counted and read by the program `Counter2Get`.

Another use of this input is to combine with digital input LI_60 as a single-ended incremental encoder input (A/B line). The program `Counter2Get` can also read the value of the incremental encoder.

The ATV IMC drive controller only supports 2 counters simultaneously. You can choose these 2 counters through 4 possibilities:

- `Counter1Activate` program
- `Counter2Activate` program
- HSC0, available through SoMachine configuration, and therefore configured before the program runs.
- HSC1, available through SoMachine configuration, and therefore configured before the program runs.

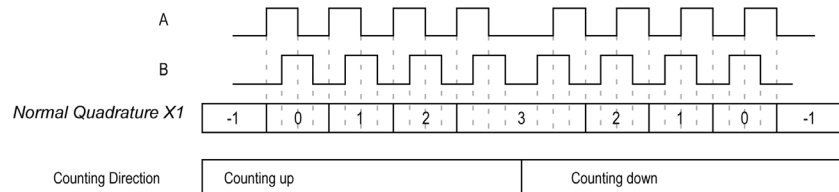
An error appears if you try to configure more than 2 counters simultaneously.

If the LI_59 or LI_60 are used by an HSC configuration (HSC0 or HSC1), the `Counter2Activate` returns an error.

Input Parameters

Name	Data Type	Description
<code>bExecute</code>	BOOL	TRUE = The parameter at <code>ucMode</code> is activated.
<code>ucMode</code>	BYTE	0 = Input 59 is digital I/O 1 = Input 59 works as a counter LI_59 and LI_60 work as an encoder interface (in a Normal Quadrature x1 mode). NOTE: To change mode between 1 and 2, you must pass by mode 0.

Normal Quadrature x 1 mode diagram:



Output Parameter

Name	Data Type	Description
bDone	BOOL	TRUE = The parameter at ucMode is successfully activated.

Counter2Get

Functional Description

This program reads the number of rising edge on LI_59, or the encoder counts since the function Counter2Activate has been executed.

Input Parameter

Name	Data Type	Description
bExecute	BOOL	TRUE = Executes the function. The output liValue of the function block is updated with the current counter or encoder value.

Output Parameters

Name	Data Type	Description
liValue	DINT	Counter or incremental encoder value read at the last execution of the function.
bDone	BOOL	TRUE = The output liValue of the function block was been updated. FALSE = The output liValue of the function block was not updated.

Counter2Set

Functional Description

This program sets the counter2 value of the ATV IMC. This function block can be used irrespective of the mode of Counter2Activate.

Input Parameters

Name	Data Type	Description
bExecute	BOOL	TRUE = Executes the function.
liValue	DINT	The value written to the counter.

Output Parameter

Name	Data Type	Description
bDone	BOOL	TRUE = The liValue is successfully written to the counter.

2.2 CycleTimeMinMaxGet

CycleTimeMinMaxGet

Functional Description

This program reads the minimum, maximum, and the current cycle time in ms of the first cycle task configured on SoMachine. For example, the 'MAST' cycle task in the Template.

Output Parameters

Name	Data Type	Description
cycle_time_min	TIME	The value of minimum cycle time
cycle_time_max	TIME	The value of maximum cycle time
cycle_time	TIME	The value of current cycle time

2.3 CycleTimeSet

CycleTimeSet

Functional Description

This program works in conjunction with the program `CycleTimeStatusGet`.

This program defines the cycle time of the first cycle task configured on SoMachine that you do not want to be exceeded. If the desired cycle time is exceeded, the program `CycleTimeStatusGet` returns an error.

Input Parameters

Name	Data Type	Description
<code>bExecute</code>	BOOL	TRUE = The cycle time is updated with <code>ucValue</code> . FALSE = The existing cycle time is used.
<code>ucValue</code>	BYTE	The cycle time is in milliseconds (5...100).

Output Parameter

Name	Data Type	Description
<code>bDone</code>	BOOL	FALSE = The <code>ucValue</code> is out of range (5...100).

2.4 CycleTimeStatusGet

CycleTimeStatusGet

Functional Description

This program works in conjunction with the program `CycleTimeSet`.

This program can be used to evaluate if your cycle time is below the cycle time you require with the `CycleTimeSet` program.

Output Parameter

Name	Data Type	Description
<code>bError</code>	BOOL	TRUE = The desired cycle time is exceeded.

2.5

LEDSet

LEDSet

Functional Description

This program is used to set the state of the user LED (*see ATV IMC Drive Controller, Hardware Guide*) in the application program (green only).

The ATV IMC board has 1 configurable user LED.

NOTE: Prefer using the PLCSystem `SetLEDBehavior` (*see ATV IMC Drive Controller, System Functions and Variables, ATV-IMC PLCSystem Library Guide*) function to control your LED.

