

Laboratorium
Aparatury Automatyзації

Instrukcje do ćwiczeń

Autorzy:

Jerzy Chojnacki

Krzysztof Oprędkiewicz

Janusz Teneta

Łukasz Więckowski

Witold Głowacz

AGH, Kraków 2017

Spis treści

Ćwiczenie 0. SIEMENS S7-1200 z panelem operatorskim – konfiguracja sprzętu	3
Ćwiczenie 1. Układ stabilizacji ciśnienia (sterownik TURCK).....	7
Ćwiczenie 2. Układ regulacji temperatury (SIEMENS)	20
Ćwiczenie 3. Sterowanie układem lewitacji powietrznej.	27
Ćwiczenie 4. Konfiguracja i uruchomienie sieci przemysłowej PROFINET (SIEMENS)...	40
Ćwiczenie 5. Dobór nastaw regulatorów przemysłowych.	46
Ćwiczenie 7. SIEMENS S7 1200- funkcje definiowane przez użytkownika –język drabinkowy (LD).....	55
Ćwiczenie 8. Pakiet VisiDac do realizacji bezpośredniego sterowania cyfrowego.....	58
Materiały pomocnicze do ćwiczenia 8.....	61
Ćwiczenie 9. Układ sterowania ogniów słonecznych.	70
Materiały pomocnicze do ćwiczenia 9.....	74
Dodatek 1. Laboratoryjny obiekt regulacji.	78
Literatura	80

Ćwiczenie 0. SIEMENS S7-1200 z panelem operatorskim – konfiguracja sprzętu

Cel ćwiczenia

Zapoznanie się ze środowiskiem TIA PORTAL: konfiguracja hardware'u i uruchomienie prostego programu sterowania logicznego z aplikacją SCADA.

Proponowany sposób wykonania ćwiczenia:

1. Konfiguracja sprzętu
 - Uruchomić TIA, utworzyć nowy projekt pod dowolną nazwą.
 - Dokonać przykładowej konfiguracji sprzętu dla budowanej aplikacji zgodnie z konfiguracją rzeczywistych sterowników w laboratorium. Konfiguracja jest podana w tabeli 1.1
 - CPU należy dodać na etapie kreacji nowego projektu, jest ona do znalezienia w zakładce: **Add new device\SIMATIC S7 1200 CPU** i tam znaleźć właściwą.
 - Panel HMI: **Add new device\HMI\ SIMATIC SIMATIC Basic Panel** i tam znaleźć właściwy (nr w tabeli 1.1).
 - Pozostałe moduły należy znaleźć w katalogu w zakładce **Hardware catalog** (jest po prawej stronie z boku). Mogą one być dołączane po wyborze opcji „Device Configuration”, switch po wyborze opcji „Topology view”

UWAGA! Podczas wyboru każdego elementu zwracać uwagę na numer katalogowy bo tylko on jednoznacznie określa dany element!

Tabela 0.1. Konfiguracja sprzętu

Slot nr	Funkcja modułu	Oznaczenie	Numer katalogowy 6ES7 ...
1	Jednostka centralna	CPU	212-1BE40-0XB0
	Moduł wyjść PWM (mocowany bezpośrednio w slotcie na CPU) UWAGA! Ten element wstawiamy tylko przy tych sterownikach, które mają zainstalowany ten moduł! (Informacja od prowadzących).	DQ 4x24VDC	222-1BD30-0XB0
	Panel Operatorski	HMI	6AV2 123-2DB03-0AX0
2	Wejścia analogowe	AI4xRTD	231-5PD32-0XB0
3	Wyjścia analogowe	AQ2x14BIT	232-4HB32-0XB0
	Switch sieciowy	CSM 1277 SIMATIC NET	6GK7-277-1AA10-0AA0

Dokonać konfiguracji kolejno poszczególnych modułów:

Jednostka centralna (slot 1):

Konfiguracja portu sieciowego: Profinet Interface [X1]: dodać nową podsieć, wybrać opcję ustawienia adresu IP w projekcie (jest proponowany domyślny, zostawiamy go).

Czas cyklu CPU: pozostawiamy domyślny 150 [ms].

Wejście PWM (UWAGA! Tylko przy tych stanowiskach z zainstalowanym modułem PWM): aktywujemy generator PWM1, sygnał typu PWM, czas cyklu 100[ms], podstawowe wypełnienie: 50%, adres wyjścia: %Q4.0, zapisujemy i pamiętamy identyfikator sprzętu: 265 (będzie potrzebny do aktywacji FB współpracującego z tym generatorem – ćwiczenie z lewitacją powietrzną).

Moduł wejść analogowych (slot 2):

Na wszystkich kanałach ustawić typ czujnika: termometr rezystancyjny połączony 4 przewodowo, Pt 100 standard Range, skala w stopniach Celsjusza, filtracja: słaba, sygnalizacja uszkodzenia przewodu – nieaktywna. Sprawdzić zakresy adresów wejściowych: (%IW) od 96 do 103

Moduł wyjść analogowych (slot 3)

W obu kanałach w zakładce AQ2 /Outputs ustawić typ sygnałów wyjściowych: napięcie 0-10V, przy STOP CPU przełączyć na wartość zastępczą: 2.5V. W zakładce „I/O Tags” odczytać i zapamiętać adresy poszczególnych wyjść, powinny być: (%QW) od 112 do 115.

UWAGA: Tu jest podana tylko wstępna konfiguracja. Przy poszczególnych ćwiczeniach będzie ona musiała być rozbudowywana (będzie to podane w instrukcjach).

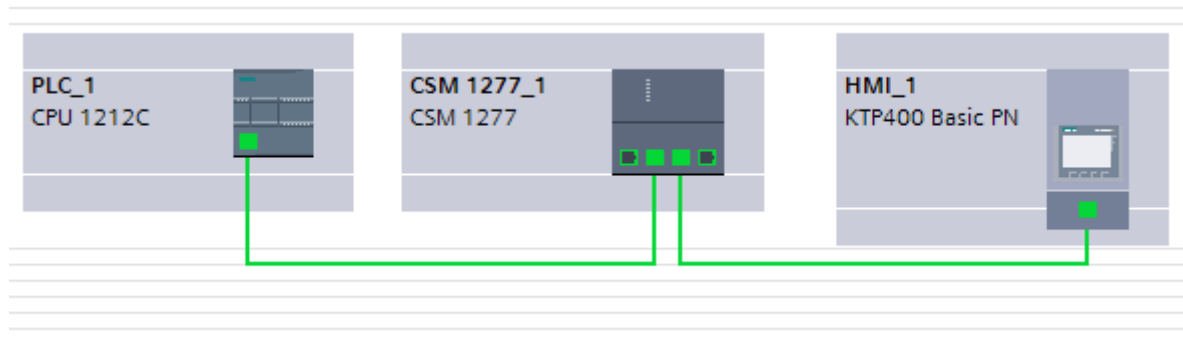
Panel operatorski

Na etapie konfiguracji sprzętu nie wymaga on dodatkowych czynności.

Switch sieciowy

Nie wymaga dodatkowych działań podczas konfiguracji sprzętu.

Po wprowadzeniu wszystkich elementów należy je połączyć w sieć zgodnie ze schematem:



2. Prosty program sterowania logicznego

Program ma być zbudowany, załadowany i uruchomiony na rzeczywistym zestawie sterownikowym z panelem, testy należy wykonać z użyciem panelu.

Algorytm sterowania

Otworzyć tabelę z nazwami symbolicznymi: PLC Tags” i zdefiniować nazwy zmiennych użytych do testów. Jak adresy wejść wpisać np. %M0.0, %M0.1, %M0.2, %M0.3 i nazwy odpowiednio: „start” „gotów” „awaria” „stop”. Są to zmienne wewnętrzne PLC, które będą ustawiane poprzez przyciski na panelu (W typowej sytuacji są to wejścia binarne). Jako adresy wyjściowe należy użyć: %Q0.0 - „napęd ON”, jako %Q0.1 - „praca”. Wyjścia mają być możliwe do obserwacji na panelu poprzez animacje odpowiednich elementów graficznych, np lampek.

W projekcie otworzyć plik źródłowy bloku organizacyjnego OB1 (Ob1 - Main). Z wykorzystaniem styków i cewek zrealizować funkcje logiczne:

Napęd ON = gotow & start & NOT awaria praca=napęd ON

WSKAZÓWKI: iloczyn logiczny realizujemy jako szeregowe połączenie styków, sumę realizujemy jako połączenie równoległe. Zmienna wejściowa wprost to styk normalnie otwarty, zmienna zanegowana to styk normalnie zamknięty. Zmienna wyjściowa wprost to cewka zwykła, zmienna wyjściowa zanegowana to cewka negująca. Każde wyjście to osobny szczebel w schemacie drabinkowym.

Aplikacja SCADA na panelu operatorskim

Dodać do projektu panel operatorski (HMI_1) i otworzyć domyślny ekran aplikacji (Screen_1). Zdefiniować na tym ekranie możliwość zadawania wartości zmiennych **gotow**, **start** i **awaria**.

Przykładowo:

Zmienną **gotow** przypisać do standardowego przycisku (Libraries/Buttons-and-Switches/Master Copies/Pushbutton (wybrać jeden). Otworzyć własności przycisku (zaznaczyć go, przycisnąć przycisk „float” obok „properties”) powiązać go z **gotow**.

Zmienną **start** powiązać z przyciskiem funkcyjnym F1: Properties/Events/Press Key: Edit bits/SetBit **start**.

Do zmiennej **awaria** przypisać grzybek bezpieczeństwa (Pushbutton Emergency).

Zmienną wyjściową **napęd_ON** powiązać z lampką sygnalizacyjną zbudowaną samodzielnie np. z podstawowego obiektu typu elipsa: (Float/basic objects): Animations/Appearance: zdefiniować **napęd_ON** i z wartością 0 skojarzyć kolor np. czerwony a z wartością 1 skojarzyć kolor zielony. Dodać podpisy do zmiennych z użyciem elementarnego pola tekstowego. W przypadku podpisu zmiennej **napęd_ON** zdefiniować, żeby podpis pojawiał się tylko wtedy, gdy napęd jest uruchomiony: „napęd włączony”: add new animation/visibility/ i zakres od 1 do 1. W przypadku pozostałych zmiennych podpisy pozostawić statyczne (nie wiązać ich z żadnymi animacjami).

Zbudowany program załadować na sterownik i panel, sprawdzić działanie zadając odpowiednie wartości na wejścia i obserwując wyjścia. Działanie programu należy obserwować dwoma metodami (muszą oczywiście dać ten sam wynik):

1. Poprzez obserwację działania elementów na ekranie animujących wartości wyjścia,
2. Poprzez obserwację w trybie „On Line” zachowania się napisanego programu w oknie, w którym został wpisany.

Alternatywna realizacja algorytmu sterowania (jak wystarczy czasu)

Zrealizować funkcję logiczną opisaną wcześniej z użyciem cewek: ustawiającej i kasującej, przy założeniu, że: załączenie odbywa się z użyciem przycisku **start**, a wyłączenie z użyciem przycisku „stop” skojarzonego ze zmienną **stop**. Zmienną **napęd_ON** przy starcie należy powiązać z cewką ustawiającą S (Set), a przy zatrzymaniu z cewką kasującą R (Reset). Zaobserwować różnice w działaniu: przy wersji pierwszej warunek musi być spełniony cały czas, w wersji 2 należy załączyć jednym przyciskiem, a wyłączyć drugim.

Zakres wiadomości na kolokwium

Elementy organizacyjne oprogramowania wg normy: funkcje (FC) i bloki funkcyjne (FB) oraz charakterystyczne dla systemu SIEMENS: bloki organizacyjne (OB), Bloki Danych (DB) i Typy Danych PLC. Zasady adresacji danych w systemie SIEMENS. Typy danych i zmiennych w systemach PLC. Podstawowe elementy języka drabinkowego: styki, cewki i komparatory oraz ich interpretacja.

Sprawozdanie

Sprawozdanie powinno zawierać: schemat i opis konfiguracji sprzętu, schemat drabinkowy realizacji zadania oraz opis zmiennych aplikacji SCADA. Do przygotowania sprawozdania można wykorzystać narzędzia w TIA PORTAL pozwalające na wygenerowanie pdf'a z konfiguracji sprzętu oraz plików źródłowych oprogramowania: otworzyć plik źródłowy bloku OB1 lub ekranu 1 w HMI odpowiednio, zaznaczyć opcję print, własności: compact. Drukarka: Microsoft XPS Document Writer, druk do pliku we wskazanej lokalizacji.

Ćwiczenie 1. Układ stabilizacji ciśnienia (sterownik TURCK)

Cel ćwiczenia

Głównym celem ćwiczenia jest konfiguracja oraz uruchomienie układu do stabilizacji ciśnienia w zbiorniku o pojemności 20 litrów.

Zakres

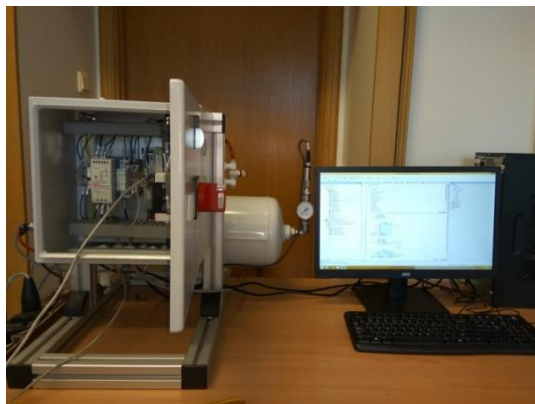
1. Konfiguracja sterownika TURCK w środowisku CODESYS, w tym konfiguracja modułów analogowych I/O oraz definiowanie zakresów pomiarowych i charakterystyki przetwarzania przetwornika TURCK PT006R-11-LI3-H113.
2. Implementacja w języku SCL funkcji oraz bloków funkcyjnych dla tworzonego układu sterowania w tym implementacja bloku regulatora PID.
3. Sprawdzenie działania regulatora w wersji P/ PI/ PD.
4. Przykład realizacji wymiany danych procesowych w oparciu o przemysłową sieć PROFINET (opcjonalnie, jak wystarczy czasu).

Aparatura użyta do budowy stanowiska

- sterownik TURCK BL 20 PG EN V3
- moduł wejścia analogowego TURCK BL20-1AI pracującym w zakresie 0/4-20mA,
- moduł wyjścia analogowego TURCK BL20-2AO pracującym w zakresie -10/0V-10V DC.
- zawór proporcjonalny FESTO typu 5/3 NC, MPYE-5-1/8-LF-010-B,
- przetwornik ciśnienia TURCK PT006R-11-LI3-H1131, piezorezystancyjny, pracujący w zakresie 0-6bar,
- zbiornik ciśnienia FESTO 20l.

Stanowisko zostało zaprezentowane na Rys.1.1. Szczegółowy opis budowy oraz działania stanowiska znajduje się w pracy inżynierskiej ^[1].

Uwaga! DTR wszystkich wymienionych urządzeń znajdują się w katalogu korespondującym z nr ćwiczenia laboratoryjnego.



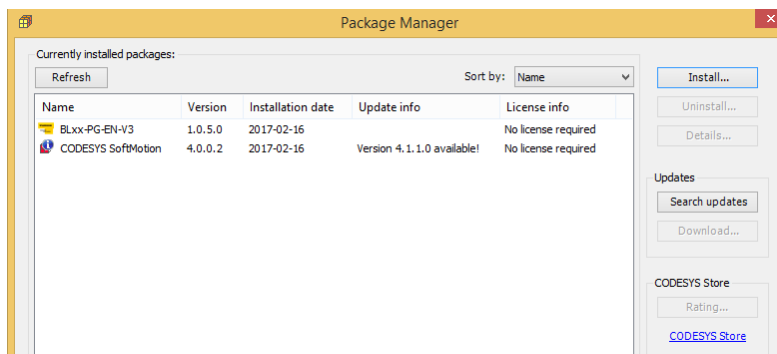
Rys.1.1. Stanowisko do stabilizacji ciśnienia ze sterownikiem TURCK BL-20

¹ Damian Sady – „Stanowisko do kontroli ciśnienia”- praca inż., Kat. Automatyki i Inż. Biomedycznej AGH, 2017r.

Proponowany sposób wykonania ćwiczenia

1. Instalacja sprzętu TURCK w środowisku CODESYS

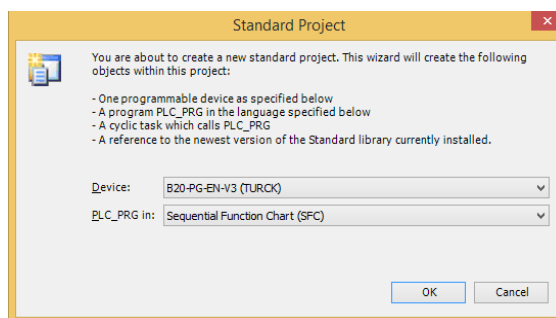
Instalację nowego sprzętu dokonujemy korzystając z opcji: Tools/PackageManager (Rys.1.2). Uwaga!: Podczas instalacji należy wskazać plik BLxx-PG-EN-V3_V1.0.5.0 znajdujący się w katalogu po rozpakowaniu archiwum. Plik można pobrać ze strony producenta sprzętu.



Rys.1.2. Instalacja nowego sprzętu w środowisku CODESYS.

2. Konfiguracja sprzętu

Uruchomić oprogramowanie CODESYS i utworzyć nowy projekt wybierając opcje Standard project (Rys.1.3). Projekt można zapisać pod dowolną nazwą.



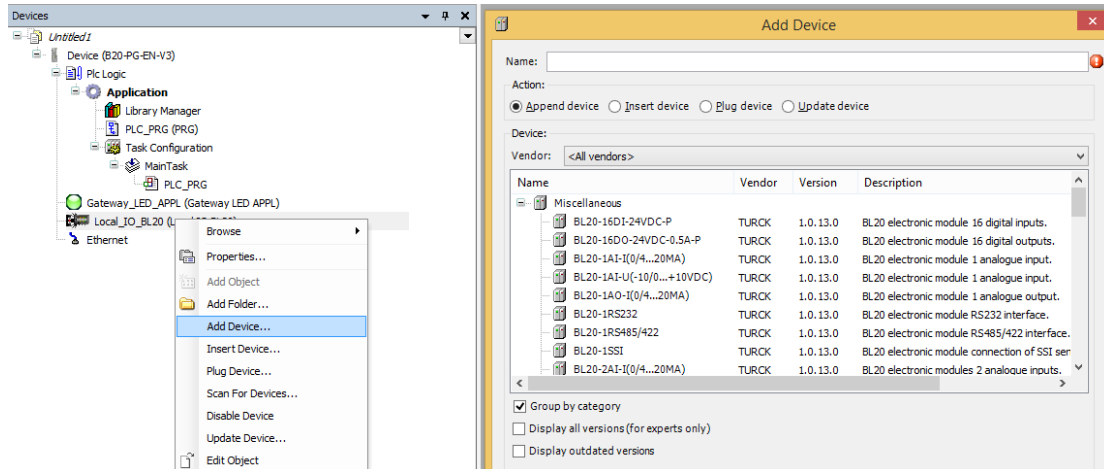
Rys.1.3. Kreator projektu w środowisku CODESYS

Dokonać przykładowej konfiguracji sprzętu dla budowanej aplikacji zgodnie z konfiguracją rzeczywistą stanowiska. Konfiguracja jest podana w tabeli 1.1.

Tabela 1.1. Konfiguracja sprzętu

Lp.	Funkcja /modułu	Oznaczenie	Numer katalogowy
1	Jednostka centralna	BL20-PG-EN-V3	B20-PG-EN-V3(TURCK)
2	Wejście analogowe	1x AI (0/4...20mA)	BL20-1AI-I(0/4...20MA)
3	Wyjście analogowe	2x AO(-0/0...+10VDC)	BL20-2AO-U(-0/0...+10VDC)

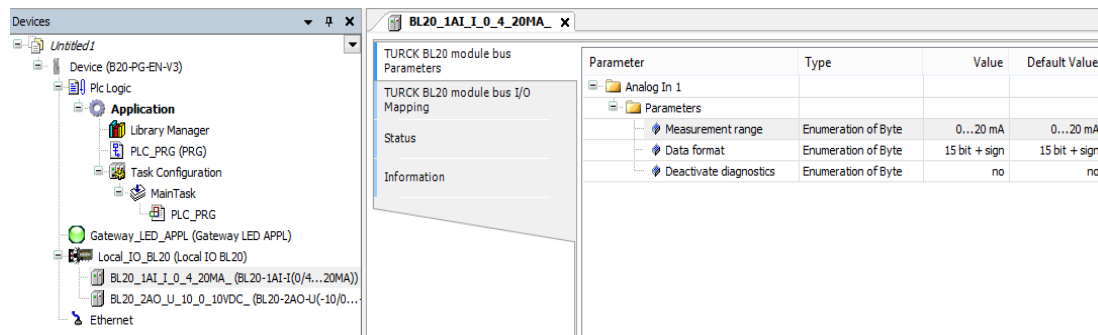
Dodanie modułów I/O realizujemy wywołując polecenie Add Device w widoku drzewa projektu, klikając lewym klawiszem na bramę komunikacyjną Local_IO_BL20 (Rys.1.4).



Rys.1.4. Dodawanie modułów rozszerzeń I/O do sterownika TURCK BL-20

Konfigurację modułów (Rys.1.5), wywołujemy klikając kolejno na ikony poszczególnych modułów I/O (drzewo projektów/ podkatalog Local_IO_BL20):

- wejście analogowe 1x AI (0/4...20mA) : **zakres 4-20mA, 15bit +sign**, adres wejścia: **%IW0**
- wyjście analogowe 2xAO(-10/0...+10VDC): zakres: (-10/0...+10VDC moduł nie wymaga konfiguracji. Zapamiętać należy adres dla wyjścia pierwszego: **%QW1**



Rys.1.5. Konfiguracja modułów I/O sterownika TURCK BL-20.

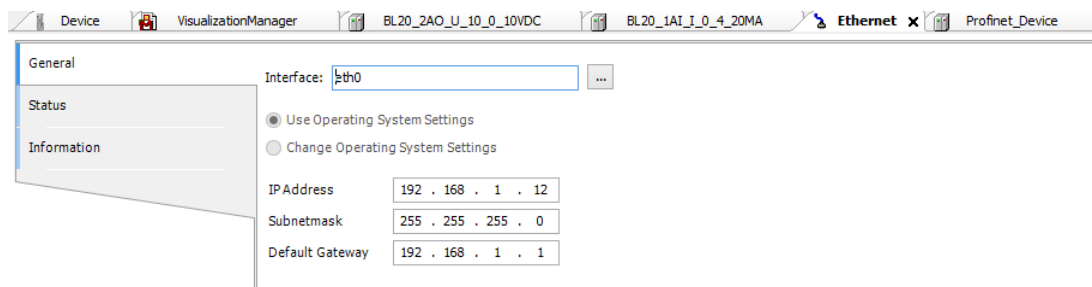
Konfiguracja (sprawdzenie) adresu IP sterownika TURCK.

Adres IP sterownika Turck **192.168.1.12**, ustawiamy przełącznikiem obrotowym na obudowie bramy komunikacyjnej BL20 (Rys.1.6).



Rys.1.6. Sposób ustawiania adresu IP dla sterownika TURCK BL-20

Ponadto w ustawieniach Ethernet /General należy sprawdzić ustawienia konfiguracyjne sieci (Rys. 1.7)



Rys.1.7. Konfiguracja bramy BL-20 (ustawienia konfiguracyjne sieci).

Program sterujący

Koncepcja programu głównego polega na stworzeniu funkcji(FUN) oraz bloków funkcyjnych (FB), odpowiedzialnych za:

- odczyt z wejścia analogowego i skalowanie sygnału z przetwornika pomiarowego **AnalogRead(FUN)**,
- formowanie sygnału wyjściowego dla zaworu proporcjonalnego **AnalogWrite (FUN)**,
- blok obliczeniowy regulatora PID - **pidREG (FB)**,
- blok z możliwością wyboru sterowania - ręcznego (stała zadana wartość wyjściowa) lub automatycznego - **selREG(FB)**.

Sposób postępowania przy tworzeniu algorytmu sterowania:

Utworzyć funkcje (FUN) oraz bloki funkcyjne (FB),zgodnie z poniższymi listingami:

- a) Odczyt z wejścia analogowego – funkcja **AnalogRead(FUN)**. Przykładowa implementacja funkcji została pokazana na Rys. 1.8.

```

1  FUNCTION AnalogRead : REAL
2  VAR_INPUT
3      AI1:WORD;           //wartość wejściowa z wejścia analogowego
4  END_VAR
5  VAR
6      AI11:REAL;
7      MAX_IN: REAL;     //max zakresu pomiarowego
8  END_VAR
9
10
11  AI11:= WORD_TO_REAL(AI1); //konwersja typ word to real
12  %QW51:= AI1;           //PROFIBUS -> PV
13  MAX_IN:= 6.0;         //maksymalna wartość zmierzona
14  AnalogRead:= AI11 * (MAX_IN/32767); //Return -> PV in [bar]

```

Rys.1.8. Implementacja funkcji **AnalogRead(FUN)** – realizującej odczyt z wejścia analogowego.

b) Sterowanie wyjściem analogowym - funkcja **AnalogWrite (FUN)** – Rys.1.9

```

1  FUNCTION AnalogWrite : INT
2  VAR_INPUT
3      pidOutput:REAL;
4  END_VAR
5  VAR
6      tempOut:REAL;
7  END_VAR
8
9
10 // formowanie sygnału wyjściowego dla zaworu proporcjonalnego
11 tempOut := pidOutput * (32767 / 100);
12 AnalogWrite:= REAL_TO_INT(tempOut); //Return outputValue in INT

```

Rys.1.9. Implementacja funkcji **AnalogWrite(FUN)**

c) Implementacja regulatora prędkościowego PID – blok **pidREG (FB)** – Rys.1.10
Uwaga! Blok regulatora PID można opisać w postaci zależności (1.1) – alg. prędkościowy.

$$m_k = m_{k-1} + k_p \left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right) e_k - k_p \left(1 + 2\frac{T_d}{T}\right) e_{k-1} + k_p \frac{T_d}{T} e_{k-2} \quad (1.1)$$

$$\text{gdzie : } K_1 = k_p \left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right) \quad K_2 = -k_p \left(1 + 2\frac{T_d}{T}\right) \quad K_3 = k_p \frac{T_d}{T}$$

```

1  FUNCTION_BLOCK PIDreg
2  VAR_INPUT
3      PV:REAL; //pomiar
4      SP:REAL; //wartość zadana
5      KC:REAL; //człon proporcjonalny
6      Ti:REAL; //człon całkujący
7      Td:REAL; //człon różniczkujący
8      T:REAL;
9  END_VAR
10 VAR_OUTPUT
11     out_PID:REAL; //
12 END_VAR
13 VAR
14     e:REAL; //uchyb regulacji
15     Kp1:REAL;
16     Kp2:REAL;
17     Kp3:REAL;
18     out1:REAL;
19     e1:REAL;
20     e2:REAL;
21 END_VAR
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Rys.1.10. Implementacja bloku funkcyjnego (FB) – regulator PID

- d) utworzyć blok wyboru sterowania – **selREG(FB)**, używając języka FBD lub SCL.

Wywołanie funkcji/bloków funkcyjnych w środowisku CodeSYS.

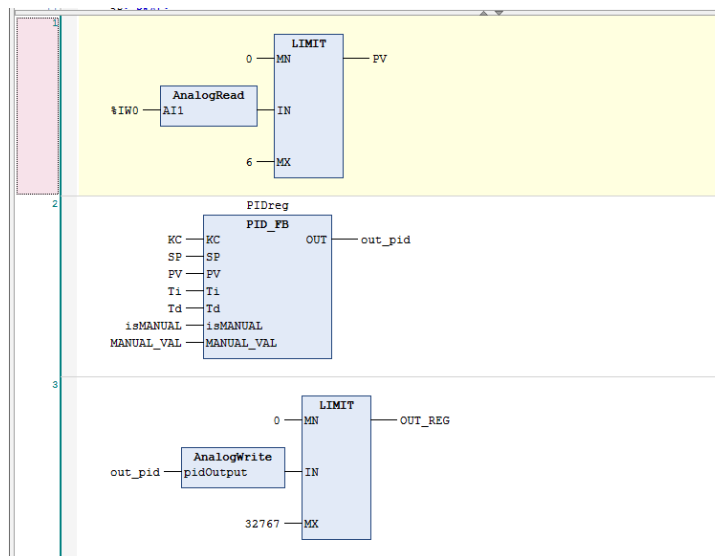
W celu utworzenia sekwencji wywołania, utworzone przez użytkownika bloki funkcyjne muszą zostać połączone w jednym bloku programowym **PLC(PRG)**, wywoływanym cyklicznie w głównym programie (MainTask). W tym celu należy:

- utworzyć blok programowy **PLC(PRG)** – zgodnie z Rys. 1.11.

