



(Driver 66)

DDHC Data File Driver

User's Manual



DRIVER ID 66 DOCUMENTATION NOTES

Der er siden kommet et nyere format: IGSSTAG, som giver mulighed for at du kan "nøjes" med at arbejde med IGSS objekt navne.

Dette format er i mine øjne noget mere brugervenligt at arbejde med. Nedenfor kan du se en beskrivelse af hvordan det skal sættes op, samt hvordan formatet på input filen skal være.

Ring endelig, hvis du har problemer med at få det "i gang".

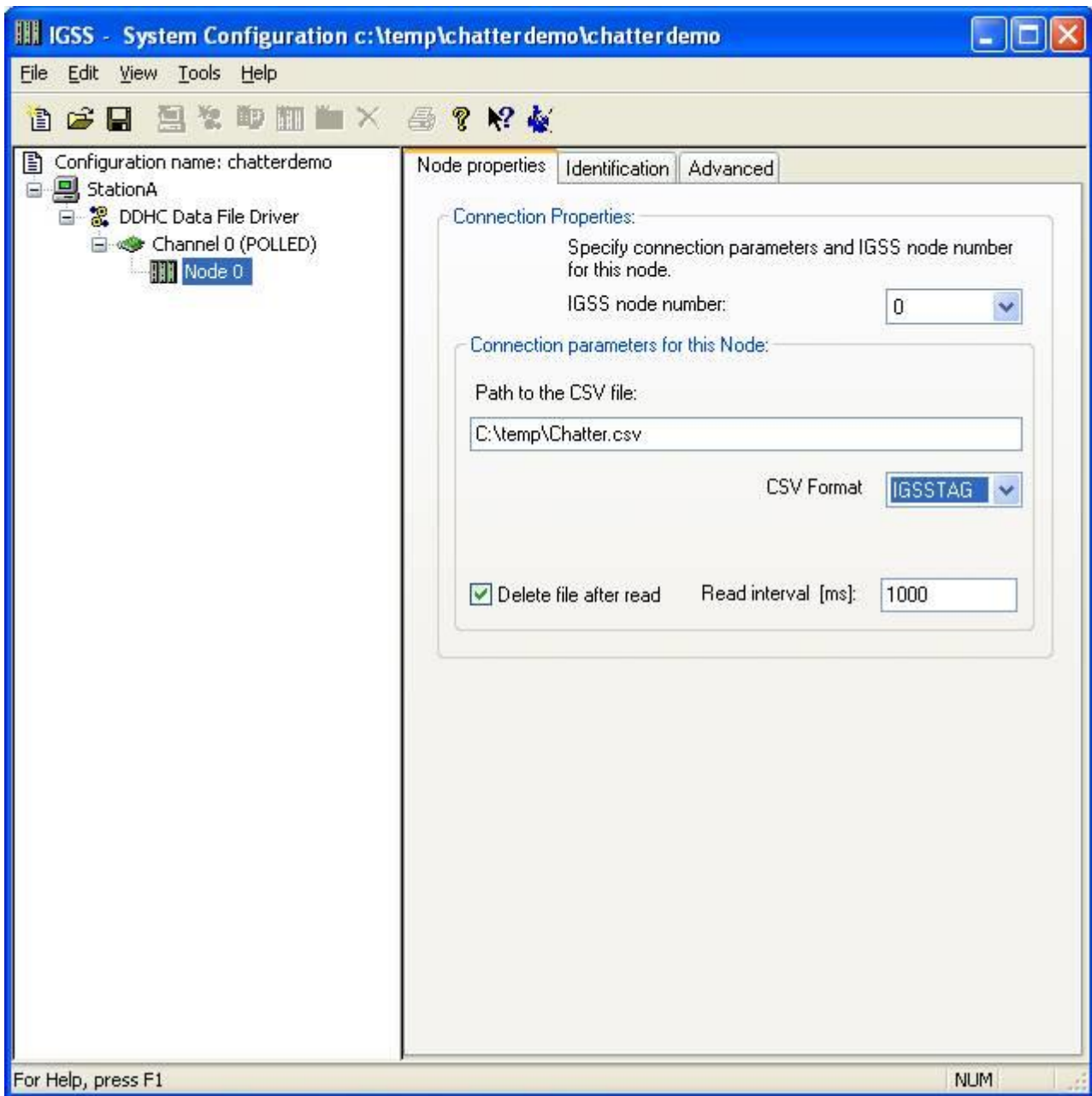
Med venlig hilsen

IGSS

Start med at oprette et nyt projekt. Til projektet tilknyttes driveren: 7TDHFILE:



På denne driver oprettes der en kanal, og under kanalen oprettes en node:



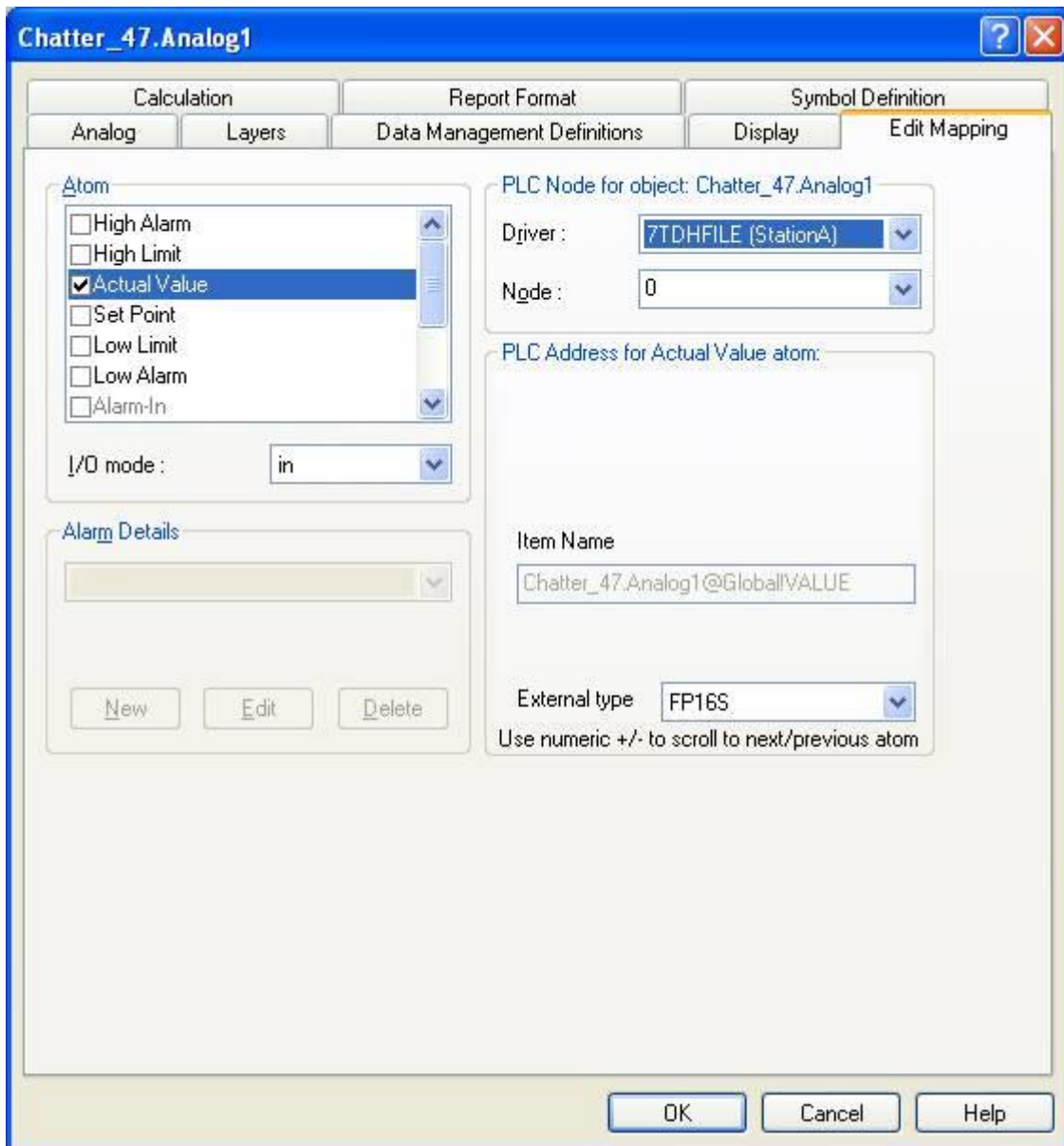
For hver node specificerer man navnet og formatet på .CSV filen, som driveren skal forsøge at læse, samt hvor ofte den skal forsøge at læse, og om filen skal slettes efter læsning.

Navnet på filen bestemmer i selvfølgelig selv. Der skal være læse/skrive adgang for den bruger der afvikler IGSS.