Input Parameters

Name	Data Type	Description
bExecute	BOOL	TRUE = The state of the LED is updated with the value of bLedOn. FALSE = The existing state of the LED is maintained.
bLedOn	BOOL	TRUE = The LED will illuminate green.

Output Parameter

Name	Data Type	Description
bDone	BOOL	TRUE = The state of the LED is successfully updated with the value of bLedOn.

2.6 ReadSwitch

ReadSwitch

Functional Description

This program reads the value of the switch block (see *ATV IMC Drive Controller, Hardware Guide*).

Input Parameter

Name	Data Type	Description
ucValue	BYTE	Bit 0 = switch1 Bit 1 = switch2 Bit 2 = switch3 Bit 3 = switch4

2.7 RealTimeClockGet

RealTimeClockGet

Functional Description

This function reads the real time clock.

NOTE: After replacing the battery, the clocks initial value must be set using the function `RealTimeClockSet` or through SoMachine into the services tab of the target (Write local time or Synchronize with date/time tool of local).

Input Parameter

Name	Data Type	Description
bDmy	BOOL	TRUE = Starts the function.

Output Parameter

Name	Data Type	Description
RealTimeClockGet	DATE_AND_ TIME	Format: Year-Month-Day-Hour:Minute:Second Example: dt#2003-03-27-14:22:45

2.8 RealTimeClockSet

RealTimeClockSet

Functional Description

This function can be used to set or adjust the real time clock.

NOTE: This must be done after replacing the battery or the first time the ATV IMC board is powered.

Input Parameter

Name	Data Type	Description
SetDateAndTime	DATE_AND_ TIME	Format: Year-Month-Day-Hour:Minute:Second Example: dt#2010-06-16-14:22:45

Output Parameter

Name	Data Type	Description
RealTimeClockSet	BOOL	TRUE = The time of the clock has been set.

2.9 Synchronized Task

SyncTaskPeriodSet

Functional Description

This program sets the `SyncTask` period (see *ATV IMC Drive Controller, Programming Guide*).

The value will be rounded down to a multiple of 2.

For example:

- With a `Period` set to 3, the task is executed every 2 ms.
- With a `Period` set to 5, the task is executed every 4 ms.

NOTE: A watchdog exception is triggered if the effective execution time of the associated task is greater than the `SyncTask` period.

Input Parameter

Name	Data Type	Description
Period	WORD	The range of period is 0...65535.

2.10 WatchDogSet

WatchDogSet

Functional Description

This program is used to set the hardware watchdog (*see ATV IMC Drive Controller, Programming Guide*) timeout value. The possible values are between 100 and 800 ms. If the actual cycle time exceeds the watchdog time, a **Reset** on the ATV IMC drive controller is triggered. The hardware watchdog is reloaded into the background task at the same priority level as the freewheel task.

NOTE: See also the ATV IMC SysLib library function `Set_WatchDog` (which allows setting the watchdog value to 15 s) to set the hardware watchdog value.

Input Parameters

Name	Data Type	Description
bExecute	BOOL	TRUE = The watchdog time is updated with uiValue. FALSE = The existing cycle time is used. In case no watchdog time is set, the watchdog time will be 20 s.
uiValue	UINT	The watchdog time is in milliseconds (100..800).

Output Parameter

Name	Data Type	Description
bDone	BOOL	TRUE = The new time for watchdog is set. FALSE = The uiValue is out of range.

Display Setting

3

Overview

This chapter describes the dynamic display setting function blocks of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

The display settings can also be set with SoMachine. All the display settings (see *ATV IMC Drive Controller, Programming Guide*) are set automatically to the settings defined with SoMachine when the ATV IMC controller transitions to a RUNNING state.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
3.1	Customizable List	42
3.2	DisplayBitFieldSet	49
3.3	DisplayLanguageGet	50
3.4	DisplayMenuTextSet	51
3.5	DisplayNumericSet	52
3.6	DisplayTextSet	54
3.7	Predefined List	55

3.1 Customizable List

What's in this Section?

This section contains the following topics:

Topic	Page
DisplayListElementTextSet	43
DisplayList_1_Set	44
DisplayList_2_Set	46
DisplayList_3_Set	47
DisplayList_4_Set	48

DisplayListElementTextSet

Functional Description

This program is used to define the 4 available lists for the Altivar graphic display which can be activated by the following functions:

- DisplayList_1_Set (see page 44)
- DisplayList_2_Set (see page 46)
- DisplayList_3_Set (see page 47)
- DisplayList_4_Set (see page 48)

Input Parameters

Name	Data Type	Description
ListIndice	BYTE	Indice of the list.
ElementIndice	BYTE	Indice of the element to be customized.
TextElementLong	STRING (9 characters max.)	Long string to be displayed on the graphic display.
TextElementShort	STRING (5 characters max.)	Short string to be displayed if the parameter is selected to be displayed on the top bar (5 characters max.).

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The program DisplayListElementTextSet was not executed correctly.

