



CRON Time Based Schedule Trigger

DT80 Firmware V9.00
May 2011



CONTENTS

CRON - Time Based Schedule Trigger 3

 Syntax 3

 Valid data ranges..... 4

 Examples..... 4

 Example 1 - Trigger schedule at 9:00:00 4

 Example 2 - Trigger schedule every minute between 9:00: and 17:00..... 5

 Example 3 - Trigger schedule every minute between 9:00 and 17:00 Monday to Friday 5

 Example 4 - Trigger schedule 10 seconds before the minute 5

 Example 5 - Trigger schedule at the end of each working week 6

 Example 6 - Trigger schedule at the end of each month..... 6

 Example 7 – Block/range sampling 1 6

 Example 8 – Block/range sampling 2 7

 Example 9 - List sampling 1 7

 Example 10 - List sampling 2 7

 Example 11 - List of ranges 7

 Example 12 – Interval sampling 1 8

 Example 13 - Interval sampling 2 8

 Example 14 - Interval sampling 3 8

 Error messages 9

 E148 - Time trigger - invalid characters in trigger 9

 E149 - Time trigger - one or more trigger fields overrange 9

 E150 - Time trigger - illegal extra characters in one or more fields..... 9

 E151 - Time trigger - 'skip' value overrange in one or more fields 10

 E152 - Time trigger - invalid characters after '/' in one or more fields..... 10



CRON - Time Based Schedule Trigger

With the release of Version 9 firmware for the DT80 is a new time based schedule trigger that is based on the Unix CRON time scheduler. This powerful new schedule trigger type will greatly simplify many existing tasks and allow a greater flexibility in how and when a schedule is triggered.

CRON allows the user to trigger a schedule:

- At a specified time of day (e.g. At 9:00:00)
- On a particular day of the month (e.g. First day of the month)
- Run between specified hours in the day (e.g. between the hours of 9am to 5 pm)
- Run between specified days of the week (e.g. Between Monday and Friday)
- Run at a specified time before the minute or hour. (e.g 1 minute before the hour)
- Run at times listed (e.g. 1 am, 2 am, 4 am, 8 am etc.)
- Run at time built from the combinations above.

Syntax

`RA"Name"(<Store File>)[Sec:Min:Hr:Day:Month:Day of Week]`

Where:

- `a` = Schedule identifier
- `Name` = Schedule name
- `<Store File>` = Store file definition
- `[` = Start of CRON expression
- `Sec` = Second of the minute
- `Min` = Minute of the hour
- `Hr` = Hour of the day
- `Day` = Day of the month
- `Month` = Month of the Year
- `Day of Week` = Day of the week
- `]` = End of CRON expression

e.g.

`RA"Schedule_1" ("b:", ALARMS:OV:100KB:W60, DATA:OV:1MB) [*:*:9-17:*:*:1-5]`

This example will sample once per second between the hours of 9 and to 5 pm Monday to Friday.



Valid data ranges

Each field in the Colon (:) separated list has a range of valid inputs

- Second of the minute 0 - 59
- Minute of the hour 0 - 59
- Hour of the day 0 - 23
- Day of the month 1 - 31
- Month of the year 1 - 12
- Day of the Week 0 - 7 Note: Sunday is 0 or 7

Each field can also accept:

- A list of values 1,2,4,8,16
- A range of values 9-17
- A list of ranges 0-4,8-12
- All in the range *
- By steps of */2 or 0-23/2

Notes:

- Fewer than 6 fields may be specified. e.g. [0:9] is identical to [0:9::*:*] and means 9 minutes and 0 seconds past the hour, every hour, every day of the month, every month and every day of the week.
- Fields may not be empty. e.g. [0:9::*:*] is not allowed.
- Must use 24 hour clock. e.g. 9 pm is not allowed, use 21
- Relative time addressing not allowed.
- The step size is up to the maximum range allowed for field type.
- Times are synchronized to midnight by the DT80 internal clock.

Examples

Example 1 - Trigger schedule at 9:00:00

Returns the daily maximum, minimum and average of the air temperature at 9:00:00 every day for a PT100 RTD sampled every 1 minute.