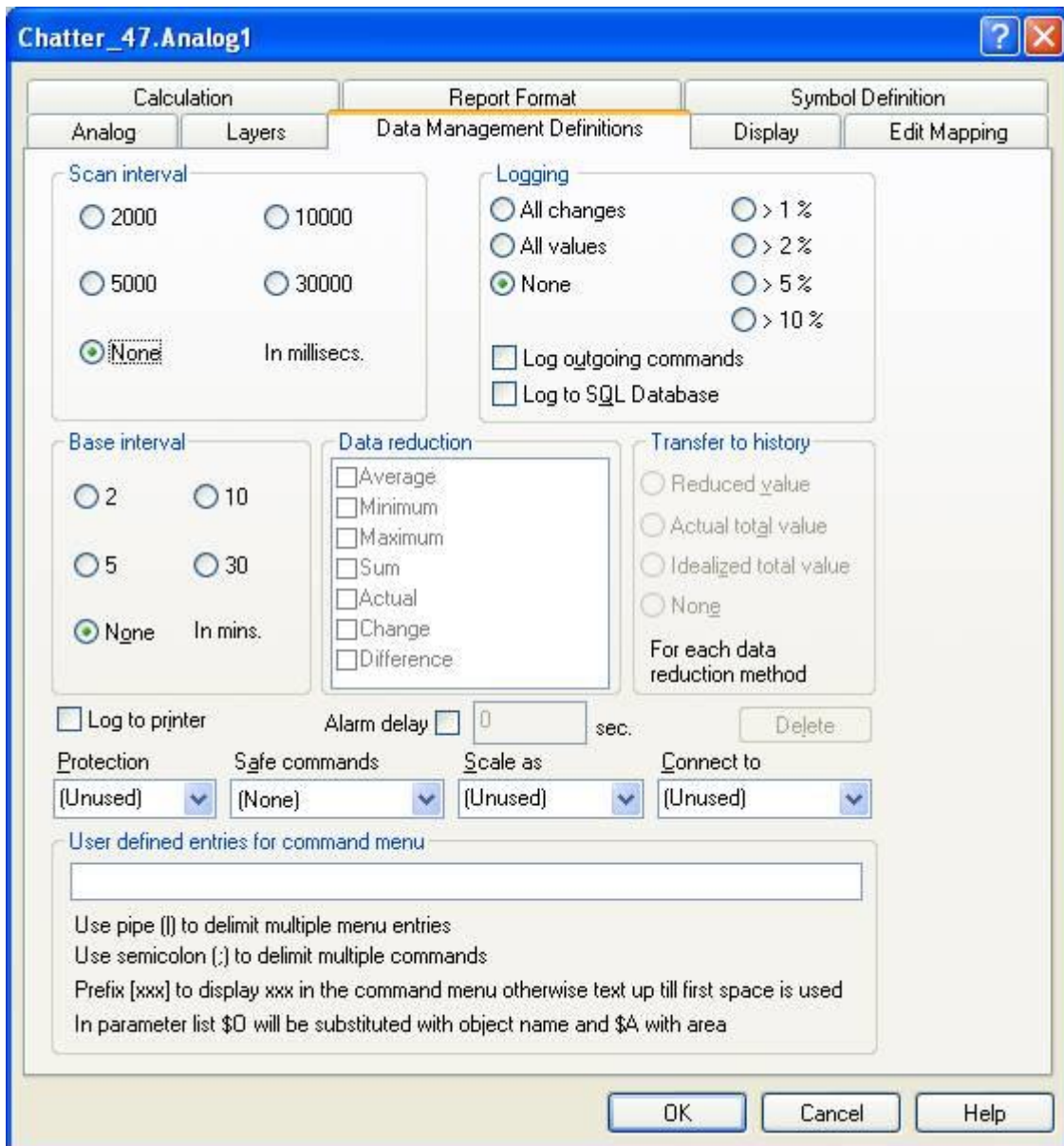
CSV format ***skal*** sættes til "IGSSTAG".

Dette er alt der skal gøres for opsætning af driveren. Nu starter vi definition og opretter et par objekter:

Her er vist et eksempel på et analogt objekt:



Navnet på objektet er "Chatter_47.Analog1". Der er kun defineret atom, nemlig "Aktuel Værdi" atomet. Til højre i dialogen vælger vi driver til 7TDHFILE og node til 0. Herefter vil feltet "Item Name" vise det navn, som vi om lidt skal bruge for at sende data ind til dette atom fra .CSV filen. I "Data Management Definitions" fanen skal scan interval stå til "None". Dette er default:



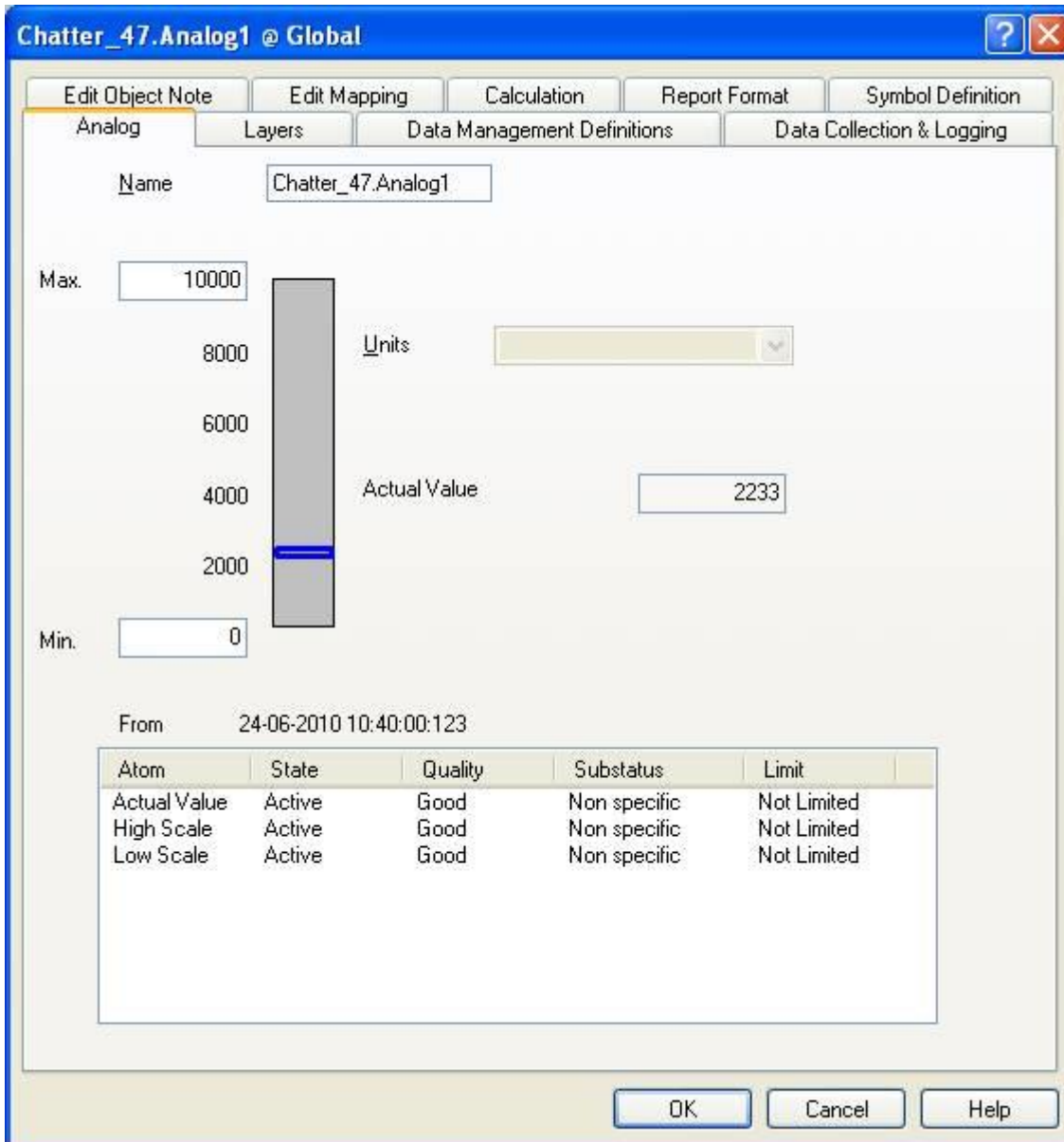
Vi installerer projektet (File->Install Configuration).

