



**AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE**

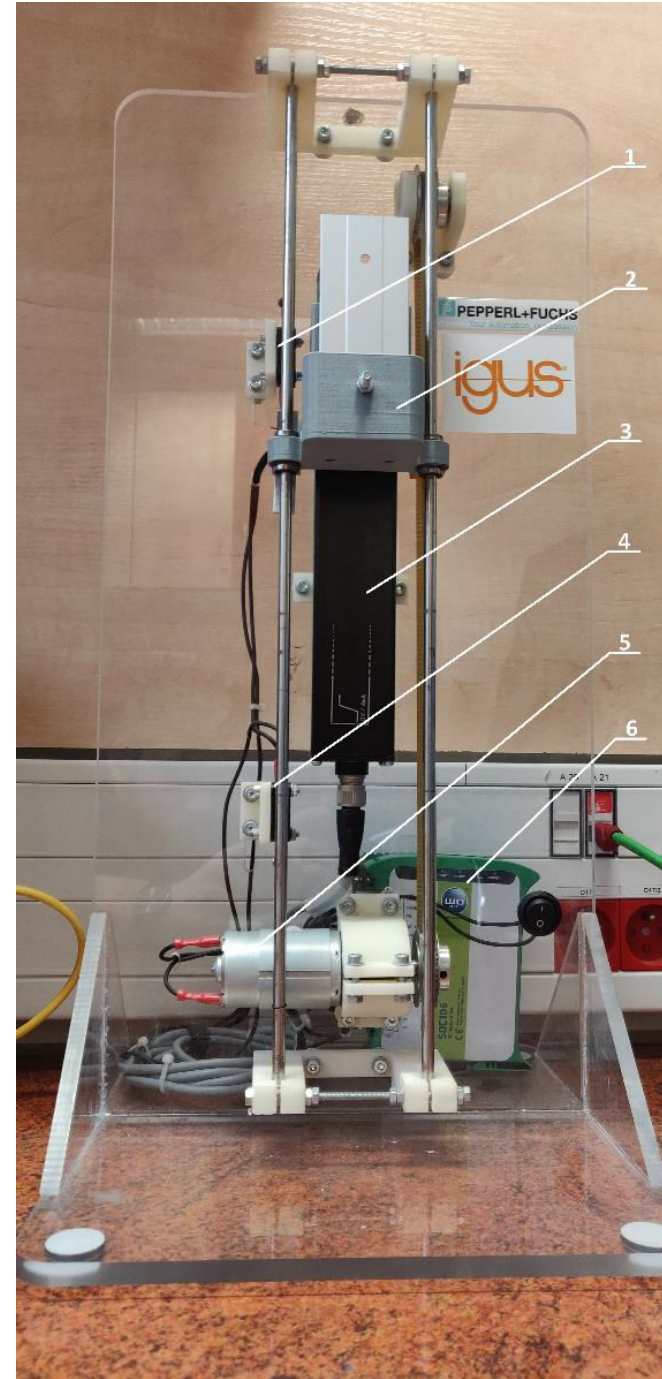
**AGH UNIVERSITY OF SCIENCE
AND TECHNOLOGY**