DisplayList_1_Set

Functional Description

This program creates and displays a new line of information in the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the **list 1**.

Indice corresponds to the `Display_Ox` used (for example, `Indice = 1` corresponds to `Display_001`), then `Display_001` value is the index of the text displayed of the **list 1**.

The text of the 4 lists is defined through the `DisplayListElementTextSet` (see page 43) program.

Input Parameters

Name	Data Type	Description
Indice	WORD	The indice is also the order in which the text is displayed into the menu. Index of the parameter (1 stands for <code>Display_001</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (see <i>ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_1_Set</code> was not executed correctly.

Example

List 1 has the following values:

- 1 Toto1
- 2 Toto2
- 3 Toto3

List 2 has the following values:

- 1** Totob1
- 2** Totob2
- 3** Totob3

```
Display_O01:=2;
```

```
DisplayList_1_Set (Indice:=1);.
```

```
// the ATV IMC graphic display will display Toto2.
```

```
Display_O02:=3;
```

```
DisplayList_2_Set (Indice:=2);.
```

```
// the ATV IMC graphic display will display Totob3.
```

```
Display_O13:=1;
```

```
DisplayList_1_Set (Indice:=13);.
```

```
// the ATV IMC graphic display will display Toto1.
```

```
Display_O14:=2;
```

```
DisplayList_2_Set (Indice:=14);.
```

```
// the ATV IMC graphic display will display Totob2.
```

The graphic display of the Altivar will display:

- Toto2
- Totob3
- Toto1
- Totob2

DisplayList_2_Set

Functional Description

This program creates and displays a new line of information in the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the **list 2**.

Indice corresponds to the `Display_Ox` used (for example, `Indice = 1` corresponds to `Display_001`) then `Display_001` value is the index of the text displayed of the **list 2**.

The text of the 4 lists is defined through the `DisplayListElementTextSet` (see page 43) program.

Input Parameters

Name	Data Type	Description
Indice	WORD	The indice is also the order in which the text is displayed into the menu. Index of the parameter (1 stands for <code>Display_001</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (see <i>ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_2_Set</code> was not executed correctly.

Example

Refer to the Example (see page 44) of the `DisplayList_1_Set` program.

DisplayList_3_Set

Functional Description

This program creates and displays a new line of information in the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the **list 3**.

Indice corresponds to the `Display_Ox` used (for example `Indice = 1` corresponds to `Display_001`) then `Display_001` value is the index of the text displayed of the **list 3**.

The text of the 4 lists is defined through the `DisplayListElementTextSet` (see page 43) program.

Input Parameters

Name	Data Type	Description
Indice	WORD	The indice is also the order in which the text is displayed into the menu. Index of the parameter (1 stands for <code>Display_001</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (see <i>ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_3_Set</code> was not executed correctly.

Example

Refer to the Example (see page 44) of the `DisplayList_1_Set` program.

DisplayList_4_Set

Functional Description

This program creates and displays a new line of information into the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the **list 4**.

Indice corresponds to the `Display_Ox` used (for example, `Indice = 1` corresponds to `Display_001`) then `Display_001` value is the index of the text displayed of the **list 4**.

The text of the 4 lists is defined through the `DisplayListElementTextSet` (see page 43) program.

Input Parameters

Name	Data Type	Description
Indice	WORD	The indice is also the order in which the text is displayed into the menu. Index of the parameter (1 stands for <code>Display_001</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (see <i>ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_4_Set</code> was not executed correctly.

Example

Refer to the Example (see page 44) of the `DisplayList_1_Set` program.

3.2 DisplayBitFieldSet

DisplayBitFieldSet

Functional Description

This program creates and displays a new line of information into the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displays a bit field.

Indice corresponds to the `Display_Ox` used, for example, `Indice = 1` corresponds to `Display_001` value.

Input Parameters

Name	Data Type	Description
Indice	WORD	The indice is also the order in which the text is displayed into the menu. Index of the parameter (1 stands for <code>Display_001</code> , and so on).
Visibility	BOOL	FALSE = The line is not displayed in the PROGRAMMABLE CARD menu. TRUE = The line is displayed in the PROGRAMMABLE CARD menu.
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (<i>see ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayBitFieldSet</code> was not executed correctly.

3.3 DisplayLanguageGet

DisplayLanguageGet

Functional Description

This program returns the language chosen in the local drive. If a new language has been chosen, then the boolean `NewLanguage` is set to TRUE.

Output Parameters

Name	Data Type	Description
Language	WORD	Word of the language
NewLanguage	BOOL	TRUE = New language selected in the drive

3.4 DisplayMenuTextSet

DisplayMenuTextSet

Functional Description

This program is used to set up the name of the **PROGRAMMABLE CARD** menu.

Input Parameters

Name	Data Type	Description
TextmenuLong	STRING (13 characters max.)	Name of the menu to be displayed on the graphical display.
TextmenuShort	STRING (4 characters max.)	Name of the menu to be displayed on the local (7 segments) display.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function DisplayMenuTextSet was not executed correctly.