Herefter opretter vi .CSV filen. I dette eksempel hedder den: C:\temp\Chatter.csv

I denne fil skrives nu linien:

[Chatter_47.Analog1@Global!VALUE;2010-06-24](#) 08:40:00.123;2233;good

Vi starter IGSS projektet og gemmer filen i c:\temp\Chatter.csv og et øjeblik efter er værdien inde i systemet med tids stempel etc.:



Så langt så godt...

Syntax'en for .CSV er således:

```
// Syntax:
//
// <ObjectName>[@<AreaName>][!<AtomName>];<DateTime>;<Value>;<Quality>]
//
// Default for <AreaName> is "Global"
// Default for <AtomName> is "Value" for ANALOG, "State" for DIGITAL object types.
//
// Format for <DateTime> is in ANSI format:
//
// YYYY:MM:DD hh:mm:ss[.mmm]
//
// If <DateTime> is empty the current time is used.
//
// Format for <Value> uses point separation. Eg. 117.11 or just 118 for the value 118.000
```

```
//  
// For string objects the <Value> has to be a double quoted string e.g. "My String Value" or "This is a string"  
//  
// <Quality> is a string: "Good", "Bad" or "uncertain"  
//
```

Som det ses, er objektnavn obligatorisk, hvorimod område (AreaName) og atom (AtomName) er optionelle, såfremt man objektet ligger i området "Global", eller man skal sende værdier ind til default atomerne ("Value" for analoge, og "State" for digitale). I ovenfor viste eksempel kunne linien derfor lige så godt have set således ud:

```
Chatter_47.Analog1;2010-06-24 08:40:00.123;2233;good
```

Kvalitetsangivelsen af data er også optionel. Hvis denne udelades antager driveren, at kvaliteten er "good". Dvs. vi kan også udelade denne i vores eksempel.

```
Chatter_47.Analog1;2010-06-24 08:40:00.123;2233
```

Hvis vi også udelader tidsstempel, så vil driveren bruge PC'ens tid (nu-tid) som tidsstempel. Formatet for dette ser således ud:

Tidsstemplet angives ***ALTID*** i UTC tid, så der ikke er bøv! med sommer/vinter tid.