Pozycjonowanie z napędem DC laboratorium

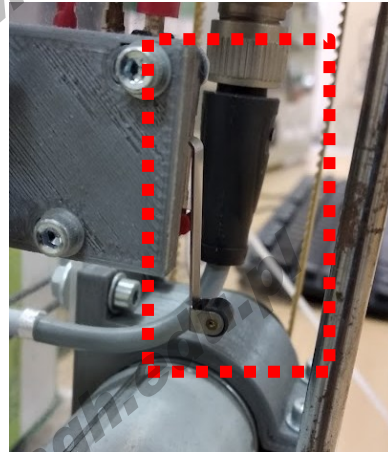
Stanisław Flaga

stanislaw.flaga@agh.edu.pl

Stanowisko

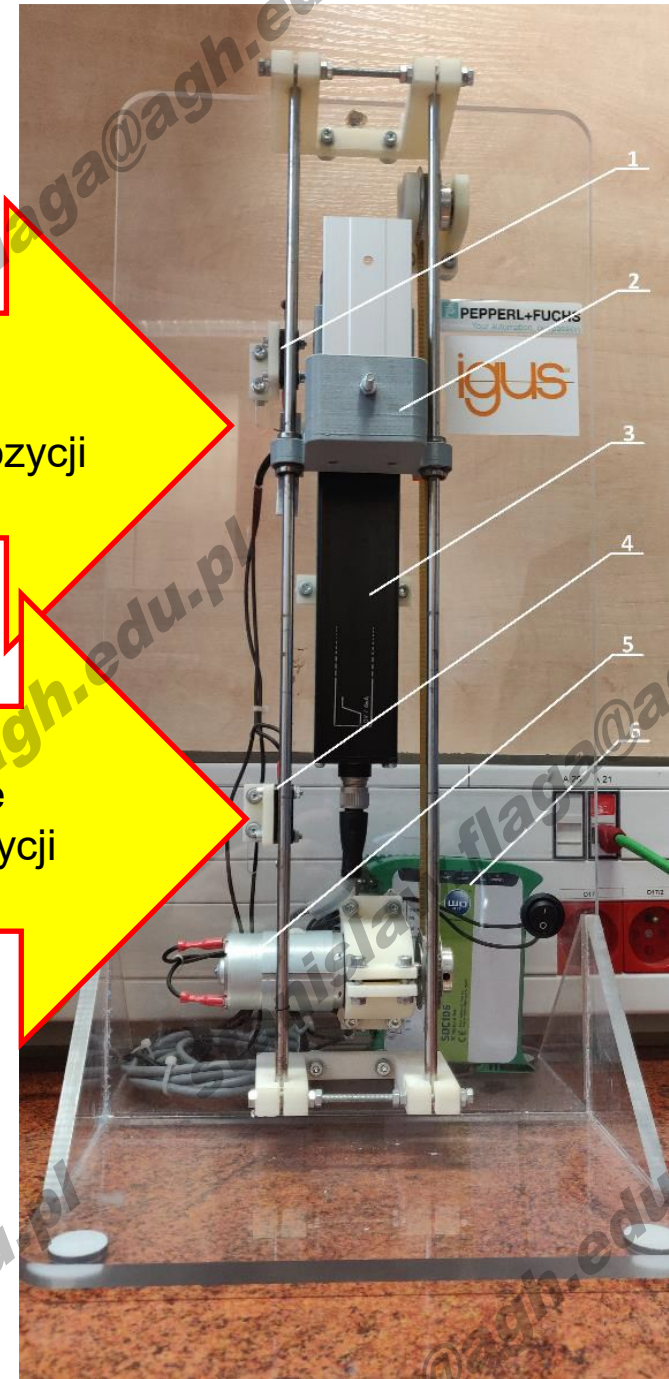


Krańcówki fizyczne



Sprzętowe zabezpieczenie maksymalnej pozycji górnej

Sprzętowe zabezpieczenie minimalnej pozycji dolnej



UWAGA

Stany tych krańcówek **NIE SĄ WIDZIANE** na wejściach PLC

Krańcówki fizyczne

UWAGA: Jeżeli skutkiem złego działania programu **wjedziesz w jedną z tych krańcówek** rozłączysz obwód „Start” sterownika silnika DC i **musisz RĘCZNIE** wyjechać z pozycji zabronionej

NIE WOLNO CIĄGNAĆ ZA PASEK !!!!!!!!



Musisz obracać kołem na wale silnika



Zadanie 1.

Temat

Napisz program którego główne funkcjonalności to:

- po trzymaniu w pozycji wciśniętej **S2** przemieszczanie karetki w górę lub w dół – w zależności od wybranego kierunku,
- wybór/zmiana kierunku: w górę – impuls na **S1**, w dół impuls na **S3**,
- przyciski **S1**, **S2**, **S3** powinny zostać zdublowane na panelu HMI,
- **karetką nie może przekroczyć** pozycji ustalonych krańcówek wirtualnych,
- panel HMI powinien pozwolić na wpisanie nastawy prędkości (**QW80**) maksymalna dopuszczalna nastawa (liczba wpisana do rejestru – **przelicz na napięcie**) wynosi **9000**,
- na HMI wyświetl aktualną nastawę w Voltach oraz pozycję w mm z rozdzielczością co 0,1 mm

Uwaga

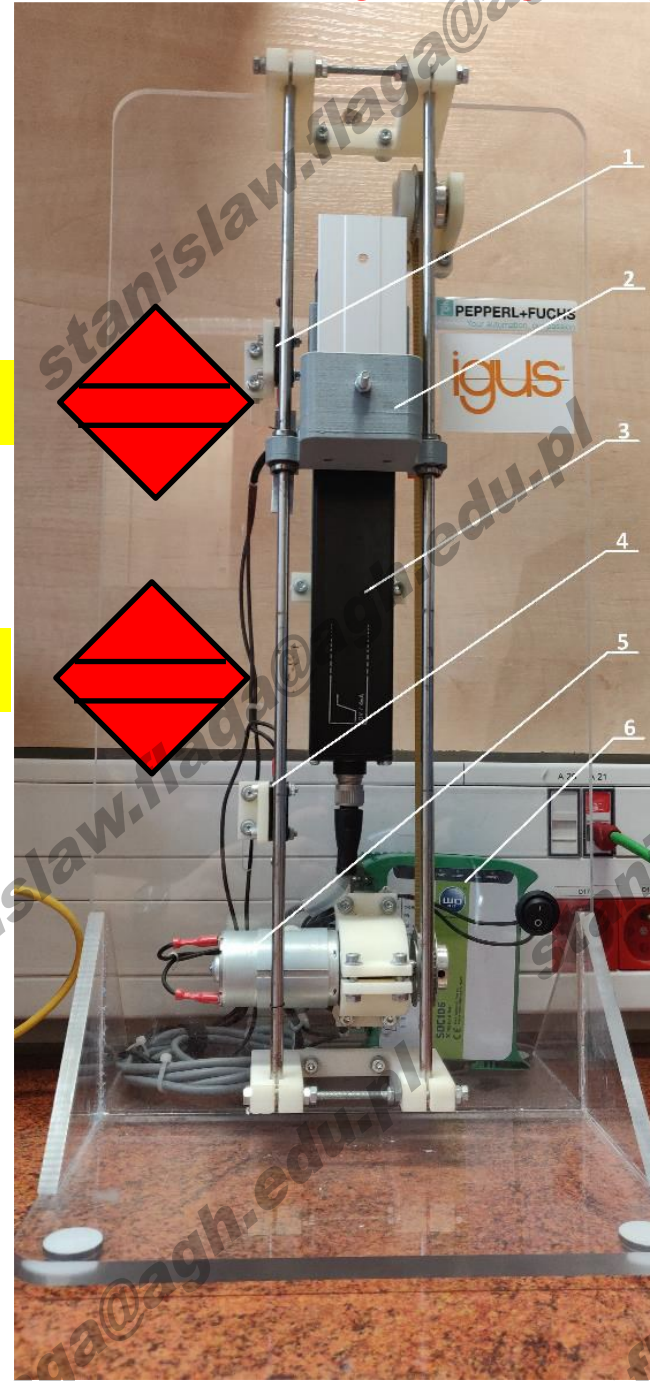
1. Zamaskuj wartość odczytaną z potencjometru i przetwornika drogi wiedząc, że przetwornik A/D jest 10 bitowy.
2. **$V_{max} \leq 9000$.**

Upewnij się, że działają wirtualne krańcówki- będziesz musiał samodzielnie je zaprogramować

