

Kewin Gałuszka nr indeksu 241624 Kacper Starościak nr indeksu **grupa 1**

1. Cel ćwiczenia

Zapoznanie się z obsługą systemu InTouch oraz jego funkcjonalnościami.

2. Wstęp teoretyczny

InTouch jest systemem SCADA (z ang. Supervisory Control And Data Acquisition). Oznacza to, iż jest on systemem informatycznym, który nadzoruje przebieg procesu technologicznego lub produkcyjnego. Głównymi funkcjami takiego oprogramowania obejmują zbieranie aktualnych danych, ich wizualizację, sterowanie zamierzonym procesem oraz alarmowanie i archiwizację danych.

Aplikacja stworzona w programie InTouch komunikuje się z programem komunikacyjnym Simulate. Program ma ten ma za zadanie emulowanie proces sterowany przez rzeczywiste urządzeniem, którym może być np. serownik programowalny PLC.

3. Opis czynności

- 1. Utworzyliśmy nową aplikację. Kolejno wywołaliśmy polecenie "Nowe okno" w którym ustawiliśmy nazwę okna oraz jego wymiary, kolor i położenie.
- 2. Utworzyliśmy obiekt graficzny "prostokąt z zaokrąglonymi rogami" poprzez wybranie tego narzędzia z zestawu narzędzi "Rysowanie". Obiekt graficzny przedstawia zbiornik do gromadzenia gotowego mleka czekoladowego.

P	:h	01	co	ola	ate	2 П	nil	k																																														
1			-						-				-					-	-	-						•						-	•	-		-	-	0.00			-							1		2 .	- 2			
1	se.	±	8t	ent.	83	±	iet:	1	3	tR	test	533	±	et:	at s	÷ 1	+ : +		S.C.	+	tes	t:	: :	e Ret	:::::	S.	±	te s	t : :	. :	e Re	est.	83	± 1	et:e	at s	e	e Re	to t	83	± 1	te e	1.57		1.0	11 to	3	± 1	to	• •	. 11	80	+ -	
÷	-	÷		. +	64	÷	÷.+-	+	ä -	÷	÷ · •		÷	÷	+		+ +	+	14	÷	÷	+		-	• +	6	÷	÷	+		6. ÷	. +	÷.	÷	+-	+	4	6.6	+	6	÷	÷	+ +			. +	3	te,	+	+ +	+		+	
1	9	ž	÷		83	÷	÷	•	3	÷	÷. ;		÷	÷	•					÷	÷	•		÷	•	8	÷	÷÷	•	3	÷	•	÷.	÷	1	-	_	_		_			_	_		~	3	+	+		÷	÷	•	4
÷	4	2	1	÷.	÷.	- 22	1	•	4	233			- 22	120		4			4	-22	20	•		2012	22	2	2	20	•	2	202		24	23	1											Y	2	2			- 2	1		4
	3	÷	сt:	-	83	+	iete:	10	3	±12	test	533	+	et:	at s		100	100	83	+	at s	t 53		a ne	-	33	+1	te s	ts:		e Re	:25	83	+1												-	3	±17	to	1.51	+	(tetr		
+	4	÷	1	. +	64	÷	÷÷	+	ä.	+00	+ - +		÷	÷	+		+ +	+	1	÷	+-	+ -		-	• +	84	÷	÷	+	4	6. ÷	. +	÷.	÷													3	÷.	+-	+ +	÷	. + -	+	
1	3	÷	÷			÷	÷	•	а.	÷	÷. ;		÷	÷	•					÷	÷	•		÷		3	÷	÷÷	•		÷		÷.	÷													3	+	+		÷	+	•	4
÷	4	2	- 2	÷.	1	- 22	1	•	2	2.5			- 22	123					24	-22	20	•		2012	22	2	2	-	•	2 3	202	-	2	23													14	2			- 2	1		4
	1	±	et:	:22	83	+	iete:	ate:	3	±12	test	533	+	et:	at s		101	100	83	+	at s	t 53	: :	195	:25	83	+1	te s	ts:		e e e	:05	83	± 1												1	3	±17	to	1.51	+	1000		
÷	4	÷	1	. +	64	÷	÷÷	+	ā.	÷	+ + +		÷	÷	+	4	+ +	+	1	÷	÷	+		-	• +	84	÷	÷	+	-	ê. ê	- +	1	÷													3	÷.	+-	+ +	÷	- ÷ ·	+	4
	3	÷	÷			÷	÷	•	З.	÷	÷. ;		÷	÷	•	4				÷	÷	•		÷		8	÷	÷÷	•		÷		÷.	÷													3	+	+ -		÷	+	•	4
1	4	2		×.	<u>_</u>	- 22	1		4	233		1	- 22	120		4			24	-22	1	• •		2012	22	24	2	10	. 7	2 3	202	22	24	23													14	2			- 2	1		4
	3	±	et:	ent.	83	±	i dete	ate:	е.	±18	test	533	±	et:	at s		1.1		83	+	et e	t 53		a Ret	:::::	83	±	te s	ts:		e de	:25	83	±1												-	3	±	to	1.1	. 21	257		
Ŧ	4	÷	1	. +	64	÷	÷	+	3	÷	÷ • •		÷	÷	+	4	+ +	+	1	÷	÷	+ -		÷.+	• +	84	÷	÷	+	4	es é	- +	1	÷													3	÷.	+-	+ +	÷	- ÷ ·	+	
1	3	÷	Ŧ			÷	Ŧ	•	З.	÷	÷. i		÷	÷	÷.,	4				÷	÷	•		÷		÷.	÷	÷	•		÷		÷.	÷													3	+	+ -		÷	+	•	4
÷	4	2	- 1	22	÷.	- 22	1		4	233			- 22	- 23	47	4			24	-22	20	•		2012	84	24	2.7	20		2 3	202	-	24	23													14	2.5			- 2	1		4
+	33	÷	Rt.	ent.	83	+	inte N	ate:	з.	± R	test	533	+	et:	at s		100	100	83	+	at s	tes:		185	-	33	+1	te s	tes:		e e e	ant.	83	± 1												-	3	± 1	to	1.11	. 1	100		
÷	4	÷	1	. +	6	÷	÷	+	4	+===	+ + +		÷	÷	+	-	+ +	+	1	÷	÷	+ -		-	• +	84	÷	÷	+	-	6. ÷	. +	1	÷													4	÷.	+-	+ •	÷		+	
		ŝ	÷		83	÷	÷		3	÷	÷. i		÷	÷		4				÷	÷	•		÷		3	÷	÷	•		÷		÷.	÷													3	+	+		÷	+	•	4
÷	4	2	1	2014	÷.	- 22	1		4	237			- 22	123		4			<u>a</u>	-22	27	•		2012	22	4	1	20	•	2 3	2012	22	24	20	ų,											Ł	14	2.5			- 2	1		a - 1
	1	+	D.		83	+	int:	1	3	+ 2	test		+	Re:	at s		+ +		33	+	tes	÷		- De	-	33	±	te s	t		e ne	10.1	33	+	+	-	-	11.0	Sec.		-	1200	Theres.	-	-	1.	3	+ -	t: 1	• •	+	100	+ -	
+	-	÷		+	24	÷	. ÷.	+	4	+===	t: +		÷	+	+		+ +	+	24	÷	+-	+ -		-	. +	8	te	+	+		-	. +	1	÷	+	+	4 4	6.1.A	+	1	+	+	+ +			. +	4	t:	+	+ +	+		+	
	4	÷	÷			÷	÷		4	÷	+		÷	÷						÷	+ -			÷		2	+	+	•		+ +		24	÷	+		4 4	10	÷. ;	24	÷	+ -		i i	+		3	+	+		÷	+		
•	1.	4	1		1	- 22	-		14	2.5		1	- 22	-		4			2.	2	14			1.1.2	1	G.	2		. 7	2 3	192	1	a.	2.5			4 4	233	100	2.	2.5	1					14	2			-			4
			-			+	ion:	-		+10				ore	-		41794										+		****					+ 11		-		-			+11				in the			+17				-		÷.,