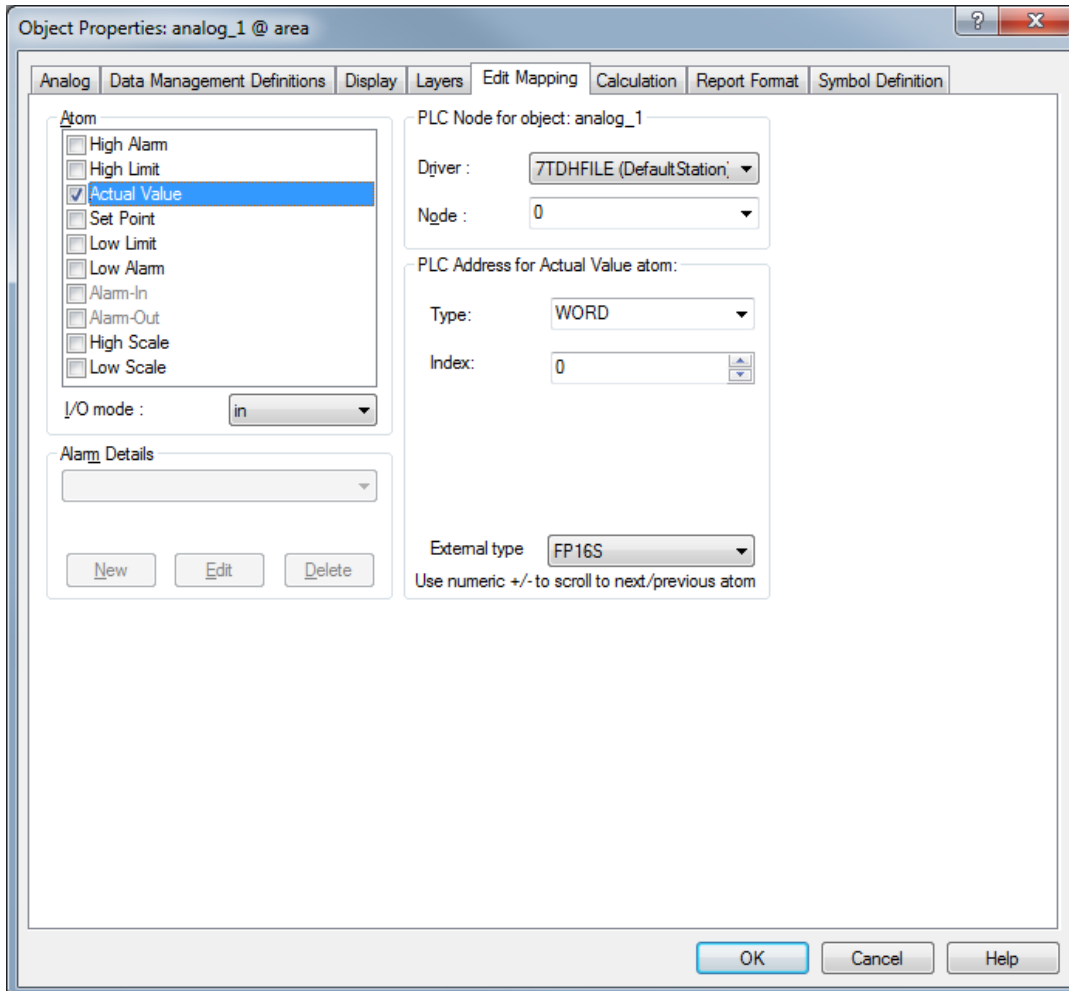
```
Chatter_47.Analog1;;2233
```

Hvis der skal decimaler på den analoge værdi (f.eks. 2233.345), da skal disse decimaler angives med punktum som decimal separator.

Digitale- og streng objekter behandles på nøjagtig samme måde.

INSTALLATION FOR .CSV FORMAT DH

Foregående installation er som ovenstående men hvor CSV format angives til DH. Specielt for DH er følgende:



Syntax'en for .CSV for DH er således:

```
// Syntax:  
//  
// <Type>,<Offset>,<Value>,<Date>,<Time>  
//  
// Format for <Type> are values 0 to 3 equal to WORD, DWORD, FLOAT and STRING  
//  
// Format for <Offset> is a value defining Index for the object in IGSS.  
//  
// Format for <Value> are regarding type 0 (WORD): 111, type 1 (DWORD): 1111111,  
// type 2 (FLOAT): 11.11 (or just 11 for 11.00) and type 3 (STRING): "text info"  
//  
// Format for Date is following  
//  
// DD-MMM-YYYY where MMM is Danish or English spelling JAN, FEB, MAR etc.  
//  
// Format for Time is hh:mm:ss.mmmm  
//
```


For ovenstående IGSS objekt med Index 0 er vist type 0 WORD.

Alle 5 objekter <Type>, <Offset>, <Value>, <Date> og <Time> er obligatoriske.

Tidsstempet angives i normaltid, og driver for DH omregner selv til UTC tid. Der kan anvendes typer WORD, DWORD, FLOAT og STRING.

Decimaler angives ved FLOAT med punktum som decimal separator.

Eksempel på indhold af .CSV for type DH:

```
0, 0, 28859, 14-SEP-2004, 12:22:59.156
1, 0, 288474443, 14-SEP-2004, 12:22:59.156
2, 0, 247.3, 14-SEP-2004, 12:22:59.156
2, 24, 247.4, 14-SEP-2004, 12:22:59.156
2, 23, 231.5, 14-SEP-2004, 13:32:59.15
2, 22, 220.75, 14-SEP-2004, 13:32:59.15
2, 39, 14.7041, 14-SEP-2004, 13:32:59.15
3, 0, "UA521", 27-AUG-2009, 11:15:01.123
3, 30, "UA534", 27-AUG-2009, 11:15:01.123
```

Forklaring på eksempel:

Første værdi for db:0 woffs:0 (WORD) er 28859 dato 14-SEP-2004 kl. 12:22:59.156.