3.5 DisplayNumericSet

DisplayNumericSet

Functional Description

This program creates and displays a new line of information in the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displays a numeric value.

Indice corresponds to the `Display_Ox` used, for example, `Indice = 1` corresponds to `Display_001` value.

Input Parameters

Name	Data Type	Description
Indice	WORD	The index is also the order in which the text is displayed into the menu. Index of the parameter (1 stands for <code>Display_001</code> , and so on).
Visibility	BOOL	FALSE = The line is not displayed in the PROGRAMMABLE CARD menu. TRUE = The line is displayed in the PROGRAMMABLE CARD menu.
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (<i>see ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Signed	BOOL	FALSE = Not signed. TRUE = Signed. NOTE: <code>Display_Ox</code> in SoMachine are global values (WORD) and are not signed. When viewing the <code>Display_Ox</code> values in the Local Display -> I/O Mapping tab, you always have the absolute value displayed.
Minimum	WORD	The minimum value of the <code>Display_Ox</code> .

Name	Data Type	Description
Maximum	WORD	The maximum value of the Display_Ox.
Default	WORD	The default value of the Display_Ox.
Decimal	BYTE	This is used for the format of decimal digits of the Display_Ox. 0 = No comma (000) 1 = One number after comma (0,0) 2 = Two number after comma (0,00)

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function DisplayNumericSet was not executed correctly.

3.6 DisplayTextSet

DisplayTextSet

Functional Description

In the **PROGRAMMABLE CARD** menu, the default name of a new line of information generated by a `Display...Set` function is `Display_Ox` followed by its value.

This program allows to modify this name.

Input Parameters

Name	Data Type	Description
Indice	WORD	Index of the <code>Display_Ox</code> (between 1 and 50).
TextParameter	STRING (13 characters max.)	New name of the variable <code>Display_Ox</code> for the line.
TextUnit	STRING (4 characters max.)	The unit of the <code>Display_Ox</code> .

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayTextSet</code> was not executed correctly.

3.7

Predefined List

What's in this Section?

This section contains the following topics:

Topic	Page
DisplayList_NO_YES_DONE_Set	56
DisplayList_NO_YES_Set	57
DisplayList_ON_OFF_Set	58
DisplayList_POS_NEG_Set	59
DisplayList_START_STOP_Set	60

DisplayList_NO_YES_DONE_Set

Functional Description

This program creates and displays a new line of information into the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the list NO_YES_DONE.

Indice corresponds to the `Display_Ox` used, then `Display_O01` value is the index of the text displayed of the list NO_YES_DONE (for example, `Indice = 1` corresponds to `Display_O01 = YES`).

The possible choices are:

- 0 NO
- 1 YES
- 2 DONE

Input Parameters

Name	Data Type	Description
Indice	WORD	Index of the <code>Display_Ox</code> (1 stands for <code>Display_O01</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (<i>see ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_NO_YES_DONE_Set</code> was not executed correctly.

DisplayList_NO_YES_Set

Functional Description

This program creates and displays a new line of information into the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the list NO_YES.

Indice corresponds to the `Display_Ox` used, then `Display_O01` value is the index of the text displayed of the list NO_YES (for example, `Indice = 1` corresponds to `Display_O01 = YES`).

The possible choices are:

- 0 NO
- 1 YES

Input Parameters

Name	Data Type	Description
Indice	WORD	Index of the <code>Display_Ox</code> (1 stands for <code>Display_O01</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (<i>see ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_NO_YES_Set</code> was not executed correctly.

DisplayList_ON_OFF_Set

Functional Description

This program creates and displays a new line of information into the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the list ON_OFF.

Indice corresponds to the `Display_Ox` used, then `Display_001` value is the index of the text displayed of the list ON_OFF (for example, `Indice = 1` corresponds to `Display_001 = OFF`).

The possible choices are:

0 NO

1 OFF

Input Parameters

Name	Data Type	Description
Indice	WORD	Index of the <code>Display_Ox</code> (1 stands for <code>Display_001</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (<i>see ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_ON_OFF_Set</code> was not executed correctly.

DisplayList_POS_NEG_Set

Functional Description

This program creates and displays a new line of information into the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the list POS_NEG.

Indice corresponds to the `Display_Ox` used, then `Display_O01` value is the index of the text displayed of the list POS_NEG (for example, `Indice = 1` corresponds to `Display_O01 = NEG`).

The possible choices are:

0 POS

1 NEG

Input Parameters

Name	Data Type	Description
Indice	WORD	Index of the <code>Display_Ox</code> (1 stands for <code>Display_O01</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (<i>see ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_POS_NEG_Set</code> was not executed correctly.

DisplayList_START_STOP_Set

Functional Description

This program creates and displays a new line of information into the **PROGRAMMABLE CARD** menu of the Altivar graphic display.