 Za pomocą narzędzia "Wielokąt" narysowaliśmy trójkąt, element zaworu. Kolejno skopiowaliśmy go, odwróciliśmy za pomocą narzędzia "Odbicie lustrzane w poziomie". Do rysunku dodaliśmy trzpień zaworu poprzez narzędzie "Prostokąt". Za pomocą narzędzia "Przesuń pod spód" przesunęliśmy trzpień pod warstwę trójkątów. Jako pokrętło użyta została elipsa narysowana za pomocą narzędzia o tej samej nazwie.



4. Zaznaczyliśmy wszystkie stworzone obiekty trzymając klawisz "Shift". Wybraliśmy narzędzie "Utwórz symbol" z zestawu "Położenie" przez co scaliliśmy pojedyncze elementy graficzne w jeden obiekt. Dodaliśmy linię łączącą zawór ze zbiornikiem, zwiększyliśmy jej grubość poprzez zaznaczenie jej i dokonaniu wyboru w menu "Linia". Następnie powtórzyliśmy wcześniejsze kroki, aby dodać ją na stałe do obiektu – zaworu. Za pomocą myszki przesunęliśmy zawór do zbiornika.



 Następnie skopiowaliśmy obiekt zaworu z rurą, powieliliśmy go. Położenie obiektów skorygowaliśmy za pomocą narzędzia "Wyrównaj do lewej" z zestawu narzędzi "Położenie". Oba zawory zostały przesunięte pod spód celem osiągnięcia oczekiwanej estetyki.



6. Celem zwiększenia czytelności, powiększyliśmy zbiornik i ponownie ustawiliśmy położenie zaworów w analogiczny do wcześniejszego sposób.



7. Utworzyliśmy pola tekstowe za pomocą narzędzia "Tekst". Wybraliśmy czcionkę "Comic Sans" ze względu na jej nienaganną estetykę. Utworzone opisy zaworów oraz zbiornika zostały przesunięte za pomocą myszy w miejsca docelowe.