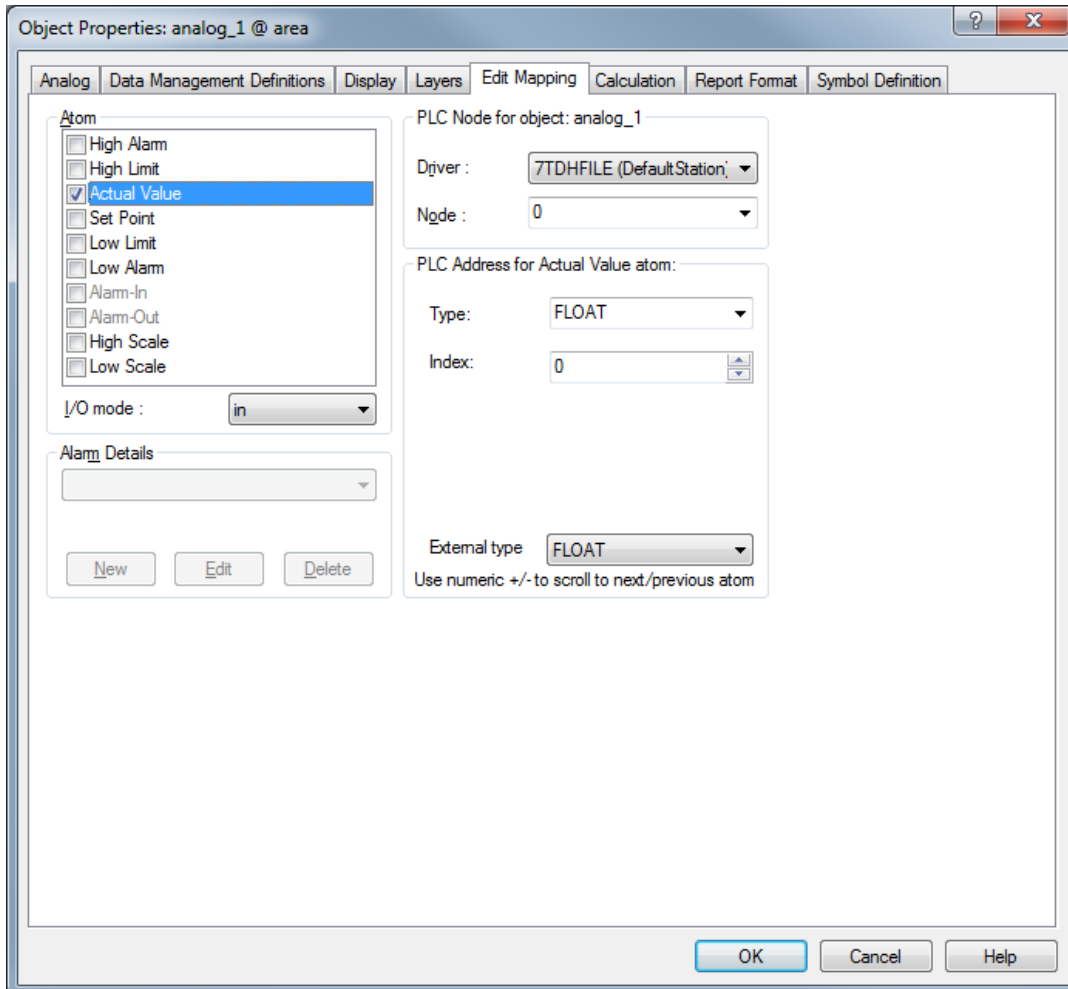
Anden værdi for db:1 woffs:0 (DWORD) er 288474443 dato 14-SEP-2004 kl. 12:22:59.156.

Tredje værdi for db:2 woffs:0 (FLOAT) er 247.3 dato 14-SEP-2004 kl. 12:22:59.156.

Ottende værdi for db:3 woffs:0 (STRING) er "UA521" dato 27-AUG-2009 kl. 11:15:01.123.

INSTALLATION FOR .CSV FORMAT WI

Foregående installation er som ovenstående, men hvor CSV format angives til WI. Specielt for WI er følgende:



Syntax'en for .CSV for WI er således:

```
// Syntax:
//
// <Object>[,<Date>,<Time>,<Seconds>]
//
// Formats for <Object> are following
//
// 'time' for setting timestamp followed by <Date><Time><Seconds>
//
// floating value for IGSS object db:2 woffs:0 (only float object). Timestamp increases 15 minutes after
// each floating value.
//
// NO_DATA for increasing timestamp 15 minutes
//
// 'ch' for character following
//
// 'text' for text following
//
```

```
// Format for Date is DD-MM-YYYY
//
// Format for Time is hh:mm
//
// Format for Seconds ss[.mmm]
//
// 'time' has to set in .CSV before driver for WI accepts input values
//
```

For ovenstående IGSS objekt med Index 0 er vist FLOAT, og dette kan kun anvendes ved .csv format WI.

Objektet <Object> er altid obligatorisk, hvorimod objekterne <Date>, <Time> og <Seconds> kun er krævet ved objektet 'time'.

Tidsstempet 'time' angives i normaltid, og driver for WI omregner selv til UTC tid.

Der anvendes automatisk et tidsinterval på 15 minutter mellem hver værdi. Ønskes kortere/længere tidsinterval skal tidstempel sendes lige før hver værdi.

Decimaler angives med punktum som decimal separator.

Eksempel på indhold af .CSV for type WI:

```
'text','Title:Water Industry Standard Data File'
'text','Site: hagelbiergvei'
'ch',1,'Flow','l/s'
'time',19/12/2006,07:15,15.123,'Minute',5952
NO_DATA
NO_DATA
NO_DATA
NO_DATA
291.796.875
-0.01953125
-0.05859375
0.02734375
0
0
```

Forklaring på eksempel:

Første 3 linier 'ch', 1 og 'text','Title:Water Industry Standard Data File' er dummy fyld som kommentarer, og som ikke bliver anvendt for type WI.

