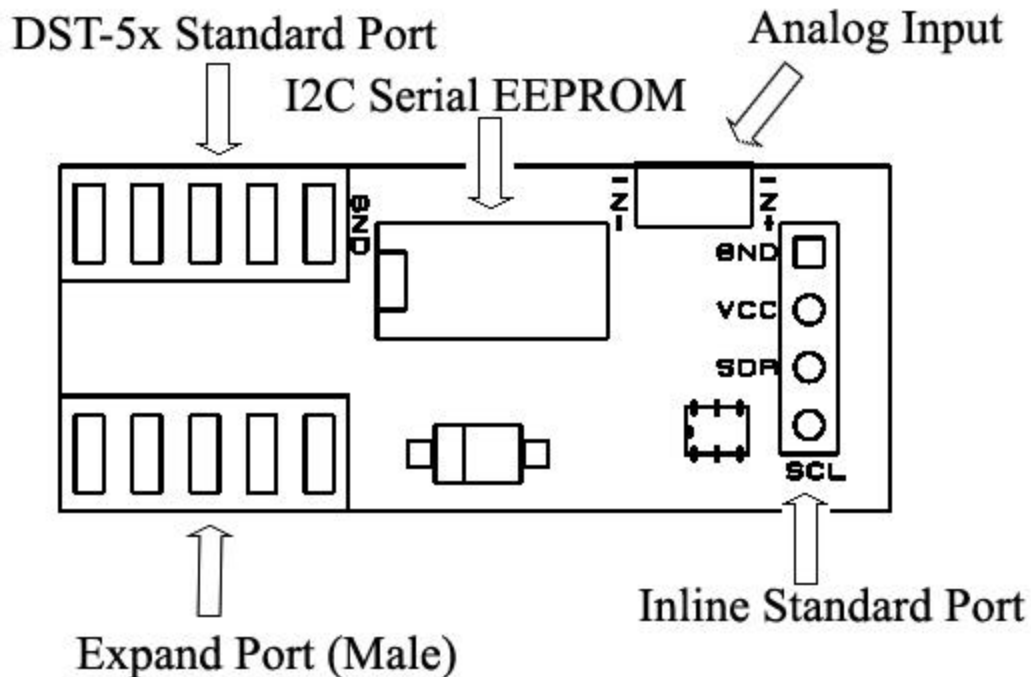


## DQI-01 DELTA DATA ACQUISITION INTERFACE 16 BIT ANALOG TO DIGITAL CONVERTER

DQI-01 adalah merupakan salah satu dari Data Acquisition Interface produksi Delta Electronic yang berfungsi sebagai Konverter Analog to Digital dengan resolusi 16 Bit. DQI-01 menggunakan teknik antar muka I2C sehingga hanya diperlukan 2 jalur I/O saja untuk mengakses modul ini yaitu melalui SDA dan SCL. Pada modul ini tersedia DST-5x Standard Port yang digunakan apabila modul ini terhubung pada Modul DST-51 ataupun DST-52 sedangkan Inline Standard Port digunakan apabila modul ini terhubung ke Modul ST-51, DST-R8C, ST-2051 dan modul-modul Sistem Mikrokontroler dari Delta Electronic lain yang menggunakan Inline Standard Port.



### Spesifikasi:

- DST-5x Standard Port
- Inline Standard Port
- 1Kb I2C Serial EEPROM Include
- Continuous Self Calibration
- Low Noise: 4uVp-p
- Programmable Gain Amplifier

### Pengalaman Modul DQI-01

Modul DQI-01 mempunyai kemampuan untuk diparalel hingga 8 modul dalam satu jalur bus I2C. Agar modul ini dapat diparalel maka alamat-alamat dari IC ADC yang ada pada modul ini harus diatur saling berbeda antara yang satu dengan yang lain, mulai dari AD0, AD1, AD2, AD3, AD4, AD5, AD6 dan AD7. Hal ini dapat dilihat pada IC 6 pin yang ada pada modul ini.

Kode IC	I2C Address
AD0	1001 000x
AD1	1001 001x
AD2	1001 010x
AD3	1001 011x
AD4	1001 100x
AD5	1001 101x
AD6	1001 110x
AD7	1001 111x

Setiap modul dialamati dengan sebuah kode yang disebut I2C Address. Pada I2C Address ini juga terdapat 1 bit pada bagian LSB yang berfungsi untuk membedakan mode yang sedang dilakukan adalah merupakan mode Write = 0 atau Mode Read = 1. 4 bit, yaitu bit 7 hingga bit 4 dari I2C Address adalah merupakan alamat IC ADC tersebut dalam jajaran komponen-komponen I2C sedangkan bit 3 hingga bit 1 adalah merupakan alamat dari IC ADC yang digunakan.

