Förslag till kommunikation PLC - PC

Cejn

==========

Version	Datum	Sign	Kommentar
1.0	2013-10-14	AME	Initial version
1.1	2013-10-14	AME	Ändrade hexadecimalt till decimalt
1.2	2013-10-14	AME	Ändrade protokollet
1.3	2013-12-13	AME	La till timeout i protokollet
1.4	2014-03-06	AME	La till väntetid vid byte av sändning/mottagning
1.5	2016-11-22	AME	La till de koder som används hos Cejn
1.6	2017-12-14	AME	La till förklaringar i exempel tabellen samt i kommandotabellen

Intro

Den lokala PLC (i märkmaskinen) kommunicerar med linjen via PROFINET. Att använda PROFINET för kommunikationen mellan PLC och PC är komplicerat och kostsamt, och då vi endast ska skicka artikelnummer till PC verkar det vara *overkill*

Kommunikation via IO

Det enklaste sättet att kommunicera är att använda IO. En Adam modul med 8 in / 8 ut IO:n räcker. T ex ADAM-6052, eller ADAM-6060.

Beskrivning av kommunikationen

Kommunikationen bygger på att man skickar ett tecken (ASCII) i taget.

Man startar med en startkod (STX - ASCII(2)), och sedan skickas koden för det fält som ska skickas och sedan skickar man fältet ett tecken i taget. För att avsluta skickar man en stoppkod (ETX - ASCII(3)). Mottagaren skickar alltid tillbaka samma värde som den har fått, som en koll på att allt är korrekt. Mellan varje *telegram* skickas en

kvittens. Kvittensen är alla signaler höga, dvs 255 eller 11111111.

Exempel

Om vi har artikelnummer: **123456789**

1. PLC startar med att skicka en startkod **2**
2. PC svarar med samma kod
3. PLC kvitterar med **255**
4. PC kvitterar med **255**
5. PLC skickar koden för artikelbyte **10**
6. PC svarar med samma kod
7. PLC kvitterar med **255**
8. PC kvitterar med **255**
9. PLC skickar nu första tecknet (1) som ASCII **49**
10. PC svarar med samma tecken
11. PLC kvitterar med **255**
12. PC kvitterar med **255**
13. PLC skickar nästa tecken (2) som ASCII **50**
14. PC svarar med samma tecken
15. PLC kvitterar med **255**
16. PC kvitterar med **255**

Detta upprepas till alla tecken har skickats.

- För att avsluta kommunikationen skickar PLC ASCII **3**
- PC svarar med samma kod

3. PLC kvitterar med **255**
4. PC kvitterar med **255**

Samma exempel med binära värden

Sender	Tx	Rx	Kommentar
PLC	00000010		STX
PC		00000010	STX
PLC	11111111		Kvittens
PC		11111111	Kvittens
PLC	00001010		10
PC		00001010	10
PLC	11111111		Kvittens
PC		11111111	Kvittens
PLC	00110001		49 "1"
PC		00110001	49 "1"
PLC	11111111		Kvittens
PC		11111111	Kvittens
PLC	00110010		50 "2"
PC		00110010	50 "2"
PLC	11111111		Kvittens
PC		11111111	Kvittens

Upprepa för övriga värden ...

Sender	Tx	Rx	Kommentar
PLC	00000011		ETX
PC		00000011	ETX
PLC	11111111		Kvittens
PC		11111111	Kvittens

Byte mellan mottagning och sändning

en sändning avslutas och en ny startar direkt:

Send	Receive	Value	Kommentar
PC	PCL	ETX	Slut på kommandot
PLC	PC	ETX	
PC	PLC	11111111	Kvittens
PLC	PC	11111111	
PLC	PC	STX	< ---- här hinner PC inte läsa 11111111

Väntetid

För att vi inte ska missa information, enligt ovan, måste vi ha en kort väntan innan vi byter mellan mottagning och sändning.

Väntetiden bör vara konfigurerbar i millisekunder. T ex 400 ms.

Timeout

I och med att både PLC och PC kan sända samtidigt så kan det inträffa! För att vi inte ska hamna i den situationen att båda parter väntar på svar från den andra måste vi ha en **timeout**. Om ett svar inte kommer inom en specificerad tid så ska *telegrammet* avbrytas och ett nytt försök göras.

Timeouten bör vara konfigurerbar i millisekunder.

Flexibilitet

Detta system är mycket flexibelt. Om det tillkommer värden man vill föra över mellan PLC och PC så behöver man bara skapa en kod för detta. Kommunikationen kan gå i båda riktningarna, dvs. PC eller PLC kan initiera kommunikation.

Längden på värdet som skickat kan variera i och med att man skickar en start och en stoppkod.

ASCII - koder

American **S**tandard **C**ode for **I**nformation **I**nterchange, är en teckenkodning som används för att representera bokstäver och andra tecken i datorer. Den omfattar 255 tecken vilket kan representeras av 8 bitar (8 IO:n)

Code Page

Då det finns flera olika varianter av ASCII måste vi bestämma vilken vi ska använda. Vi tror att Cejn använder **CP850**. OBS! Detta måste kontrolleras!

Koder i protokollet hos Cejn

Kommandon som börjar på **10** är från **PLC** och de som börjar på **20** är från **PC**.

Kommando	Parameter	Exempel
10	Artikelnr (alfa)	10 101021503
11	prov (1 eller 0)	11 1
12	Start/OK	12 1
13	Bit i läge (1 eller 0)	13 1
14	Extern startsignal	14 1
15	Extern End	15 1
16	Starta om ClmRRS	16 1
20	Setkant (en siffra 1-9)	20 1
21	Batch ej klar (1 eller 0)	21 1
22	Ready to mark	22 1

Felkoder	Parameter	Meddelande
29	1	"för artikelnumret finns inte"
29	2	"mallen hittas inte" (alltså filen med märkmallen)
29	3	"märkmallen är inte definierad" (vilket inträffar om PLC bara skickar 12 1 direkt,utan att först ha talat om vilket artikelnummer som ska märkas)
29	4	"Okänt kommando" (om PLC t ex skickar 55 123)

En liten fotnot...

Dessa signaler är separerade bara för att vara tydlig. Det skulle ju gå med bara en signal.

Kommando	Parameter	Beskrivning	Exempel
14	0 eller 1	Extern Start	14 1
15	0 eller 1	Extern End	15 1

Det ger att

Exempel	Beskrivning
14 1	Extern Start
15 0	Extern Start

och

Exempel	Beskrivning
14 0	Extern End
15 1	Extern End

Men det är ju tydligare vad man menar när man separerar dem. Och i verkligheten använder man alltid **14 1** för Extern Start och **15 1** för Extern End!