```

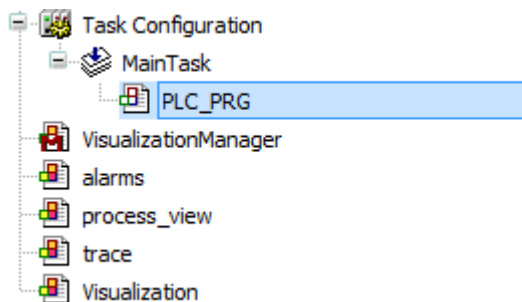
1  PROGRAM PLC_PRG
2  VAR
3  //*****//
4  AI1:WORD;
5  //*****PID*****//
6  PIDreg:PID_FB;
7  PV:REAL;
8  out_pid: REAL;
9  ERROR_out:BOOL;
10 OUT_REG:INT;
11 SP: REAL;
12 KC: REAL;
13 Ti: REAL;
14 Td: REAL;
15 isMANUAL: BOOL;
16 MANUAL_VAL: REAL;
17
18 END_VAR

```



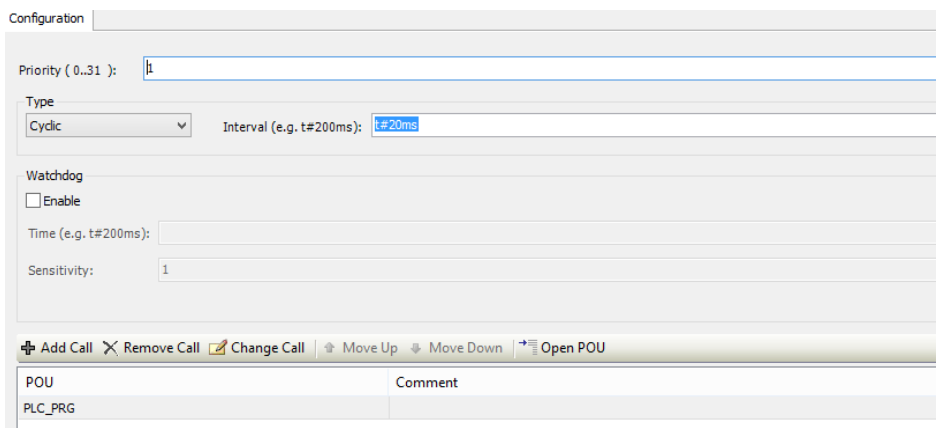
Rys.1.11. Propozycja implementacji bloku programowego (PRG) – główna pętla regulacji dla stanowiska kontroli ciśnienia.

- skonfigurować wywołanie bloku programowego **PLC(PRG)**, poprzez utworzenie w Task Configuration zadania (MainTask), Rys.1.12.



Rys.1.12. Widok programów realizowanych w MainTask

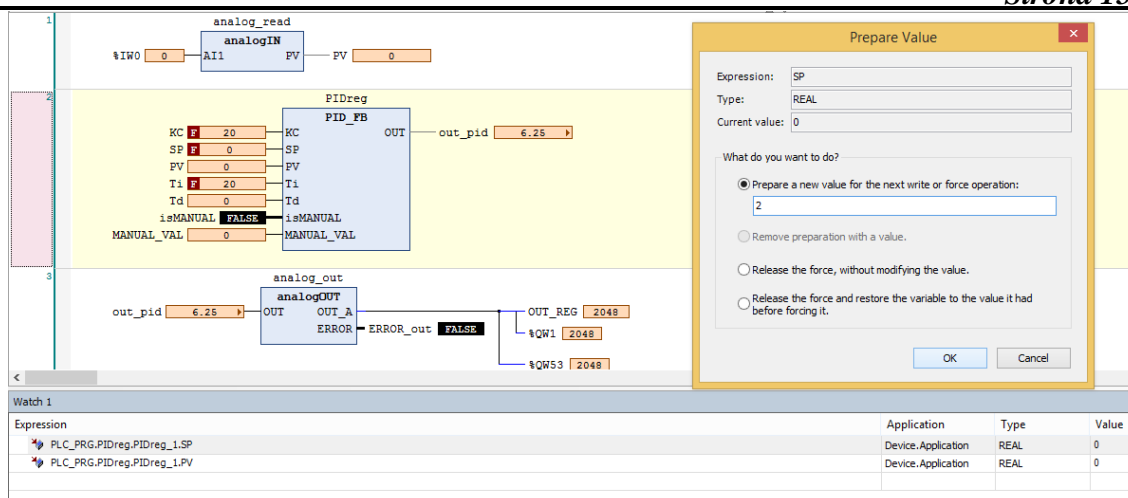
Następnie w Main Task poprzez wybór opcji (Add Object→Program Call) wskazujemy blok programowy PLC(PRG). Operację kończy konfiguracja wywołania cyklicznego zadania MainTask /Configuration. W oknie Configuration (Rys.1.13), należy ustawić wartość okresu próbkowania na 20ms oraz priorytet wywołania zadania (Priority:1, Type: **Cyclic**, Interval:t#20ms).



Rys.1.13. Sposób konfiguracji wywołania dla zadania cyklicznego PLC(PRG).

3. Sprawdzenie działania regulatora w wersji P/ PI/ PD.

Utworzony projekt należy skompilować wydając polecenie (Build/Rebuild). Program wsadowy jest automatycznie ładowany podczas wybrania opcji (Online/ Login). Środowisko Codesys, przy wykryciu nowej wersji oprogramowania, zapyta użytkownika czy załadować nowe/zmodyfikowane oprogramowanie na sterownik PLC. Podczas wykonywania programu (w trybie debugowania), możliwy jest podgląd oraz modyfikacja wartości zmiennych, w tym interesujących nas parametrów regulatora oraz wartości procesowych. Aby wymusić zmianę wartości, wprowadzamy je w polu przy interesującym nas sygnale (Rys. 1.14). Zmianę wartości inicjujemy wybierając opcję Debug/Write Values (Ctrl+F7) oraz wymuszamy poprzez wybór: Debug/Force Values (F7).



Rys.1.14. CODESYS podgląd zmiennych oraz sposób ich modyfikacja w trakcie wykonywania programu – debug.

Sprawdzenie działania regulatora w wersji **P**:

- ustawić działanie regulatora na czysto proporcjonalne ($K_d = 0$, $T_i = 0$) oraz wartość zadaną ciśnienia PV na poziomie 50% zakresu (0 - 4bar).

Zbadać charakterystykę statyczną regulatora (odczyt w oknie dla sygnału wejściowego z zakresu 0% – 60% co 10% dla $K_p = 1, 10, 20$. Sporządzić wykres tej charakterystyki.

Sprawdzenie działania regulatora w wersji **PI**:

Ustawić następujące wartości parametrów regulatora: $K_p = 10$, $T_i = 20$ [s], $K_d = 0$, wartość zadana 50% zakresu, po ustawieniu zapisać do pamięci (F7).

Odczekać, aż sygnał wyjściowy z regulatora osiągnie poziom 0%.

Zmienić skokowo sygnał wejściowy o 20% wartości (SP), zaobserwować przebieg sygnału wyjściowego z regulatora.

Sprawdzenie działania regulatora w wersji **PD**.

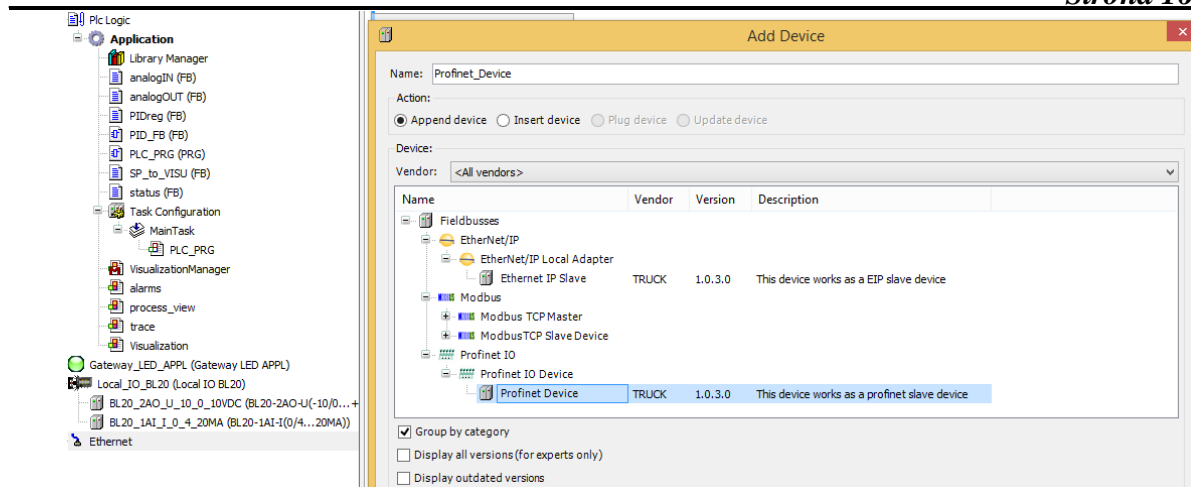
Podobnie jak w przypadku regulatora PI, ustawić następujące wartości parametrów regulatora: $K_p = 10$, $T_i = 0$ [s] $K_d = 10$, obserwować zachowanie regulatora.

Wnioski z obserwacji regulatora, w poszczególnych trybach pracy P,PI i PD należy zawrzeć w sprawozdaniu z ćwiczenia.

4. Wymiana danych procesowych w oparciu o przemysłową sieć PROFINET (opcjonalnie).

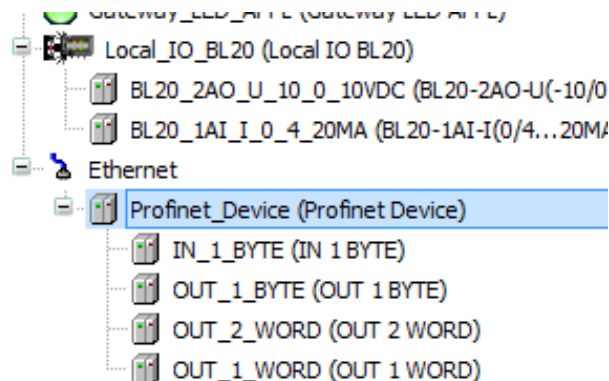
Konfiguracja sieci Profinet.

Sterownik TURCK może pracować jako urządzenie SLAVE w sieci Profinet. Interfejs Profinet, dodajemy poprzez wybór opcji (Add Device..) wybierając z drzewa projektu element *Ethernet*. Jako nazwę urządzenia, można przyjąć: proponowaną przez środowisko **Profinet_Device** (Rys.1.15).



Rys.1.15. Konfiguracja urządzenia typu slave w sieci Profibus

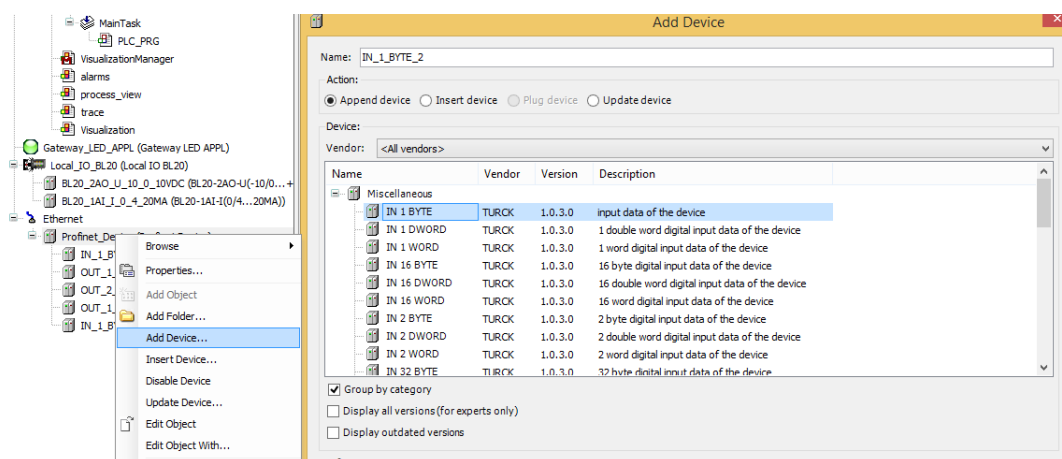
Następnie należy zdefiniować pakiety danych, które sterownik TURCK będzie wymieniał z systemem nadrzędnym (Siemens S7-1200). Interfejs Profinet w sterowniku TURCK pozwala na transfer 1024 bajtów danych (512 dane wejściowe i 512 dane wyjściowe). W dwóch bajtach (BYTE) – wymieniamy informacje o statusie, ze sterownika nadrzędnego (Rys.1.16), a także sygnały statusowe ze sterownika TURCK do S7 1200. W 3-ch słowach (WORD), przesyłamy informacje procesowe: wartość zadana (SP), wartość aktualna (PV) oraz stopień wysterowania zaworu regulacyjnego (OUT). Pakiety danych dodajemy kolejno, poprzez wybór opcji Add Device.. (drzewo projektu, Profinet_Device, podkatalog Ethernet (Rys.1.17). Organizacja pakietów danych została przedstawiona w tabeli 1.3.



Rys.1.16. Pakiety danych – sterownik TURCK BL-20 (Code SYS)

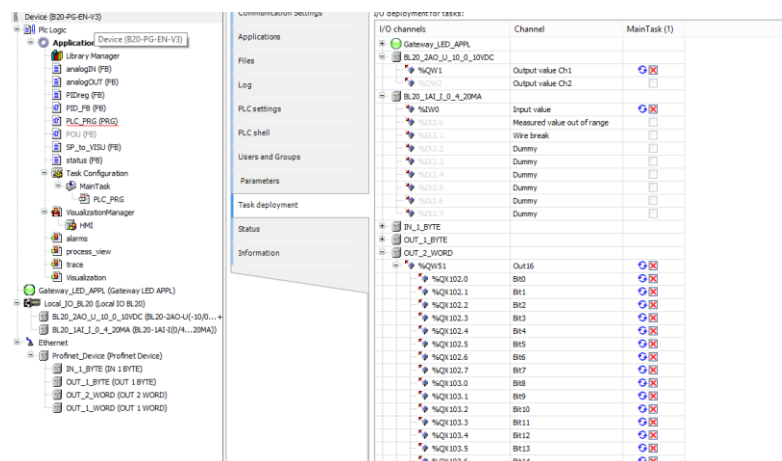
Tabela 1.3. Organizacja pakietów danych w sieci Profinet, dla sterownika TURCK.

nazwa	Typ zmiennej	Adres	Opis
OUT_1_BYTE	Byte (In8)	%QB100	
OUT0	Bool	%QX100.0	bit statusowy - "liveBit " ze sterownika TURCK
OUT1	Bool	%QX100.1	bit statusowy - wartość aktualna nie jest równa wartości zadanej
OUT2	Bool	%QX100.2	bit statusowy - sterownik pracuje bez błędów
OUT3	Bool	%QX100.3	bit statusowy - wartość aktualna = wartość zadana
OUT4	Bool	%QX100.4	bit statusowy - stanowisko w trybie sterowania ręcznego
OUT5-OUT7	Bool	%QX100.5 do 7	niewykorzystane
IN_1_BYTE	Byte (In8)	%IB100	
IN0	Bool	%IX100.0	bit statusowy - "liveBit " ze sterownika S7-1200
IN2-IN7	Bool	%IX100.1 do 7	niewykorzystane
OUT_2_WORD	Word (UINT)	%QW51	wartość ciśnienia w zbiorniku
OUT_2_WORD	Word (UINT)	%QW52	wartość ciśnienia zadanego (odczytanego z TURCK)
OUT_1_WORD	Word (UINT)	%QW53	stopień wysterowania zaworu w %



Rys.1.17. Definiowanie kontenerów dla danych dla potrzeb ich wymiany w sieci profibus.

Uwaga! Należy zmodyfikować bloki funkcyjne programu, tak aby wartości z procesu regulacji ciśnienia (PV, SP, OUT, AUTOM, MANUAL, etc.), były przepisywane do zmiennych: OUT_2_WORD, OUT_2_BYTE oraz OUT_1_WORD i OUT_1_BYTE. Ponadto w ustawieniach konfiguracyjnych bramy Device (B20-PG-EN_V3), zakładka: Task deployment (Rys.1.18), należy zaznaczyć wszystkie zmienne, które mają być aktualizowane podczas wywołania cyklicznego bloków programu (kolumna MainTask).



Rys.1.18. Aktualizacja wartości zmiennych.

Zakres wymaganych wiadomości

Elementy organizacyjne oprogramowania dla sterowników PLC, typy danych i zmiennych w systemach PLC, funkcje (FC) i bloki funkcyjne (FB) - wg normy IEC61131-3. Bloki programowe (PRG) charakterystyczne dla CODESYS. Zasady adresacji danych w sterownikach TURCK. Miejsce i rola przetworników pomiarowych w układzie regulacji. Budowa i zasada działania pojemnościowych i piezorezystancyjnych czujników ciśnienia. Cyfrowy algorytm PID w wersji pozycyjnej i przyrostowej, zalety i wady tych algorytmów. Charakterystyka sieci komunikacyjnej na przykładzie przemysłowego standardu PROFINET.

UWAGA! Przygotowując się do ćwiczenia nr 1, zaleca się zapoznanie się z funkcjami środowiska do programowania sterowników PLC - pakiet CODESYS. W ramach przygotowania do ćwiczenia, proszę wykonać konfigurację bramy TURCK BL20 w tym modułów analogowych (we/wy). Ćwiczenie aby zostało wykonane prawidłowo wymaga ścisłej współpracy z zespołem realizującym ćwiczenie na stanowisku nr 4. W ramach zadań należy m.in. sprawdzić wymianę danych w sieci przemysłowej PROFINET między sterownikami PLC: SIEMENS S7-1200 i TURCK BL20.

Literatura i dokumentacja techniczna

DTR aparatury dostępnej na stanowisku (katalog z dokumentacją).

Tomasz Gilewski: „Podstawy programowania sterowników SIMATIC S71-1200 w języku SCL”

Zakres wymaganych wiadomości.

Podstawowe wiadomości n/t zamkniętego układu regulacji temperatury: schemat blokowy zamkniętego układu regulacji i funkcje jego elementów, regulacja wprost i z inwersją, algorytmy PID, sterowanie PWM: zasada działania,

Podstawowe wiadomości n/t regulacji dwupołożeniowej: schemat układu regulacji dla obiektów statycznych i astatycznych, przebiegi wielkości regulowanej i sterowania w układzie, wpływ parametrów obiektu i przekaźnika na przebiegi wielkości regulowanej, wpływ poziomu wartości zadanej na średnią wartość przebiegu wielkości regulowanej.

Metody pomiaru temperatury z wykorzystaniem termopar oraz czujników rezystancyjnych: zakresy zastosowań poszczególnych typów czujników charakterystyki przetwarzania, układy połączeń, czynniki zakłócające pomiary.

Sprawozdanie.

Sprawozdanie powinno zawierać: uproszczony schemat układu regulacji (regulator z wyjściem sterującym i alarmowym, obiekt) opis poziomów dostępu, na których programuje się poszczególne parametry regulatora, wnioski końcowe (np. co jest przyczyną tego, że dla różnych poziomów wartości zadanej oraz różnych warunków pracy obiektu - chłodzenie lub jego brak, zostały wyznaczone różne nastawy).

Ćwiczenie 2. Układ regulacji temperatury (SIEMENS)

Cel ćwiczenia.

Zapoznanie się z budową i programowaniem sterownika SIEMENS SIMATIC S7-1200 oraz panelu operatorskiego SIMATIC HMI Basic w celu realizacji układu regulacji temperatury za pomocą pakietu narzędziowego SIMATIC STEP 7.

Opis aparatury.

Dokumentacja techniczna użytkownika sterownika SIEMENS SIMATIC S7-1200.

Budowa stanowiska laboratoryjnego.



Rys. 2.1. Stanowisko pracy.



Rys. 2.2. Panel operatorski

- Sterownik SIMATIC SIEMENS S7-1200 z wyjściem przekaźnikowym oraz panel operatorski SIMATIC HMI Basic.
- Obiekt regulacji typu cieplnego. Jest nim walec aluminiowy zamocowany w oporowym wkładzie grzejnym o mocy 400 W. Spirala grzejna jest zasilana impulsowo z wyjścia sterownika (poprzez przekaźnik mocy SSR) sygnałem sieciowym 230 V. Współczynnik wypełnienia sygnału sterującego jest modulowany poziomem sygnału wyjściowego sterownika. Na obu końcach walca są symetrycznie zamocowane dwa termometry Pt100. Jeden z nich jest połączony z wejściem sterownika, a drugi z gniazdami na przedniej ścianie obudowy. Czujnik Pt100 podłączony jest do sterownika w systemie czteroprzewodowym. Wystające z wkładu końce walca są karbowane w celu zwiększenia intensywności chłodzenia. Obiekt umieszczony jest w metalowej obudowie. W prawej ścianie obudowy jest wbudowany wentylator uruchamiany w celu dodatkowego chłodzenia obiektu.

Wykonanie ćwiczenia.

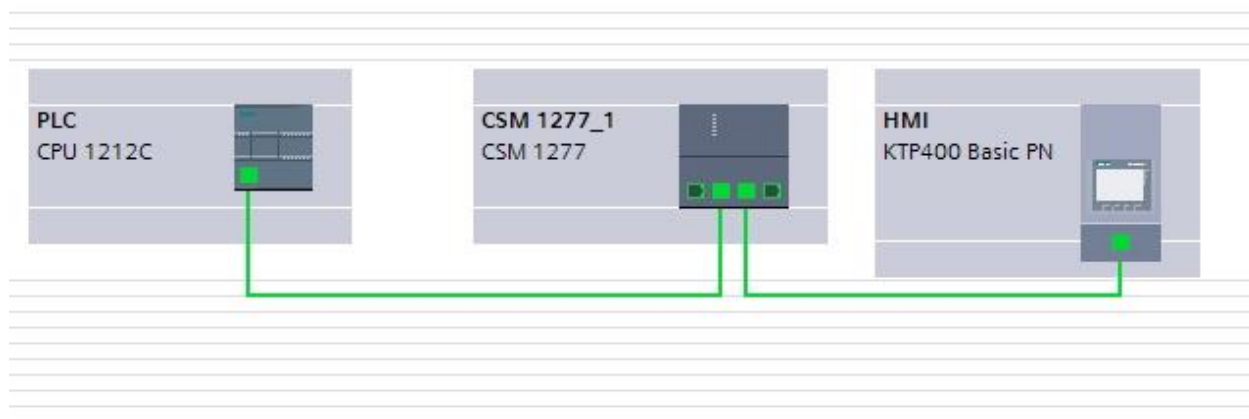
Skonfiguruj stanowisko (sprzęt i nazwy symboliczne zmiennych) wg. danych pokazanych na poniższych rysunkach i w tabelach.



Rys. 2.3. Konfiguracja stanowiska

Tabela 2.1. Konfiguracja sprzętu

Slot nr	Funkcja modułu	Oznaczenie	Numer katalogowy 6ES7 ...
1	Jednostka centralna	CPU	212-1BE40-0XB0
	Moduł wyjść PWM (mocowany bezpośrednio w slocie na CPU)	DQ 4x24VDC	222-1BD30-0XB0
2	Wyjścia analogowe	AQ2x14BIT	232-4HB32-0XB0
3	Wejścia analogowe	AI4xRTD	231-5PD32-0XB0
	Switch sieciowy	CSM 1277 SIMATIC NET	6GK7-277-1AA10-0AA0
	Panel Operatorski	HMI	6AV2 123-2DB03-0AX0

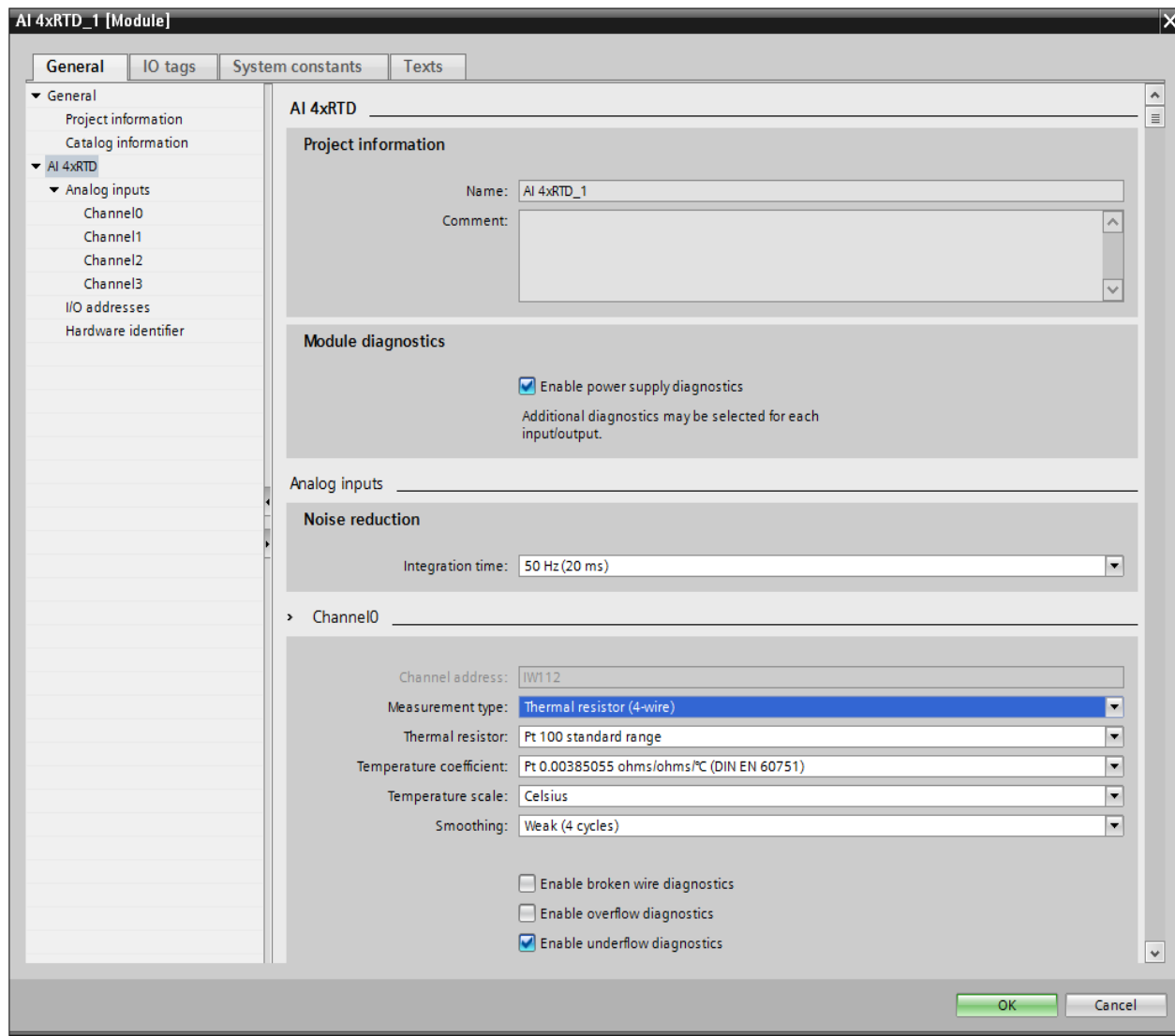


Rys. 2.4. Sieć

Tabela 2.2. Tablica nazw symbolicznych

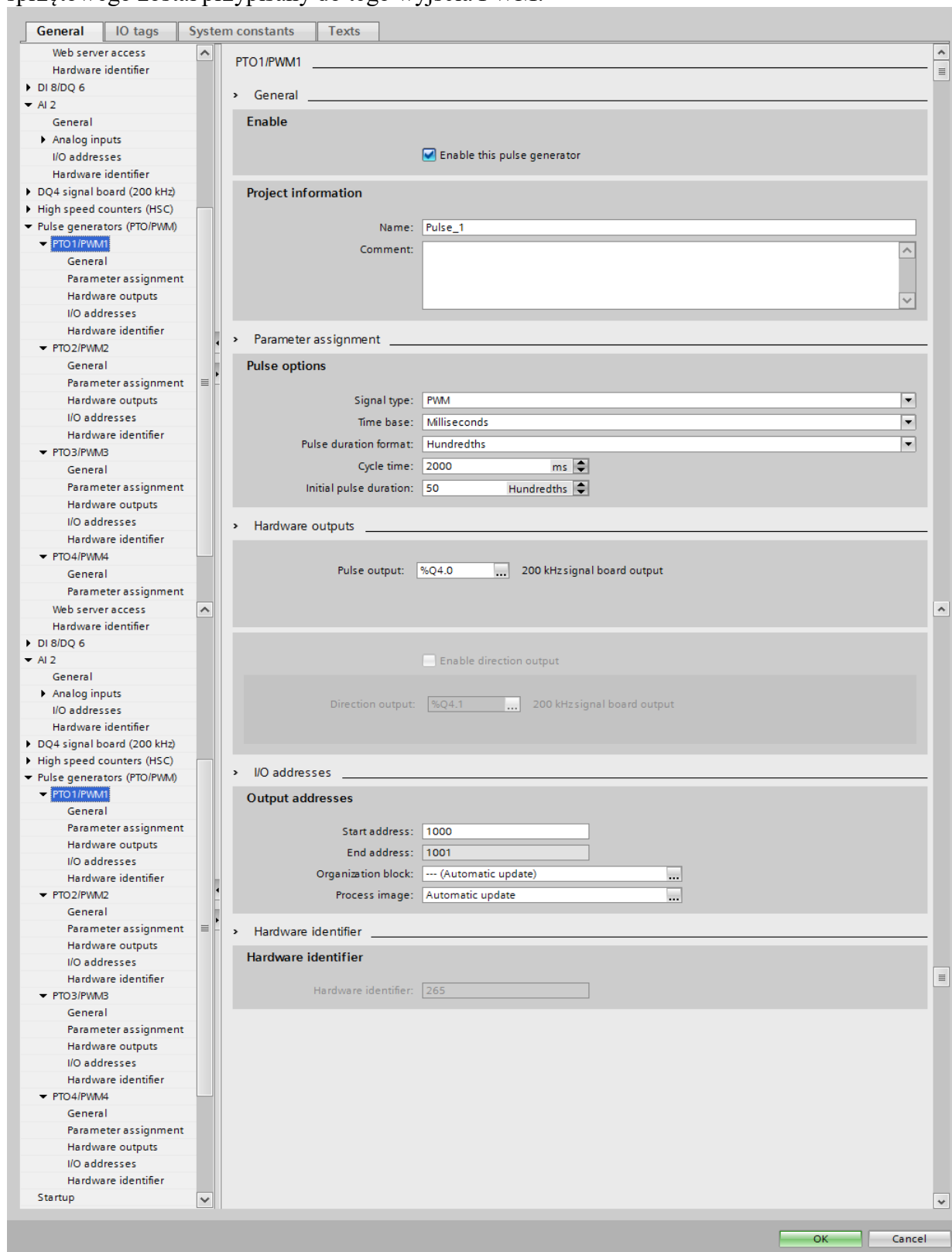
PLC tags							
	Name	Tag table	Data type	Address	Retain	Visibl...	Acces...
1	RTD_Input0	Default tag table	Word	%IW112	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	RTD_Input0_Tag	Default tag table	Int	%MW6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	PWM_Output0	Default tag table	Int	%QW1000	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	Heater_Value	Default tag table	Int	%MW2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	RLO_Fan	Default tag table	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	Fan_ON	Default tag table	Bool	%M128.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	Manual	Default tag table	Bool	%M128.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	poziom_0	Default tag table	Int	%MW10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	poziom_1	Default tag table	Int	%MW12	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	SP	Default tag table	Int	%MW14	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	<Add new>				<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Skonfiguruj wejście RTD (Drzewo Projektu -> PLC [CPU1212C AC/DC/Rly] -> Local Modules -> AI 4xRTD_1) zgodnie z parametrami pokazanymi na rys 2.5. W dokumentacji (S7-1200 Easy Book) sprawdź w jaki sposób jest reprezentowana temperatura odczytywana za pomocą czujnika Pt100 i używanego w ćwiczeniu modułu wejść analogowych.



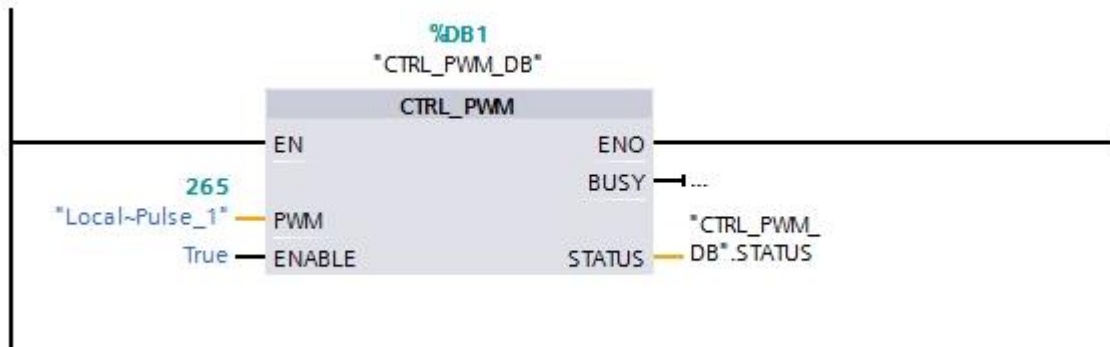
Rys. 2.5 Konfiguracja wejścia RTD

Skonfiguruj wyjście PWM1 (PLC [CPU1212C AC/DC/Rly] -> Properties -> Pulse Generators PTO1/PWM1) zgodnie z parametrami pokazanymi na rys 2.6. Sprawdź jaki numer identyfikatora sprzętowego został przypisany do tego wyjścia PWM.



Rys. 2.6. Konfiguracja wyjścia PWM

W Program blocks / Main [OB1] umieść blok CTRL_PWM i odpowiednio go skonfiguruj (zgodność numeru identyfikatora sprzętowego wyjścia PWM).



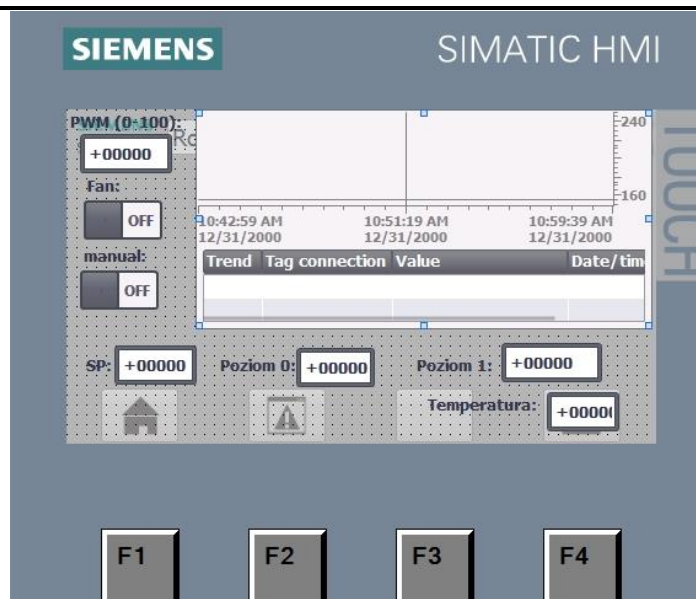
Rys. 2.7. Blok CTRL_PWM w bloku OB1.