## Register Config DQI-01

BIT	7	6	5	4	3	2	1	0
NAME	ST/BSY	0	0	SC	DR1	DR0	PGA1	PGA0

ST/BSY: Menuliskan logika 1 pada bit ini akan mengakibatkan proses konversi dimulai. Pada mode read, bit ini akan berlogika 1 apabila proses konversi ADC belum selesai, dan akan berlogika 0 pada saat konversi berakhir.

SC: Logika 1 pada bit ini menandakan single conversion dan logika 0 menandakan bahwa mode yang digunakan adalah continuous conversion

DR1 dan DR0:

DR1	DR0	DATA RATE
0	0	128SPS
0	1	32SPS
1	0	16SPS
1 <sup>(1)</sup>	1 <sup>(1)</sup>	8SPS <sup>(1)</sup>

NOTE: (1) Default Setting.

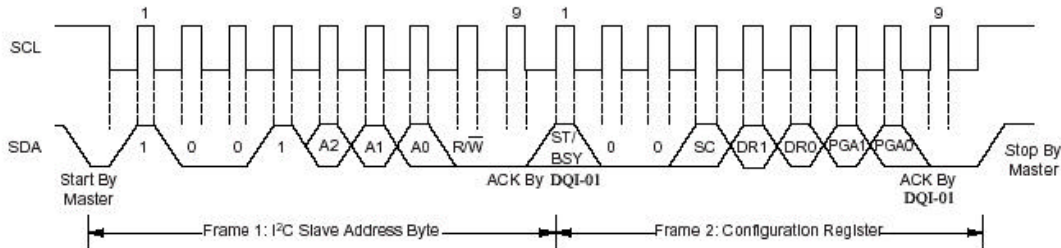
PGA1 dan PGA0:

PGA1	PGA0	GAIN
0 <sup>(1)</sup>	0 <sup>(1)</sup>	1 <sup>(1)</sup>
0	1	2
1	0	4
1	1	8

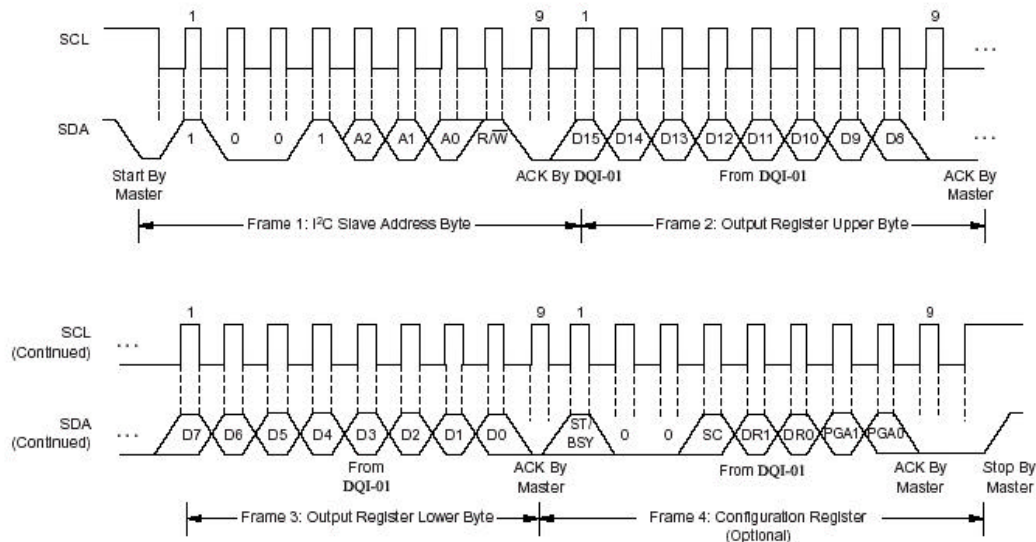
NOTE: (1) Default Setting.

## Akses DQI-01

Akses ke DQI-01 terdiri dari 2 macam yaitu Penulisan Data dan Pembacaan Data. Proses penulisan data pada modul ini adalah penulisan data ke Register Config untuk mengatur mode kerja dari Modul DQI-01.



