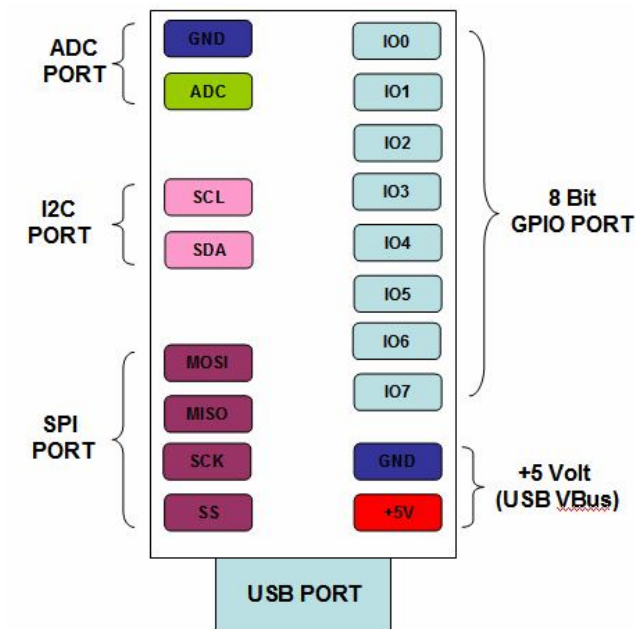


USB HIDClass <-> IOConverter DLL



Blok diagram USB HID <-> IO Converter berdasarkan fungsi

Berikut adalah fungsi-fungsi yang ada pada file [HID_IOConverter.dll](#)

GPIO_Port_Out (ByVal Handle&, ByVal DataBits As Byte) As Long

- **Val Handle&** adalah handle USB HID <-> IO Converter ini.
- **DataBits** adalah data yang kita kirimkan ke port GPIO USB HID <-> IO Converter.
- Fungsi ini digunakan untuk menulis data pada 8 bit port IO, 1 akan menjadikan port IO berlogika high (tegangan VCC 5V) dan 0 akan menjadikan port IO berlogika low (tegangan Ground (0 Volt))

Contoh;

```
GPIO_Port_Out(Handle, 255)
```

Perintah ini akan menjadikan 8 bit port IO berlogika high semua, dimana dapat digunakan untuk menyalakan LED secara langsung jika dihubungkan dengan LED ke ground melalui resistor.

GPIO_Port_In (ByVal Handle&, ByRef DataRead As Byte) As Long

- Fungsi ini digunakan untuk membaca 8 bit port IO. Misalnya digunakan untuk mendeteksi penekanan tombol jika dihubungkan dengan tombol Push-Button.

Contoh;

```
Data = GPIO_Port_In (Handle)
```

Perintah ini akan membaca logic Port IO.

GPIO_DDR_Out (ByVal Handle&, ByVal DataBits As Byte) As Long

- Fungsi ini untuk menulis data pada DDR (*data direction register*) 8 bit port IO. Untuk menjadikan 8 bit port IO sebagai port output, DDR harus high dan untuk menjadikannya sebagai port input kita harus menulis DDR-nya berlogika low.

Contoh;

`GPIO_DDR_Out(Handle,&h0F)`

Perintah ini akan menjadikan bit 0-3 sebagai port input dan bit 4-7 sebagai port output.

GPIO_DDR_In (ByVal Handle&, ByRef DataRead As Byte) As Long

- Fungsi ini digunakan untuk membaca keadaan terakhir register DDR 8 bit port IO.

Set_Bit (ByVal Handle&, ByVal DataBits As Byte) As Long

- Fungsi ini digunakan untuk membuat bit tertentu pada 8 bit port IO, port I2C dan port SPI berlogika high dengan parameter alamat portnya ada pada file module `HIDI_DLL_Interface.bas`

Contoh;

`Set_Bit(Handle, GPIO_Port1)`

Perintah ini akan membuat pin IO1 akan berlogika high.

`Set_Bit(Handle, I2C_SCL_Port)`

Perintah ini akan membuat pin SCL berlogika high.

Clear_Bit (ByVal Handle&, ByVal DataBits As Byte) As Long

- Fungsi ini digunakan untuk membuat bit tertentu pada 8 bit port IO, port I2C dan port SPI berlogika low dengan parameter alamat portnya ada pada file module `HIDI_DLL_Interface.bas`

Contoh;

`Set_Bit(Handle, GPIO_Port2)`

Perintah ini akan membuat pin IO2 akan berlogika low.

In_Bit (ByVal Handle&, ByVal DataBits As Byte, ByRef DataInByte As Byte) As Long

- Fungsi ini digunakan untuk membaca pin pada alamat port tertentu.

Contoh;

`In_Bit(Handle, GPIO_Port2, Data_In)`

- Perintah ini akan membaca logika pin IO2. Hasil pembacaannya akan di simpan di variabel `Data_In`. Misal pin IO2 ini dihubungkan dengan tombol push-button ke Ground, maka jika push-button ditekan `Data_In` akan bernilai 0, sebaliknya jika tidak ditekan akan bernilai 1.

I2C_Init (ByVal Handle&) As Long

I2C_Stop (ByVal Handle&) As Long

I2C_Start (ByVal Handle&) As Long

I2C_Write_Byte (ByVal (ByVal Handle&, ByRef DataRead As Byte) As Long

I2C_Write_Byte_rACK (ByVal (ByVal Handle&, ByRef DataRead As Byte) As Long

I2C_Read_Byte_ACK (ByVal Handle&, ByRef DataRead As Byte) As Long
I2C_Read_Byte_NACK Lib (ByVal Handle&, ByRef DataRead As Byte) As Long

- Fungsi-fungsi diatas digunakan untuk komunikasi I2C. Fungsi-fungsi ini akan di bahas pada contoh aplikasi I2C protokol dan implementasinya.

SPI_Write (ByVal Handle&, ByVal DataWrite As Byte) As Long
SPI_Read (ByVal Handle&, ByRef DataRead As Byte) As Long
SPI_Init (ByVal Handle&) As Long

- Fungsi-fungsi diatas digunakan untuk komunikasi SPI. Fungsi-fungsi ini akan di bahas pada contoh aplikasi SPI protokol dan implementasinya.

ADC_Read (ByVal Handle&, ByRef DataInWord As Integer) As Long

- Fungsi ini digunakan untuk membaca nilai tegangan pada kaki ADC dan mengkonversinya menjadi data digital 10bit.

Contoh;

ADC_Read(Handle, ADC_Data)

- Perintah ini akan membaca nilai tegangan pada kaki ADC. Hasil pembacaannya adalah 10 bit nilai tegangan akan di simpan di variabel ADC_Data.

Contoh2 aplikasi akan mempraktekkan penggunaan fungsi-fungsi diatas dengan membuat program-program aplikasi untuk mengontrol suatu alat secara langsung atau berkomunikasi dengan perangkat elektronik lainnya, misalnya untuk menyalakan LED, membaca tombol push button, membaca sensor digital melalui komuniaksi I2C atupun SPI, pemograman mikrokontroler dan membaca tegangan melalui ADC.