W programie OB1 umieść bloki realizujące odczyt temperatury i przeliczające ten odczyt na stopnie Celsjusza (z dokładnością do jednego stopnia).

Stwórz funkcję (FC) realizującą regulator dwupołożeniowy bez histerezy. Wejściem tej funkcji ma być zadana wartość temperatury, aktualna wartość temperatury, wartość sterowania w stanie „0” oraz wartość sterowania w stanie „1”. Wynikiem działania tej funkcji ma być sterowanie przypisane do stanów „0” i „1” : jeśli wartość zmierzona jest równa lub większa od wartości zadanej to sterowanie przyjmuje poziom „0”; w przeciwnym przypadku sterowanie ma poziom „1”. Wyjście z tej funkcji jest podawane na sprzętowe wyjście PWM1 więc może ono przyjmować wartości z zakresu 0..100.

W programie OB1 utwórz możliwość nadpisywania sterowania wyliczonego przez funkcję regulatora dwupołożeniowego wartością ustawianą ręcznie. Możliwość ta ma być aktywowana przełącznikiem (sterowanie ręczne/automatyczne) umieszczonym na panelu operatorskim.

Zbuduj system SCADA na panelu operatorskim. Powinny się na nim znaleźć: przełącznik trybu pracy (ręczny/automatyczny), włącznik wentylatora, pola do zadawania wartości zadanej, sterowania w stanie „0”, sterowania w stanie „1”, wartości sterowania w trybie ręcznym, pole odczytu aktualnej temperatury oraz wykres trendu zmian temperatury. Wszystkie parametry powinny być aktualizowane na panelu operatorskim co 1 sekundę,



Rys. 2.8. Przykładowy ekran HMI

Eksperymenty pomiarowe.

Należy zbadać wpływ wartości zadanej oraz wartości sterowania (w stanie „0” i „1”) na stabilizację temperatury. **Wszystkie eksperymenty należy przeprowadzać przy włączonym na stałe wentylatorze.**

- Ustaw (z poziomu panelu operatorskiego) sterowanie w stanie „0”= 0% a w stanie „1”= 100 %. Ustaw temperaturę zadaną na 50°C. Zarejestruj (ręcznie – na kartce) oscylacje temperatury (temperaturę minimalną i maksymalną) wokół wartości zadanej w stanie ustalonym. Eksperyment powtórz dla temperatur zadanych 150°C i 250°C.
- Ustaw temperaturę zadaną równą 150°C a następnie zarejestruj oscylacje temperatury wokół wartości zadanej dla sterowań w stanie „1” równych 40% i 70%. W obu przypadkach sterowanie w stanie „0” ma być równe 0%.

Zakres wymaganych wiadomości.

Podstawowe wiadomości n/t programowania w języku drabinkowy LD (bloki kodu i adresacja), zamkniętego układu regulacji temperatury: schemat blokowy zamkniętego układu regulacji i funkcje jego elementów, sterowanie PWM: zasada działania,

Podstawowe wiadomości n/t regulacji dwupołożeniowej: schemat układu regulacji dla obiektów statycznych i astatycznych, przebiegi wielkości regulowanej i sterowania w układzie, wpływ parametrów obiektu i przekaźnika na przebiegi wielkości regulowanej, wpływ poziomu wartości zadanej na średnią wartość przebiegu wielkości regulowanej.

Metody pomiaru temperatury z wykorzystaniem termopar oraz czujników rezystancyjnych: zakresy zastosowań poszczególnych typów czujników charakterystyki przetwarzania, układy połączeń, czynniki zakłócające pomiary.

Sprawozdanie.

Sprawozdanie ska powinno zawierać: opis konfiguracji stanowiska, schemat drabinkowy realizacji zadania, listing realizacji funkcji regulatora dwupołożeniowego, opis zmiennych aplikacji SCADA, widok panelu operatorskiego, tabelaryczne zastawienie wyników eksperymentów pomiarowych (wraz z ich dyskusją) oraz wnioski końcowe.

Ćwiczenie 3. Sterowanie układem lewitacji powietrznej.

Cel ćwiczenia i opis stanowiska

Celem ćwiczenia jest konfiguracja programowa i testy działania bazującego na PLC systemu sterowania układem lewitacji powietrznej. Stanowisko laboratoryjne (rys 3.1) składa się ze sterownika SIEMENS SIMATIC S7-1200 oraz przezroczystej rury, w której porusza się piłeczka ping-pong'owa. Na dolnym końcu rury umieszczony jest laserowy czujnik odległości a na górnym końcu wentylator napędzany silnikiem elektrycznym DC. Wentylator zasysa powietrze przez rurę (od dołu ku górze) powodując unoszenie piłeczki. Na rurze umieszczona jest skala metrowa pozwalająca na odczyt pozycji piłeczki. Rura prawie na całym obwodzie osłonięta jest czarnym papierem, aby zmniejszyć wpływ światła z otoczenia na wskazania laserowego czujnika odległości oraz zabezpieczyć osoby realizujące ćwiczenie przed ewentualnymi odbiciami wiązki laserowej. Na przodzie rury znajduje się szczelina pozwalająca na określenie pozycji piłeczki.



Rys. 3.1 Widok stanowiska laboratoryjnego

Etapy realizacji ćwiczenia:

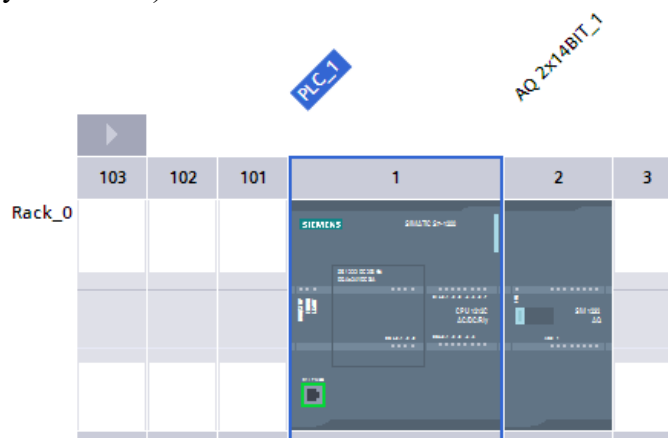
1. Utworzenie nowego projektu w TIA PORTAL oraz odpowiednie skonfigurowanie sterownika SIEMENS SIMATIC S7-1200 obejmujące konfigurację sprzętową sterownika, a w szczególności:
 - wejścia analogowego
 - wejścia dyskretnego
 - wyjścia PWM
 - wejścia szybkiego licznika w układzie pomiaru częstotliwości
 - konfigurację panelu operatorskiego
2. Zapoznanie się z sposobem działania laserowego czujnika odległości
3. Realizacja sterowania prędkością obrotową silnika DC (wentylatora) za pomocą sygnału PWM
4. Pomiar prędkości obrotowej silnika
5. Budowa układu stabilizacji pozycji piłeczki z wykorzystaniem regulatora PID (z opcją sterowania ręcznego)
6. Budowa aplikacji SCADA na panelu operatorskim z wizualizacją: pozycji piłeczki, wartości sygnału PWM, prędkości obrotowej wentylatora, parametrów regulatora PID

oraz wartości sterowania ręcznego. Z poziomu panelu operatorskiego ma być możliwa zmiana wartości sterowania ręcznego oraz parametrów regulatora. Na panelu ma się znaleźć przełącznik trybów pracy układu (automatyczny/ręczny) oraz wyłącznik blokujący wysyłanie sygnału PWM do wentylatora.

7. Automatyczny dobór nastaw regulatora PID w oparciu o narzędzia wbudowane w TIA PORTAL.
8. Przygotowanie sprawozdania.

Konfiguracja stanowiska

Na początku ćwiczenia należy przeprowadzić odpowiednią konfigurację stanowiska w TIA PORTAL. Można do tego wykorzystać projekt stworzony w ćwiczeniu nr 0 (wczytując go z nośnika pamięci USB i po niezbędnych modyfikacjach zapisując pod nową nazwą) lub otworzyć całkowicie nowy projekt w TIA PORTAL i całą konfigurację stworzyć od nowa (nie zalecane ze względu na potrzebny na to czas).

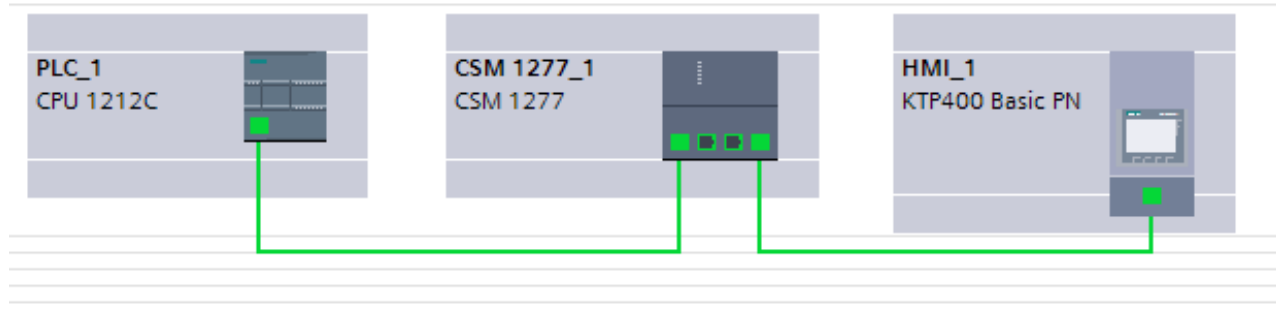


Rys. 3.2 Układ modułów w poszczególnych slotach

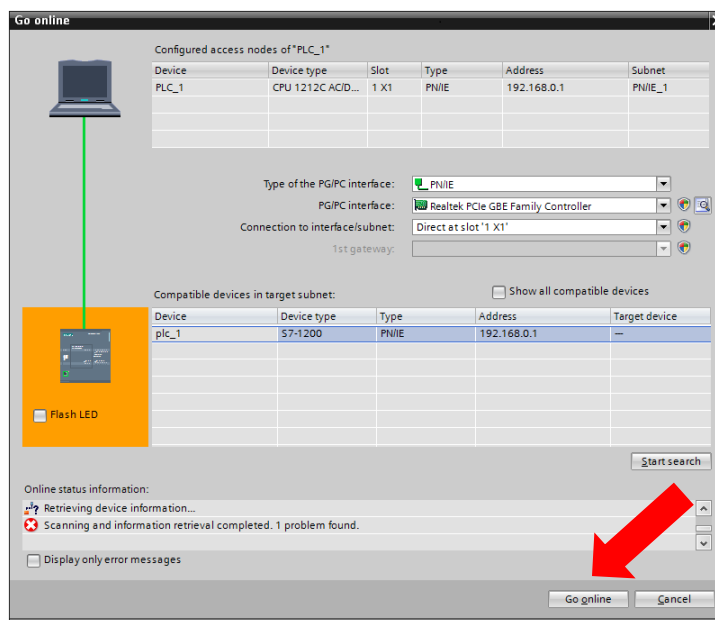
Tabela 3.1 Konfiguracja sprzętowa stanowiska

Slot nr.	Funkcja modułu	Oznaczenie	Numer katalogowy 6ES7 ...
1	Jednostka centralna	CPU 1212C AC/DC/RLY	212-1BE40-0XB0
	Moduł wyjść PWM (mocowany bezpośrednio w slotcie na CPU)	DC 200kHz DQ 4x24VDC	222-1BD30-0XB0
2	Wyjścia analogowe	SM 1232 AQ AQ2x14BIT	232-4HB32-0XB0
	Switch sieciowy	CSM 1277 SIMATIC NET	6GK7-277-1AA10-0AA0
	Panel Operatorski	KTP-400	6AV2 123-2DB03-0AX0

Analogicznie jak w ćwiczeniu 0 dokonać konfiguracji portów sieciowych (rys. 3.3-3.5):



Rys. 3.3 Topologia połączeń sieciowych



Rys. 3.4 Okno konfigurowania adresów sieciowych



Rys. 3.5 Zatwierdzenie nowego adresu

Zdefiniować tablicę nazw symbolicznych (PCL Tags) wg. poniższego wzoru:

	Name	Tag table	Data type	Address	Retain	Visibl...	Acces...
1	czujnik_odleglosci	Default tag table	Word	%IW64	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	pozycja_kulki	Default tag table	Word	%MWO	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	PWM_Strat	Default tag table	Bool	%M128.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	Zakres_ok	Default tag table	Bool	%M128.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	Poza_zakresem	Default tag table	Bool	%IO.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	Stan_PWM	Default tag table	Word	%MW2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	Zadany_PWM	Default tag table	Word	%MW4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	Wartosc_PWM	Default tag table	Word	%QW1000	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	Manual_ON	Default tag table	Bool	%M128.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	War_zadana	Default tag table	Real	%MD6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	Ster_manual	Default tag table	Real	%MD10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	obroty	Default tag table	DWord	%ID1004	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	RPM	Default tag table	DWord	%MD14	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	bład_poz	Default tag table	Real	%MD20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	<Add new>				<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Rys. 3.6 Tablica nazw symbolicznych

Laserowy czujnik odległości OADK 25U748/S14C

Użyty w ćwiczeniu czujnik odległości działa w zakresie od 10 do 100 cm. Pomiaru odległości dokonuje w oparciu o pomiar kąta padania na czujnik wiązki laserowej odbitej od przedmiotu, do którego mierzona jest odległość. Dokładność pomiaru oraz jego niepewność w funkcji odległości podane są w nocie katalogowej tego czujnika. Sygnałem wyjściowym z czujnika jest napięcie w zakresie 0-10V. Sygnał ten jest podłączony do wejścia analogowego **AI.0** sterownika (adres **%IW64**). Czujnik posiada również wyjście dyskretne. Jest ono aktywowane (osiąga stan wysoki) jeśli odległość do mierzonego przedmiotu jest mniejsza niż 10cm lub większa niż 100cm. Wyście dyskretne czujnika podłączone jest do wejścia dyskretnego DI0.0 sterownika (adres **%IO.0**).

W oknie programu głównego Main [OB1] należy umieścić elementy języka drabinkowego pozwalające na odczyt obu wymienionych wyżej sygnałów (analogowego i dyskretnego), a następnie na panelu operatorskim umieścić wskaźniki pozycji kulki (pole cyfrowe i bargraf) oraz wskaźnik (np. lampkę sygnalizacyjną) pokazującą, że piłeczka znajduje się poza zakresem pomiarowym czujnika).

Jeśli powyższe czynności zostały wykonane, to należy zwrócić się do opiekuna o przekonfigurowanie stanowiska do opcji ręcznej zmiany położenia piłeczki (usunięcie głowicy z wentylatorem oraz umieszczenie w rurze piłeczki przymocowanej na lince. Następnie - trzymając w ręce linkę - należy ręcznie zmieniać pozycję piłeczki w całym zakresie ruchu z krokiem 5cm. Za pozycję piłeczki należy przyjąć pozycję jej dolnej krawędzi (najniższy punkt). W każdej z pozycji należy zanotować odczytaną i wyświetloną na panelu operatorskim pozycję piłeczki. Pomiaru należy wykonać dwukrotnie (raz przy ruchu od dołu do góry, drugi raz przy ruchu od góry ku dołowi). Na podstawie wykonanych pomiarów należy przygotować do sprawozdania odpowiednie wykresy a ich przebieg skomentować odnosząc się do parametrów czujnika umieszczonych w nocie katalogowej. W sprawozdaniu należy również umieścić zrzuty ekranu obrazujące stworzony panel operatorski oraz program w języku drabinkowym.

Po wykonaniu tej części należy poprosić opiekuna o ponowne przekonfigurowanie stanowiska do układu automatycznego sterowania pozycją piłeczki.

Sterowanie prędkością obrotową wentylatora DC oraz pomiar jego prędkości obrotowej.

Zagadnienia realizowane w tej części ćwiczenia są bardzo szczegółowo opisane w rozdziale 9 książki J.Kwaśniewskiego :”Sterowniki SIMATIC S7-1200 w praktyce inżynierskiej.”

Przypisanie fizycznych wyjść i wejść do poszczególnych generatorów PTO/PWM oraz liczników HSC jest stałe i nie można go zmieniać programowo.

Opis	Domyślne przypisanie wyjść	Impuls	Kierunek
PTO 1	Wyjścia CPU	Q0.0	Q0.1
	Wyjścia płytki sygnałowej	Q4.0	Q4.1
PWM 1	Wyjścia CPU	Q0.0	–
	Wyjścia płytki sygnałowej	Q4.0	–
PTO 2	Wyjścia CPU	Q0.2	Q0.3
	Wyjścia płytki sygnałowej	Q4.2	Q4.3
PWM 2	Wyjścia CPU	Q0.2	–
	Wyjścia płytki sygnałowej	Q4.2	–

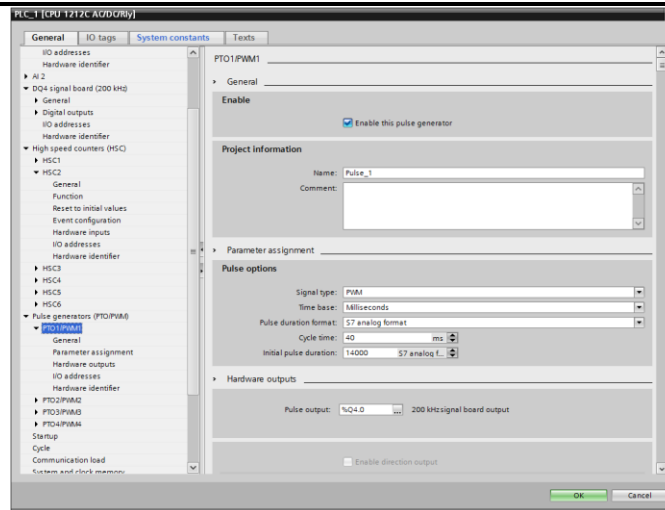
Rys. 3.7 Powiązanie wyjść fizycznych z poszczególnymi generatorami PTO/PWM (zaznaczono użyte w ćwiczeniu wyjście)

Ze względu na wykorzystanie wejścia dyskretnego DI0.0 do obsługi sygnału błędu pomiaru z czujnika laserowego pomiar prędkości obrotowej pomiar prędkości obrotowej wentylatora zrealizowano w oparciu o wejście dyskretno DI0.2 i szybki licznik HSC2.

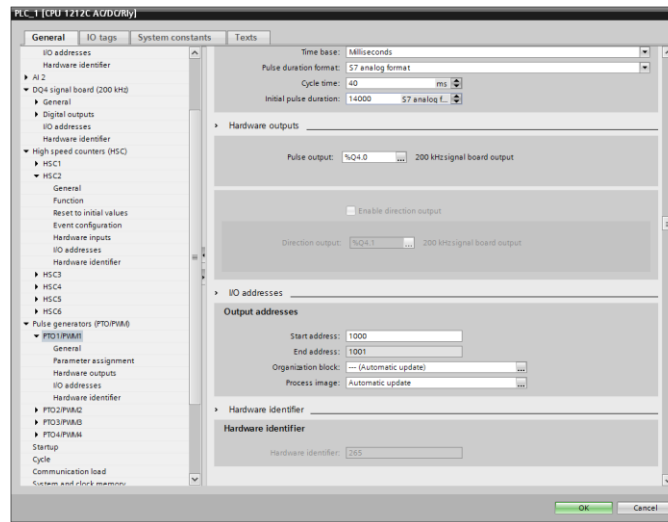
HSC	Wejścia CPU (0.x)							Wejścia SB (4.x) ³				
	0	1	2	3	4	5	6	7	0	1	2	3
HSC 1 ¹	1-fazowe	C	[d]	[R]					C	[d]	[R]	
	2-fazowe	CU	CD	[R]					CU	CD	[R]	
	AB-kwadraturowe	A	B	[R]					A	B	[R]	
HSC 2 ¹	1-fazowe	[R]	C	[d]					[R]	C	[d]	
	2-fazowe	[R]	CU	CD					[R]	CU	CD	
	AB-kwadraturowe	[R]	A						[R]	A		
HSC 3	1-fazowe				C	[d]	[R]					
	2-fazowe				CU	CD	[R]					
	AB-kwadraturowe				A	B	[R]					
HSC 4	1-fazowe				[R]	C	[d]					
	2-fazowe				[R]	CU	CD					
	AB-kwadraturowe				[R]	A	B					
HSC 5 ²	1-fazowe							C	[d]	[R]		
	2-fazowe							CU	CD	[R]		
	AB-kwadraturowe							A	B	[R]		
HSC 6 ²	1-fazowe							[R]	C	[d]		
	2-fazowe							[R]	CU	CD		
	AB-kwadraturowe							[R]	A	B		

Rys. 3.8 Powiązanie fizycznych wejść z szybkimi licznikami (zaznaczono użyte w ćwiczeniu wejście)