This new line displayed is a text from the list START_STOP.

Indice corresponds to the `Display_Ox` used, then `Display_O01` value is the index of the text displayed of the list START_STOP (for example, `Indice = 1` corresponds to `Display_O01 = STOP`).

The possible choices are:

0 START

1 STOP

Input Parameters

Name	Data Type	Description
Indice	WORD	Index of the <code>Display_Ox</code> (1 stands for <code>Display_O01</code> , and so on).
AppliLock	BOOL	FALSE = The application and all channels of the drive can modify the <code>Display_Ox</code> value. TRUE = Only the ATV IMC can write the value of <code>Display_Ox</code> .
RunLock	BOOL	FALSE = <code>Display_Ox</code> can be modified at any time. TRUE = <code>Display_Ox</code> cannot be written when the drive is in Run mode.
Save	BOOL	FALSE = Not saved. TRUE = Parameter saved into the UserLib global variable: <code>Saved_Display_Ox[]</code> . The template (<i>see ATV IMC Drive Controller, Programming Guide</i>) provides a way to restore the saved values.
Default	WORD	Default <code>Display_Ox</code> value of the list.

Output Parameter

Name	Data Type	Description
Error	WORD	1 = The function <code>DisplayList_START_STOP_Set</code> was not executed correctly.

Drive Control

4

Overview

This chapter describes the drive control programs of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
DriveFunctionC41•	62
DriveRunForward	63
DriveRunReverse	64
DriveStopFast	65
DriveStopFreeWheel	66
DriveStopInjdc	67
DriveStopQuick	68
DriveStopRamp	69

DriveFunctionC41•

Functional Description

This program is used to enable/disable a bit on the **CMD** (command) register of the local drive.

There are 5 DriveFunctionC41•:

- DriveFunctionC411
- DriveFunctionC412
- DriveFunctionC413
- DriveFunctionC414
- DriveFunctionC415

These bits of the **CMD** register can be linked to specific functions configured in your local drive.

The drive management is performed through the use of the **MANDATORY_AT_EACH_CYCLE** function (*see page 99*). The **CMD** register is updated at the end of the **on_SYNC** event even if no task is associated to this event.

Input Parameter

Name	Data Type	Description
Activate	BOOL	TRUE = Activates the function set on C41• bit FALSE = Deactivates the function set on C41• bit

DriveRunForward

Functional Description

This program does a **Run Forward** command to the local drive.

The drive management is performed through the use of the MANDATORY_AT_EACH_CYCLE function (*see page 99*). The function is executed at the end of the `on_SYNC` event even if no task is associated to this event.

NOTE: This function is disabled if the local drive is in **IO mode**.

DriveRunReverse

Functional Description

This program does a **Run Reverse** command to the local drive.

The drive management is performed through the use of the MANDATORY_AT_EACH_CYCLE function (*see page 99*). The function is executed at the end of the `on_SYNC` event even if no task is associated to this event.

NOTE: This function is disabled if the local drive is in **IO mode**.

DriveStopFast

Functional Description

This program does a **Fast Stop** command to the local drive.

The drive management is performed through the use of the MANDATORY_AT_EACH_CYCLE function (*see page 99*).

NOTE: This function is disabled if the local drive is in **IO mode**.

DriveStopFreeWheel

Functional Description

This program does a **Free Wheel Stop** command to the local drive.

The drive management is performed through the use of the MANDATORY_AT_EACH_CYCLE function (*see page 99*). The function is executed at the end of the `on_SYNC` event even if no task is associated to this event.

NOTE:

- This function can be used to unlock the drive after a `DriveStopQuick` call or if the **STOP** button of the Altivar graphic display has been pressed.
- This function is disabled if the local drive is in **IO mode**.

DriveStopInjdc

Functional Description

This program does a **DC Injection Stop** command to the local drive.

The drive management is performed through the use of the MANDATORY_AT_EACH_CYCLE function (*see page 99*).

NOTE: This function is disabled if the local drive is in **IO mode**.

DriveStopQuick

Functional Description

This program does a **Quick Stop** command to the local drive.

The drive management is performed through the use of the MANDATORY_AT_EACH_CYCLE function (*see page 99*). The function is executed at the end of the on_SYNC event even if no task is associated to this event.

NOTE:

- When DriveStopQuick has been done, a call to DriveStopFreeWheel is needed to unlock the drive. The **STOP/RESET** button of the Altivar graphic display can also unlock the drive.
- This function is disabled if the local drive is in **IO mode**.

DriveStopRamp

Functional Description

This program does a **Ramp Stop** command to the local drive.

After a stop ramp, the state of the drive is still RUN.

The drive management is performed through the use of the MANDATORY_AT_EACH_CYCLE function (*see page 99*). The function is executed at the end of the `on_SYNC` event even if no task is associated to this event.

NOTE: This function is disabled if the local drive is in **IO mode**.