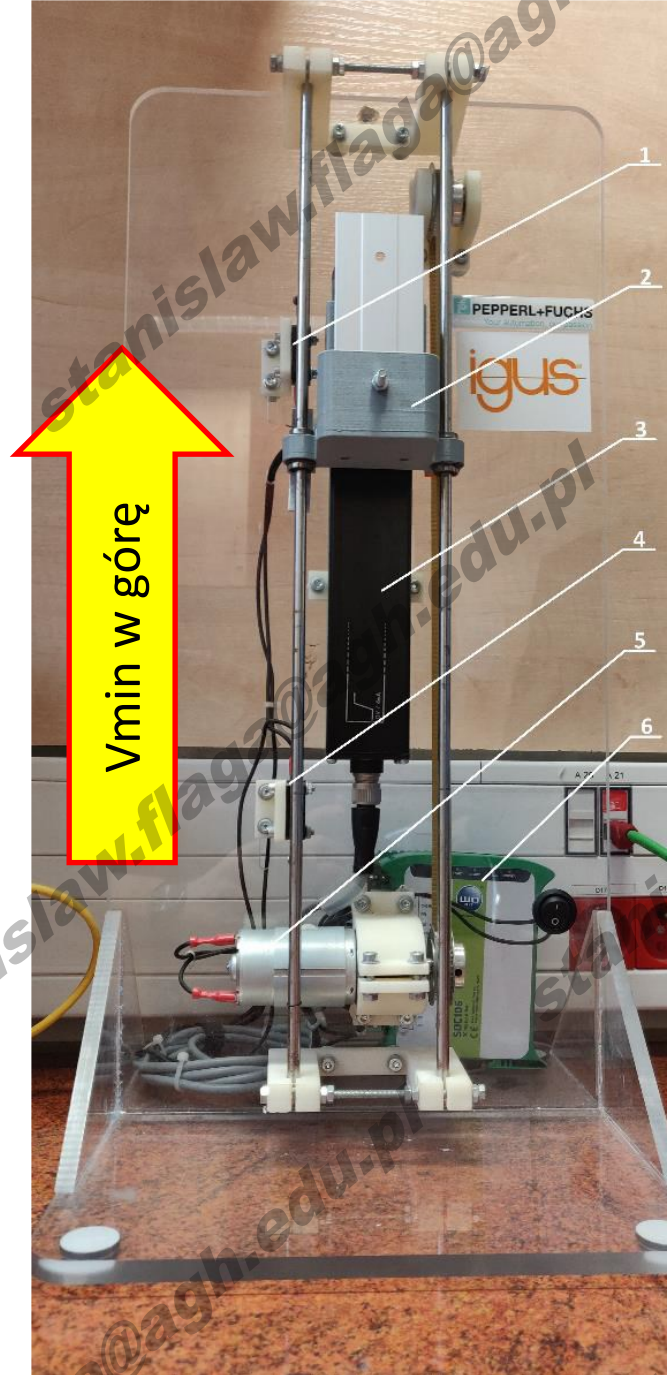


Wirtualna krańcówka górna

Wirtualna krańcówka dolna



Doświadczalnie odszukaj najmniejszą wartość podaną na QW80 przy której wózek porusza się „płynnie” w górę i w dół