Sterowanie wentylatora podłączone jest (poprzez stopień mocy) do wyjścia PTO1/PWM1 sterownika. Na samym początku należy to wyjście odpowiednio skonfigurować. W tym celu w oknie drzewa projektu należy kliknąć prawym klawiszem myszy na zakładkę jednostki centralnej (PLC[CPU]), kliknąć opcję **Properties** i w otwartym po lewej stronie oknie wybrać zakładkę **Pulse Generators (PTO/PWM)**. W tym miejscu należy skonfigurować PTO1/PWM1 w tryb wyjścia PWM z parametrami pokazanymi na rys. 3.9 i 3.10 (typ sygnału: **PWM**, podstawa czasu: **milisekundy**, format wartości pulsu: **S7 analog format**; okres cyklu **40ms**).

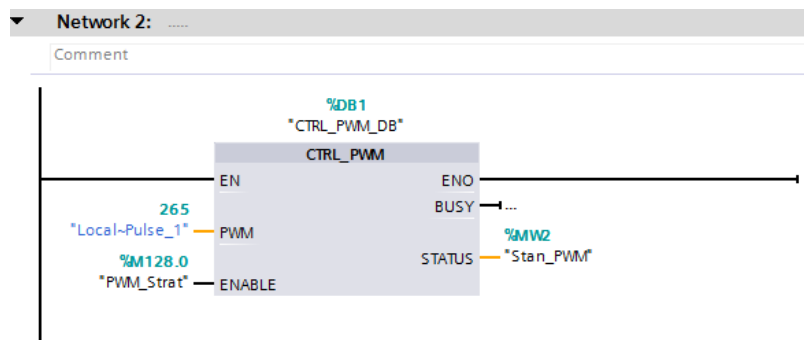


Rys. 3.9 Konfiguracja wyjścia PWM cz.1



Rys. 3.10 Konfiguracja wyjścia PWM cz.2

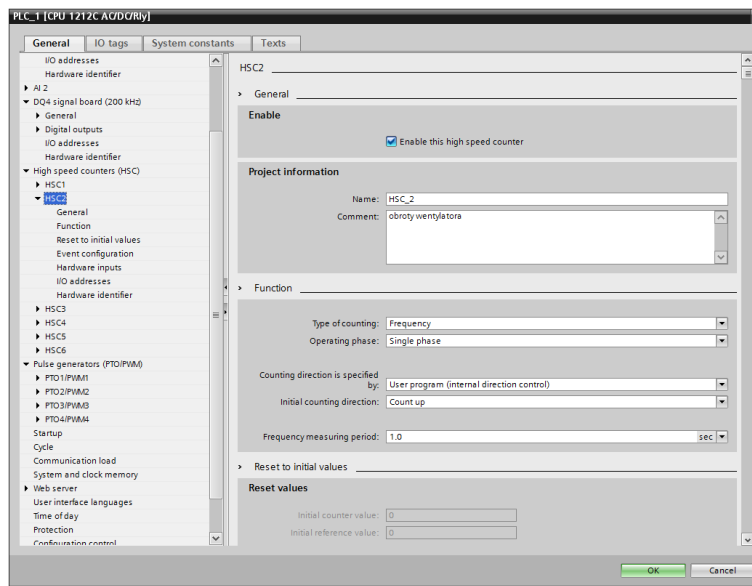
W programie głównym (OB1) należy umieścić blok kontroli wyjścia PWM.



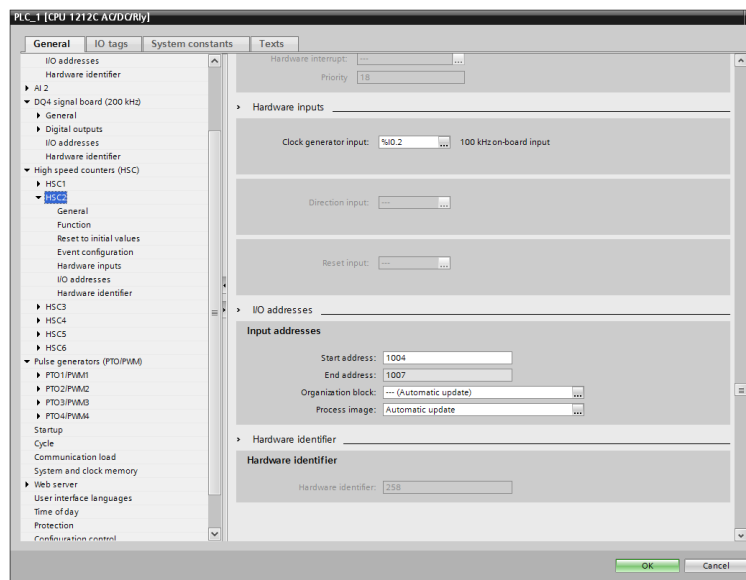
Rys. 3.11 Blok kontroli wyjścia PWM

Wejście „Enable” bloku „CTRL_PWM_DB” należy połączyć ze zmienną wewnętrzną „PWM_Start”, której wartość powinna być ustalana przełącznikiem na panelu operatorskim. Pozwoli to na wyłączenie wentylatora w dowolnym momencie niezależnie od działania reszty programu.

Konfigurację szybkiego licznika HSC należy przeprowadzić analogicznie jak konfigurację generatora PWM. W tym celu w oknie drzewa projektu należy kliknąć prawym klawiszem myszy na zakładkę jednostki centralnej (PLC[CPU]), kliknąć opcję **Properties** i w otwartym po lewej stronie oknie wybrać zakładkę **High Speed Counters (HSC)**. Następnie należy wprowadzić parametry licznika HSC2: typ zliczania: **częstotliwość**, tryb pracy: **jedna faza**, kierunek zliczania: **wyberany programowo**, początkowy kierunek zliczania: **do góry**, okres zliczania: **1 sek.** (rys 3.12 i 3.13).



Rys. 3.12 Konfiguracja szybkiego licznika HSC2 cz.1



Rys. 3.13 Konfiguracja szybkiego licznika HSC2 cz.2

Do poprawnej współpracy szybkiego licznika HSC2 z sygnałem pochodzącym od wentylatora użytego w ćwiczeniu niezbędna jest zmiana domyślnej filtracji sygnału na wejściu dyskretnym DI0.2 z wartości 6.4 ms na 1.6 ms. Konfigurację przeprowadza się w drzewie projektu ((PLC[CPU]), kliknąc opcję **Properties** i w otwartym po lewej stronie oknie wybrać zakładkę **DI8/DQ6 Digital Inputs**, następnie **Channel2**, i zmienić wartość parametru **Input Filters** z 6.4 ms na 1.6 ms.

Po odpowiednim skonfigurowaniu obsługi sygnałów sterowania i pomiaru prędkości wentylatora należy sprawdzić poprawność ich działania. Aby to zrobić trzeba na panelu operatorskim umieścić pola cyfrowe do zadawania wartości sterowania PWM oraz odczytu prędkości obrotowej wentylatora. Początkowo można bezpośrednio odczytywać częstotliwość sygnału pojawiającego się na wejściu DI0.2 (wejście licznika HSC2) . Jednak po sprawdzeniu poprawności odczytów należy je przeliczyć (poprzez odpowiedni blok w programie głównym) na „obroty na minutę” (RPM). **Przy przeliczaniu trzeba uwzględnić fakt, że wentylator podaje dwa impulsy na jeden obrót wirnika!**

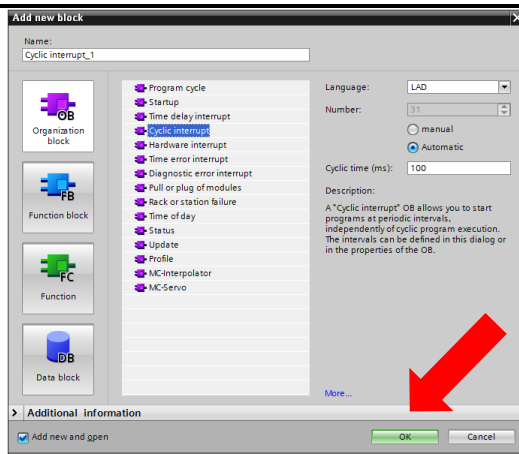
Mając możliwość zadawania sterowania PWM oraz odczytu prędkości obrotowej wentylatora (RPM) należy przebadać zachowanie się całego układu sterowania tj. odnaleźć prędkość obrotową wentylatora (i powiązane z nią sterowanie PWM). przy której piłeczka zaczyna podnosić się do góry, oraz prędkość przy której piłeczka zaczyna opadać. Badając układ należy zwrócić uwagę na następujące cechy układu sterowania:

- minimalna wartość sterowania PWM, przy której zaczyna obracać się wentylator,
- maksymalna prędkość obrotowa wentylatora,
- histereza między wartościami sterowania (i prędkości obrotowej wentylatora) powodującymi unoszenie się i opadanie piłeczki.

Obserwacje i wnioski z tej części ćwiczenia należy umieścić w sprawozdaniu.

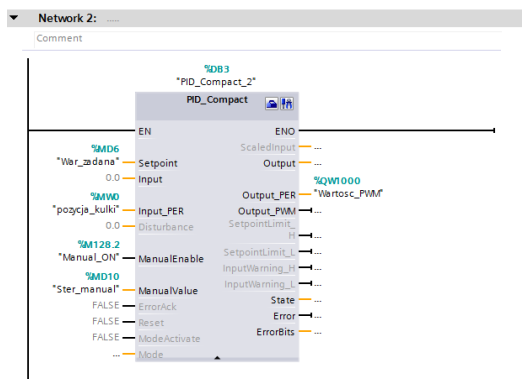
Zastosowanie regulatora PID.

Kolejnym etapem ćwiczenia jest zastosowanie regulatora PID (z opcją sterowania ręcznego) do stabilizacji pozycji piłeczki. Regulator PID umieszczamy w bloku przerw cyklicznych [OB30]. Aby utworzyć blok przerw cyklicznych należy w drzewie projektu wybrać kolejno **PLC_1[CPU1212C AC/DC/Rly]** -> **Program blocks** -> **Add new block** i w oknie, które się otworzy wybrać **Organization block** -> **Cyclic interrupt**. Dla tworzonego bloku należy pozostawić domyślny parametr okresu wyzwalania czyli **100 ms** (rys. 3.14).



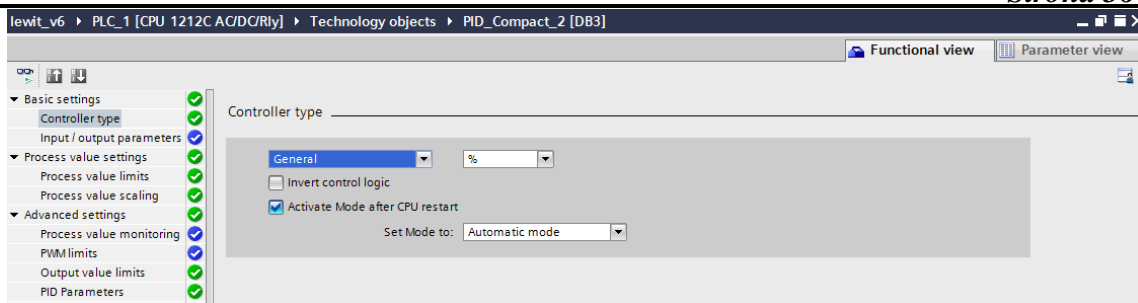
Rys. 3.14 Dodawanie i konfigurowanie bloku przerw cyklicznych

Po stworzeniu bloku przerw cyklicznych należy w nim umieścić regulator PID. Przypisanie nazw symbolicznych do poszczególnych wejść i wyjść regulatora pokazano na rys. 3.15.

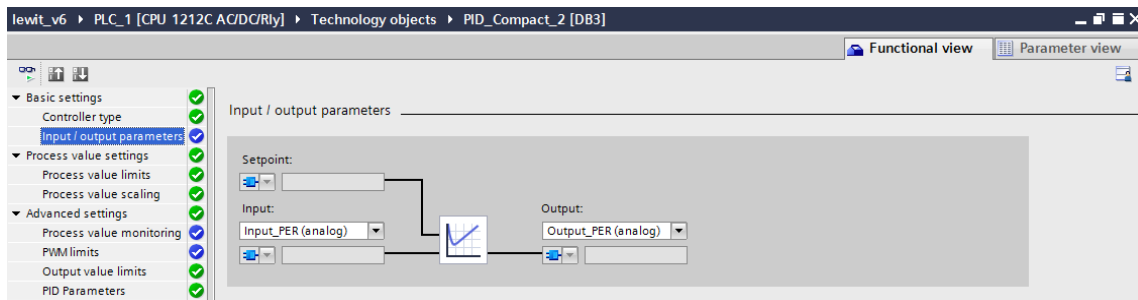


Rys. 3.15 Konfiguracja regulator PID w bloku przerw cyklicznych OB30

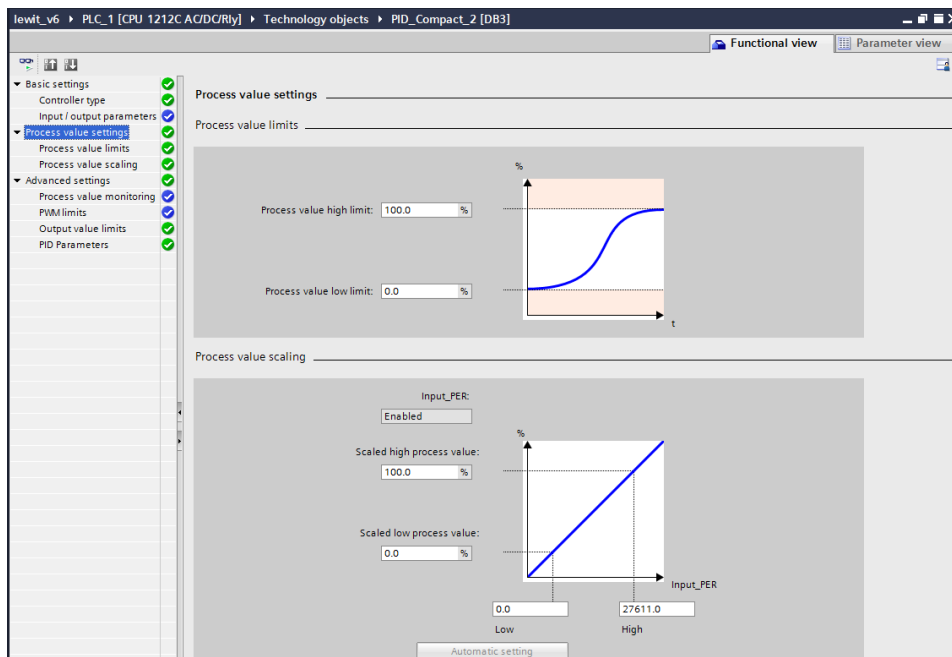
Należy również odpowiednio skonfigurować sam regulator PID (rys. 3.16). Wejście i wyjście z tego regulatora powinno być typu „PER” (rys. 3.17) z odpowiednim skalowaniem (rys. 3.18) oraz ograniczoną wartością wyjściową (rys. 3.19). Na rysunku 3.19 pokazano również wstępne wartości nastaw regulatora PID. Ograniczenie wartości wyjściowej wynikające z jednej strony z minimalnego sterowania PWM powodującego obrót wentylatora oraz z minimalnej prędkości obrotowej powodującej unoszenie się piłeczki a z drugiej strony z ograniczenia prędkości obrotowej aby (przy zbyt dużym sterowaniu) piłeczka zbyt szybko nie wyszła poza górny zakres pomiaru czujnika odległości. Eksperymentalnie ustalono, że górnym ograniczeniem sterowania powinna być jego wartość o ok. 20% przekraczająca sterowanie powodujące początek unoszenia się piłeczki do góry. Warto zaznaczyć, że w przypadku wejść typu „PER” regulator dokonuje wewnętrznego przeliczenia sygnału na skalę procentową (od 0 do 100%). **W takiej skali należy również zadawać regulatorowi wartość sterowania ręcznego.** Wyjście regulatora należy przypisać do zmiennej symbolicznej „Wartosc_PWM” (adres %QW1000) będącej bezpośrednią wartością przekazywaną na wyjście PWM1. **Należy również usunąć z programu ręczne nadawanie wartości „Wartosc_PWM” ponieważ może to skutkować nadpisywaniem wartości wyliczonej przez regulator PID.**



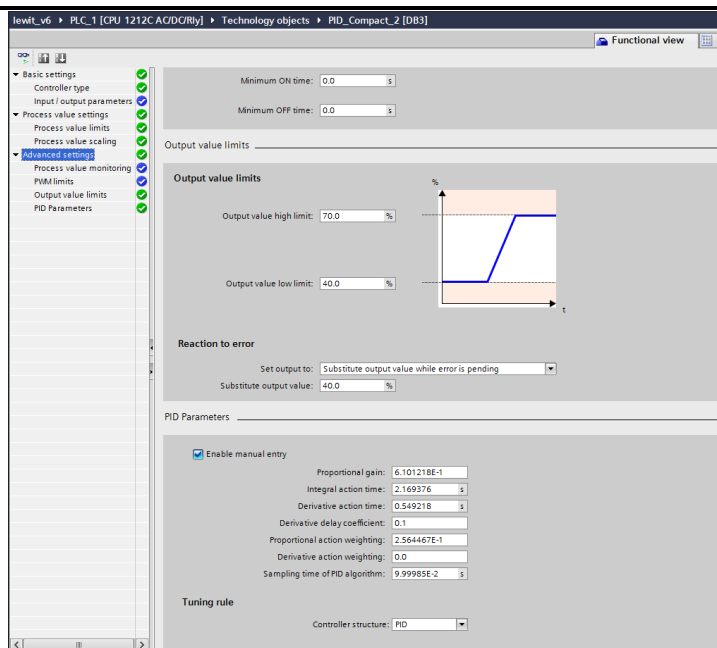
Rys. 3.16 Wybór typu regulatora PID



Rys. 3.17 Definiowanie rodzaju wejść i wyjść regulatora PID

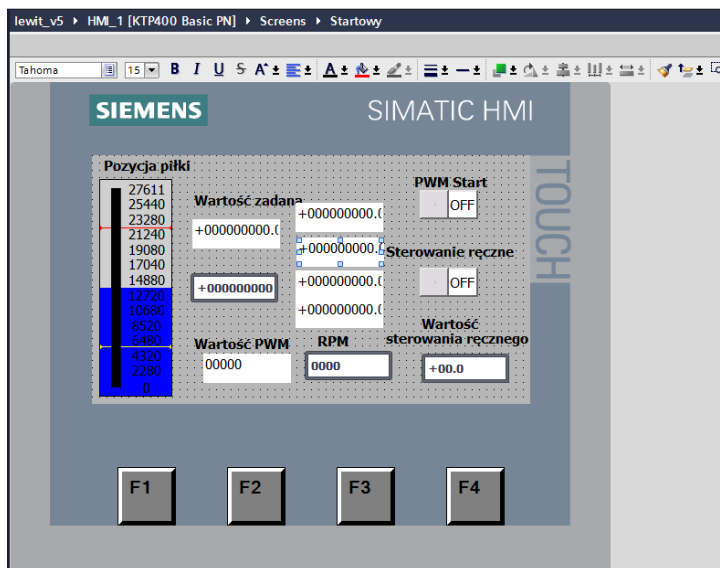


Rys. 3.18 Skalowanie wartości wejścia regulatora PID



Rys. 3.19 Ograniczenie zakresu wartości wyjściowej oraz wstępne nastawy regulatora PID

Na panelu operatorskim należy umieścić pola do ustawiania wartości zadanej, parametrów regulatora, wartości sterowania ręcznego oraz wizualizacji aktualnego położenia piłeczki, prędkości obrotowej wentylatora i uchybu sterowania (wymaga to zastosowania również odpowiedniego bloku w programie głównym). Widok przykładowego panelu operatorskiego pokazano na rys. 3.20.



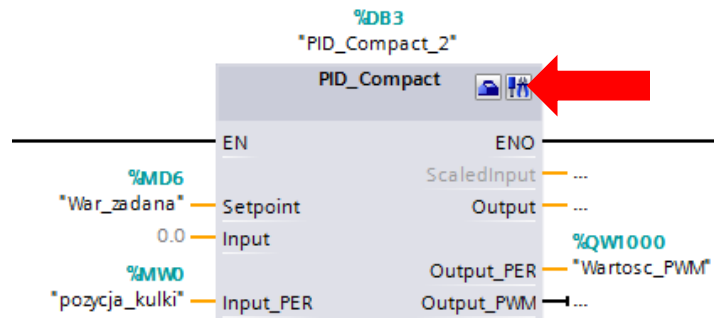
Rys. 3.20 Przykładowy panel operatorski SCADA dla układu sterowania lewitacją powietrzną

Na tym etapie należy przeprowadzić szereg testów funkcjonalnych zbudowanego układu regulacji. Zmiany parametrów należy dokonywać z panelu operatorskiego. Zmieniając skokowo wartość zadaną należy obserwować zmianę pozycji piłeczki oraz stabilność utrzymywania pozycji (wartość uchybu). Po przejściu na sterowanie ręczne należy powtórzyć procedurę

wyszukiwania wartości sterowania, przy którym piłeczka zaczyna się podnosić/zaczyna opadać/nie zmienia pozycji. Tym razem wartość sterowania będzie określona w skali od 0 do 100%. Wnioski z obserwacji należy umieścić w sprawozdaniu.

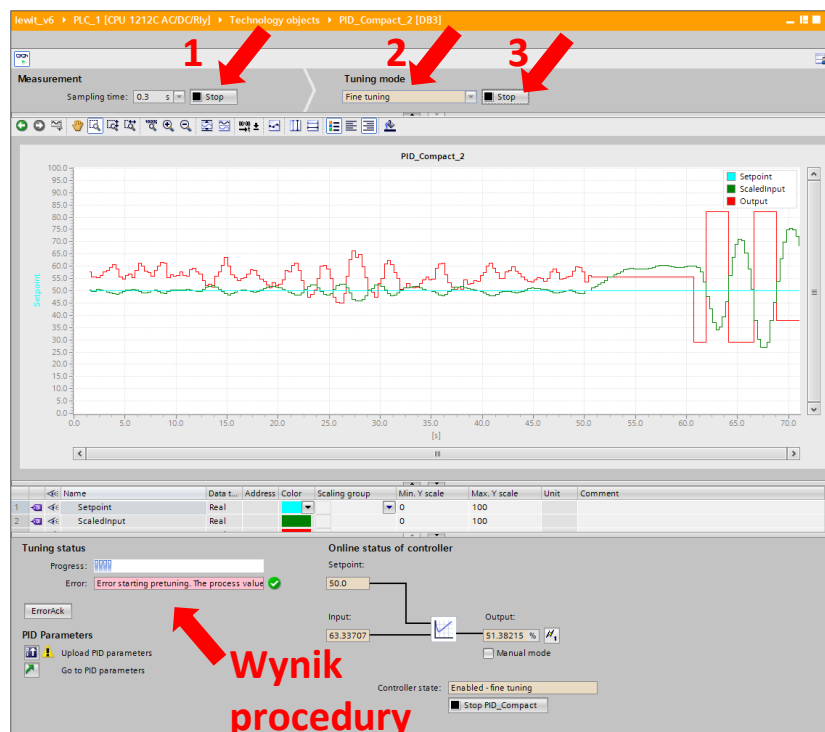
Automatyczny dobór nastaw regulatora PID

Używany w ćwiczeniu sterownik PLC pozwala na stosowanie procedury automatycznego doboru nastaw regulatora PID. W tym celu na bloku „PID_compact” należy kliknąć odpowiedni symbol (rys. 3.21).



Rys. 3.21 Uruchomienie procedury samostrojzenia regulatora PID

W wyniku tego działania otworzy się okno procedury samostrojzenia (rys. 3.22). Są do dyspozycji dwie procedury: wstępna (Pretuning) oraz dokładana (Fine tuning). **W dokumentacji sterownika należy sprawdzić kiedy, której z nich można użyć!** Kolejność działań jest następująca: włączamy monitorowanie parametrów procesu (1), wybieramy rodzaj procedury (2), uruchamiamy procedurę (3).



Rys. 3.22 Automatyczny dobór nastaw regulatora PID

Przebieg procedury automatycznego doboru nastaw regulatora PID należy udokumentować i umieścić w sprawozdaniu, a efekt działania nowych nastaw regulatora przetestować tak samo jak w przypadku nastaw wstępnych i opisać to w sprawozdaniu.

Zakres wiadomości na kolokwium.

Charakterystyka sterownika Siemens Simatic S7-1200 (w konfiguracji używanej w ćwiczeniu) od strony sygnałów wejściowych i wyjściowych. Podstawy programowania w języku drabinkowym, elementy oprogramowania PLC znormalizowane: funkcje (FC) i bloki funkcyjne (FB) oraz charakterystyczne dla systemu SIEMENS: bloki organizacyjne (OB), Bloki Danych (DB) i Typy Danych PLC. Zasada sterowania silnikami prądu stałego za pomocą sygnału PWM. Metoda pomiaru prędkości obrotowej (oraz kierunku obrotu) z wykorzystaniem wejść szybkich liczników. Zasada działania laserowego czujnika odległości OADK 25U748/S14C.

Sprawozdanie.

Sprawozdanie powinno zawierać opis funkcjonalny stanowiska ćwiczeniowego – użyte elementy i sposób ich połączenia, skomentowane programy w języku drabinkowym realizujące poszczególne etapy ćwiczenia, wykresy pomiarowe czujnika odległości i ich odniesienie do danych katalogowych tego czujnika, widoki panelu operatorskiego na poszczególnych etapach realizacji ćwiczenia. Należy też opisać własne wnioski wypracowane podczas ręcznego i automatycznego sterowania pozycją piłeczki oraz przebieg i efekty działania procedury automatycznego doboru nastaw regulatora PID.

Literatura i dokumentacja techniczna.

J.Kwaśniewski :”Sterowniki SIMATIC S7-1200 w praktyce inżynierskiej.”
S7-1200 Easy Book Manual 04/2012

Ćwiczenie 4. Konfiguracja i uruchomienie sieci przemysłowej PROFINET (SIEMENS)

Cel ćwiczenia

Celem ćwiczenia jest konfiguracja oraz uruchomienie rozproszonego systemu sterowania połączonego z użyciem sieci przemysłowej Profinet. Sterownik SIEMENS S7-1200 oraz HMI (panel operatorski), będzie wykorzystany do wizualizacji procesu automatycznej regulacji stanowiska do kontroli ciśnienia, wyposażonego w inny sterownik PLC tj. TURCK BL 20 PG EN V3. Wymiana danych pomiędzy systemami PLC będzie realizowana z wykorzystaniem sieci przemysłowej Profinet.

Proponowany sposób wykonania ćwiczenia:

1. Konfiguracja sprzętu
 - Uruchomić TIA, otworzyć nowy projekt pod dowolną nazwą.
 - Dokonać przykładowej konfiguracji sprzętu dla budowanej aplikacji zgodnie z konfiguracją rzeczywistą stanowiska. Konfiguracja jest podana w tabeli 4.1.

Tabela 4.1. Konfiguracja sprzętu

Slot nr	Funkcja modułu	Oznaczenie	Numer katalogowy 6ES7 ...
1	Jednostka centralna	CPU	212-1BE40-0XB0
2	Panel operatorski	HMI	6AV2 123-2DB03-0AX0
3	Wyjścia analogowe	AQ2x14BIT	232-4HB32-0XB0
4.	Sterownik TURCK BL 20		
	Switch sieciowy	CSM 1277 SIMATIC NET	6GK7-277-1AA10-0AA0

UWAGA! Podczas wyboru każdego elementu zwracać uwagę na numer katalogowy bo tylko on jednoznacznie określa dany element!

Dokonać konfiguracji kolejno poszczególnych modułów:

Jednostka centralna (slot 1):

Konfiguracja portu sieciowego: Profinet Interface [X1]: dodać nową podsieć, wybrać opcję ustawienia adresu IP w projekcie wymagane jest ustawienie adresu IP dla sterownika S-1200 – IP: **192.168.1.45**.

Zaznaczyć opcję : generate Profinet device name automatically (plc_1)

Czas cyklu CPU: pozostawiamy domyślny 150 [ms].

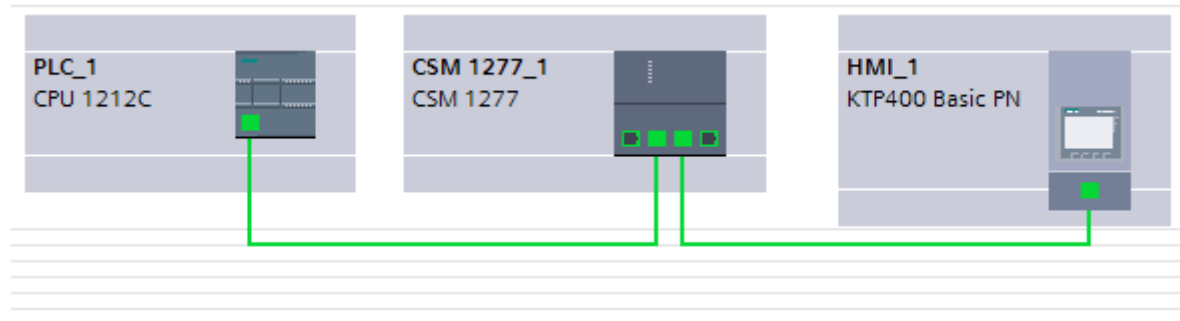
Panel operatorski KTP 400

Na etapie konfiguracji sprzętu poza przypisaniem nr IP: **192.168.1.2** panel nie wymaga dodatkowych czynności konfiguracyjnych.

Switch sieciowy

Nie wymaga dodatkowych działań podczas konfiguracji sprzętu.

Po wprowadzeniu wszystkich elementów należy je połączyć w sieć zgodnie ze schematem (Rys.4.1).



Rys.4.1. Schemat połączenia urządzeń.

Sterownik TURCK BL 20

W katalogu sprzętu (Hardware catalog) szukamy urządzenia o nazwie „CODESYS3 generic PROFINET Device”, o nazwie symbolicznej CDS3 PN DEVICE. Jeśli w środowisku nie znaleziono sterownika, wymagana jest instalacja plików konfiguracyjnych, tzw. GSDML. Pliki konfiguracyjne można pobrać ze strony producenta lub skorzystać z przygotowanej paczki danych tj. TURCK-BL20_PROFINET.zip udostępnionej na pulpicie).

Instalację nowego sprzętu dokonujemy korzystając z opcji: Options/Manage general station description files (GSD).