CRON expression:

Schedule A - [0:0:9]

```

BEGIN"9AMRPT"
RS1M      'SET STATISTICAL SAMPLE RATE TO ONCE PER MINUTE.
RA[0:0:9] 'SET SCHEDULE TO SCAN AT 9AM EVERY DAY
1PT385 ("AIRTMP~DEGC",MX) ("AIRTMP~DEGC",MN) ("AIR TEMP~DEGC",AV)
END

```



Example 2 - Trigger schedule every minute between 9:00: and 17:00

Sample a current loop device once per minute between the hours of 9 am to 5 pm seven days a week

CRON expression:

Schedule A - [0:*:9-17]

```

BEGIN"9TO5"
RA[0:*:9-17] 'TRIGGER ON ZERO SECOND, EVERY MINUTE BETWEEN HOURS 9 TO 17
  1L("DEMAND~%")
END

```

Example 3 - Trigger schedule every minute between 9:00 and 17:00 Monday to Friday

Sample a current loop device once per minute between the hours of 9 am to 5 pm Monday to Friday.

CRON expression:

Schedule A - [0:*:9-17:*:1-5]

```

BEGIN"9TO5WEEK"
RA[0:*:9-17:*:1-5]
  1L("DEMAND~%")
END

```

Example 4 - Trigger schedule 10 seconds before the minute

Uses schedule A to turn on the power to a 4 to 20 mA sensor 10 seconds before the minute. Schedule B then reads the sensor then turns of the power. This gives the sensor 10 seconds of warm up time before being read.

CRON expressions:

Schedule A - [50]
(Run at 50 seconds into the minute)

Schedule B - [0]
(Run on the minute)

```

BEGIN"PWRON"
RA[50] 'RUN ON SECOND #50
  1SSPWR=1 'TURN ON THE POWE
RB[0] 'RUN ON SECOND #0
  1L("FLOW RATE~%") 'READ THE SENSOR
  1SSPWR=0 'TURN OFF THE POWER
END

```



Example 5 - Trigger schedule at the end of each working week

Sample a current loop device once per minute between the hours of 9 am to 5 pm, Monday to Friday. On Sunday a weekly report is generated at midnight Sunday morning then the weekly tally is reset

CRON expressions:

Schedule A - [0:*:9-17:*:*:1-5]
(Run 9am to 5 pm Monday to Friday)

Schedule B - [0:0:0:*:*:0]
(Run report Midnight Sunday morning)

```
BEGIN"WKRPT"  
RA[0:*:9-17:*:*:1-5]  
    1L ("KW", +=1CV)  
RB[0:0:0:*:*:0]  
    1CV ("WEEK TOTAL~KW", R)  
END
```

Example 6 - Trigger schedule at the end of each month

Sample a current loop device once per minute between the hours of 9 am to 5 pm, Monday to Friday. A monthly report is generated at midnight on the first of each month then the monthly tally is reset

CRON expressions:

Schedule A - [0:*:9-17:*:*:1-5]
(Run 9am to 5 pm Monday to Friday)

Schedule B - [0:0:0:1]
(Run report at Midnight 1st of month)

```
BEGIN"MTHRPT"  
RA[0:*:9-17:*:*:1-5]  
    1L ("KW", +=1CV)  
RB[0:0:0:1]  
    1CV ("MONTH_TOTAL~KW", R)  
END
```

Example 7 – Block/range sampling 1

Sample a current loop device every second for the first 15 seconds of every minute.

CRON expressions:

Schedule A - [1-15]
(Sample second in the range of 1 to 15)

```
BEGIN"15SEC"  
RA[1-15]  
    1L ("KW", +=1CV)  
END
```



Example 8 – Block/range sampling 2

Sample a current loop device every second for the first 15 minutes of every hour.

CRON expressions:

Schedule A - [*:1-15]

(Sample every second of each minute in the range of 1 to 15)

```
BEGIN"15MIN"  
RA[*:1-15]  
  1L("FLOW", +=1CV)  
END
```

Example 9 - List sampling 1

Sample a current loop device on the minute for each minute listed.