Drive Functions

5

Overview

This chapter describes the drive functions programs of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
DriveConsistencySet	72
DriveFaultDatation	73
DriveFaultExternalSet	74
DriveFaultReset	75

DriveConsistencySet

Functional Description

This program enables or disables the consistency control of the local drive regarding its configuration parameters.

For more information about consistency control, refer to your drive programming manual.

It is necessary to disable the consistency control:

- when modifying multiple parameters of the drive configuration.
- to allow multiple configuration parameters writing at the same time.

Input Parameter

Name	Data Type	Description
bOn	BOOL	TRUE = Disables the consistency control of the parameter. FALSE = Enables the consistency control of the parameters.

WARNING

UNINTENDED EQUIPMENT OPERATION

Enable the consistency control by using the function block `DriveConsistencySet` (`bOn = FALSE`) as soon as the data transfer is finished.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

DriveFaultDatation

Functional Description

This program enables the display of the `TIME` and `DAY` for when the errors were recorded in the **Diagnostic** menu of the graphic display.

Input Parameter

Name	Data Type	Description
Activate	BOOL	TRUE = Activates the function.

DriveFaultExternalSet

Functional Description

This program reports an external error and places the local drive in an error state.

Input Parameter

Name	Data Type	Description
bOn	BOOL	TRUE = Sets the drive in error state and an external error indication is generated. FALSE = Removes the external error indication, the drive stays in error state.

NOTE: If you want to reset the external error, bOn has to be released by setting it to FALSE and call the `DriveFaultReset` function (*see page 75*).

DriveFaultReset

Functional Description

This program resets the errors of the local drive.

An error is reset if the indication of the detected error is previously cleared and if `Fr1` = `APP`. The drive management is performed through the use of the `MANDATORY_AT_EACH_CYCLE` function (*see page 99*). The function is executed at the end of the `on_SYNC` event even if no task is associated to this event.

Output Parameter

Name	Data Type	Description
<code>bStatus</code>	<code>BOOL</code>	TRUE = Resets the detected errors of the drive.

Drive Parameter

6

Overview

This chapter describes the drive parameter programs of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
6.1	Aperiodic_request	78
6.2	ApplicationName	81
6.3	Cyclic_configuration	82

6.1 Aperiodic_request

What's in this Section?

This section contains the following topics:

Topic	Page
DriveParameterRead1	79
DriveParameterWrite1	80

DriveParameterRead1

Functional Description

This program reads one parameter of the drive.

Input Parameters

Name	Data Type	Description
uiAddress	WORD	Address of the drive parameter to be read.
bExecute	BOOL	On rising edge, starts the function block execution.

Output Parameters

Name	Data Type	Description
bError	BOOL	TRUE = An error occurred.
bBusy	BOOL	TRUE = Indicates that the function block execution is in progress.
bDone	BOOL	TRUE = Indicates that the parameter is read. Function block execution is finished.
uiValue	WORD	Value read at the address uiAddress.
wCodeReq	WORD	Reserved.

WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use the Drive Parameter (*see page 77*) function in a POU linked to the freewheel task.
- Do not execute the program without setting uiAddress.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

DriveParameterWrite1

Functional Description

This program writes one parameter of the local drive.

Input Parameters

Name	Data Type	Description
uiAddress	WORD	Address of the drive parameter to be read.
uiValue	WORD	Value to be written.
bExecute	BOOL	On rising edge, starts the function block execution.

Output Parameters

Name	Data Type	Description
bError	BOOL	TRUE = An error occurred.
bBusy	BOOL	TRUE = Indicates that the function block execution is in progress.
bDone	BOOL	TRUE = Indicates that the parameter is written. Function block execution is finished.
wCodeReq	WORD	Reserved.

WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use the Drive Parameter (*see page 77*) function in a POU linked to the freewheel task.
- Do not execute the program without setting `uiAddress`.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

6.2 ApplicationName

ApplicationName

Functional Description

This program sends the application name to the drive in order to display it on the keypad (**1.11 Identification**).

Input Parameters

Name	Data Type	Description
sName	STRING(16)	Name of the application.
bExecute	BOOL	On rising edge, starts the function block execution.

Output Parameters

Name	Data Type	Description
bError	BOOL	TRUE = An error occurred.
bBusy	BOOL	TRUE = Indicates that the function block execution is in progress.
bDone	BOOL	TRUE = Indicates that the application name is updated. Function block execution is finished.

WARNING

UNINTENDED EQUIPMENT OPERATION

Only use the Drive Parameter functions (*see page 77*) in a POU linked to the freewheel task.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

6.3 **Cyclic_configuration**

What's in this Section?

This section contains the following topics:

Topic	Page
DriveParameterCyclicReadSet	83
DriveParameterCyclicWriteSet	85

DriveParameterCyclicReadSet

Functional Description

When using SoMachine, you can define up to 8 drive registers (*see ATV IMC Drive Controller, Programming Guide*) to be exchanged between the ATV IMC controller and the local drive.