For linie 4 sættes timestamp til 19/12/2006 kl 07:15:15.123 (udelades .123 sættes denne af driver til .000).

Følgende 4 gange NO_DATA rykker timestamp frem til dato 19/12/2006 kl. 08:15:15.123

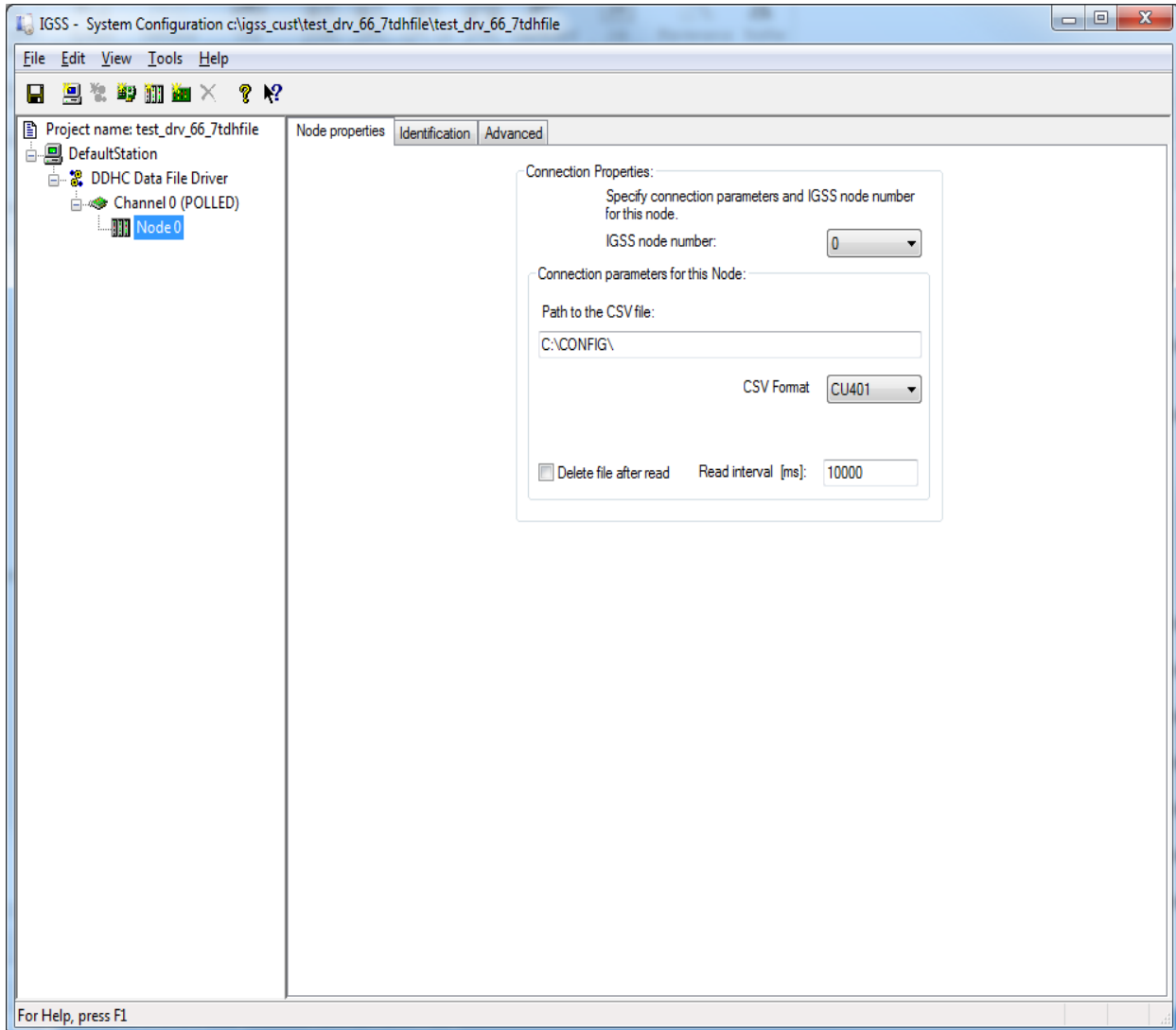
Første værdi for db:2 woffs:0 er 291.796 dato 19/12/2006 kl. 08:15:15.123.

Anden værdi for db:2 woffs:0 er -0.01953125 dato 19/12/2006 kl. 08:30:15.123.

Tredje værdi for db:2 woffs:0 er -0.05859375 dato 19/12/2006 kl. 08:45:15.123.