Przyjmij, że V_{max} zadane na QW80 ≤ 9000

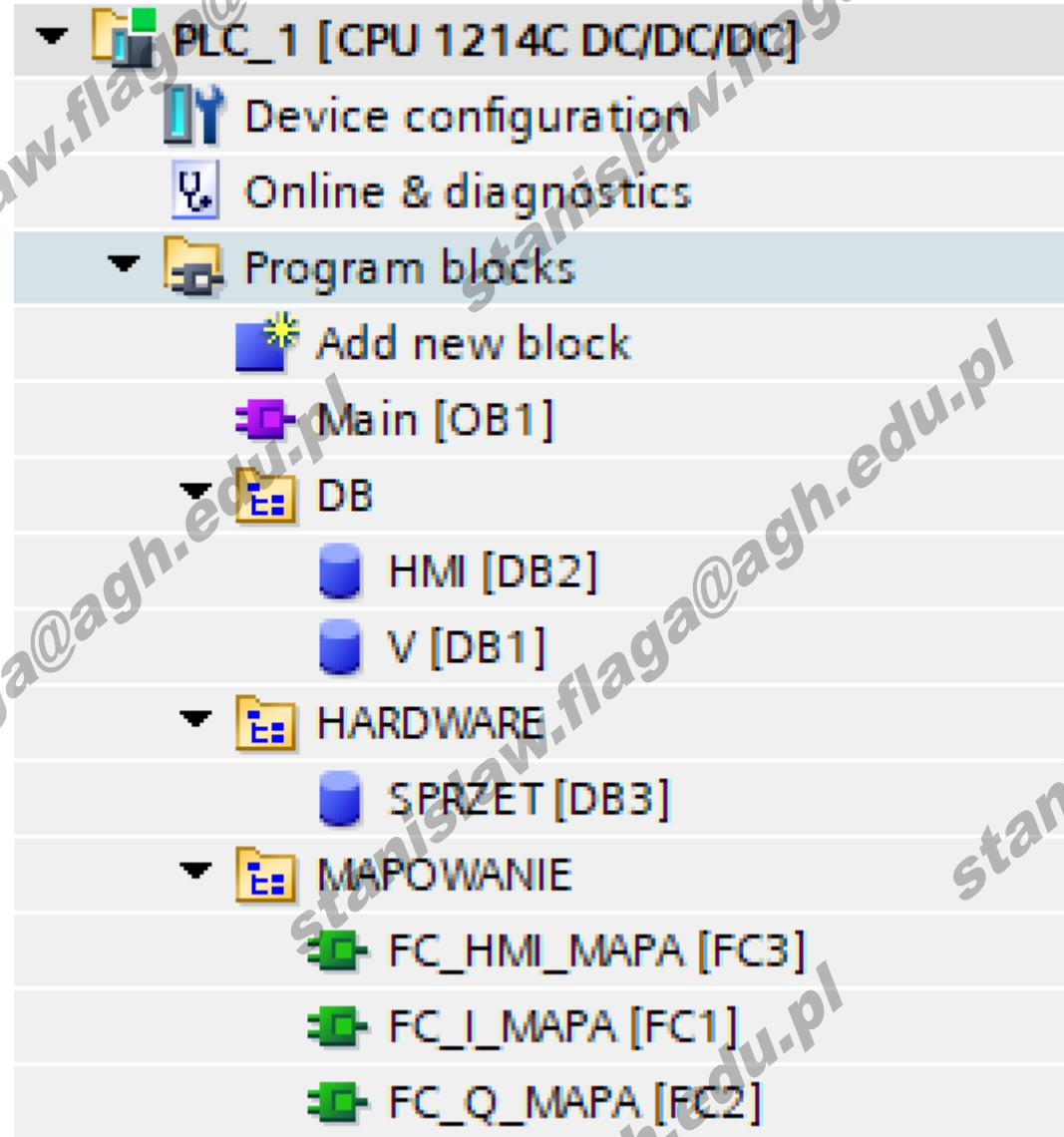
V_{max} dla QW ≤ 9000



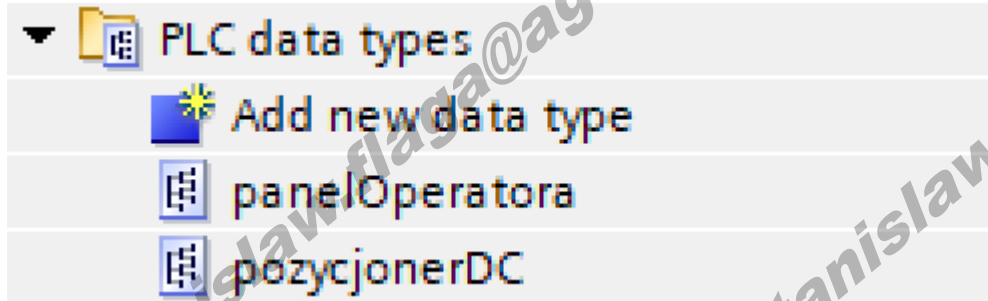
Przypisania wejść / wyjść PLC, **zbuduj tablicę tagów**









Name	Data Type	Logical Address	Comment
zadajnik	Int	%IW64	Wejście napięciowe AI0 - potencjometr
pozycja	Int	%IW66	Wejście napięciowe AI1 - przetwornik przemieszczenia
predkosc	Int	%QW80	Wyjście napięciowe - zadawanie prędkości obrotowej silnika – AQ0
S1	Bool	%i0.0	
S2	Bool	%i0.1	
S3	Bool	%i0.2	
driverDCstart	Bool	%Q1.0	Na wejście START sterownika silnika DC
driverDCdir	Bool	%Q1.1	Na wejście DIR sterownika silnika DC: 1-góra, 0-dół
H1	Bool	%Q0.0	
H2	Bool	%Q0.1	
H3	Bool	%Q0.2	

Utwórz strukturę programu



Zdefiniuj dwa typy



panelOperatora			
		Name	Data type
1		S1	Bool
2		S2	Bool
3		S3	Bool
4		H1	Bool
5		H2	Bool
6		H3	Bool
7		zadajnik	Int
8		zadajnik_01mm	Int
9		zadajnikZnormalizowany	Real

Poruszanie prosty projekt – zdefiniowane typy

- PLC data types
 - Add new data type
 - panelOperatora
 - pozycjonerDC

pozycjonerDC			
	Name	Data type	Default value
1	start	Bool	false
2	dir	Bool	false
3	predkosc	Int	0
4	predkosc_jednostka	Int	0
5	status	Int	0
6	pozycja	Int	0
7	pozycja_01mm	Int	0
8	pozycjaZnormalizowana	Real	0.0
9	limit_dolny_01mm	Int	0
10	limit_gorny_01mm	Int	0



Zdefiniuj DB: V, HMI


HMI				
	Name	Data type	Start value	Retain
	Static			<input type="checkbox"/>
	driverDCstart	Bool	false	<input type="checkbox"/>
	driverDCdir	Bool	false	<input type="checkbox"/>
	driverDCpoz	Int	0	<input type="checkbox"/>
	driverDCpoz_01mm	Int	0	<input type="checkbox"/>
	zadajnikWartosc	Int	0	<input type="checkbox"/>
	zadajnik_01mm	Int	0	<input type="checkbox"/>
	HMI_S1	Bool	false	<input type="checkbox"/>
	HMI_S2	Bool	false	<input type="checkbox"/>
	HMI_S3	Bool	false	<input type="checkbox"/>














DB

- HMI [DB2]
- V [DB1]

V			
	Name	Data type	Start value
	Static		
	pozycjoner01_start	Bool	false

DB SPRZET

 HARDWARE
 SPRZET [DB3]

SPRZET				
	Name	Data type	Start value	Retain
	Static			<input type="checkbox"/>
	pozycjoner01	"pozycjonerDC"		<input checked="" type="checkbox"/>
	start	Bool	false	<input checked="" type="checkbox"/>
	dir	Bool	false	<input checked="" type="checkbox"/>
	predkosc	Int	0	<input checked="" type="checkbox"/>
	predkosc_jednostka	Int	0	<input checked="" type="checkbox"/>
	status	Int	0	<input checked="" type="checkbox"/>
	pozycja	Int	0	<input checked="" type="checkbox"/>
	pozycja_01mm	Int	0	<input checked="" type="checkbox"/>
	pozycjaZnormaliz...	Real	0.0	<input checked="" type="checkbox"/>
	limit_dolny_01mm	Int	320	<input checked="" type="checkbox"/>
	limit_gorny_01mm	Int	1980	<input checked="" type="checkbox"/>
	panel_01	"panelOperatora"		<input type="checkbox"/>

FC1 - interfejs



Constant			
01	MASKA_10_BIT	Int	2#1111111111100000
01	ANALOG_I_MIN	Int	0
01	ANALOG_I_MAX	Int	27648
01	POZYCJA_MIN_01mm	Int	0
01	POZYCJA_MAX_01mm	Int	2000