This program defines the drive registers to be read by the ATV IMC controller.

When the MANDATORY_AT_EACH_CYCLE is executed, up to 8 values are read in the drive register, those values are stored in the DRIVE_PI1 to DRIVE_PI8 variables.

WARNING

UNINTENDED EQUIPMENT OPERATION

Only use the Drive Parameter functions (*see page 77*) in a POU linked to the freewheel task.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Input Parameters

Name	Data Type	Description
addrDRIVE_PI1	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI1 variable.
addrDRIVE_PI2	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI2 variable.
addrDRIVE_PI3	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI3 variable.
addrDRIVE_PI4	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI4 variable.
addrDRIVE_PI5	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI5 variable.
addrDRIVE_PI6	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI6 variable.

Name	Data Type	Description
addrDRIVE_PI7	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI7 variable.
addrDRIVE_PI8	WORD	Address of the drive parameter register. The value of the register is stored in the DRIVE_PI8 variable.
bExecute	BOOL	TRUE = Executes this function.

Output Parameters

Name	Data Type	Description
bError	BOOL	TRUE = An error occurred.
bDone	BOOL	TRUE = Function block execution is finished.

DriveParameterCyclicWriteSet

Functional Description

When using SoMachine, you can define up to 8 drive registers to be exchanged between the ATV IMC controller and the local drive.

This program defines the drive registers to be written by the ATV IMC controller.

When the MANDATORY_AT_EACH_CYCLE is executed, up to 8 values are written in the drive register, those values are provided by the DRIVE_PO1 to DRIVE_PO8 variables.

WARNING

UNINTENDED EQUIPMENT OPERATION

Only use the Drive Parameter functions (*see page 77*) in a POU linked to the freewheel task.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Input Parameters

Name	Data Type	Description
addrDRIVE_PO1	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO1 variable.
addrDRIVE_PO2	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO2 variable.
addrDRIVE_PO3	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO3 variable.
addrDRIVE_PO4	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO4 variable.
addrDRIVE_PO5	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO5 variable.
addrDRIVE_PO6	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO6 variable.
addrDRIVE_PO7	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO7 variable.
addrDRIVE_PO8	WORD	Address of the drive parameter register. The value of the register is provided by the DRIVE_PO8 variable.
bExecute	BOOL	TRUE = Executes this function.

Output Parameters

Name	Data Type	Description
bError	BOOL	TRUE = An error occurred.
bDone	BOOL	TRUE = Function block execution is finished.

Drive Status



Overview

This chapter describes the drive status programs of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
DrivecomStateGet	88
DriveStatusGet	89

DrivecomStateGet

Functional Description

This program reads the **DriveCom** register of the local drive.

Output Parameters

Name	Data Type	Description
bNotReadyToSwitchOn	BOOL	TRUE = Drive state: not ready to switch ON
bSwitchOnDisabled	BOOL	TRUE = Drive state: switch ON disabled
bReadyToSwitchOn	BOOL	TRUE = Drive state: ready to switch ON
bSwitchedOn	BOOL	TRUE = Drive state: switched ON
bOperationEnabled	BOOL	TRUE = Drive state: operation enable
bMalfunction	BOOL	TRUE = Drive state: malfunction
bMalfunctionReactionActive	BOOL	TRUE = Drive state: malfunction reaction active
bQuickStopActive	BOOL	TRUE = Drive state: quick stop active

DriveStatusGet

Functional Description

This program reads the local drive internal status register **ETA**.

Output Parameters

Name	Data Type	Description
bReadyToSwitchOn	BOOL	TRUE = Drive state: ready to switch ON
bSwitchedOn	BOOL	TRUE = Drive state: switched ON
bOperationEnabled	BOOL	TRUE = Drive state: operation enable
bMalfunction	BOOL	TRUE = Drive state: malfunction
bVoltageDisabled	BOOL	TRUE = Voltage disabled
bQuickStop	BOOL	TRUE = Drive state: quick stop active
bSwitchedOnDisabled	BOOL	TRUE = Drive state: switched ON disabled
bAlarm	BOOL	TRUE = Alarm present
bForcedLocal	BOOL	TRUE = Forced local mode absent
bSteadyState	BOOL	TRUE = Reference reached. Output frequency (RFR) = Reference frequency (FRH)
bRefExceeded	BOOL	TRUE = Reference exceeds (< LSP or > HSP)
bStopKeypad	BOOL	TRUE = Stop from keypad STOP key
bDirection	BOOL	TRUE = Reverse rotation FALSE = Forward rotation
wETA	WORD	ETA register value. For more detailed information, refer to Altivar Communication Manual (<i>see page 10</i>).