Uwaga!: Podczas instalacji należy wskazać plik GSDML-V2.3-TURCK-CDS3_PN_Device-20151208-010322.xml znajdujący się w katalogu po rozpakowaniu archiwum.

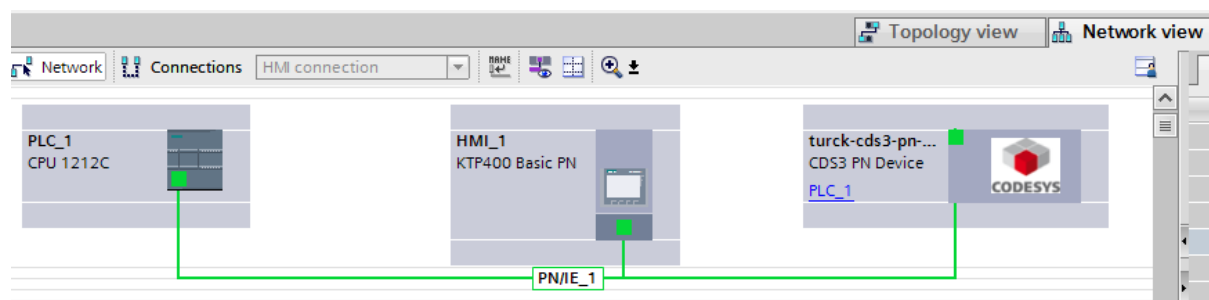
Konfiguracja sieci Profinet (sterownik TURCK)

W sieci Profinet najważniejszym parametrem jest unikalna nazwa urządzenia oraz właściwy adres IP („Assign device name/IP address”). Dla bramy TURCK-cds3-pn-device, cechy te konfigurujemy w: Device configuration/General/ PROFINET interface [X1]/ Ethernet addresses:

UWAGA! Dla sterownika TURCK należy ustawić następujący adres IP oraz nazwę:

IP: 192.168.1.12/ PROFINET device name: TURCK_BL

Po skonfigurowaniu wszystkich elementów systemu, należy je połączyć w sieć zgodnie ze schematem (Rys.4.2) zakładka Network view.

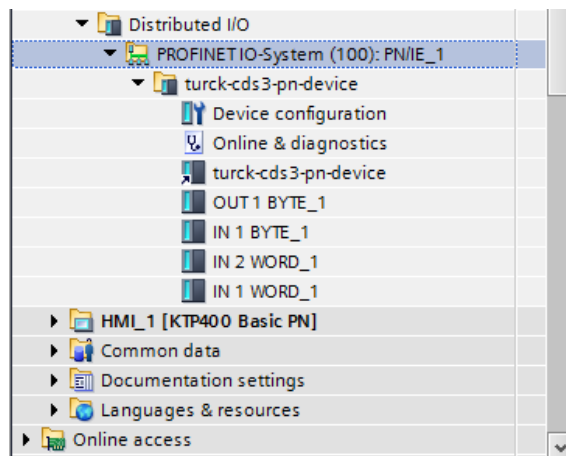


Rys.4.2. Schemat połączeń urządzeń sieci PROFINET.

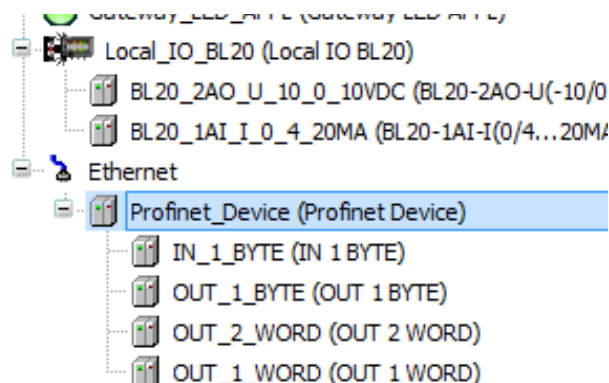
Wymiana danych w sieci Profinet

Sieć Profinet pozwala na transfer 1024 bajtów danych (512 dane wejściowe i 512 dane wyjściowe). W projekcie zdefiniowano wymianę danych w dwóch bajtach (BYTE) – sygnały statusowe, „liveBIT” ze sterownika s7 1200 do sterownika TURCK BL20, a także ze sterownika TURCK do S7 1200, oraz 3 słowach (WORD), dla sygnałów: wartość zadana, aktualna i stopień wysterowania zaworu regulacyjnego. Dane muszą zostać skonfigurowane w systemach które chcą je wymieniać, przeciwnie tzn. dane wejściowe ze środowiska Codesys są danymi

wyjściowymi w TIA. Zamieszczone poniżej rysunki (Rys. 4.3 i 4.4), przedstawiają skonfigurowane pakiety danych sieci Profinet w dwóch sterownikach.



Rys.4.3. Pakiety danych dla sterownika S7-1200 (TIA Portal V13)



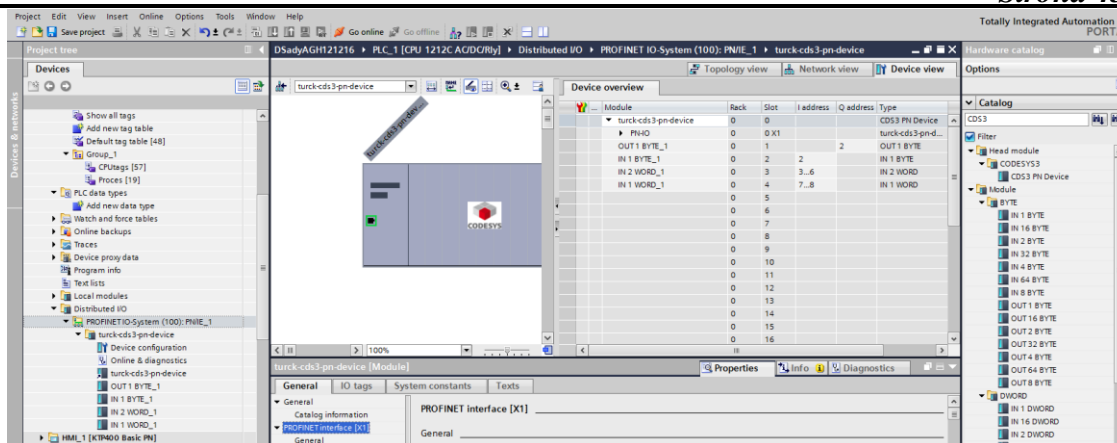
Rys.4.4. Pakiety danych – sterownik TURCK BL-20 (Code SYS)

Aby zapewnić dostęp do danych ze sterownika TURCK, w środowisku TIA Portal, dla urządzenia sieciowego TURCK-CDS3_PN_Device należy, zainstalować odpowiednie moduły (przeciągając je z sąsiadującego okna katalogu sprzętu z podkatalogu *Module* - podkatalog sterownika CDS3 PN Device), zgodnie z tabelą 4.2.

Tabela 4. 2. Konfiguracja sprzętu dla sterownika TURCK

Module	Rack	Slot	I address	O address
OUT 1 BYTE_1	0	1		2
IN 1 BYTE_1	0	2	2	
IN 2 WORD_1	0	3	3..6	
IN 1 WORD_1	0	4	7..8	

Po wprowadzeniu konfiguracji – okno Device overview, dla sterownika TURCK-CDS3_PN_Device, powinno wyglądać jak na Rys.4.5.



Rys.4.5. Device overview, dla sterownika TURCK (TIA Portal V13)

2. Program sterujący

Koncepcja programu polega na stworzeniu funkcji (FC) odpowiedzialnej za komunikację ze sterownikiem TURCK. Blok będzie wywoływany w OB1, pętli głównej programu. Dostęp do zmiennych procesowych ma umożliwić optymalizowany blok danych. W bloku tego typu to sterownik decyduje o alokacji zmiennych w pamięci, dostęp do danych odbywa się tylko poprzez symboliczne nazwy elementów bloku.

Sposób postępowania przy tworzeniu algorytmu sterowania

Otworzyć tabelę z nazwami symbolicznymi „PLC Tags” i zdefiniować nazwy zmiennych użytych do testów (tabela 4.3).

Tabela 4.3. Nazwy symboliczne (Tag-i) sterownika PLC S7-1200

Tag	Typ zmiennej	Adres	Opis
In0	Bool	%I2.0	bit statusowy - “liveBit ” ze sterownika TURCK
In1	Bool	%I2.1	bit statusowy - wartość aktualna nie jest równa wartości zadanej
In2	Bool	%I2.2	bit statusowy - sterownik pracuje bez błędów
In3	Bool	%I2.3	bit statusowy - wartość aktualna = wartość zadana
In4	Bool	%I2.4	bit statusowy - stanowisko w trybie sterowania ręcznego
In5	Bool	%I2.5	
In6	Bool	%I2.6	
In7	Bool	%I2.7	
Out0	Bool	%Q2.0	bit statusowy - “liveBit ” ze sterownika S7-1200
Out1	Bool	%Q2.1	
Out2	Bool	%Q2.2	
Out3	Bool	%Q2.3	
Out4	Bool	%Q2.4	
Out5	Bool	%Q2.5	
Out6	Bool	%Q2.6	
Out7	Bool	%Q2.7	
DQ0	Bool	%Q0.5	wyjście fizyczne sterownika S7-1200 wykorzystane jako indyktor komunikacji ze sterownikiem TURCK
Pessure_PV	word	%IW3	wartość ciśnienia w zbiorniku
Pessure_SP	word	%IW5	wartość ciśnienia zadanego (odczytanego z TURCK)
Valve_PV	word	%IW7	stopień wysterowania zaworu w %

Utworzyć blok danych (DB) przechowujący w pamięci (w strukturze TURCK) informacje o zmiennych procesowych i statusie sterownika TURCK, np.:

- TURCK.liveBit (bool),
- TURCK.mode(bool),
- TURCK.alarm(bool),
- TURCK.PV (real),
- TURCK.SP(real),
- TURCK.VALVE(real),
- etc.

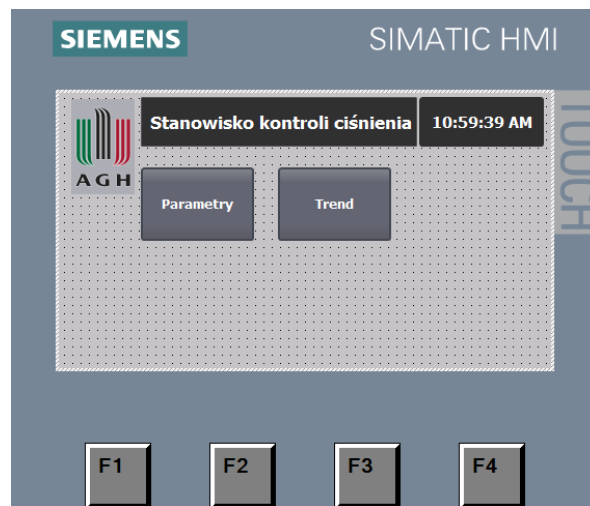
Utworzyć funkcję (FC), której zadaniem jest alokacja zmiennych procesowych ze sterownika TURCK w pamięci sterownika S7-1200. Blok funkcyjny stworzyć z użyciem języka LAD lub FBD. Utworzona funkcja FC nie zwraca wartości (void). UWAGA! Dla poprawnego odczytu wartości ciśnienia aktualnego (PV) i zadanego (SP), należy dokonać konwersji (z użyciem instrukcji CALCULATE), zgodnie z zależnością: $OUT := (IN1 * IN2) / IN3$. Zakres wartości ciśnienia mierzonego to 0-6 [bar]. Stopień wystawiania zaworu wyrażamy w [%] (0-100).

3. Aplikacja SCADA na panelu operatorskim HMI KTP400 w środowisku TIA Portal

Zadanie ma na celu sprawdzenie poprawności działania funkcji komunikacyjnych w oparciu o sieć Profinet, oraz wizualizację danych procesowych z „oddalonego” stanowiska kontroli ciśnienia.

Otworzyć tabelę z nazwami symbolicznymi ”HMI Tags” i powiązać je z danymi procesowymi (struktura TURCK w bloku danych DB).

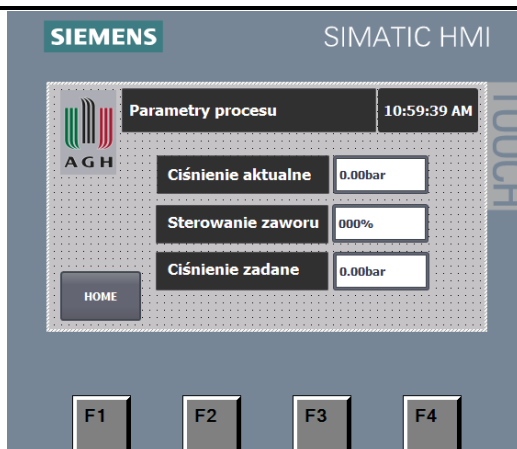
Dodać do projektu panel operatorski (HMI_1) i utworzyć domyślny ekran aplikacji (Home). Przykładowy panel startowy aplikacji został pokazany na Rys.4.6.



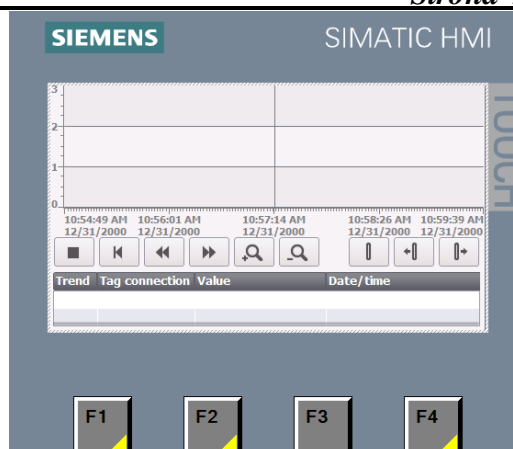
Rys.4.6. Przykład ekranu startowego HMI

Zdefiniować na tym ekranie możliwość wyboru pod-ekranów np. za pomocą przycisków funkcyjnych (Properties/Events/Press Key: Edit bits/ SetBit) umożliwić przejście do 2-ch pod-ekranów odpowiedzialnych za:

- a) podgląd wartości z procesu regulacji ciśnienia w zbiorniku np.: PV, SP, VALVE). W tym celu powiązać zmienne HMI_Tag z użyciem elementarnego pola I/O field – ekran parametry (Rys. 4.7),
- b) obserwację przebiegów czasowych z wartości zmiennych procesowych (SP, PV). Ten krok wymaga dodania i skonfigurowania kontrolki *Trend View* – ekran trend (Rys. 4.8).



Rys.4.7. Ekran parametry



Rys.4.8. Ekran trend.

Zakres wiadomości na kolokwium

Elementy organizacyjne oprogramowania dla sterowników PLC, wg normy: funkcje (FC) i bloki funkcyjne (FB) oraz charakterystyczne dla systemu SIEMENS: bloki organizacyjne (OB), Bloki Danych (DB) i Typy Danych PLC. Zasady adresacji danych w systemie SIEMENS. Typy danych i zmiennych w systemach PLC. Charakterystyka sieci komunikacyjnej na przykładzie przemysłowego standardu Profinet. Sposoby wymiany danych w sieci komunikacyjnej Profinet.

Sprawozdanie

Sprawozdanie powinno zawierać: schemat i opis konfiguracji sprzętu, dokładny opis realizacji zadań wymiany danych oraz alokacji zmiennych dla użytych sterowników PLC. Należy także udokumentować wygląd ekranów z utworzonego podczas ćwiczenia interfejsu użytkownika (HMI) oraz zaprezentować zarejestrowany przebieg czasowy dla wybranych zmiennych procesowych.

Literatura i dokumentacja techniczna

Tomasz Gilewski: „Podstawy programowania sterowników SIMATIC S7-1200 w języku SCL”

Ćwiczenie 5. Dobór nastaw regulatorów przemysłowych.

Cel ćwiczenia.

Zapoznanie się z niektórymi metodami strojenia i samostrojenia regulatorów przemysłowych na podstawie metod: Zieglera-Nicholsa, Astroma-Hagglunda, Takahashi'ego. Zapoznanie się z praktyczną realizacją procedury samostrojenia w seryjnie produkowanym regulatorze.



Fot. 5.1 Stanowisko z regulatorem EFTRONIK XS

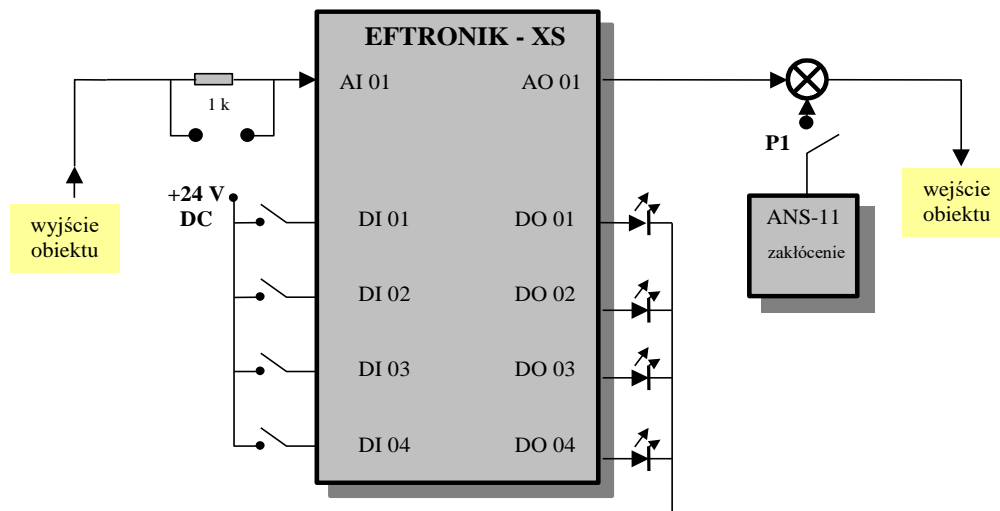
Opis aparatury.

DTR regulatora EFTRONIK XS.

Obiekt regulacji – Dodatek 1 instrukcji.

Opis stanowiska.

Uproszczony schemat połączeń elektrycznych stanowiska jest pokazany na rys. 5.1.



Rys. 5.1. Uproszczony schemat stanowiska z zamkniętym układem regulacji.

Ma ono postać szafy z umieszczonym w jej górnej części regulatorem, włącznikiem sieciowym oraz wskaźnikami stanu wejść i wyjść binarnych i przełącznikami służącymi do zadania sygnału „1” lub „0” na wejścia binarne.

W dolnej części szafy jest umieszczony obiekt regulacji typu cieplnego (opis obiektu – zob. Dodatek 1). Na ścianie czołowej obiektu znajduje się zespół przełączników (6) służących do wyboru wyjścia obiektu do badań. Poniżej znajduje się wskaźnik analogowy wyskalowany w % (7) pokazujący sterowanie bezpośrednio podawane na obiekt. Obok wskaźnika znajduje się zadajnik ANS-11 (8), z którego za pośrednictwem przycisku P1 można podawać zakłócenie na wejście obiektu.

Wykonanie ćwiczenia.

1. Uruchomienie stanowiska, sprawdzenie poprawności działania wszystkich elementów.
 - wszelkie zgłaszane alarmy należy zlikwidować podając logiczną „1” na 4-te wejście dyskretne regulatora.

Dla jednego z 4 wyjść obiektu dobrać nastawy ręcznie i automatycznie:

2. Ręczny dobór nastaw regulatora metodą Zieglera – Nicholasa.
 - Ustawić regulator na pracę proporcjonalną poprzez ustawienie $T_i = 0$ oraz $T_d = 0$.
 - Ustawić skalę bargrafu sygnału uchybu na wartość $\pm 10\%$.
 - Znaleźć wzmocnienie krytyczne układu zamkniętego na podstawie przebiegów sygnału wyjściowego układu zamkniętego.

Uwaga:

Wzmocnienie krytyczne jest w zakresie od 2.1 do 2.5 . Należy więc początkowo ustawić $P=2.0$, podać na 30 sekund skok zakłócający z zadajnika ANS-11 (czerwony przycisk), oraz zarejestrować odpowiedź układu z prędkością przesuwu rejestratora 50sek/cm. (trzy okresy oscylacji). Po malejącej amplitudzie przebiegu widać będzie iż wzmocnienie jest zbyt małe. W następnym kroku należy zbadać drugie skrajne wzmocnienie czyli $P=2.6$ (parametry rejestracji jak poprzednio). Tym razem amplituda zwiększa się, co oznacza przekroczenie granicy stabilności. I tak w kolejnych iteracjach zmieniając wzmocnienie z krokiem początkowym równym 0.2 (czyli $P= 2.2, 2.4, 2.3 \dots$) należy wyznaczyć K_{kryt} . Obserwując przebieg trzeba również zwrócić uwagę na wartości sterowań wychodzących z regulatora – w przypadku bardzo szybkich zmian od 0 do 100% (faktycznie sterowanie dwupołożeniowe) wystąpią też oscylacje o stałej amplitudzie, ale spowodowane jedynie fizycznymi ograniczeniami sygnału sterującego. Świadczyć to będzie o zbyt dużej wartości wzmocnienia. Podawanie skoku zakłócającego przy kolejnych próbach nie jest wskazane. Przy sprawnym prowadzeniu ćwiczenia zgodnie z instrukcją, oscylacje na wyjściu z obiektu nie powinny wygasnąć. Gdyby jednak do tego doszło należy powtórzyć 30-sto sekundowy skok zakłócający.

- Po osiągnięciu granicy stabilności – oscylacje nietłumione (nie wymaga się tutaj idealnej precyzji) trzeba zarejestrować z prędkością przesuwu 10 sek/cm. w celu dokładnego odczytu ich okresu.
- Na podstawie odnalezionego K_{kryt} i T_{osc} wyznaczyć nastawy regulatora zgodnie z wzorami Zieglera – Nicholasa.
- Wprowadzić obliczone nastawy do pamięci regulatora.
- Zarejestrować na rejestratorze odpowiedź zamkniętego układu regulacji na skok zakłócenia o wartości 30% na wejściu obiektu (prędkość rejestracji 50 sek./cm.). Skok zakłócający podaje się tym razem na stałe – początkowo widoczny wzrost sygnału wyjściowego z obiektu jest spowodowany automatycznie do poziomu sprzed podania skoku

poprzez odpowiednią zmianę sterowania z regulatora. Pełny obraz zmian daje porównanie sygnału sterowania z regulatora i całkowitego sygnału wejściowego obiektu. Gdy sygnał powróci do pierwotnego poziomu i w miarę się ustabilizuje (pełne minimalne oscylacje będą występowały zawsze) należy wyłączyć skok zakłócający i zarejestrować przebieg tak jak poprzednio. W tym przypadku będzie to też zakłócenie ale o przeciwnej polaryzacji.

3. Automatyczny dobór nastaw regulatora metodą Astroma – Hagglunda.

- Doprowadzić układ do położenia równowagi (np. na nastawach wyznaczonych ręcznie),
- Ustawić parametry procedury samostrojzenia zgodnie z poniższą tabelą:

Adres	Wartość	Opis
4138	0001	Maksymalny czas trwania 3 etapu identyfikacji, określony przez zastępczą stałą czasową obiektu.
4139	0020	Dopuszczalna amplituda wahań sygn. sterującego
4140	0050	Dopuszczalny zakres wahań wielkości mierzonej przed eksperymentem (wartość wpisana dzielona jest przez 10).
4141	0050	Dopuszczalny zakres wahań wielkości mierzonej podczas eksperymentu.
4142	0001	Nastawy wyliczane dla reg. PID
4143	0001	Nastawy wprowadzane automatycznie.

- Uruchomić procedurę samostrojzenia (oczywiście bez podawania skoku zakłócającego). Przebieg sygnału wyjściowego układu rejestrować na rejestratorze(50sek./cm.). Na wykresie zaznaczyć poszczególne etapy procedury.
- Zwrócić uwagę na wartość sterowania pod koniec drugiego etapu i przez cały trzeci etap samostrojzenia.
- Odczytać z regulatora otrzymane wartości nastaw, porównać z nastawami otrzymanymi metodą Zieglera- Nicholasa.
- Odczekać na ustabilizowanie się wyjścia obiektu.