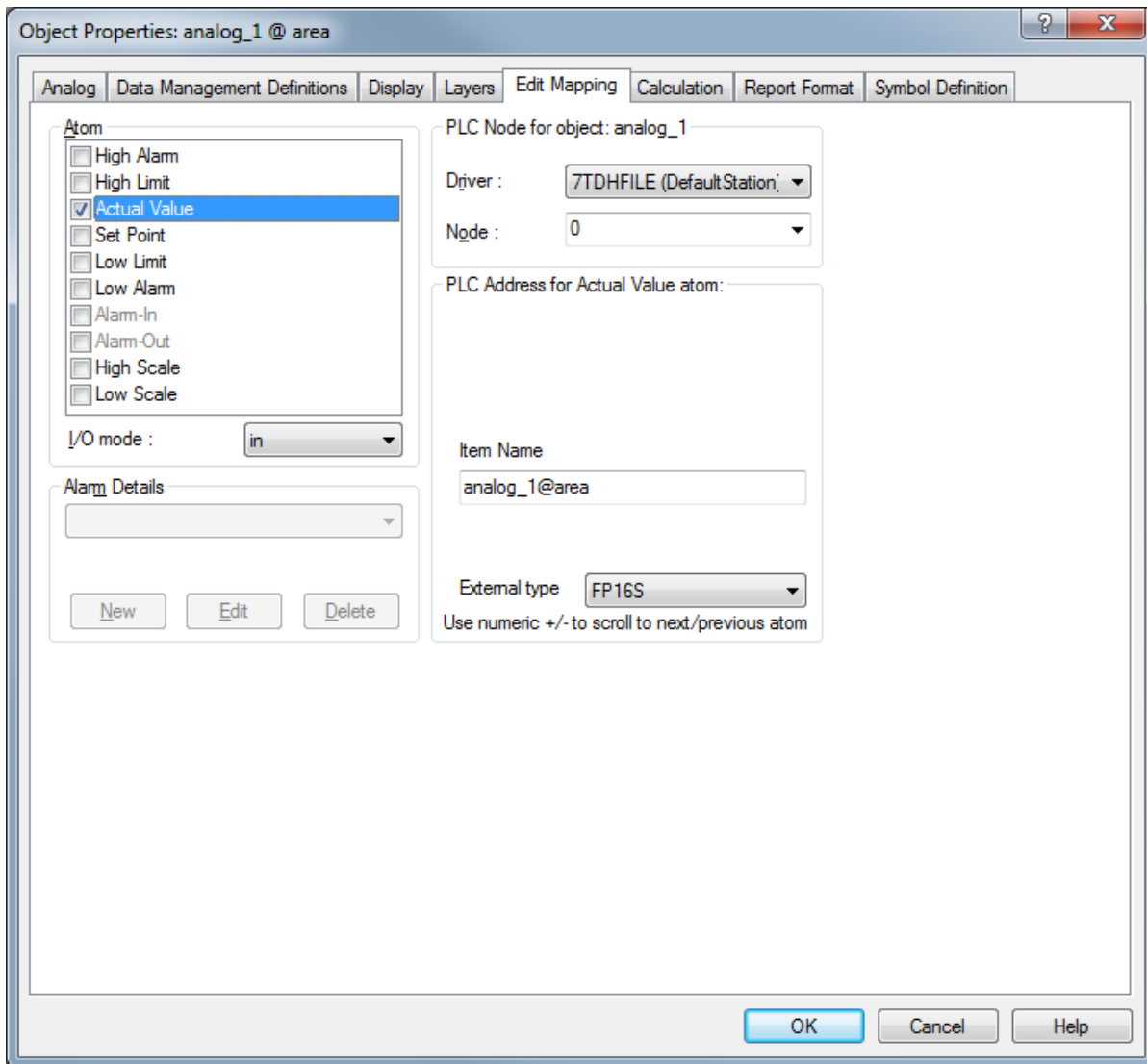
INSTALLATION FOR .CSV FORMAT CU401

Foregående installation er som ovenstående, men hvor CSV format angives til CU401, og der er ikke angivet noget filnavn. Specielt for CU401 er følgende:



For hver node angives placeringen af stien for filerne, som driver 66 ved csv format CU401 anvender. Formatet på filnavnet skal være 20130724.csv, dvs. år, måned og dag.

Bemærk, at de 2 første linier i hver fil kan anvendes til kommentarer og springes over af driver 66 ved CSV format CU401.



Syntax'en for .CSV for CU401 er således:

```
// Syntax:
//
// <ObjectName>,<DateTime>,<Value>,<Quality>
//
//
// Format for <DateTime> is in ANSI format:
//
// YYYY:MM:DD hh:mm:ss[.mmm]
//
// Format for <Value> uses dword 11111 or float with point separation 118.11 or
// just 118 for the value 118.000
//
// <Quality> is a string specified as "good". Driver only uses quality good
//
```

For ovenstående er vist IGSS objekt analog_1@area.

Alle 4 objekter <ObjectName>, <DateTime>, <Value> og <Quality> er obligatoriske. Tidsstempet angives i normaltids, og driver for CU401 omregner selv til UTC tid.

Tidsstempolet angives i normaltids, og driver for CU401 omregner selv til UTC tid

Eksempel på indhold af .CSV for type CU401

```
information 1  
information 2  
analog_1@area,2013-07-25 16:35:27.123,23456,good  
analog_1@area,2013-07-25 16:38:42.473,78901,good
```

Forklaring på eksempel:

De 2 første linier er for information om indholdet af .CSV filen og anvendes ikke af driver. Bemærk driver anvender ikke de 2 første linier i .CSV filen for type CU401.

Hvis der skal decimaler på den analoge værdi (f.eks. 23456.89), skal disse angives med punktum som decimal separator.

Driver 66 for .csv format CU401 anvender for <Quality> kun værdien good, men denne skal angives.