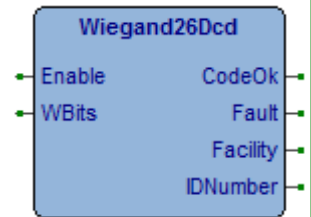


1.1.21 Wiegand26Dcd, decodes the Wiegand 26 bits code

Type	Library
FB	eLLabUtyLib_C050

Questo FB decodifica un codice Wiegand a 26 bits. Il FB può essere eseguito come una funzione, occorre fornire in **WBits** il codice Wiegand acquisito. In uscita se il codice è corretto sarà attivo **CodeOk** ed avremo in **Facility** ed in **IDNumber** il codice decodificato.

In caso di errore nella decodifica viene attivata per un loop di programma l'uscita **Fault**.



- Enable** (BOOL) Abilitazione FB, attivandolo viene eseguita la decodifica del codice.
- WBits** (DWORD) Codice Wiegand da decodificare.
- CodeOk** (BOOL) Si attiva se decodifica correttamente eseguita.
- Fault** (BOOL) Attivo per un loop se errore decodifica.
- Facility** (USINT) Facility code decodificato.
- IDNumber** (UDINT) ID number decodificato.

Codici di errore

In caso di errore si attiva l'uscita **Fault**, con [SysGetLastError](#) è possibile rilevare il codice di errore.

Codice	Descrizione
10068100	Codice Wiegand in WBits non è corretto ha più di 26 bits.
10068110	Even parity errata.
10068120	Odd parity errata.

Esempi

Viene eseguita l'acquisizione di un lettore TAGS RFID connesso agli ingressi Di00 e Di01 di un modulo CPU SlimLine. Il valore del TAG acquisito è visualizzabile da debug.

Definizione variabili

	Name	Type	Address	Array	Init value	Attribute	Description
1	LInp	SysGetPhrDI	Auto	No		..	Acquisizione ingressi logici
2	Data	BOOL	Auto	[0..1]		..	Appoggio ingressi dato
3	BTime	UDINT	Auto	No		..	Buffer time (uS)
4	DStrobe	BOOL	Auto	No		..	Data strobe
5	Value	DWORD	Auto	No		..	Digital input value
6	DWiegand	DWORD	Auto	No		..	Wiegand data
7	BWiegand	USINT	Auto	No		..	Wiegand bits
8	WDecoder	Wiegand26Dcd	Auto	No		..	Wiegand decoder

Esempio ST (PTP114A660, ST_Wiegand26Dcd)

```
(* Program initialization. *)

IF (SysFirstLoop) THEN
  DWiegand:=SysSetTaskLpTime(TaskID:=ID_TASK_FAST, Time:=150);
  LInp.Address:=255; (* Module address *)
  LInp.Mode:=DI_I_8_LL; (* Acquisition mode *)
END_IF;

(* Se non variazione ingressi per tempo definito bus a riposo. *)

IF (DStrobe) THEN
  IF ((SysGetSysTime(TRUE)-BTime) > 100000) THEN
    DStrobe:=FALSE; (* Data strobe *)

    (* Eseguo decodifica codice Wiegand. *)

    IF (BWiegand = 26) THEN
      WDecoder(Enable:=TRUE, WBits:=DWiegand); (* Wiegand decoder *)
    END_IF;
  END_IF;

  (* Acquisizione e controllo se variazione ingressi. *)

  LInp(); (* Acquisizione ingressi logici *)
  IF (LInp.Value = Value) THEN RETURN; END_IF;
  Value:=LInp.Value; (* Digital input value *)
  BTime:=SysGetSysTime(TRUE); (* Buffer time (uS) *)

  (* Se variazione ingressi dopo bus a riposo start dati. *)

  IF NOT(DStrobe) THEN
    DStrobe:=TRUE; (* Data strobe *)
    BWiegand:=0; (* Wiegand bits *)
    DWiegand:=0; (* Wiegand data *)
  END_IF;

  (* Gestione "D0" ogni transizione è un bit con valore "0". *)

  IF (TO_BOOL(LInp.Value AND 16#01) <> Data[0]) THEN
    Data[0]:=NOT(Data[0]); (* Appoggio ingresso "D0" *)
    IF NOT(Data[0]) THEN DWiegand:=(DWiegand*2); BWiegand:=BWiegand+1; END_IF;
  END_IF;

  (* Gestione "D1" ogni transizione è un bit con valore "1". *)

  IF (TO_BOOL(LInp.Value AND 16#02) <> Data[1]) THEN
    Data[1]:=NOT(Data[1]); (* Appoggio ingresso "D1" *)
    IF NOT(Data[1]) THEN DWiegand:=(DWiegand*2)+16#00000001; BWiegand:=BWiegand+1; END_IF;
  END_IF;

```