



Aggregate Driver *Manual*

ASKOM[®] and **asix**[®] are registered trademarks of ASKOM Spółka z o.o., Gliwice. Other brand names, trademarks, and registered trademarks are the property of their respective holders.

All rights reserved including the right of reproduction in whole or in part in any form. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without prior written permission from the ASKOM.

ASKOM sp. z o. o. shall not be liable for any damages arising out of the use of information included in the publication content.

Copyright © 2005, ASKOM Sp. z o. o., Gliwice



ASKOM Sp. z o. o., ul. Józefa Sowińskiego 13, 44-121 Gliwice,
tel. +48 (0) 32 3018100, fax +48 (0) 32 3018101,
<http://www.askom.com.pl>, e-mail: office@askom.com.pl

1. Aggregate Driver

1.1. Driver Use

Aggregate driver allows definition of variables, values of which are generated as a result of calculations performed on other variables of **asix** system (source variables). To calculate the aggregate, the driver uses archived values of source **asix** variables. Using of archived values allows to prevent any discontinuities in the event of **asix** system restart.

1.2. Declaration of Transmission Channel

The syntax of declaration of transmission channel in ASMEN module section is as follows:

Channel_Name = AGGREGATE

where:

Channel_Name - channel name in ASMEN section.

1.3. Addressing the Process Variables

The address part of variable declaration takes the following form:

aggregate_name aggregate_parameters

where:

aggregate_name - name of aggregate;

aggregate_parameters - aggregate parameters, delimited with white space.

The driver may realize the following aggregates:

| Aggregate name | Way of calculations |
|-------------------|--|
| Average (Average) | weighted average of source variable in calculation period is obtained as a result; |
| Max (max) | maximum value of source variable in calculation period is obtained as a result; |
| Min (min) | minimum value of source variable in calculation period is obtained as a result. |

Parameters of above aggregates, take the following form:

*Variable_name:Archive_Type Period Threshold [A] [L[*lower_limit*]:[*upper_limit*]]*

where:

Variable_name - **NAME OF SOURCE VARIABLE WHICH IS CONNECTED WITH AGGREGATE**

Archive_type - one letter code determining the type of archive, in which source variable values are saved;

Period - calculation period of aggregate;

Threshold - minimum number of correct measures, in percentages, needed for aggregate calculations;

| | |
|--------------------|---|
| <i>A</i> | - it determines whether calculation performing time should be adjusted to calculation period; |
| <i>Lower_limit</i> | - lower limit value; if source variable value is lower than value of <i>lower_limit</i> , then <i>lower_limit</i> value is used instead; parameter may be used up to version 1.01.000 of the driver; |
| <i>Upper_limit</i> | - upper limit value; if source variable value is higher than value of <i>upper_limit</i> , then <i>upper_limit</i> value is used instead; parameter may be used up to version 1.01.000 of the driver; |

The *Period* parameter determines the calculation period. The calculation period is given in the same manner as time intervals specification for ASPAD module, ie. in form of numbers and units:

`<number><unit> [<number><unit> [...]]`

where:

| | |
|-----------------------------|---|
| <code><number></code> | - is quantity of given subsequently time units, |
| <code><unit></code> | - determines the time unit, which may be: |
| | s - second, |
| | m - minute, |
| | g lub h - hour, |
| | d - day (24 hours) . |

In case when unit is missing, the minute is taken as default unit of calculation period.

The result of aggregate calculations is said to be good, if quantity of correctly read samples (given in percentages) is equal to *Threshold*. Default value of *Period* parameter is 5 minutes, and *Threshold* parameter is 80 percentages. Correctness time of source variable value is taken into account for calculation of measurement correctness threshold, according to a parameterization of archiving of this variable (sampling period). It means, the threshold is calculated as a ratio of sum of all correctness times of measurements and calculation period length. In case of average, the calculation result is the average weighted in respect of correctness time of the measurement. Values of variables, for which time stamps are greater or equal to the beginning of the calculation period and lower than the end of it, are taken into consideration for aggregate calculation. Calculating the aggregate occurs at the end of calculation period.

The last, optional parameter *A* determines the time instant at which aggregate will be calculated. If the parameter is omitted, the aggregate will be calculated after each reading of source variable (in stepwise manner). If *A* is the last parameter, then time instant of aggregate calculation is adjusted to multiplicity of aggregate calculation period. Aggregate calculation result type is adjusted to the type of the conversion function given in a variable declaration.

1.4. EXAMPLE OF VARIABLE DECLARATION

```
Temp_sr, Temp-średnia, SREDNIA Temperatura:B 10 70 A, Srednie, 1, 1,
NOTHING_FP
```

The „Temperatura_sr” variable declared above is an average value of variable "Temperatura". The period, over which the variable was averaged, is 10 minutes and in order to obtain the correct value of average, at least 70 percent of correct measurements are needed. To create the average the archived values of "Temperatura" variable placed in Archive B are used. The time instant of aggregate calculation will be adjusted to

multiplicity of calculation period, i.e. calculations will be performed at 00:00:00, 00:10:00, 00:20:00 and so on.

| | | |
|-----------|---|----------|
| 1. | AGGREGATE DRIVER | 3 |
| 1.1. | DRIVER USE..... | 3 |
| 1.2. | DECLARATION OF TRANSMISSION CHANNEL | 3 |
| 1.3. | ADDRESSING THE PROCESS VARIABLES | 3 |
| 1.4. | EXAMPLE OF VARIABLE DECLARATION | 4 |