- Przeprowadzić i zarejestrować (50sek./cm) zachowanie układu przy otrzymanych nastawach dla stałego skoku zakłócającego o wartości 30% - analogicznie jak dla nastaw z metody Zieglera – Nicholasa.
- Porównać zachowanie układu przy obu zestawach nastaw.

UWAGA: Zwrócić uwagę na fakt, że wartość współczynnika filtracji sygnału wejściowego ma wpływ na wartości nastaw PID zarówno przy ręcznym , jak i automatycznym doborze nastaw! (Zastanowić się, dlaczego?)

Zakres wymaganych wiadomości.

Metody doboru nastaw regulatorów przemysłowych: metody cyklu granicznego (metoda Zieglera – Nicholasa , metoda Astroma – Hagglunda), metody oparte o znajomość parametrów odpowiedzi skokowej, wskaźniki jakości regulacji, wpływ filtracji sygnału wejściowego na wartość nastaw w regulatorze EFTRONIK XS, obsługa panelu operatorskiego w regulatorze EFTRONIK XS: funkcje wyświetlaczy i przycisków, parametry i uruchamianie procedury samostrojzenia.

Sprawozdanie.

Sprawozdanie powinno zawierać: uproszczony schemat badanego układu, otrzymane podczas eksperymentów wykresy uzupełnione o opis i układy współrzędnych, tabelki zawierające otrzymane wartości nastaw regulatora dla jednego wybranego wyjścia obiektu oraz obu metod, próbę objaśnienia różnic pomiędzy wartościami nastaw otrzymanymi obiema metodami.

Ćwiczenie 6. Asynchroniczny silnik klatkowy z falownikiem

Cel ćwiczenia.

Zapoznanie się z konfiguracją i uruchomieniem napędu elektrycznego z falownikiem.

Opis aparatury.

Opis katalogowy falownika SIEMENS MICROMASTER 420.

Instrukcja obsługi falownika SIEMENS MICROMASTER 420

Opis oprogramowania SIMANICS v4.4. STARTER

Systemy pomocy kontekstowej oprogramowania SINAMICS 4.4 STARTER

Opis stanowiska laboratoryjnego.

Silnik elektryczny z falownikiem i zespołem przełączników jest pokazany na rys. 6.1.



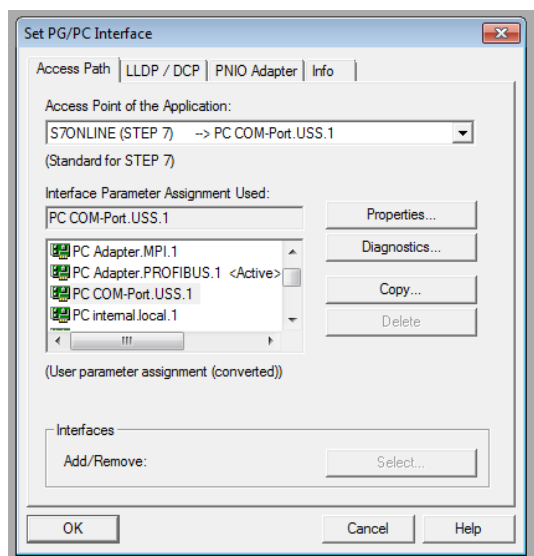
Rys. 6.1. Silnik elektryczny z falownikiem i osprzętem.

Składa się ono z następujących zasadniczych części: standardowego komputera klasy PC z konwerterem USB-RS232 i oprogramowaniem konfiguracyjnym, falownika, silnika elektrycznego oraz zespołu przełączników z potencjometrem do zadawania napięcia wejściowego. Dodatkowym elementem stanowiska jest laserowy czujnik prędkości obrotowej, używany do zdejmowania charakterystyk statycznych silnika.

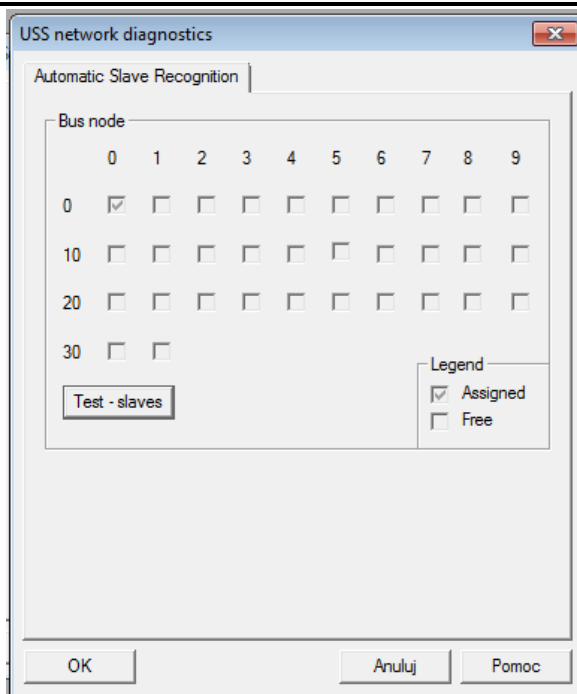
Wykonanie ćwiczenia.

Włączyć komputer i zasilanie obiektu, uruchomić oprogramowanie **SINAMICS 4.4 STARTER**.

Skonfigurować połączenie pomiędzy środowiskiem konfiguracyjnym i falownikiem: Options\set PG/PC interface. Ustawić: PC COM port USS1, po ustawieniu uruchomić diagnostykę (Diagnostics) – rys 6.2 i sprawdzić widoczność falownika (Test-slaves –rys 6.3):

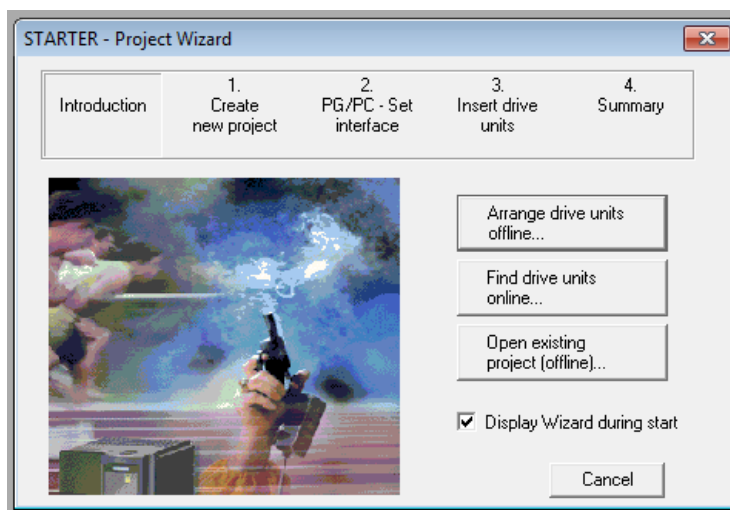


Rys 6.2. Konfiguracja połączenia komputera z falownikiem



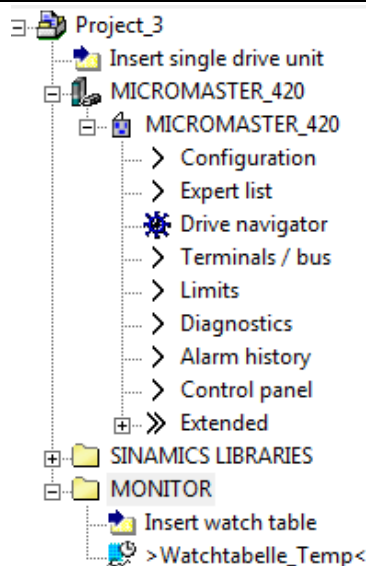
Rys.6.3. Sprawdzanie poprawności konfiguracji połączenia

Założyć nowy projekt korzystając z Wizarda, wyszukując podłączony falownik w trybie on line (Find drive units online) – zob rys 6.4.



Rys 6.4. Wizard do zakładania projektu

Skonfigurować podstawowe parametry, kolejność działań:
Wybieramy opcję: Configuration\Reconfigure drive.



Rys. 6.5. Główne menu wybranego falownika

Parametry należy ustawić następujące:

Standard: typ sieci w danym kraju: Europa, 50Hz,

Motor: dane silnika odczytać z tabliczki znamionowej,

Tryb pracy: V/f z charakterystyką liniową,

Źródło sygnałów sterujących: ustawienia fabryczne, wartość zadana prędkości: stała częstotliwość lub „analog set point” (dla sterowania z poziomu zadajnika).

Parametry główne: pozostawić wyznaczone,

Wyliczenie parametrów silnika: wyliczyć kompletne,

Połączyć komputer z falownikiem (opcja Connect to selected devices). Poprawne połączenie jest sygnalizowane komunikatem „drive ready”.

Załadować projekt do pamięci falownika (uwaga – zapis trwa dość długo i komputer może działać wtedy niestabilnie).

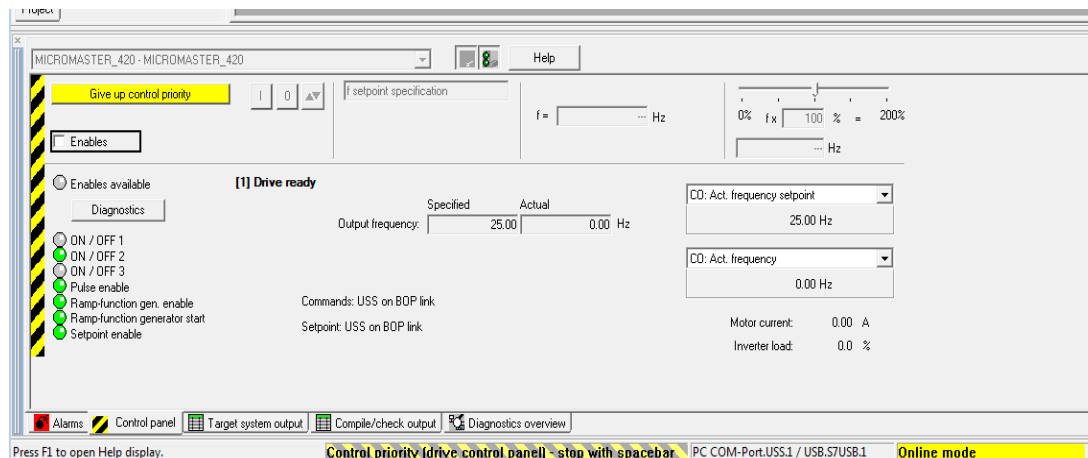
Zdefiniować tabelę do monitorowania zmiennych: opcje: Monitor/Watchtabelle, dodać do tabeli (z listy eksperta: Expert List) następujące parametry: częstotliwość: **r21**, prędkość obrotową: **r22** napięcie na wejściu analogowym: **r752** oraz napięcie na wyjściu falownika: **r25** Można też dodać inne interesujące parametry (np. stany przełączników – znaleźć w dokumentacji). Wygląd tabeli – zob. rys rys.6.6.

>Watchtabelle_Temp<							
Namespace							
Name	Information	Display format	Status value	Control value	Unit	Condition	Data type
1	MICROMASTER_420.MICROMASTER_... CO: Act. frequency	DEC-10	6.055716		Hz	All	REAL
2	MICROMASTER_420.MICROMASTER_... Act. rotor speed	DEC-10	181.8201		rpm	All	REAL
3	MICROMASTER_420.MICROMASTER_... Act. input of ADC [V]	DEC-10	1.251222			All	REAL
4							

Rys.6.6. Tabela do monitorowania zmiennych

Otworzyć panel sterujący, odblokować możliwość uruchomienia napędu („Assume Control priority), zaznaczyć możliwość zadania częstotliwości bazowej, wpisać częstotliwość np. 25 Hz oraz -25 Hz uruchomić napęd i sprawdzić działanie (rys 6.7)

Zbadać zależność prędkości obrotowej od zadanej częstotliwości dla liniowej charakterystyki: V/f. Częstotliwość zadaną zmieniać w zakresie 5 – 50 Hz co 5Hz, prędkość obrotową odczytywać (notować niezależnie): z użyciem czujnika laserowego oraz na panelu sterującym (po prawej stronie można ustawić odczyt prędkości obrotowej). W sprawozdaniu wyznaczyć różnicę pomiędzy wartościami zmierzonymi i odczytanymi i uzasadnić, skąd się wzięła?



Rys.6.7. Panel sterujący.

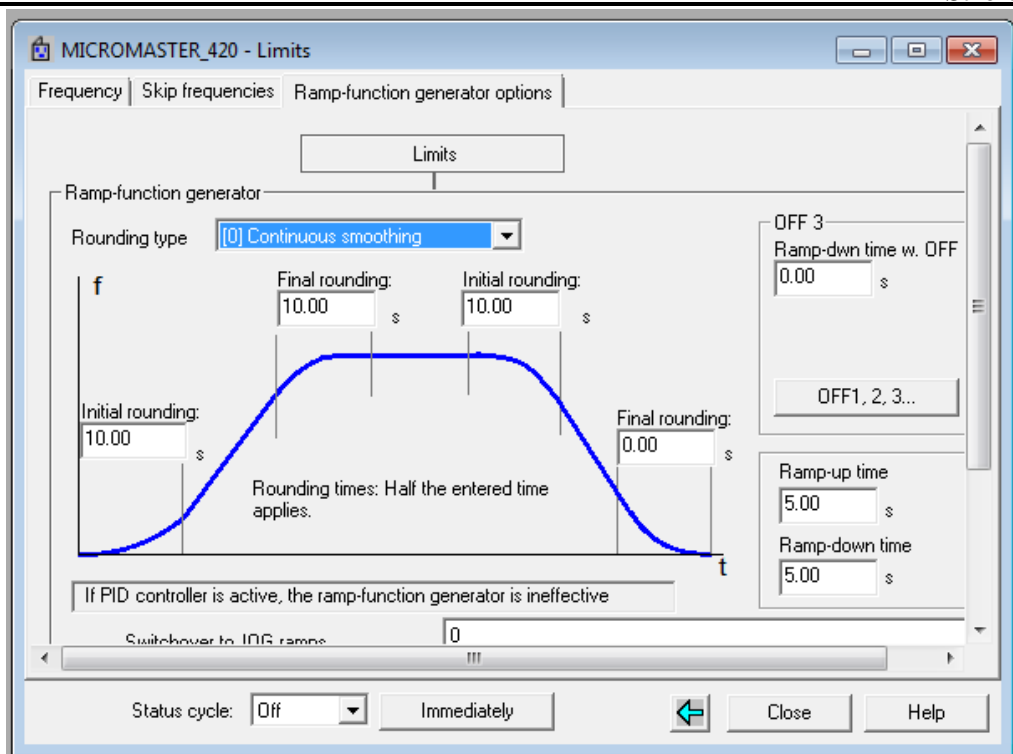
Zamknąć panel sterujący, uruchomić falownik przełącznikiem start/stop i wyznaczyć zależność pomiędzy prędkością obrotową silnika i napięciem wejściowym (zmieniając napięcie co 1[V]) dla dwóch charakterystyk V/f: liniowej i parabolicznej. Zmiana tego parametru: Drive Configuration-> Operating Mode. Pamiętać o zapisaniu zmian do pamięci falownika (download to target). Napięcie wejściowe i częstotliwość odczytywać w tabeli, prędkość obrotową z użyciem czujnika laserowego.

Sprawdzić działanie napędu podczas zmiany kierunku obrotów (przełącznik).

UWAGA:

1. Aby skorzystać z zewnętrznego zadajnika prędkości należy wybrać jako źródło sygnałów sterujących : np. USS on COM LINK + Analog set point.
2. Aktywny panel sterujący blokuje możliwość sterowania silnikiem z poziomu zespołu przełączników!

Zdefiniować różne wartości czasów rozbiegu i hamowania: Drive Navigator->Limiter->Ramp Function Generator Options (zob. rys 6.8). Ustawić wartości (rozbieg i hamowanie): 2s, 5s, 10s w różnych kombinacjach. Sprawdzić działanie poprzez pomiar czasów: rozbiegu i hamowania dla różnych ustawionych wartości.



Rys.6.8. Konfiguracja czasów rozbiegu i hamowania.

Sprawozdanie

Sprawozdanie powinno zawierać: uproszczony schemat blokowy układu, przebiegi zbadanych charakterystyk i opis wykonanych działań.

Zakres wymaganych wiadomości

Budowa, zasada działania i sterowanie pracą silnika asynchronicznego. Konstrukcja i zasada działania falownika. Zasady konfiguracji falowników SIEMENS MICROMASTER.

Ćwiczenie 7. SIEMENS S7 1200- funkcje definiowane przez użytkownika – język drabinkowy (LD)

Cel ćwiczenia:

Zapoznanie się z funkcjami definiowanymi przez użytkownika oraz językiem LD na przykładzie S7 1200.

W ramach wykonania ćwiczenia należy samodzielnie zbudować funkcję realizującą sumator dwóch liczb 2-bitowych bez znaku (zakres każdej z liczb: od 0 do 3). Sumator ma być zrealizowany z użyciem wyłącznie zmiennych binarnych. Wartość wyniku ma zakres od 0 do 6 (jest to liczba 3 bitowa).

Założmy, że liczby wejściowe są oznaczone jako **a** oraz **b**, a ich wynik jako **y**. Operacja, która ma być realizowana, może być zapisana jako:

$$y = a + b \quad (7.1)$$

Każda z tych liczb może być zapisana tak, jak w tabeli 2.1. Symbole z tabeli 2.1 będą później używane w zapisie funkcji.

Tabela 7.1

Liczb	Zapis w kodzie binarnym
a	
a	(x2 x1)
b	(x4 x3)
y	(y3 y2 y1)

Operacja sumowania z użyciem notacji z tabeli 7.1 może być zapisana następująco:

$$(y3 y2 y1) = (x2 x1) + (x4 x3) \quad (7.2)$$

Funkcja logiczna realizująca sumator może być zapisana z użyciem tabeli Karnaugh 7.2. W komórkach tabeli wynik zapisany jest jako (y3 y2 y1).

Tabela 7.2

a (x2 x1) →	(0 0)	(0 1)	(1 1)	(1 0)
b(x4 x3) ↓				
(0 0)	(0 0 0)	(0 0 1)	(0 1 1)	(0 1 0)
(0 1)	(0 0 1)	(0 1 0)	(1 0 0)	(0 1 1)
(1 1)	(0 1 1)	(1 0 0)	(1 1 0)	(1 0 1)
(1 0)	(0 1 0)	(0 1 1)	(1 0 1)	(1 0 0)

Po minimalizacji funkcji zapisanej w tabeli 7.2 (sprawdzić, czy poniższa minimalizacja jest poprawna?) poszczególne bity wyniku są wyrażone następująco:

$$y3 = x2 \cdot x4 + x4 \cdot x3 \cdot x1 + x2 \cdot x1 \cdot x3 \quad (7.3a)$$

$$y2 = \overline{x4} \cdot \overline{x3} \cdot x2 + x2 \cdot \overline{x1} \cdot \overline{x4} + \overline{x4} \cdot x3 \cdot \overline{x2} \cdot x1 + x4 \cdot \overline{x3} \cdot \overline{x2} + \overline{x2} \cdot \overline{x1} \cdot x4 + x4 \cdot x3 \cdot x2 \cdot x1 \quad (7.3b)$$

$$y1 = x3 \cdot \overline{x1} + \overline{x3} \cdot x1 \quad (7.3c)$$

Proponowany sposób wykonania ćwiczenia:

1. Uruchomić TIA PORTAL, założyć nowy projekt z konfiguracją hardware'u identyczną jak w ćwiczeniu (PLC+HMI).
2. Do projektu wstawić cztery funkcje zwracające wynik typu BOOL o nazwach zastrzeżonych: FC1, FC2, FC3, FC4 Program Blocks: Add New Block – wybrać Function i język LAD. Funkcje te mają realizować ustawienie poszczególnych bitów wyniku zgodnie z (2.3a) – (2.3c) oraz wystawienie gotowego wyniku na wyjście. Wystawienie wyniku na wyjście zrealizować poprzez odwołanie do nazwy funkcji, w tym celu należy pamiętać o zdefiniowaniu nazwy zwracanej Dla tych funkcji można zdefiniować nazwy symboliczne. Opisy tych funkcji, ich zmienne wejściowe, wyjściowe i wewnętrzne oraz sugestie nazw symbolicznych są podane w tabeli 7.3.

Tabela 7.3

Funkcja	Nazwa symboliczna	Działanie	Zmienne wejściowe	Zmienne wyjściowe	Zmienne wewnętrzne tymczasowe
FC3	y3	Wylicza bit y3 wyniku zgodnie z (2.3a)	x4 x3 x2 x1	y3	do zapisu wyników iloczynów
FC2	y2	Wylicza bit y2 wyniku zgodnie z (2.3b)	x4 x3 x2 x1	y2	do zapisu wyników iloczynów
FC1	y1	Wylicza bit y1 wyniku zgodnie z (2.3c)	x4 x3 x2 x1	y1	do zapisu wyników iloczynów
FC4	wynik	Wywołuje pozostałe funkcje i wystawia wynik na wyjście.	x4 x3 x2 x1	y3 y2 y1	Brak (zbędne)

3. Dla wszystkich funkcji zdefiniować zmienne wejściowe, wyjściowe i wewnętrzne zgodnie z sugestiami w tabeli 7.3. Następnie zapisać algorytm działania funkcji **FC1** – **FC3** zgodnie z wzorami (7.3a)-(7.3c) z użyciem języka drabinkowego.
4. W obrębie funkcji **FC0** zrealizować bezwarunkowe wywołanie pozostałych w takiej kolejności, aby ich wyjścia przypisać do wyjść funkcji FC0 zgodnie z zapisem (7.2).
5. W projekcie zdefiniować zmienne PLC (odpowiednik zmiennych reprezentowanych bezpośrednio wg normy) opisane w tabeli 7.4. Będą one używane do komunikacji z HMI, wprowadzania wejść oraz odczytu wyników działania programu.

Tabela 7.4

Nazwa zmiennej	Typ	Adres absolutny
a	BYTE	%MB0
b	BYTE	%MB1
y	BYTE	%MB2
x4	BOOL	%M1.1
x3	BOOL	%M1.0
x2	BOOL	%M0.1
x1	BOOL	%M0.0
y31	BOOL	%M2.2
y21	BOOL	%M2.1
y11	BOOL	%M2.0

UWAGA! Sens osobnego definiowania zmiennych typu BYTE oraz typu BOOL odnoszących się do tych samych obszarów pamięci (np. por %MB0 i %M0.0, %M0.1) jest taki, że można z użyciem zmiennych typu BOOL można ustawiać pojedyncze bity w zmiennej typu BYTE.

- Wywołać funkcję **wynik** w bloku organizacyjnym **OB1** i przypisać do poszczególnych wejść i wyjść tej funkcji zmienne globalne opisane w tabeli 7.4 w następujący sposób:

Tabela 7.5

Zmienna wejściowa FC4 (y)	Zmienna globalna
x4	x4
x3	x3
x2	x2
x1	x1
y3	y31
y2	y21
y1	y11

- Na ekranie HMI zdefiniować możliwość: wprowadzania wartości zmiennych globalnych **a** i **b** oraz odczytu zmiennej globalnej **y**. Można to zrobić przeciągając odpowiednie zmienne z tabeli „PLC Tags” bezpośrednio na ekran. Po przeciągnięciu pojawiają się odpowiednie okna dialogowe do wpisywania wartości.
- Załadować zbudowany program na sprzęt i sprawdzić działanie programu wpisując wartości zmiennych wejściowych i odczytując wynik na panelu operatorskim.
- (Jak wystarczy czasu). Dla zmiennych wprowadzanych z panelu zdefiniować sprawdzanie poprawności zakresu: poprawny zakres to od 0 do 3.

Zakres wiadomości na kolokwium

Zasady adresacji danych w systemie SIEMENS. Różnice pomiędzy funkcją i blokiem funkcyjnym w programowaniu PLC. Zmienne bezpośrednio reprezentowane i zmienne PLC w SIEMENSIE. Język drabinkowy: podstawowe elementy: styki i cewki, reguły interpretacji programu.

Sprawozdanie

Sprawozdanie powinno zawierać schemat drabinkowy realizacji zadania oraz opis zmiennych aplikacji SCADA.