FC1 – ciało funkcji



A

```
1 //Mapowanie wejść
2 //
3 "SPRZET".panel_01.S1 := "S1" OR "HMI".HMI_S1;
4 "SPRZET".panel_01.S2 := NOT "S2" OR "HMI".HMI_S2;
5 "SPRZET".panel_01.S3 := NOT "S3" OR "HMI".HMI_S3;
6
7 //Maskowanie 5 najmłodszych bitów na wejściach analogowych - A/D 10 bitowy
8 //Nie straci si
9 "SPRZET".pozycjoner01.pozycja := "pozycja" AND #MASKA_10_BIT;
10 "SPRZET".panel_01.zadajnik := "zadajnik" AND #MASKA_10_BIT;
11
12 //Normalizacja zadajnika i pozycji na zakres 0,0 .. 1,0
13 //
14 "SPRZET".panel_01.zadajnikZnormalizowany := NORM_X(MIN:=#ANALOG_I_MIN,
15 |                                     VALUE:="SPRZET".panel_01.zadajnik,
16 |                                     MAX:=#ANALOG_I_MAX);
17 "SPRZET".pozycjoner01.pozycjaZnormalizowana := NORM_X(MIN := #ANALOG_I_MIN,
18 |                                     VALUE := "SPRZET".pozycjoner01.pozycja,
19 |                                     MAX := #ANALOG_I_MAX);
20
21 //Przeliczenie zadajnika i pozycji na liczbę dziesiątych części mm
22 //
23 "SPRZET".panel_01.zadajnik_01mm := SCALE_X(MIN := #POZYCJA_MIN_01mm,
24 |                                     VALUE := "SPRZET".panel_01.zadajnikZnormalizowany,
25 |                                     MAX := #POZYCJA_MAX_01mm);
26 "SPRZET".pozycjoner01.pozycja_01mm := SCALE_X(MIN := #POZYCJA_MIN_01mm,
27 |                                     VALUE := "SPRZET".pozycjoner01.pozycjaZnormalizowana,
28 |                                     MAX := #POZYCJA_MAX_01mm);
29
```



FC1_MAPA [FC1]

FC2 – ciało funkcji

 FC_Q_MAPA [FC2]

```
1 "H1" := "SPRZET".panel_01.H1;  
2 "H2" := "SPRZET".panel_01.H2;  
3 "H3" := "SPRZET".panel_01.H3;  
4 "predkosc" := "SPRZET".pozycjoner01.predkosc;  
5  
6 "driverDCstart" := "SPRZET".pozycjoner01.start;  
7 "driverDCdir" := "SPRZET".pozycjoner01.dir;
```

FC3 – ciało funkcji

FC_HMI_MAPA [FC3]

```
1 //Do odczytu
2 //
3 //Driver silnika DC
4 "HMI".driverDCstart := "SPRZET".pozycjoner01.start;
5 "HMI".driverDCdir := "SPRZET".pozycjoner01.dir;
6 "HMI".driverDCpoz := "SPRZET".pozycjoner01.pozycja;
7 "HMI".driverDCpoz_0lmm := "SPRZET".pozycjoner01.pozycja_0lmm;
8
9 //"zadajnik" położenia
10 "HMI".zadajnikWartosc := "SPRZET".panel_01.zadajnik;
11 "HMI".zadajnik_0lmm := "SPRZET".panel_01.zadajnik_0lmm;
12
```



OB1

Main [OB1]

Network 1: Mapowanie wejść i ustalenie prędkości testowej

Comment

```
1 "FC_I_MAPA" ();  
2 //prędkość testowa "SPRZET".pozycjoner01.predkosc := 3200;
```

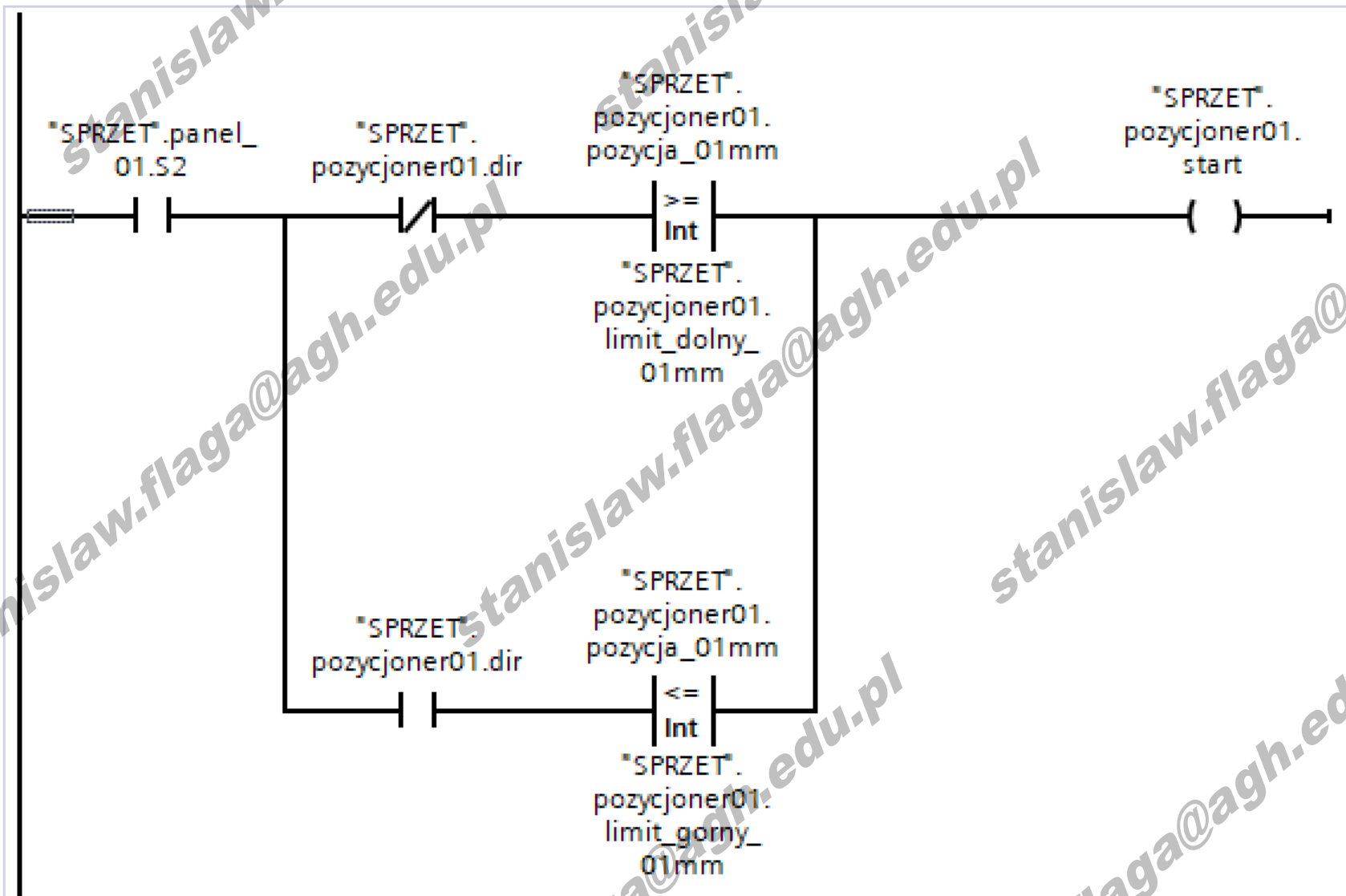
Network 2: Wybór kierunku ruchu

Comment

```
1 //SET - RESET na "SPRZET".pozycjoner01.dir  
2 "SPRZET".pozycjoner01.dir := ("SPRZET".panel_01.S1 OR "SPRZET".pozycjoner01.dir)  
3 AND NOT "SPRZET".panel_01.S3;
```