CRON expression:

Schedule A - [0:1,2,5,10,20,40]

(Sample at minutes 1, 2, 5, 10, 20 and 40 every hour)

```
BEGIN"MINLIST"  
RA[0:1,2,5,10,20,40]  
  1L("CONSOLIDATION~MM")  
END
```

Example 10 - List sampling 2

Sample a current loop device on the minute for each hour listed.

CRON expression:

Schedule A - [0:0:0,1,2,4,8,16]

(Sample at hours 1, 2, 4, 8 and 16)

```
BEGIN"HRLIST2"  
RA[0:0:0:1,2,5,10,20,40]  
  1L("CONSOLIDATION~MM")  
END
```

Example 11 - List of ranges

Sample a current loop device on the minute for each minute listed.

CRON expression:

Schedule A - [0*:1-3,6-9,12-15,18-21]

(Sample every minute between 1 am to 3 am, 6 am to 9 am, noon to 3pm and 6pm to 9pm every day)

```
BEGIN"HRRNG1"  
RA[0*:1-3,6-9,12-15,18-21]  
  1L("CONSOLIDATION~MM")  
END
```



Example 12 – Interval sampling 1

Sample temperature sensor every second minute

CRON expression:

Schedule A - [0:*/2]

(Sample every second minute. e.g. at minutes 0, 2, 4, 6, 8, 10 etc.)

BEGIN "2MIN"

RA[0:*/2]

PT385 ("TEMPERATURE")

END

Example 13 - Interval sampling 2

Sample temperature sensor 30 seconds past the minute every second minute

CRON expression:

Schedule A - [30:*/2]

(Samples at 00:00:30, 00:02:30, 00:04:30 etc.)

BEGIN "2M30S"

RA[30:*/2]

PT385 ("TEMPERATURE")

END

Example 14 - Interval sampling 3

Sample load cell on the minute every 6 hours

CRON expression:

Schedule A - [0:0:*/6]

(Samples at 00:00:00, 06:00:00, 12:00:00 and 18:00:00 every day)

BEGIN "6HR"

RA[0:0:*/6]

1BGI ("LOAD~PPM")

END



Error messages

E148 - Time trigger - invalid characters in trigger

Example 15. The code:

```
BEGIN"E148"  
RA[*:*:*:*:JUNE]  
    REFT  
END
```

Will return the message:

```
E148 - Time trigger - invalid characters in trigger at line 2 col  
3 [ ra<err>[*:*:*:*:J ]
```

Error due to the month being named. Only integer numbers are allowed in CRON schedule trigger definition.

E149 - Time trigger - one or more trigger fields overrange

Example 16. The code:

```
BEGIN"E149"  
RA[60:*:*:*]  
    REFT  
END
```

Will return the message:

```
E149 - Time trigger - one or more trigger fields overrange at line  
2 col 3 [ ra<err>[60:*:*:*] ]
```

Error due to the number of seconds entered as 60. Seconds' field valid input range is 0 to 59

E150 - Time trigger - illegal extra characters in one or more fields

Example 17. The code:

```
BEGIN"E150"  
RA[2s:*:*:*]  
    REFT  
END
```

Will return the message:

```
E150 - Time trigger - illegal extra characters in one or more  
fields at line 2 col 3 [ ra<err>[2s:*:*:] ]
```

Error due to the letter "s" after the seconds' definition. No characters are allowed.



E151 - Time trigger - 'skip' value overrange in one or more fields

Example 4. The code:

```
BEGIN"E151"  
RA[*/90:*:*:*]  
    REFT  
END
```

Will return the message:

E151 - Time trigger - 'skip' value overrange in one or more fields at line 2 col 3 [ra<err>[*/90:*:*:]

Error due to the number of seconds to skip exceeding the valid range of inputs for the seconds field (0 to 59).

E152 - Time trigger - invalid characters after '/' in one or more fields

Example 5. The code:

```
BEGIN"E152"  
RA[*/-9:*:*:*]  
    REFT  
END
```

Will return the message:

E152 - Time trigger - invalid characters after '/' in one or more fields at column 3 [ra<err>[*/-90:*:*]

Error due to the - character after the / in the seconds. Relative time addresses are not allowed.