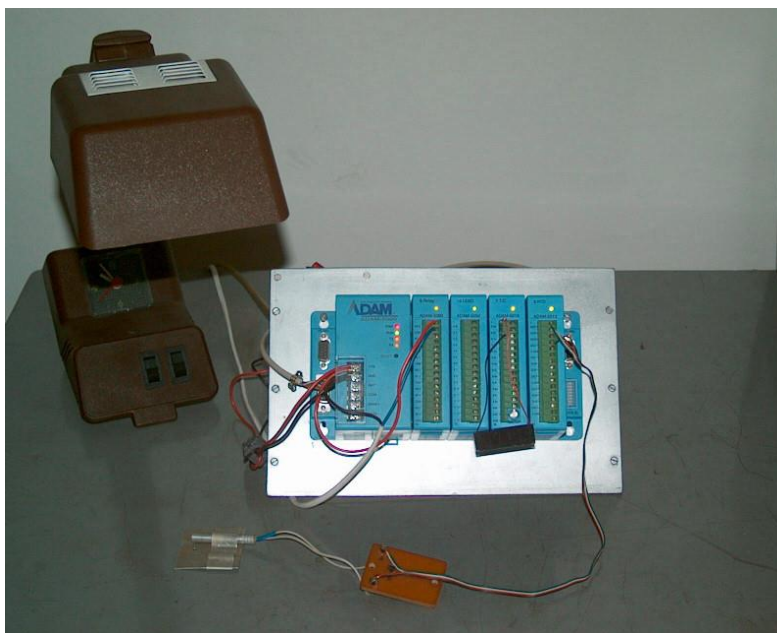
Ćwiczenie 8. Pakiet VisiDaq do realizacji bezpośredniego sterowania cyfrowego.

Cel ćwiczenia.

Zapoznanie się przykładowym oprogramowaniem pozwalającym na realizację sterowania cyfrowego w oparciu o komputer klasy PC oraz zapoznanie się z przykładem interfejsu procesowego dla komputera klasy PC.

Opis aparatury.

Podręcznik użytkownika GENIE 3.0 (plik GeniDaq.pdf), opisy techniczne modułów DAQ ADAM5000 (odpowiedni katalog na stronie internetowej)



Fot. 8.1 Moduły ADAM 5000 z podpiętymi czujnikami i lampą

Opis stanowiska.

Stanowisko laboratoryjne składa się z komputera PC oraz stacji akwizycji danych ADAM5000 wyposażonej w odpowiednie moduły we/wy.

Wykonanie ćwiczenia.

Pierwsza część ćwiczenia ma za zadanie zapoznanie studenta z techniką tworzenia aplikacji w środowisku GENIE. W tym celu należy przed zajęciami samodzielnie (w domu) zrealizować 8 zadań wprowadzających opisanych w podręczniku użytkownika Genie w rozdziale TUTORIAL. Wersja demonstracyjna programu GENIE znajduje się na stronie internetowej.

Druga część ćwiczenia polega na zbudowaniu prostej aplikacji – zaawansowanego wyłącznika zmierzchowego. Aplikacja powinna realizować następujące funkcje:

- pomiar i wyświetlanie na ekranie aktualnego poziomu oświetlenia (spodziewany zakres napięciowy sygnału 0 – 10 mV)

- włączenie lampy na pewien zadany czas, gdy poziom oświetlenia spadnie poniżej wartości progowej
- wizualizacja czasowego przebiegu aktualnego oświetlenia i zadanego progu zadziałania
- możliwość niezależnego od czujnika stałego włączenia lampy (przycisk)
- możliwość zablokowania automatu włączającego lampę (przycisk)
- pomiar temperatury otoczenia z termometru PT-100

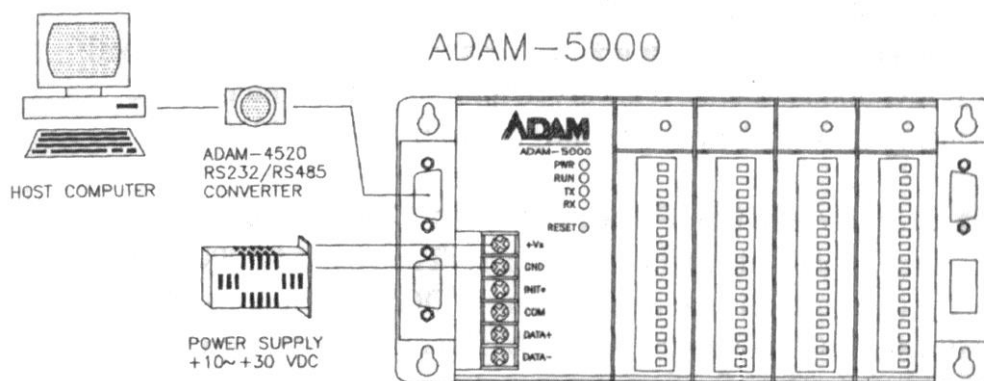
Wszelkie parametry kontrolne zadawane mają być z poziomu panelu operatorskiego za pomocą odpowiednich elementów:

- czas włączenia lampy ustawiany za pomocą cyfrowego pola sterowania (Numeric Control Display Item) – wartość początkowa 5sek., skok 1sek., maksymalnie 50 sek.
- wartość progowa ustalana za pomocą suwaka (Slider Control Display Item)

Działanie strategii powinno być następujące: jeśli oświetlenie spadnie poniżej ustawionego progu, lampa zapala się na określony czas. Jeśli podczas świecenia lampy poziom oświetlenia wzrośnie powyżej zadanego progu – lampa ma zgasnąć. Przy wciśniętym przycisku blokady automatu spadek oświetlenia poniżej progu nie włącza lampy. Niezależny włącznik lampy ma najwyższy priorytet i niezależnie od poziomu oświetlenia może włączyć lampę.

Uwaga: Należy tak ustawić lampę, aby nie oświetlała czujnika – generowałyby to ciągłe oscylacje w układzie.

Typowy schemat połączeń sprzętowych dla ADAM'ów 5000 przedstawia poniższy rysunek



Rys. 8.1 Podłączenie modułu ADAM5000 do komputera PC

W naszym przypadku korzystamy tylko z jednej stacji bazowej ADAM 5000 co pozwala na bezpośrednie podłączenie do komputera poprzez złącze RS232. Od strony funkcjonalnej nie zmienia to niczego – wszelkie ustawienia dla magistrali RS485 zachowują swoją ważność.

Stacja bazowa ADAM 5000 posiada w sieci RS485 adres 02h i wyposażona jest w następujące moduły kontrolno pomiarowe:

- ADAM5060 – dyskretne wyjścia przekaźnikowe (gniazdo 0)
- ADAM5018 - wejścia analogowe (gniazdo 1)
- ADAM5013 – wejścia termometru oporowego PT100 (gniazdo 3)

Lampa zasilana z sieci 230V została podłączona poprzez zewnętrzny (zabezpieczający) przekaźnik do wyjścia 4 modułu ADAM5060. Uaktywnienie tego wyjścia powoduje włączenie lampy.

Czujnik oświetlenia to małe fotoogniwo. Prąd zwarciovowy fotoogniwa jest proporcjonalny do oświetlenia. Napięcie proporcjonalne do prądu zwarciovowego podawane jest na wejście 0 modułu ADAM 5018.

Do zerowego kanału modułu ADAM 5013 podpięto termometr oporowy PT100. Służy on do pomiaru temperatury otoczenia, a umieszczony w polu światła lampy doskonale pokazuje jak dużo ciepła ona dostarcza.

Jako rozszerzenie (nieobowiązkowe) podstawowego zakresu ćwiczenia proponuje się realizację dwupołożeniowego układu stabilizacji temperatury – lampa to grzejnik a termometr to obiekt z czujnikiem.

Interesujące mogą być próby bezpośredniej komunikacji z ADAM'em za pomocą znaków ASCII. Proponuje się tutaj wykorzystanie w Genie bloku RS232 i odczytanie temperatury z wewnętrznego termometru w module ADAM5018. Wszelkie informacje i kody rozkazów znajdują się w dokumentacji technicznej modułów.

Zakres wymaganych wiadomości.

Zasady tworzenia aplikacji w środowisku Genie 3.0, budowa, działanie i sposoby programowania modułów zdalnej akwizycji danych i sterowania (szczególnie ADAM5000, ADAM5060, ADAM5017, ADAM5018), zasady budowania rozproszonych systemów sterowania w oparciu o magistralę RS485.

Sprawozdanie.

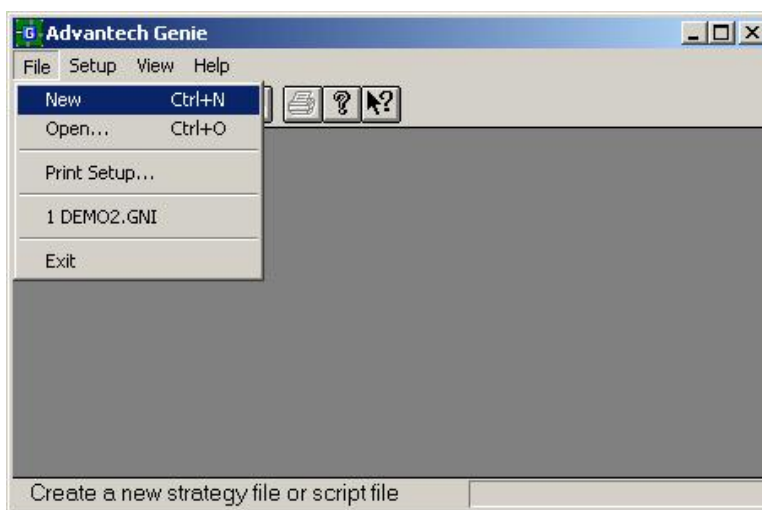
Sprawozdanie powinno zawierać krótki opis pakietu Genie 3.0 , jego zalety i wady. Szczegółowo należy opisać przebieg drugiej części ćwiczenia (budowa prostej aplikacji) umieszczając odpowiednie rysunki obszaru roboczego i panelu operatorskiego.

Materiały pomocnicze do ćwiczenia 8.

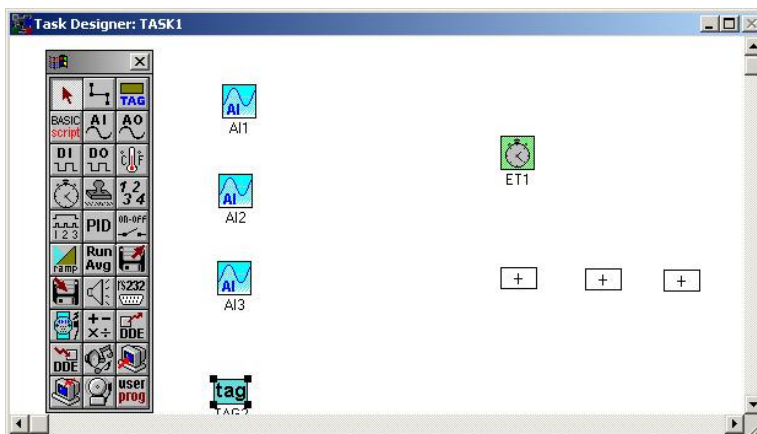
Aby przedstawić pewne zagadnienia związane z realizacją ćwiczenia zostanie zaprezentowana realizacja „krok po kroku” przykładowej strategii w środowisku Advantech Genie. Zadania realizowanej aplikacji są następujące:

- budowa stopera odliczającego „w dół” zadany czas
- sygnalizacja włączenia odliczania
- prezentacja ilości sekund pozostałych do wyłączenia stopera
- wizualizacja trzech, odpowiednio przeskalowanych przebiegów z wejść analogowych
- sterowanie przebiegiem strategii (wstrzymanie/wznowienie)
- przełączanie pomiędzy kilkoma panelami operatorskimi

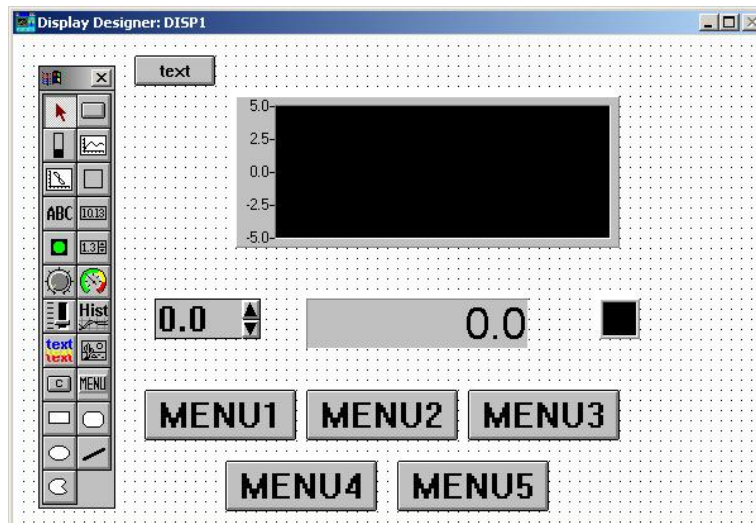
Zaczynamy od uruchomienia programu Genie i stworzenia nowej strategii (File->New).



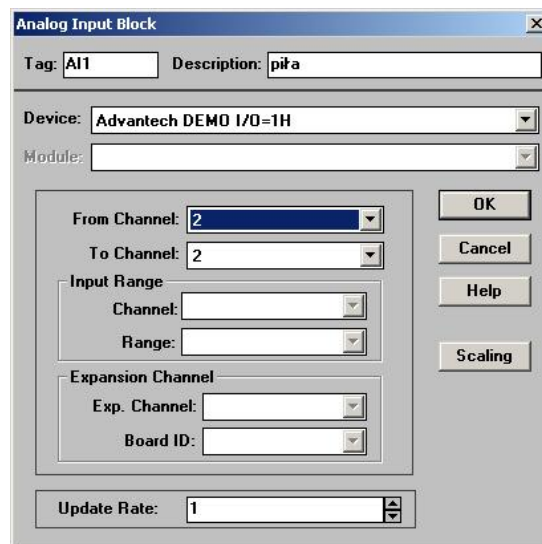
W obszarze TASK DESIGNER'a umieszczamy kolejno wszelkie potrzebne nam ikony: wejść analogowych AI, zegarka, TAG'ów i operacji matematycznych SOC. Umieszczenie odpowiedniej ikony polega na kliknięciu jej symbolu w obszarze TOOLBOX'a, a następnie wskazaniu jej położenia w obszarze TASK'a (kursor ma wtedy postać krzyża) i ponownym kliknięciu.



Teraz pora zająć się panelem operatorskim. Umieszczamy na nim przycisk do uruchamiania stopera (Conditional Button), Trend Graph do czasowej wizualizacji przebiegów, Numeric Control – zadajnik czasu stopera, Numeric/String Display – będzie pokazywał pozostałą ilość sekund, Indicator – wskaże włączenie stopera oraz kilka przycisków MENU dla sterowania przebiegiem strategii. Zalecane jest aby na każdym etapie tworzenia strategii korzystać z informacji pomocniczych (HELP'ów) dotyczących używanych elementów.



Jako sygnały analogowe wykorzystamy przebiegi demonstracyjne z urządzenia Advantech DEMO I/O. Jest ono przypisane do adresu 1H i posiada trzy kanały: 0 – sinus, 1 – prostokąt, 2- piła. Wszystkie przebiegi mają amplitudę od -5 do 5 . Dwukrotne kliknięcie ikony AI pozwala na dostęp do pól konfiguracyjnych. Poszczególnym wejściom analogowym przypisujemy odpowiedni kanał urządzenia. Warto też w polu Description wpisać nazwę sygnału – ułatwia to później szybką orientację w strategii.



Aby poznać w pełni możliwości konfiguracyjne wejść analogowych zajmiemy się skalowaniem przebiegów. Docelowo mają one mieć następujące zakresy: sinus – od 0 do 1, prostokąt – od 2 do

2.5 a pila od 3 do 5. W tym celu na ikonie odpowiedniego wejścia analogowego naciskamy klawisz Scaling.

SCALING

Enable scaling

Input Range: min value (n1) max value (m1)
-5 5

Scale to: min value (n2) max value (m2)
2 2.5

OK Cancel

result = $n2 + (input - n1) \times ((m2 - n2) / (m1 - n1))$

W menu SCALING zaznaczamy pole Enable scaling i wypełniamy odpowiednimi wartościami pozostałe pola. Rysunek prezentuje skalowanie przebiegu prostokątnego. Wartości wejściowe to $-5/5$ a wyjściowe $2/2.5$. Zatwierdzenie wprowadzonych wartości następuje poprzez naciśnięcie przycisku OK. Powtórne wciśnięcie tego przycisku w menu konfiguracyjnym kończy konfigurację danego wejścia analogowego.

W następnej kolejności trzeba skonfigurować zegarek. Nazwijmy go (Description) STOPER, każdy zliczać czas upływający (Timer Type – Elapsed time) wyrażany w sekundach (Resolution – Second) i wyłączmy zapętlenie (Cycle – None).

Timer Block

Tag: ET1 Description: STOPER

Timer Type
 Time Of Day
 Elapsed time

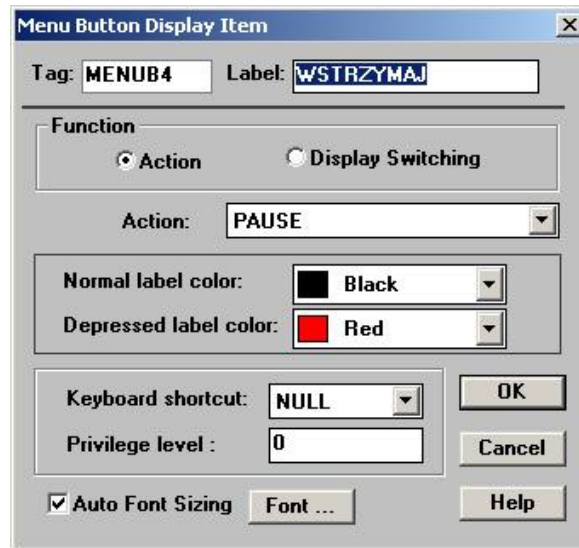
Resolution
 Second
 Tick (0.1 Sec)

Cycle
 None
 Yearly
 Monthly
 Weekly
 Daily
 Hourly
 Minutely

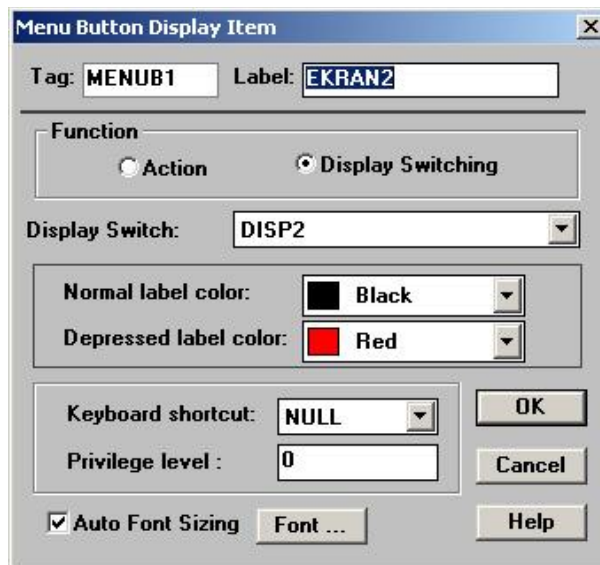
OK
Cancel
Help

Reset from:

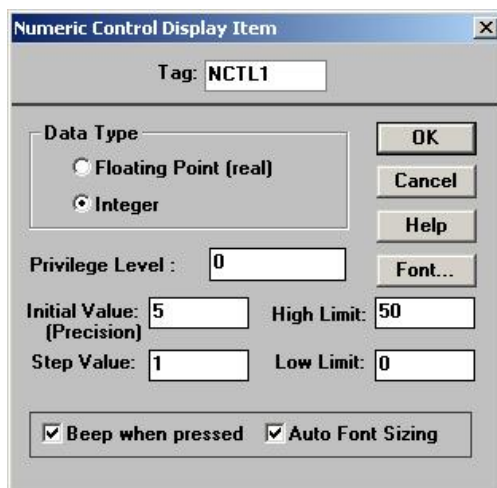
Teraz warto wrócić go Display'a i ponazywać na nim wszelkie przyciski. Zaczynamy od przycisków MENU. Trzy z nich będą odpowiednio wstrzymywały, wznowiały i kończyły działanie strategii. Opisujemy je więc właściwymi nazwami i przypisujemy prawidłową akcję. Może wystąpić konieczność zwiększenia wymiarów klawiszy aby ich nazwy były w pełni widoczne.



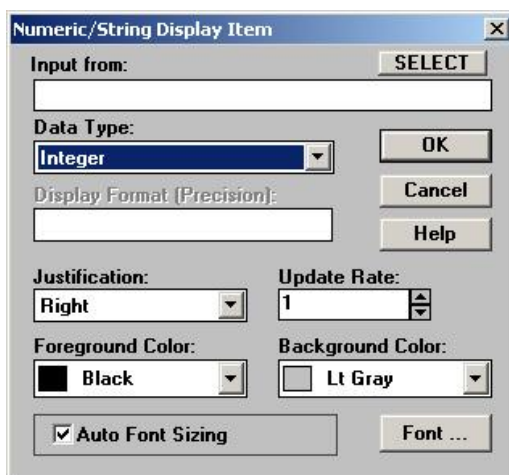
Pozostałe dwa klawisze MENU będą służyły do przełączania ekranów. Aby było to możliwe należy zdefiniować dodatkowe Display'e. Robimy to poprzez menu robocze FILE → ADD/DELETE → ADD DISPLAY. Dwukrotne powtórzenie tej operacji powinno zaowocować zaistnieniem w strategii trzech ekranów: DISP1, DISP2 i DISP3. Teraz można już przypisać odpowiednim przyciskom przełączanie na właściwy ekran.



Przy przycisku Conditional Button ograniczymy się na razie do nadania mu nazwy START. Numeric Control – zadajnik czasu działania stopera ustawmy następująco: liczby całkowite (Integer), zakres zmian od 0 do 50, krok 1, wartość początkowa 5.

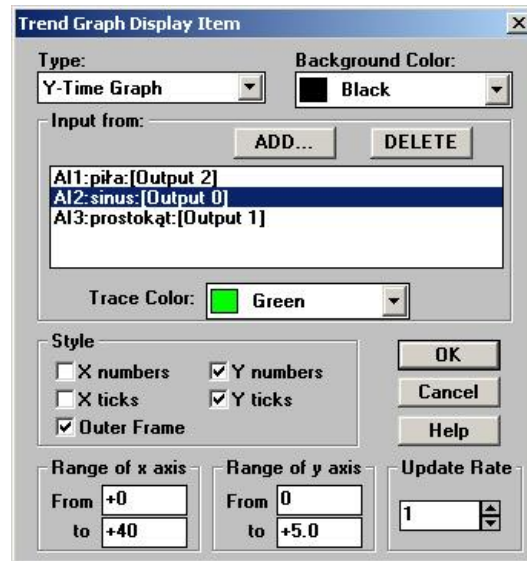


Pole wyświetlacza cyfrowego (Numeric/String Display) należy przestawić na liczby całkowite.

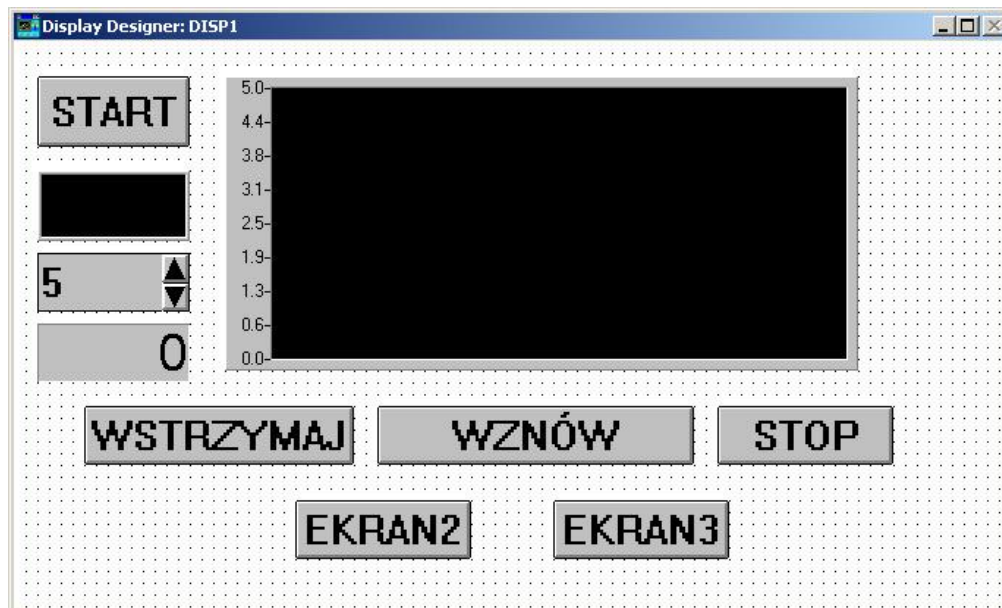


Nieco bardziej złożona jest konfiguracja Trend Graph'a. Po pierwsze trzeba określić zakresy osi : pionowa od 0 do 5, pozioma właściwie dowolna – można zostawić domyślne 0 do 40. Następnie używając przycisku ADD trzeba podpiąć sygnały i nadać im kolory.



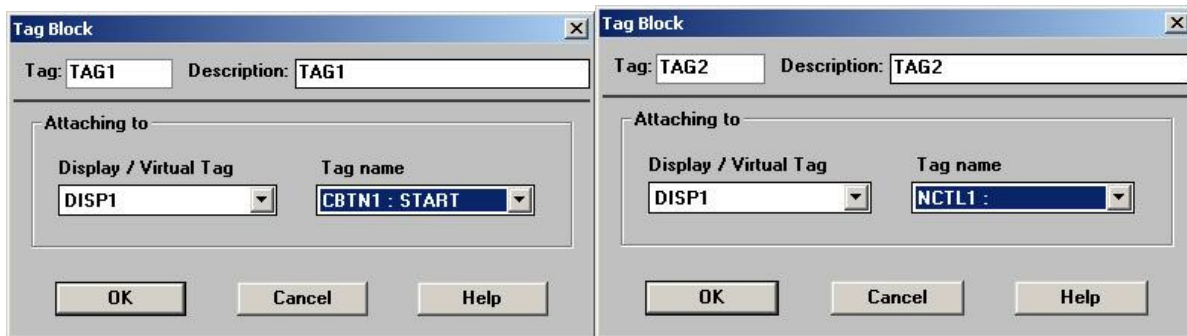


Po wykonaniu opisanych wyżej czynności i odpowiednim ustawieniu wielkości poszczególnych elementów nasz DISP1 powinien wyglądać następująco:

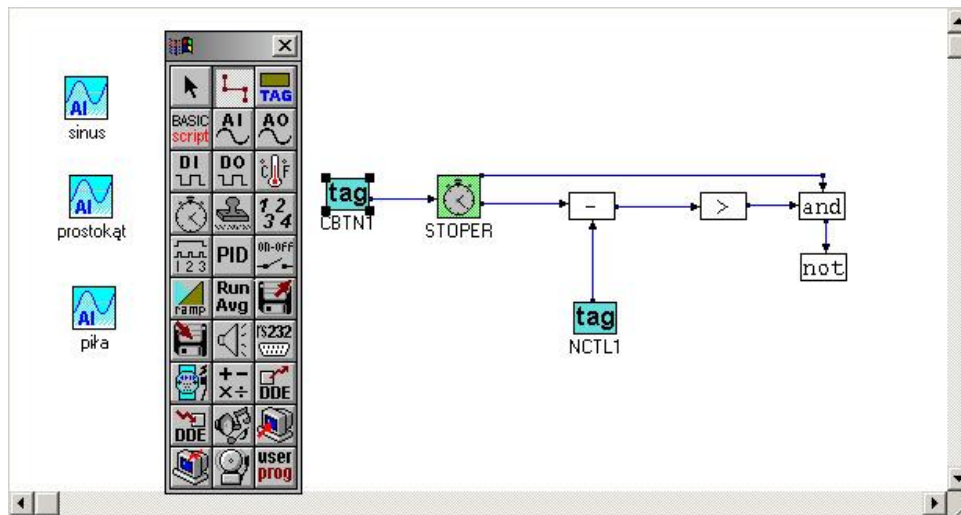


Wygląd i zawartość pozostałych dwóch ekranów pozostawia się inwencji studentów. Można na nich nawet nic nie umieszczać, choć wskazane jest zainstalowanie przycisków MENU do przełączania ekranów.

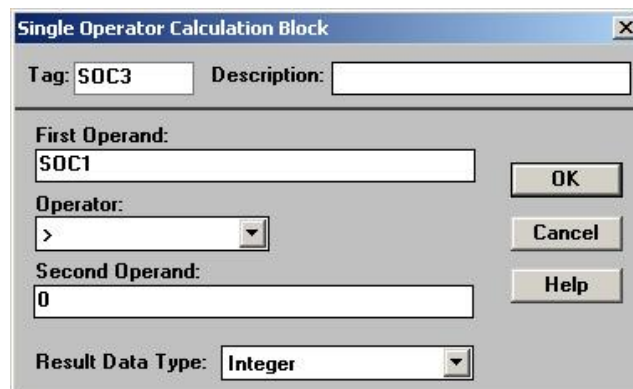
Najwyższa pora na odpowiednie podpięcie sygnałów na TASK'u. W pierwszej kolejności należy przypisać TAG'om elementy z DISP1, a mianowicie przycisk START i zadajnik wartości cyfrowej.



Następnie uaktywniając ikonę przewodu z TOOLBOX'a trzeba połączyć odpowiednie bloki.

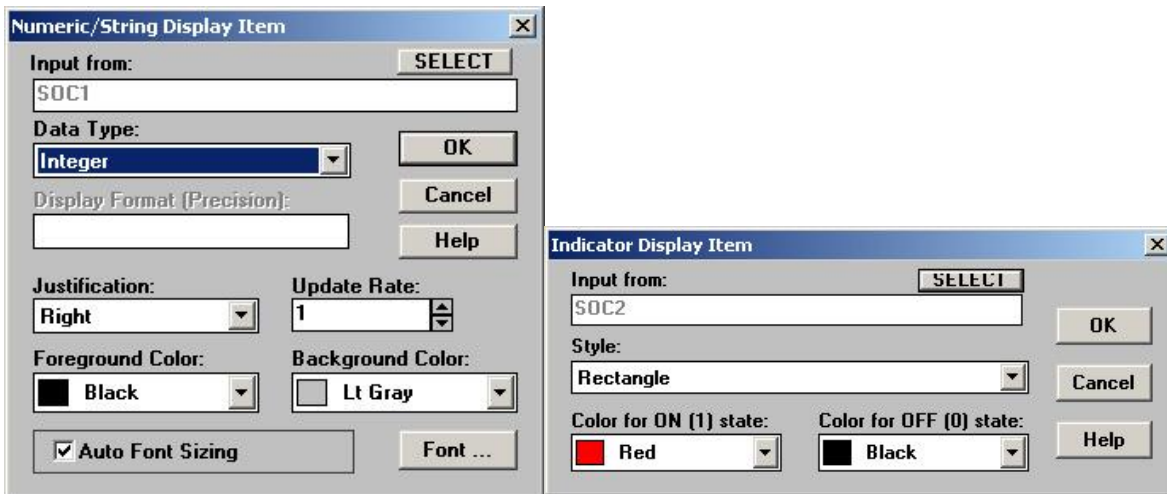


TAG powiązany z przyciskiem STRAT należy podpiąć do zerującego (Reset) wejścia zegarka. W pierwszej operacji matematycznej (SOC1) należy od TAG'a połączonego z cyfrowym polem zadajnika z DISP1 odjąć wartość wyjścia zegarka. Następnie w SOC2 trzeba sprawdzić czy wynik SOC2 jest większy od 0.

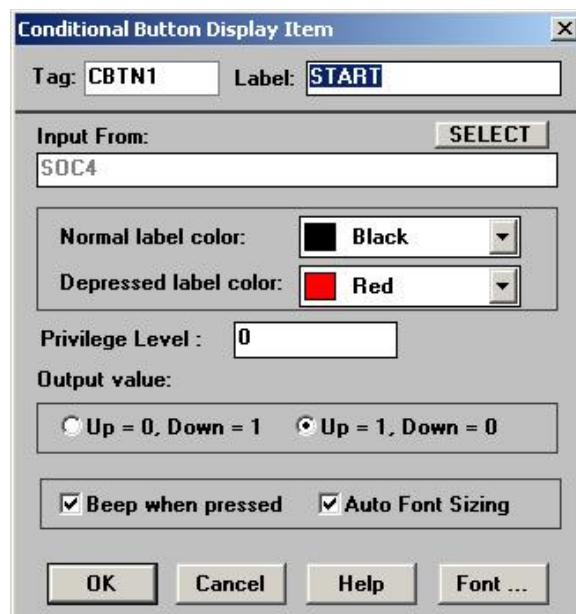


Podpinając wyjście zegarka i SOC2 do SOC3 sprawdzamy czy prawdą jest, że wyjście zegarka jest większe od zera i mniejsze od zadanej wartości sekund. Operacja negacji ustawiona w SOC4 konieczna jest dla prawidłowego wyłączenia przycisku START.

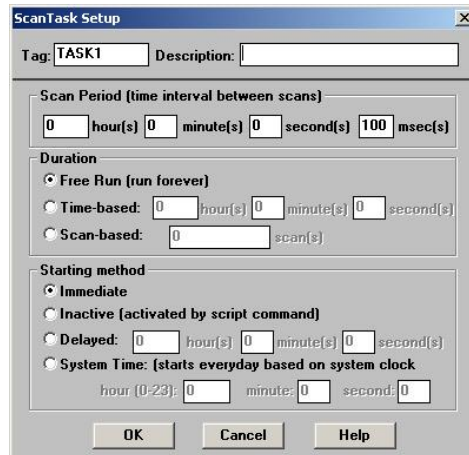
Powracamy do DISP1 i kończymy konfigurację sygnałów. Do wyświetlacza wartości cyfrowych podłączamy SOC1 (różnica wartości zadanego czasu i wyjścia z zegarka) a do pola wskaźnika wynik SOC2.



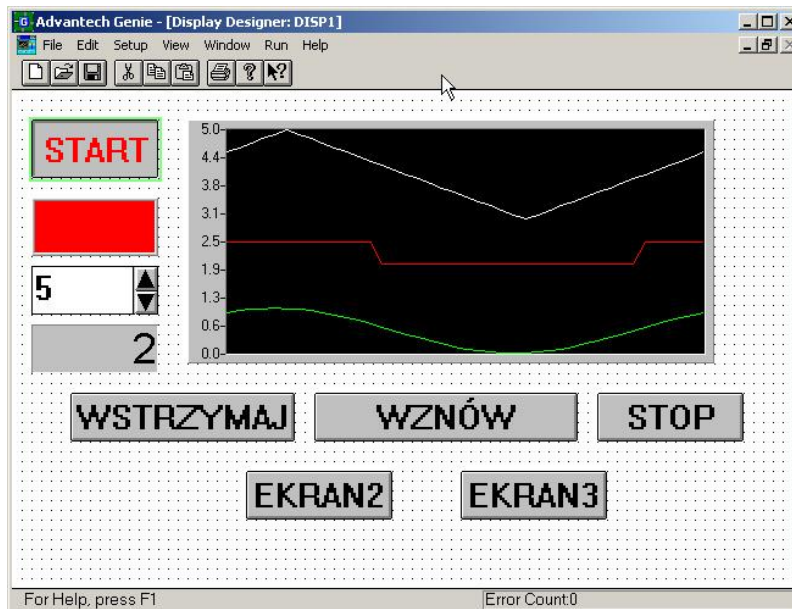
Pozostaje jeszcze skonfigurowanie przycisku warunkowego START. Ponieważ zerowanie zegarka aktywowane jest stanem wysokim, należy odwrócić działanie przycisku START – gdy nie jest wciśnięty daje „1”. Wejście warunkowe przycisku należy pobrać z SOC4.



Aby nasza strategia działała dość szybko zmieniamy w SETUP->TASK PROPERTIES okres skanowania na 100ms.



Po zapisaniu na dysk naszej strategii można ją uruchomić. Po uruchomieniu powinna ona wyglądać tak:



Wskazane są wszelkiego rodzaju eksperymenty rozbudowujące działanie opisanej strategii. Szerokim polem do popisu są dwa niewykorzystane ekrany.

Ćwiczenie 9. Układ sterowania ogniw słonecznych.

Cel ćwiczenia.

Stworzenie zaawansowanej aplikacji sterującej ruchem w nadążnym systemie ogniw słonecznych.

Jako sprzęt pośredniczący między komputerem a obiektem starowania wykorzystano moduły DAQ rodziny ADAM4000. Aplikacja sterująca tworzona jest w środowisku VisiDAQ .

Opis aparatury.

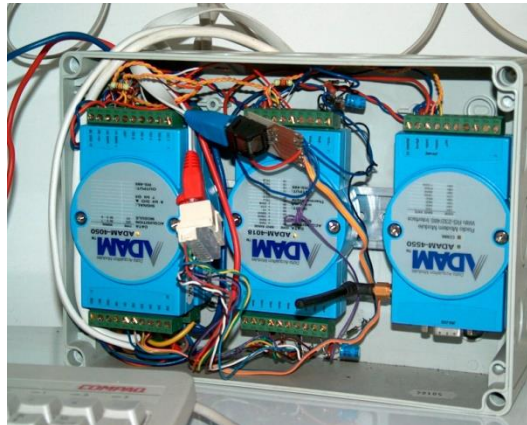