Network 3: Zabezpieczenie przed przekroczeniem dozwolonych pozycji

Comment





OB1

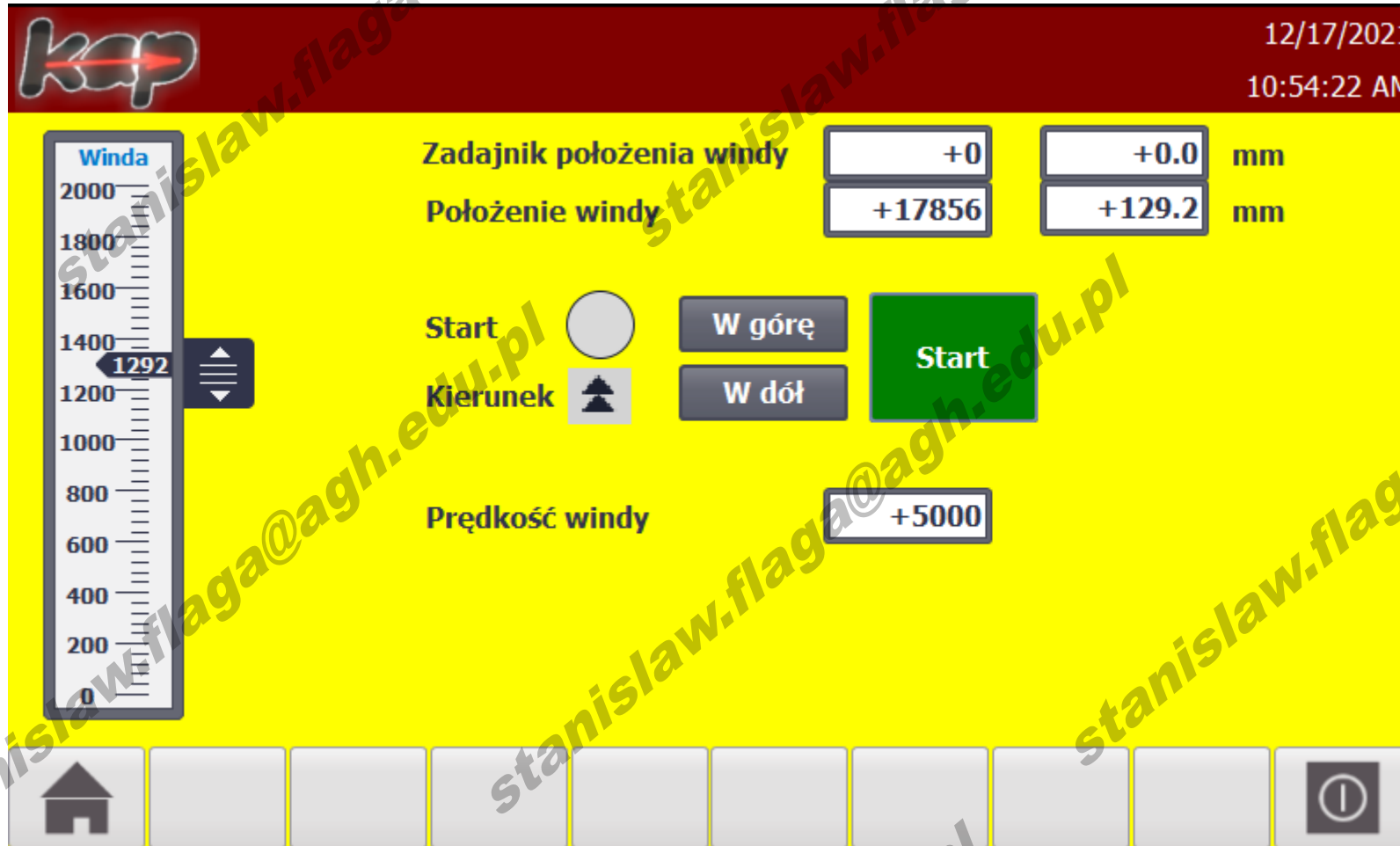
Main [OB1]

Network 4: Mapowanie Wyjść fizycznych i zmiennych dla HMI

Comment

```
1 "FC_Q_MAPA" ();  
2 "FC_HMI_MAPA" ();
```

Poruszanie prosty projekt – PANEL HMI



Poruszanie prosty projekt – PANEL HMI

Properties	Animations	Events	Texts
Click			
Press		SetBitWhileKeyPressed	
Release		Tag (Input/output)	HMI_HMI_S2
Activate		Bit	0
Deactivate		<Add function>	

Zadajnik położenia windy mm

Położenie windy mm

Start

Kierunek

W górę

W dół

Prędkość windy

Properties	Animations	Events	Texts
Click			
Press		SetBitWhileKeyPressed	
Release		Tag (Input/output)	HMI_HMI_S1
Activate		Bit	0
Deactivate		<Add function>	

Properties	Animations	Events	Texts
Click			
Press		SetBitWhileKeyPressed	
Release		Tag (Input/output)	HMI_HMI_S3
Activate		Bit	0
Deactivate		<Add function>	