Drive Velocity, Torque, Position, and PI



8

Overview

This chapter describes the drive velocity, torque, position, and PI functions of the ATV IMC UserLib library in association with the ATV IMC Controller and the local drive (a local drive is the drive on which the ATV IMC Drive Controller card is mounted).

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
DriveActualPositionGet	92
DriveActualTorqueGet	93
DriveActualVelocityGet	94
DriveReferencePISet	95
DriveTargetTorqueSet	96
DriveTargetVelocitySet	97
DriveVelocityResolutionSet	98

DriveActualPositionGet

Functional Description

This function gives the absolute position of the encoder mounted on the drive.

The value is updated on a `on_SYNC` drive event. So, it is recommended to use it in a task associated to the `on_SYNC` event.

Input Parameter

Name	Data Type	Description
bDummy	BOOL	Not relevant.

Output Parameter

Name	Data Type	Description
DriveActualPositionGet	DWORD Minimum: 0 Maximum: 65535	Current absolute position of the encoder mounted on the drive. (Rollovers of the encoder are not managed by this function.)

Example:

```
ActualPosition:=DriveActualPositionGet(TRUE);
```

DriveActualTorqueGet

Functional Description

This function gives the current torque.

The value is updated on a `on_SYNC` drive event. So, it is recommended to use it in a task associated to the `on_SYNC` event.

Input Parameter

Name	Data Type	Description
bDummy	BOOL	Not relevant

Output Parameter

Name	Data Type	Description
DriveActualTorqueGet	WORD	Torque value ($\pm 5\%$)

Example:

```
ActualTorque = DriveActualTorqueGet (TRUE)
```

DriveActualVelocityGet

Functional Description

This function returns the absolute value of the output frequency in tenth of a Hz applied to the motor.

The value is updated on a `on_SYNC` drive event. So, it is recommended to use it in a task associated to the `on_SYNC` event.

Input Parameter

Name	Data Type	Description
bDummy	BOOL	Not relevant.

Output Parameter

Name	Data Type	Description
DriveActualVelocityGet	INT	Absolute value of the output frequency (tenth of a Hz).

Example:

```
ActualVelocity = DriveActualVelocityGet (TRUE)
```

DriveReferencePISet

Functional Description

This function sets the PI regulator consign. This parameter is only taken into account by the local drive if the PID feedback is assigned.

The value is updated with the use of the MANDATORY_AT_EACH_CYCLE function (see page 99).

Input Parameter

Name	Data Type	Description
uiConsigne	WORD	Consign for the PI (0...10000).

Output Parameter

Name	Data Type	Description
DriveReferencePISet	BOOL	TRUE = If uiConsigne is out of the range (0...10000).

DriveTargetTorqueSet

Functional Description

This function sets the target torque. The local drive has to be configured to support the torque reference. For more detailed information, refer to Altivar Programming manual (*see page 10*).

The value is updated on a `on_SYNC` drive event. So, it is recommended to use it in a task associated to the `on_SYNC` event.

Input Parameter

Name	Data Type	Description
<code>iConsigne</code>	INT	In tenth of a percent of the motor nominal torque (0...1000 for 0 %...100 %).

Output Parameter

Name	Data Type	Description
<code>DriveTargetTorqueSet</code>	BOOL	Not relevant.

DriveTargetVelocitySet

Functional Description

This function sets the speed reference for the local drive in tenth of a Hz (by default). The value is updated on a `on_SYNC` drive event. So, it is recommended to use it in a task associated to the `on_SYNC` event.

Input Parameter

Name	Data Type	Description
<code>iConsigne</code>	INT	-5000...5000 corresponds to -500...500 Hz with a resolution of 0.1 Hz (by default). NOTE: The <code>DriveVelocityResolutionSet</code> can change the frequency resolution.

Output Parameter

Name	Data Type	Description
<code>DriveTargetVelocitySet</code>	BOOL	Not relevant.

DriveVelocityResolutionSet

Functional Description

This program changes the target velocity resolution.

The value is updated with the use of the MANDATORY_AT_EACH_CYCLE function (see page 99).

Input Parameter

Name	Data Type	Description
bOn	BOOL	bOn = 0, maximum value = 10 times the maximum frequency of the local drive. For example, iConsigne = 5000 corresponds the frequency to 500 Hz. bOn = 1, iConsigne = 32767 corresponds to the frequency TFR (Top Frequency Register) set in the local drive.

MANDATORY_AT_EACH_CYCLE



9

MANDATORY_AT_EACH_CYCLE

Functional Description

This program maintains cyclic communication with the local drive. To have cyclic communication with the local drive, this program has to be called once in each program cycle of a freewheel task.

With each call, the current values are exchanged between the ATV IMC controller and local drive.

Input Parameter

Name	Data Type	Description
xInitState	BOOL	FALSE = Enables data exchange between the local drive and the ATV IMC controller.

Output Parameter

Name	Data Type	Description
bError	BOOL	TRUE = Error occurred in communication with the local drive.