Podręcznik użytkownika VisiDAQ (plik VisiDaq.pdf), opisy techniczne modułów DAQ ADAM4000 (odpowiedni katalog na stronie internetowej)



Fot. 9.1 Model nadążnego układu ogniw fotowoltaicznych

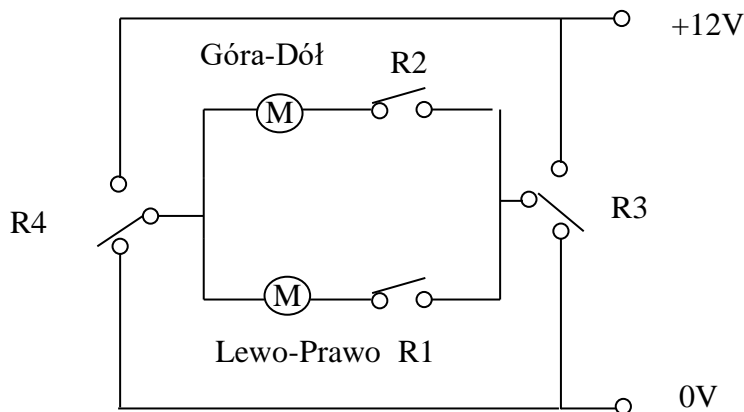
Opis stanowiska.

Stanowisko laboratoryjne składa się z modelowego układu śledzenia pozycji słońca, pozwalającego na niezależny obrót baterii słonecznych wokół osi pionowej (azymut) i poziomej (elewacja). Układ mechaniczny posiada kierunkowy czujnik oświetlenia, czujniki aktualnej pozycji fotoogniw oraz krańcowe czujniki i wyłączniki ruchu (działające całkowicie niezależnie od sterowania i zabezpieczające układ przed uszkodzeniem). Silniki zasilane są prądem stałym o napięciu 12V. Sterowanie silnikami odbywa się za pomocą modułu przekaźnikowego ADAM4060, sygnały napięciowe z czujnika oświetlenia i czujnika położenia fotoogniw podawane są na wejścia modułu ADAM4018 natomiast dyskretne sygnały z wyłączników krańcowych odbiera moduł ADAM4050. Wszystkie moduły połączone są wspólną magistralą RS485 i dalej za pośrednictwem modemu radiowego z komputerem.



Fot. 9.2 Moduły kontrolno-pomiarowe ADAM4000

Sposób sterownia silnikami przedstawiono na rys.9.1 .



Rys. 9.1 Układ sterowania silnikami

R1-R4 – przekaźniki modułu ADAM4060

Odczyty sygnałów z obiektu:

-oświetlenie padające z góry	ADAM4018 (adres 01)	kanał 0
-oświetlenie padające z dołu	ADAM4018 (adres 01)	kanał 1
-oświetlenie padające z lewa	ADAM4018 (adres 01)	kanał 2
-oświetlenie padające z prawa	ADAM4018 (adres 01)	kanał 3
-pozycja w pionie	ADAM4018 (adres 01)	kanał 4
-pozycja w poziomie	ADAM4018 (adres 01)	kanał 5
-sygnalizacja górnego wył. krańcowego	ADAM4050 (adres 03)	bit 3
-sygnalizacja dolnego wył. krańcowego	ADAM4050 (adres 03)	bit 2
-sygnalizacja lewego wył. krańcowego	ADAM4050 (adres 03)	bit 1
-sygnalizacja prawego wył. krańcowego	ADAM4050 (adres 03)	bit 0

Sygnalizacja jest w logice odwróconej – normalnie logiczna „1”, a gdy wyłącznik zadziała to logiczne „0”.

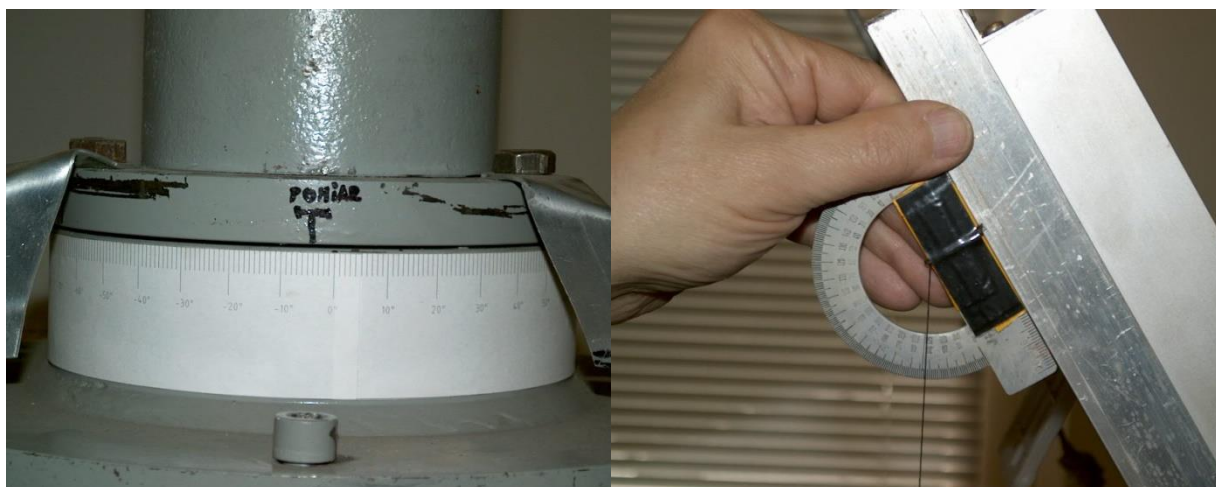
Ustawienie wyjść modułu ADAM4060 (adres 02) przy sterowaniu silnikami:

RUCH/PRZEKAŻNIK	R1 (bit 0)	R2 (bit 1)	R3 (bit 2)	R4 (bit 3)
Do góry	0	1	0	1
Na dół	0	1	1	0
W lewo	1	0	1	0
W prawo	1	0	0	1

Oczywiście jest to tylko fragment możliwych kombinacji ustawień przełączników. Pozostałe niech będą tematem własnych przemyśleń Czytelnika w ramach przygotowań do wykonania tego ćwiczenia.

Wykonanie ćwiczenia.

Ćwiczenie polega na stworzeniu aplikacji w środowisku VisiDAQ nadzorującej pracę nadążnego układu fotoogniw. Aplikacja powinna umożliwiać ręczne sterowanie systemem. Na panelu operatorskim powinny znaleźć się wskaźniki obrazujące aktualne ustawienie fotoogniw, zadziałanie wyłączników krańcowych, odczyty z czujnika oświetlenia i klawisze do sterowania ręcznego.



Fot. 9.3 Skala kąta azymutu i sposób pomiaru kąta elewacji

Sygnały napięciowe z czujników położenia należy przeliczyć na pozycję azymutu i elewacji wyrażoną w stopniach. Odpowiednia stopniowa skala dla kąta azymutu znajduje się na obudowie urządzenia. Do pomiaru kąta elewacji należy użyć prostego kątomierza grawitacyjnego.

Procedura skalowania powinna wyglądać następująco:

- za pomocą klawiszy sterowania ręcznego na panelu operatorskim doprowadzić ogniwa w skrajne lewe-górne położenie (zadziałają sygnalizatory wyłączników krańcowych)
- odczytać i zanotować wartości napięć z czujników położenia (dokładność dwa miejsca po przecinku)
- odczytać i zanotować wartości kątów azymutu i elewacji z odpowiednich skali (uwaga: kąty azymutu mogą przyjmować również wartości ujemne)
- powtórzyć czynności dla skrajnego prawego-dolnego położenia ogniw
- zanotowane wartości kątów i napięć wpisać w odpowiednie rubryki menu skalowania wejścia analogowego.

Przy obserwacji pomiarów oświetlenia można posłużyć się reflektorem halogenowym przymocowanym do metalowego regału.

Rozwinięciem (nieobowiązkowym) ćwiczenia jest próba stworzenia aplikacji automatycznie śledzącej źródło światła.

Sprawozdanie.

Sprawozdanie powinno zawierać krótkie omówienie istoty sterowania nadążnym układem fotowoltaicznym. Dalej należy umieścić schemat stanowiska laboratoryjnego uwzględniający wszystkie sygnały pomiarowe i sprzęt służący do ich przetwarzania. W końcowej części ma się znaleźć dokładny opis stworzonej aplikacji (również rysunki) oraz omówienie problemów i ograniczeń, które wystąpiły podczas pracy.

Zakres wymaganych wiadomości.

Zasady tworzenia aplikacji w środowisku Genie 3.0, budowa, działanie i sposoby programowania modułów zdalnej akwizycji danych i sterowania (szczególnie ADAM4018, ADAM4050, ADAM4060), zasady budowania rozproszonych systemów sterowania w oparciu o magistralę RS485.

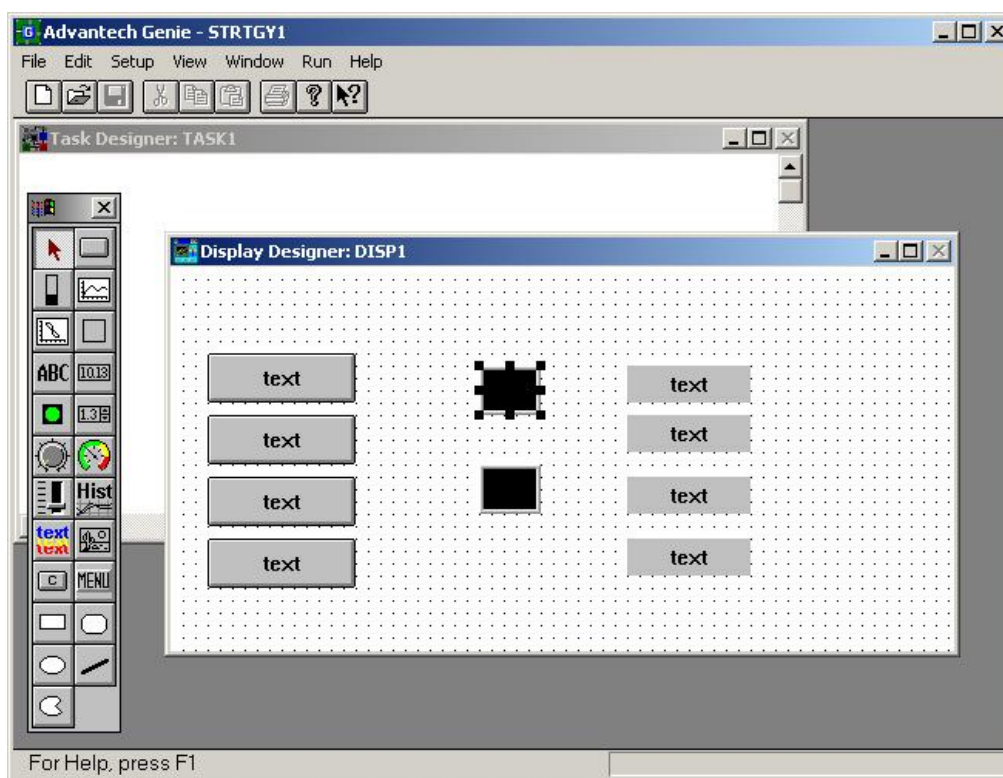
Materiały pomocnicze do ćwiczenia 9.

Zagadnienia odpowiedniego podpinania i skalowania kanałów analogowych zostały omówione w materiałach pomocniczych do ćwiczenia 8 dlatego też tym razem zajmiemy się wejściami i wyjściami dyskretnymi oraz układami logicznymi z wykorzystaniem klawiszy.

Naszym zadaniem będzie zbudowanie enkodera liczb dziesiętnych (od 0 do 3) na binarne (2 bity).

Liczbę binarną wyświetlimy na wyjścia dyskretna i kontrolnie odczytamy ją z wejść dyskretnych (wykorzystując wirtualne połączenie odpowiednich kanałów dyskretnych urządzenia ADVANTECH DEMO I/O).

Rozpoczęcie tworzenia nowej strategii zostało omówione w materiałach pomocniczych do ćwiczenia 8 dlatego zostanie tutaj pominięte. W obszarze roboczym DISP1 umieszczamy cztery klawisze (zadawanie wartości dziesiętnej), dwa pola sygnalizacyjne (sygnalizacja bitów liczby binarnej) i kilka pól tekstowych, które pozwolą czytelnie opisać panel operatorski.

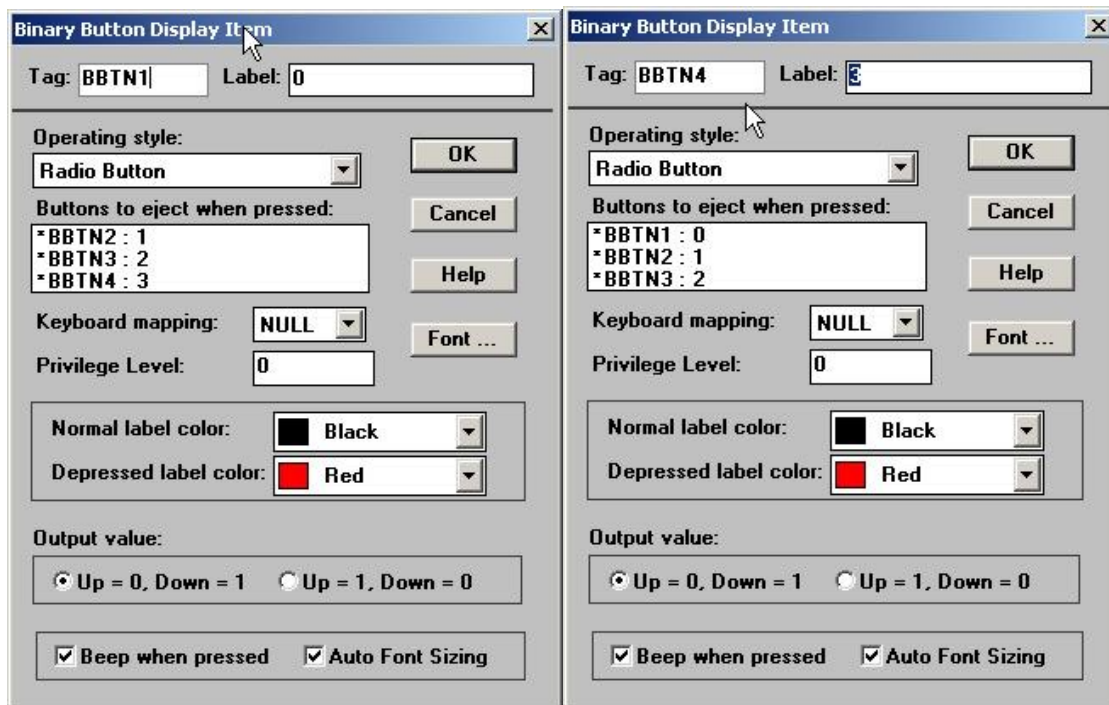


Należy się zastanowić nad sposobem przeliczania liczb z postaci dyskretnej na binarną. Zbudujmy tabelkę prawdy dla zadanego zakresu liczb:

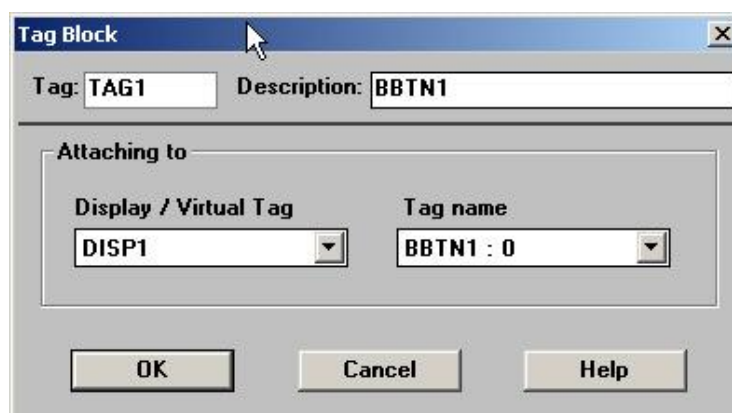
L. dziesiętna \ L. binarna	Bit 1	Bit 0
0	0	0
1	0	1
2	1	0
3	1	1

Z tabelki jasno widać, że bit 0 jest uaktywniony gdy wciśnięty jest klawisz „1” lub „3” a bit 1 gdy wciśnięte są klawisze „2” lub „3”. Wynika z tego, że poszczególne bity mogą być sterowane wartościami operacji sumy logicznej „or” właściwych sygnałów wejściowych.

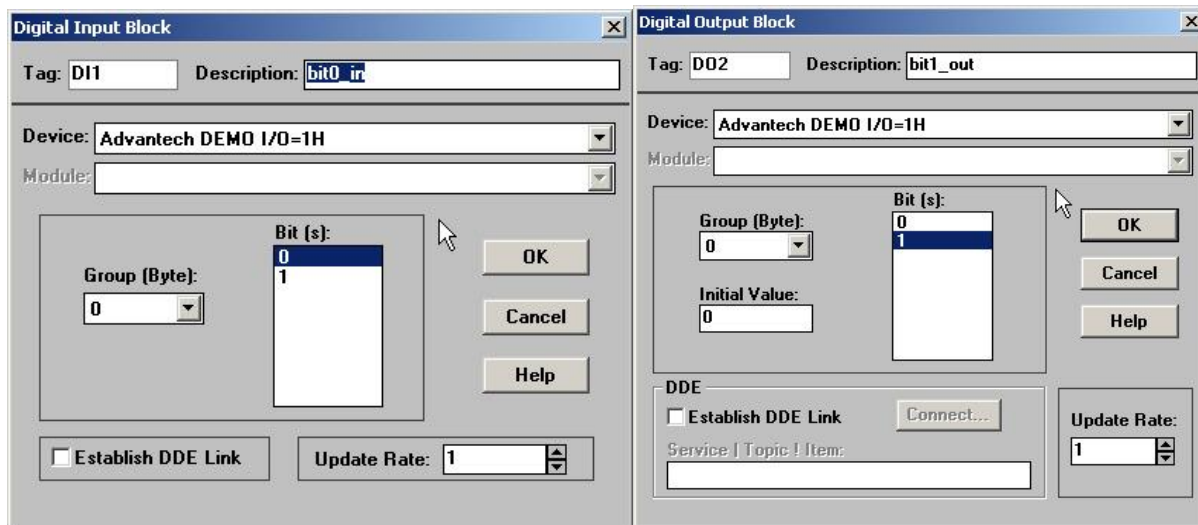
Po tych rozważaniach teoretycznych przystępujemy do konfiguracji obszaru TASK'a. Umieszczamy na nim pięć TAG'ów (do odpowiednich klawiszy), dwa bloki operacji matematycznych, dwa wejścia dyskretne i dwa wyjścia dyskretne. Warto teraz na chwilę wrócić do DISP1 i skonfigurować klawisze. Po pierwsze trzeba nadać im nazwy (0,1,2 i 3), ustawić ich typ na RADIO BUTTON (wciśnięcie jednego klawisza zwalnia pozostałe) i zaznaczyć relacje wzajemnego zwalniania.



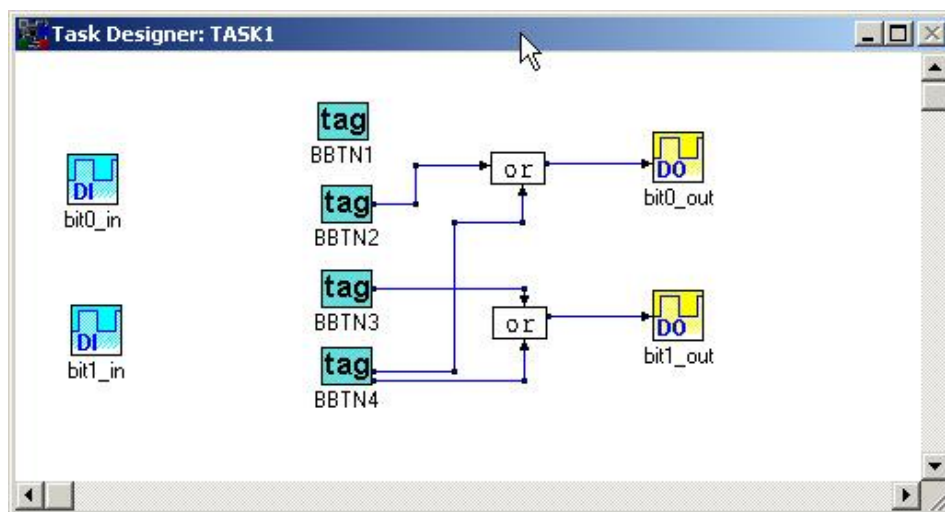
Powyższe rysunki przedstawiają konfigurację klawiszy „0” i „3”. W oknie „Buttons to eject when pressed” poprzez dwukrotne kliknięcie na nazwę zaznaczamy klawisze, które mają zostać zwolnione przy wciśnięciu klawisza obecnie konfigurowanego. Wracamy do TASK'a i łączymy TAG'i z klawiszami.



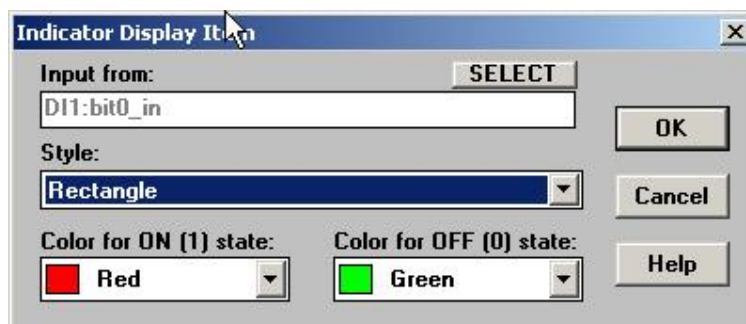
Następnie konfigurujemy bloki wejść i wyjść dyskretnych.



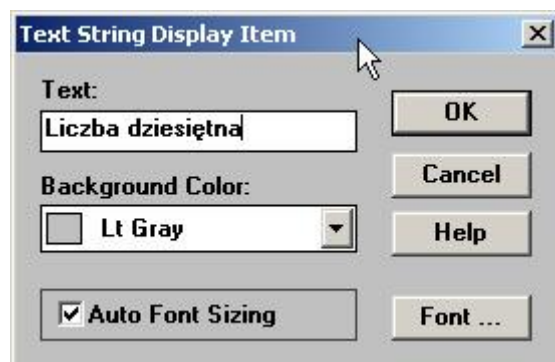
W blokach operacji matematycznych ustawiamy operację „or” i łączymy wszystkie ikony na TASK’u.



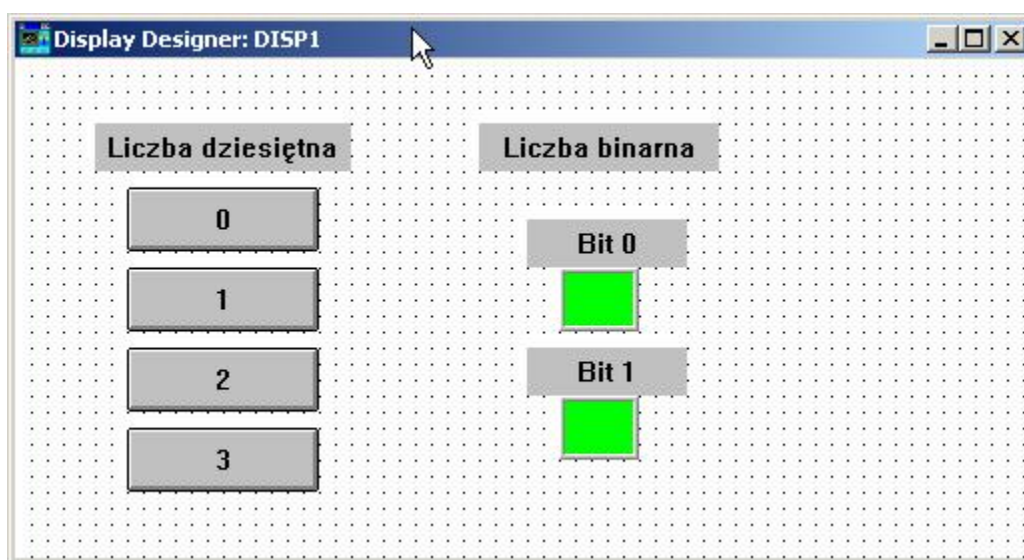
Końcowa konfiguracja DISP1 polega na podpięciu pól wskaźników logicznych do odpowiednich wejść dyskretnych i nadaniu im kolorów (aktywny-czerwony, nieaktywny-zielony).



Za pomocą pól tekstowych można jednoznacznie opisać elementy występujące na panelu operatorskim.



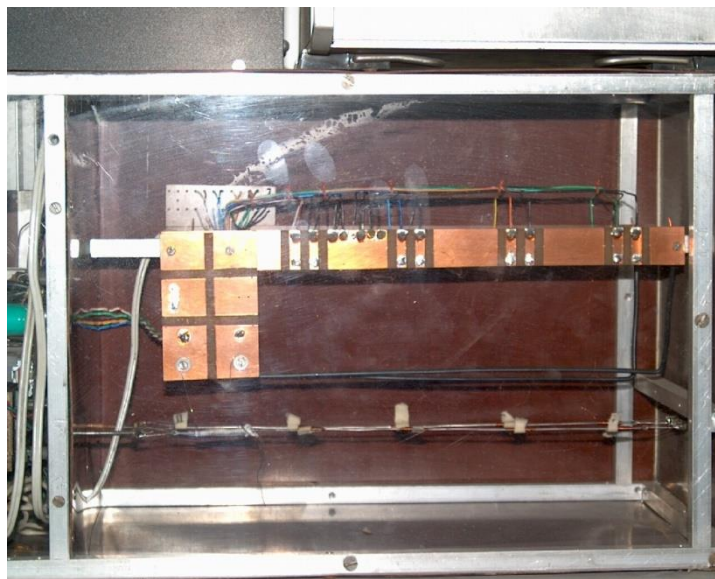
W efekcie końcowym panel operatorski może wyglądać na przykład tak:



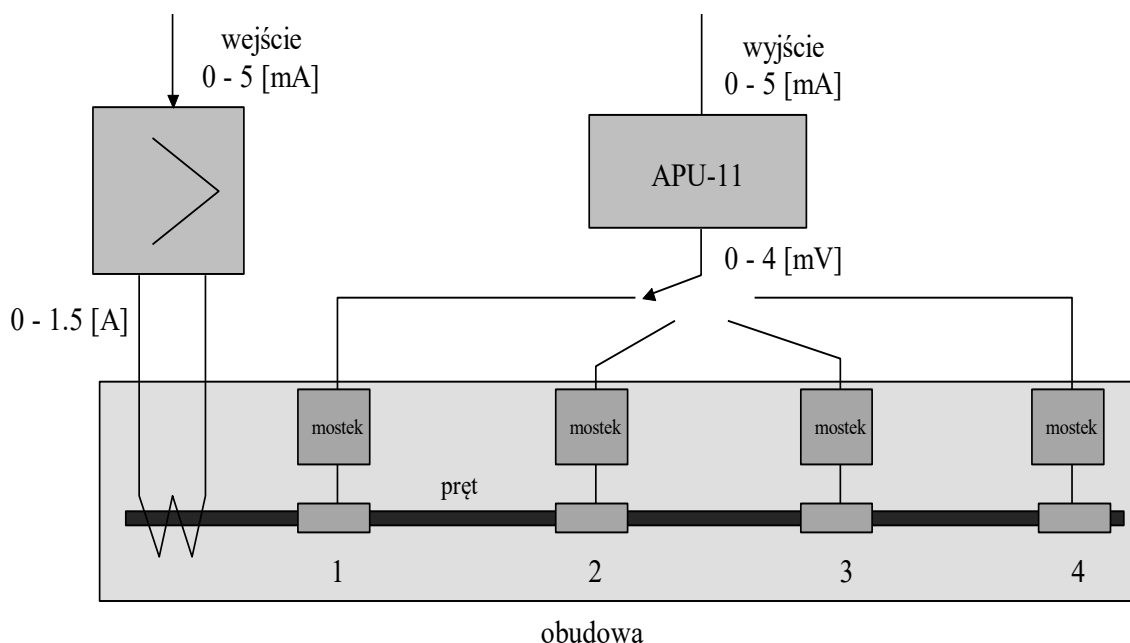
Pozostaje tylko ustawienie okresu skanowania na 100ms (SETUP->Task Properties), nagranie strategii na dysk pod dowolną nazwą i można przetestować jej działanie (RUN->START).

Dodatek 1. Laboratoryjny obiekt regulacji.

Stosowany w Laboratorium Aparatury Automatykacji eksperymentalny obiekt regulacji jest obiektem typu cieplnego, przy czym zarówno sygnał sterujący obiekt, jak i sygnał wyjściowy mają charakter elektryczny ze znormalizowanego zakresu 0 - 5 [mA]. Widok obiektu pokazany jest na fot. D1, a jego uproszczona budowa na rys. D1:



Fot. D1 Widok obiektu regulacji



Rys. D1. Uproszczony schemat obiektu laboratoryjnego.

Obiekt jest prętem metalowym (miedzianym) długości 260 [mm] umieszczonym w obudowie zapewniającej izolację termiczną od otoczenia. Obiekt jest ogrzewany elementem grzejnym zamocowanym na jednym z końców. Temperatura pręta jest mierzona termometrami rezystancyjnymi nawiniętymi w punktach: 0.3, 0.5, 0.7 oraz 0.9 długości pręta.

Sygnałem wejściowym obiektu jest sygnał 0 - 5 [mA] podawany na wejściowy wzmacniacz mocy. Na wyjściu wzmacniacza otrzymujemy sygnał z zakresu 0 - 1.5 [A] proporcjonalny do sygnału wejściowego, który jest podawany na element grzejny. Zmiana temperatury pręta powoduje zmianę rezystancji termometrów oporowych nawiniętych na pręt. Zmiana rezystancji termometru jest przetwarzana na zmianę napięcia na wyjściu mostka pomiarowego. Zmiana ta zawiera się w zakresie 0 - 4 [mV]. Napięcie z mostka jest podawane na przetwornik pomiarowy APU-11, na którego wyjściu otrzymujemy sygnał prądowy z zakresu 0 - 5 [mA], który jest sygnałem wyjściowym obiektu. Wyboru wyjścia obiektu dokonuje się przełącznikiem, poprzez przełączenie odczytu temperatury z wybranego czujnika (zob. rysunek).

UWAGA: Opiswany obiekt jest najlepiej widoczny przy stanowisku ćwiczeniowym dla ćwiczenia nr 6 (bezpośrednio sterowanie cyfrowe). W niniejszym opisie dla ułatwienia nie uwzględniono układu kompensacji temperatury, który znajduje się w rzeczywistych obiektach. Przy ćwiczeniu nr 6 **na stałe** podłączone jest wyjście obiektu nr 2 (termometr oporowy w połowie długości pręta) - nie ma możliwości przełączenia.

Z punktu widzenia automatyki jako obiekt regulacji traktowana jest całość, tj. pręt wraz z wzmacniaczem wejściowym, elementem grzejnym, termometrami oporowymi oraz przetwornikiem APU-11. Jest to uzasadnione faktem, że dynamika elementów wejściowych i wyjściowych (wzmacniacz mocy, termometry oporowe, przetwornik APU-11) jest pomijalnie mała w porównaniu z prętem.

Rozważany obiekt regulacji może być uważany za obiekt liniowy tylko w pewnym ograniczonym zakresie sygnałów wejściowych (od ok. 20% do ok. 90% wartości maksymalnej) i przy doborze punktu pracy na liniowej części charakterystyki statycznej może być opisany modelem w postaci transmitancji I rzędu z opóźnieniem mającej postać:

$$G(s) = \frac{ke^{-s\tau}}{1 + sT}$$

Parametry zastępcze tego modelu mogą być wyznaczone drogą identyfikacji obiektu na podstawie odpowiedzi na skok jednostkowy (charakterystykę skokową należy zmierzyć na liniowej części charakterystyki statycznej). Wartość wzmocnienia statycznego obiektu k jest w przybliżeniu równa: $k = 1.08$. Było to możliwe do osiągnięcia dzięki odpowiedniemu doborowi parametrów mostków pomiarowych. Wartości pozostałych parametrów zastępczych modelu będą różne w zależności od wyjścia obiektu oraz sytuacji, czy rozważamy grzanie czy chłodzenie obiektu. Ich przybliżona wartość jest równa:

Wyjście obiektu	Grzanie		Chłodzenie	
	T [s]	τ [s]	T [s]	τ [s]
1	35	8.7	37.5	10
2	46.2	17.5	51.2	18.7
3	62.5	28.7	65	30
4	72.5	41.2	77.5	42.5

Literatura

- [1] Findeisen W. “Technika regulacji automatycznej” PWN W-a 1978.
- [2] Kwaśniewski J. “Wprowadzenie do inteligentnych przetworników pomiarowych”, WNT W-a 1993.
- [3] Kwaśniewski J. “Programowalne sterowniki przemysłowe w systemach sterowania” Kraków 1999.
- [4] Bateson R. N. “Introduction to control system technology” Prentice Hall 1996.
- [5] Legierski T. i inni “Programowanie sterowników PLC” Wyd. Prac. Komp. J. Skalmierski, Gliwice 1998.
- [6] Nise N. S. “Control systems engineering” Wiley & Sons, New York 2000.
- [7] Shell E. D. Hall E. L. (editors) “Handbook of industrial automation” Marcel. Dekker, New York, Basel 2000.
- [8] Skoczowski S. “Technika regulacji temperatury” PAR 2000.
- [9] Takahashi Y. “Sterowanie i systemy dynamiczne” WNT W-a 1976.
- [10] Tuszyński K. , Walewski M. “Regulacja automatyczna w inżynierii chemicznej” WNT W – a 1983.
- [11] Trybus L. “Regulatory wielofunkcyjne” WNT W-a 1992.
- [12] Werewka J. “Systemy rozproszone sterowania i akwizycji danych, sterowniki” Wyd. Katedry Automatyki AGH, 1998.
- [13] p.r. Wiszniewski A. “Teoria sterowania. Ćwiczenia laboratoryjne” Wyd. Pol. Wr. Wrocław 1997.
- [14] J.Kwaśniewski :”Sterowniki SIMATIC S7-1200 w praktyce inżynierskiej.” Wydawnictwo BTC Legionowo 2013