Poruszanie prosty projekt – PANEL HMI

Properties Animations Events Texts

Property list

General

Appearance

Characteristics

Layout

Text format

General

Process

Tag: HMI_zadajnikWartosc

PLC tag: HMI.zadajnikWartosc

Property list

General

Appearance

Characteristics

Layout

Text format

General

Process

Tag: HMI_zadajnik_01mm

PLC tag: HMI.zadajnik_01mm

Zadajnik położenia windy

Położenie windy

+0	+0.0 mm
+17856	+129.2 mm

General

Appearance

Characteristics

Layout

Text format

General

Process

Tag: HMI_driveDCpoz

PLC tag: HMI.driverDCpoz

Properties Animations Events Texts

Property list

General

Appearance

Characteristics

Layout

Text format

General

Process

Tag: HMI_driverDCpoz_01mm

PLC tag: HMI.driverDCpoz_01mm

stanislaw.flaga@agh.edu.pl

Poruszanie prosty projekt – PANEL HMI

General

Process

Tag: SPRZET_pozycjoner01_predkosc

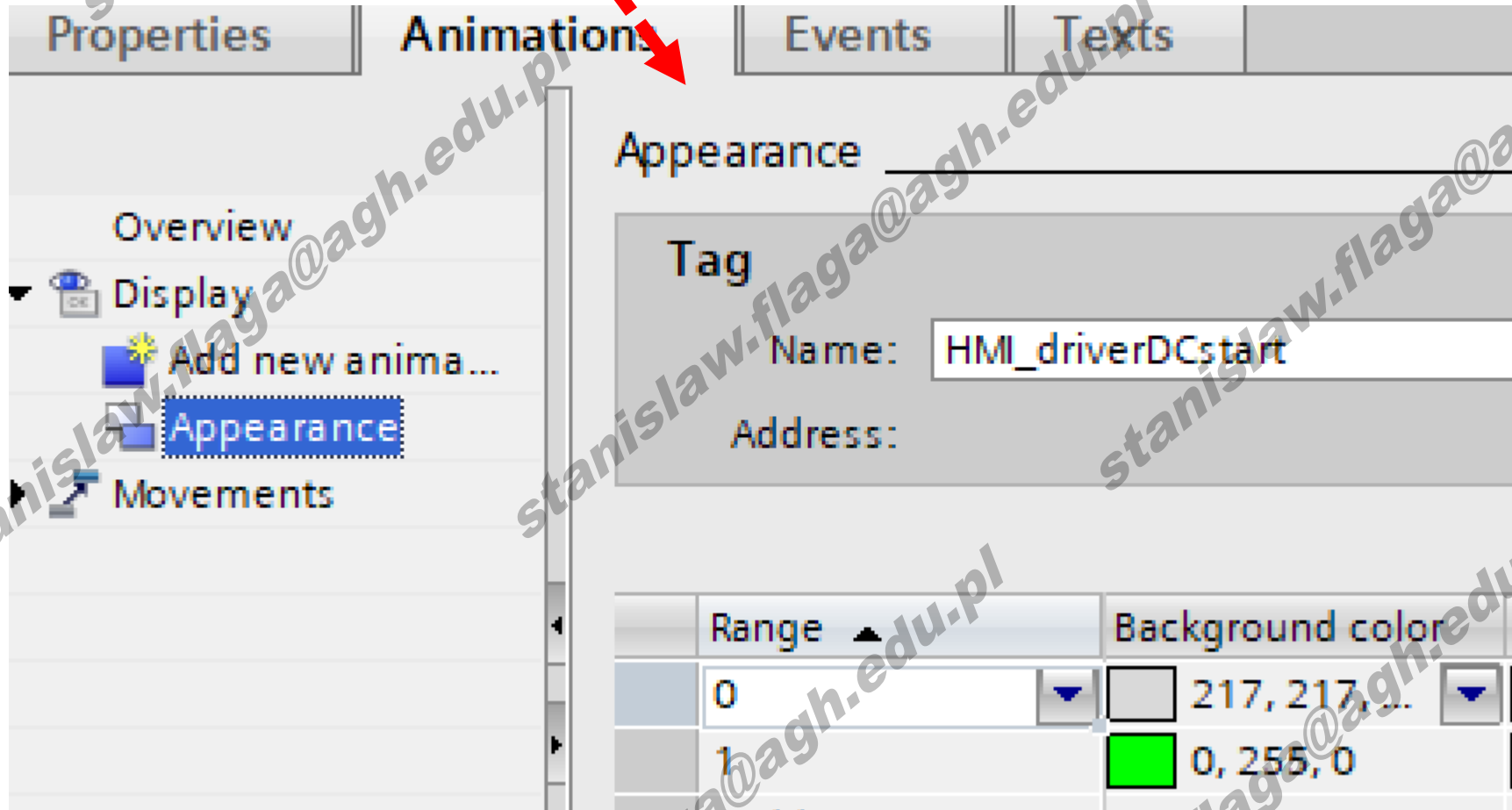
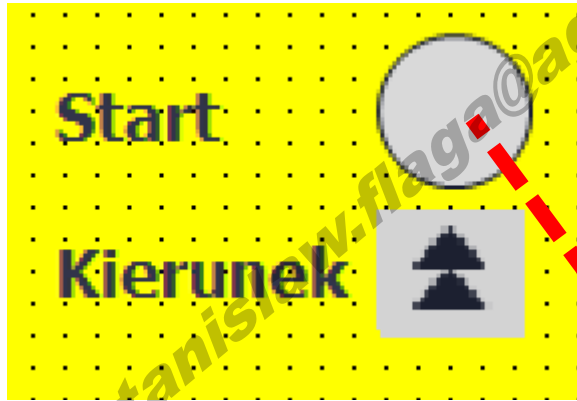
PLC tag: SPRZET.pozycjoner01.predkosc

Address:

Prędkość windy

+00000

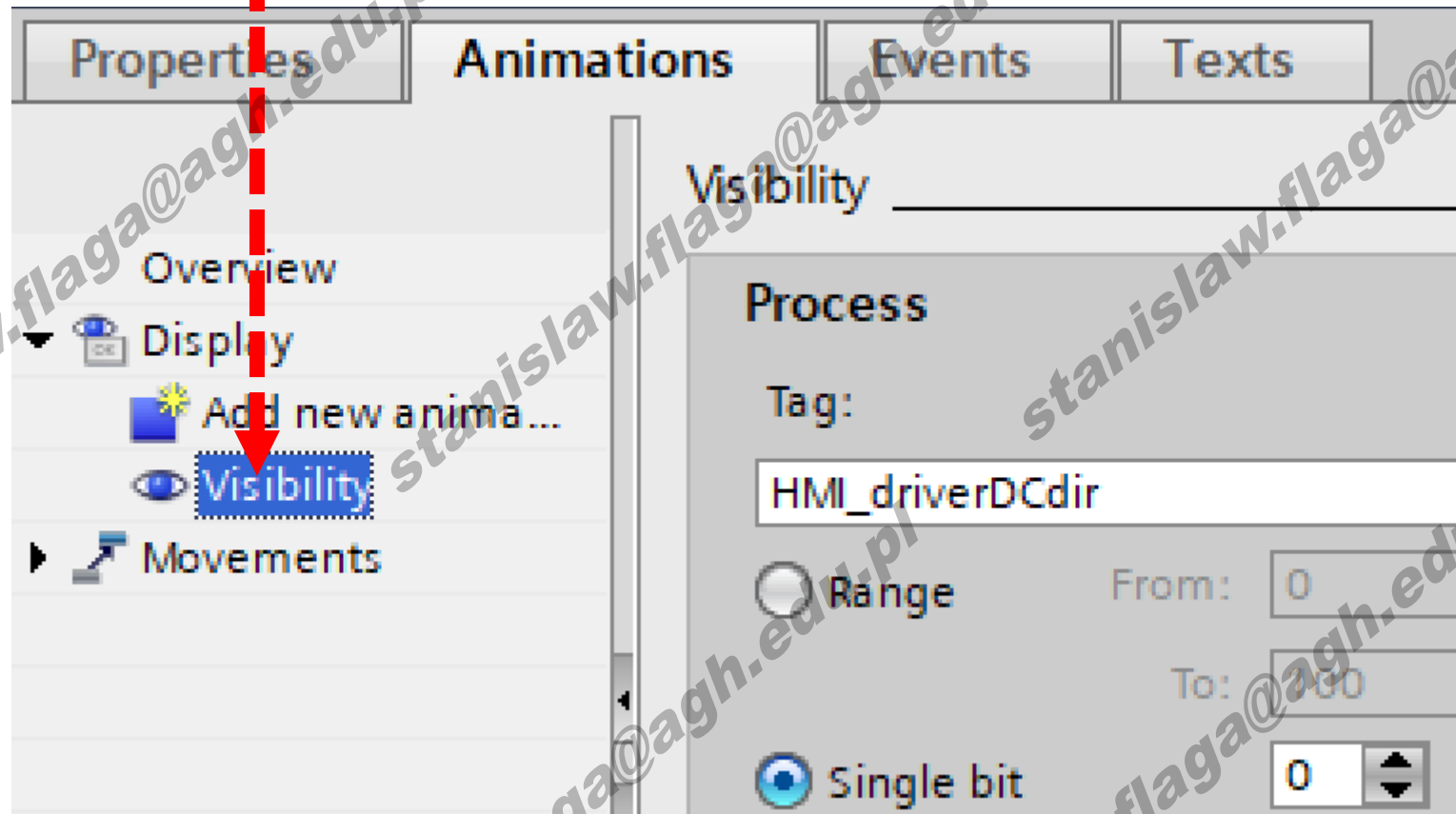
Poruszanie prosty projekt – PANEL HMI



The screenshot shows the configuration interface of the HMI software. The "Animations" tab is selected, and the "Appearance" section is active. The "Tag" field is set to "HMI_driverDCstart". Below this, a table defines the appearance for different states:

Range	Background color
0	217, 217, ...
1	0, 255, 0

Poruszanie prosty projekt – PANEL HMI



The software interface shows the 'Properties' tab selected. The 'Visibility' property is highlighted with a blue dashed border. The 'Process' section is visible, showing a tag 'HMI_driverDCdir' and a 'Single bit' control set to '0'.

Properties	Animations	Events	Texts
Overview			
Display			
Add new anima...			
Visibility			
Movements			

Visibility

Process

Tag:

HMI_driverDCdir

Range From: 0 To: 100

Single bit 0

Zadanie 2.

Temat

Napisz program którego główne funkcjonalności to:

- ręczny tryb pracy (w tej chwili realizuje go powyższy program),
- tryb automatyczny pozycjonowania karetki na wartości zadanej potencjometrem,
- zadawanie pozycji docelowej dla trybu automatycznego potencjometrem – rozdzielczość 0,1 mm,
- implementacja algorytmu regulatora typu P i wywoływania do co 200 ms (użyj przerwania zegarowego),

UWAGA

Na tym stanowisku zmieniasz wartość prędkości w zależności od uchybu wyliczonego jako różnica wartości zadanej i aktualnej.

Niezależnie od wartości uchybu nie możesz dla prędkości przekroczyć nastawy 9000. Musisz wyznaczyć minimalną nastawę prędkości przy której karetką jeszcze się przemieszcza – to zdecydowanie nie jest wartość bliska 0.

Struktura układu sterowania



Wartość zadana (pozycja docelowa) – potencjometr – napięcie na **A10 (IW64)**

iW64

iW66

Uchyb

REGULATOR

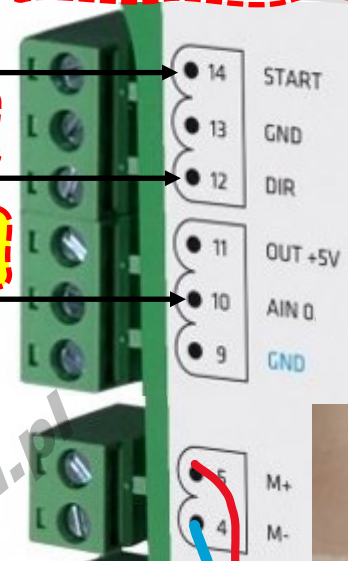
jednowymiarowy, ale:
- znak uchybu wyznacza kierunek (wyjście dwustanowe),
- moduł z uchybu wartość prędkości (wyjście analogowe)

Q1.0

Q1.1

QW80

Element wykonawczy - sterownik



Wartość zmierzona – przetwornik przemieszczenia - napięcie na **A11 (IW66)**

obiekt

Element wykonawczy