Proses penulisan data diawali dengan mengirimkan I2C Slave Address dan dilanjutkan dengan data berupa register config serta diakhiri dengan stop bit. Proses pembacaan data pada Modul DQI-01 adalah merupakan proses untuk membaca nilai tegangan analog yang telah dikonversi menjadi data digital. Proses ini diawali dengan start bit dari Master dan dilanjutkan dengan I2C Slave Address Byte. Modul DQI-01 akan membalas dengan ACK dan diikuti dengan Upper byte dan lower byte dari data yang dikonversi serta diakhiri dengan isi dari nilai register config. Proses ditutup dengan sebuah stop bit dari Master.



```

;*****
; Program Akses DQI-01 melalui Port 1
; oleh Modul DST-52 BIOS Versi 2.7
.DATA

SDA    Bit    P1.0
SCL    Bit    P1.1

Delay_1detik    EQU    0090H
ASCII_Out    EQU    00CBH
Enter_Code    EQU    00DFH

DQIAddress    EQU    90H
Buffer    EQU    50H
ROM    EQU    2000H
.Code
Org    ROM
;Reset Vector
Ajmp    Start    ;
Org    ROM+3H
;External Interrupt 0 Vector
Reti

Org    ROM+0BH
;Timer 0 Interrupt Vector
Reti
Org    ROM+13H
;External Interrupt 1 Vector
Reti    ;
Org    ROM+1BH
;Timer 1 Interrupt Vector
Reti    ;
Org    ROM+23H
;Serial Interrupt Vector
Reti    ;

Start:
Mov    R7,#DQIAddress
;Set DQI Slave Address
Mov    R4,#01
;Set DQI Device Address
Mov    DataSEE,#9CH
;Set Data yang ditulis ke Config
Lcall KirimDataI2C    ;

Loop:
Mov    R0,#Buffer
;Tentukan alamat buffer read
Mov    R5,#3
;Tentukan jumlah data yang dibaca
Mov    R4,#01
;Tentukan Dev Address
Mov    R7,#90H
;Tentukan Slave Address
Lcall BacaDataI2C    ;

Mov    R0,#Buffer
;Display data di buffer
Mov    R5,#3    ;
LoopX:
Mov    A,@R0    ;
Inc    R0    ;
Lcall ASCII_Out    ;
Djnz    R5,LoopX    ;

Lcall Delay_1detik
Lcall Enter_Code
Ajmp    Loop

.Code
Org    $
Slave_AddrSEE Ds    1

DataSEEDs    1

.Ambil_Ack:
Clr    C
Setb    SDA
Setb    SCL
Mov    C,SDA
Clr    SCL
Ret

BacaDataSEE:
Push    B
Mov    B,#08H
Clr    A

LoopBacaSEE16b:
Push    B
Rl    A
Setb    SDA
Setb    SCL
Clr    C
Mov    C,SDA
Mov    A.0,C
Clr    SCL
Pop    B
Djnz    B,LoopBacaSEE16b
Pop    B
Ret

KirimDataSEE:
Push    B
Mov    B,#8

Send8_bitloop
Rlc    A
Mov    SDA,C
Lcall Pulse_SEE
Djnz    B,Send8_bitloop
Pop    B
Clr    C
Lcall Ambil_Ack
Ret

Pulse_SEE:
Push    b
Setb    SCL
Clr    SCL
Pop    B
Ret

Buat_StartBit:
Setb    SDA
Setb    SCL
Clr    SDA
Clr    SCL
Ret

Buat_StopBit:
Clr    SDA
Setb    SCL
Setb    SDA
Clr    SCL
Ret

SendACK:
Clr    SDA
Lcall Pulse_SEE
Setb    SDA

```

```

Ret

Ret

PageReadSEE8b:
Lcall  Buat_StartBit
Mov    A,R7
;Read Mode
Setb   A.0 ;
Lcall  KirimDataSEE ;
Lcall  BacaDataSEE
Mov    DPH,A
Lcall  SendACK
Lcall  BacaDataSEE
Mov    DPL,A
Lcall  SendACK
Lcall  BacaDataSEE
Lcall  Buat_StopBit
Ret

;*****
;*****
; BACA DATA I2C
; - R5 = Jumlah data yang dibaca
; - R7 = DDI Slave Address
; - R4 = DDI Device Address
; - R0 = Alamat awal buffer

BacaDataI2C:
Acall  BuatSlaveDevAddress
;Akses ke Modul DDI-01 Dev
Address 01
Setb   A.0 ;Mode Read
Lcall  Buat_Startbit ;
Lcall  KirimDataSEE ;

LoopPageRead:
Lcall  BacaDataSEE ;Baca data
SEE
Clr    SDA ;Berikan ACK
Lcall  Pulse_SEE ;
Setb   SDA ;
Mov    @R0,A ;
Inc    R0 ;
Djnz  R5,LoopPageRead ;
Ret

;*****
;*****
; TULIS DATA I2C
; - R7 = DDI Slave Address
; - R4 = DDI Device Address
; - DataSEE = Data yang ditulis

KirimDataI2C:
Acall  BuatSlaveDevAddress
;Kirim Slave dan Device Address
Lcall  Buat_StartBit ;
Lcall  KirimDataSEE ;
Mov    A,DataSEE
;Kirim Data
Lcall  KirimDataSEE ;
Lcall  Buat_StopBit
;Stop Bit
Ret

BuatSlaveDevAddress:
Mov    A,R4
Rl     A
Mov    R4,A
Mov    A,R7
Anl    A,#11111100b
Add    A,R4

```