 Utworzyliśmy przycisk który ma na celu uruchomienie procesu. Za pomocą narzędzia "Przycisk" dodaliśmy obiekt. Po zaznaczeniu przycisku i wciśnięciu ctrl+L podmieniliśmy napis na przycisku na "Start"



 Rozpoczęliśmy proces przygotowania animacji obiektów graficznych oraz definiowana zmiennych. Wybraliśmy na początek obiekt zaworu mleka. Nacisnęliśmy klawisz enter celem uruchomienia biblioteki. Wybraliśmy kolor wypełnienia jak i kolor linii na dyskretny.



 Po naciśnięciu przycisku pojawiło się okno dialogowe. Wpisaliśmy nazwę zmiennej w polu do tego przeznaczonym. Określiliśmy kolor zielony jako prawdę(1) i czerwony jako fałsz (0)

Object type: Symbol	Prev Link Next Lin <u>k</u>	OK Cancel
7	Fill Color -> Discrete Expression	
xpression:		ОК
MILK_VALVE		Cancel
Colors 1,TRUE,On:	0,FALSE,Off:	Clear
	Tagname Undefined X Image: Define "MILK_VALVE"?	
	OK Anuluj	

11. Po wciśnięciu OK pojawia się kolejne okno dialogowe, w którym zmieniliśmy typ zmiennej na I/O Discrete.

NEAR EFAR EFAR A	
Tagname Dictionary	×
C Main 💿 Details C Alarms C Details & Alarms C Members	
New Restore Delete Save < Select >> Cancel Close	
Tagname: MILK_VALVE Type: Memory Discrete	
Group: \$System C Read only C Read Write	
Comment:	
Log Data 🗖 Log Events 🗖 Retentive Value	
Initial Value O On Off On Msg: Off Msg:	
Tag Types Memory Discrete Indirect Discrete Memory Integer I/O Discrete Memory Real I/O Real Indirect Analog Memory Message I/O Message Indirect Message Group Var Hist Trend Tag ID	×
Clear All Details Select All Clear All	

12. W polu "element" wpisaliśmy "V1" co w programie simulate oznacza otwieranie i zamykanie zaworu mleka.

13. Jako, że chcieliśmy, aby również kolor rury – linii się zmieniał, ustawiliśmy kolor linii na dyskretny.



14. Posłużyliśmy się w tym przypadku wcześniej utworzonymi ustawieniami, które zostały zastosowane do zaworu. Dwa razy kliknęliśmy pole "Wyrażenie". Pojawiła się aktualna lista zmiennych. Wybraliśmy "MILK_VALVE".

- Incar]
Tagname	Tag Type	Access Name	Alarm Group	-
 \$OperatorDomain \$OperatorDomainE \$OperatorEntered \$PasswordEntered \$PasswordEntered \$Second \$StartDdeConvers \$System \$Time \$Time \$UerifiedUserName 	System Message System Message System Message System Message System Integer System Discrete System Alarm System Integer System Message System Message			
MILK_VALVE	System Integer I/O Discrete	PLC1	\$System	
(٢
ot Field: <none></none>	•		OK	
Filter: (none)	T		Canci	el

15. Ustawiliśmy kolory zabarwienia linii.

Object type: Symbol	<u>P</u> rev Link	Next Lin <u>k</u>	Cancel
L	ine Color -> Discrete Ex	pression	
xpression:			ОК
xpression: MILK_VALVE			OK Cance

16. Analogicznie, jak w przypadku zaworu mleka ożywiliśmy grafikę zaworu syropu.

Object type: Symbol	Pre	v Link Next Lin <u>k</u>	Cancel
Touch Links	Line Color	Fill Color	Text Color
User Inputs	Discrete	Discrete	C Discrete
Discrete	Analog	C Analog	F Analog
Analog	Discrete Alarm	Discrete Alarm	Discrete Alarm
String	Analog Alarm	Analog Alarm	Analog Alarm
Sliders	Object Size	Location	Percent Fill
Vertical	E Height	C Vertical	C Vertical
Horizontal	□ Width	Horizontal	Horizontal
Touch Pushbuttons	Miscellaneous	Value Display	1
Discrete Value	Uisibility	Discrete	
Action	E Blink		
Show Window	C Orientation	C String	
Hide Window	Disable		
	T Tooltip		

Object type: Symbol	Prev Link Next Link	OK Cancel
	Fill Color -> Discrete Expression	
Expression: SYRUP_VALVE		OK Cancel
Colors	0,FALSE,Off:	Clear
	Define "SYRUP_VALVE"? OK Anuluj	

Tag Types				x
Memory Discrete Indirect Discrete Memory Integer I/O Integer Memory Real I/O Real Indirect Analog Memory Message I/O Message Group Var Hist Trend Tag ID				
ОК	Cancel	Details	Select All	Clear All

Tagname Dictionary	×
C Main 💿 Details C Alarms C Details & Ala	arms C Members
New Restore Delete Save <<	Select >>> Cancel Close
Tagname: SYRUP_VALVE	Type: I/O Discrete
Group: \$System	C Read only 💿 Read Write
Comment:	
🗖 Log Data 🗖 Log Events 🗖	Retentive Value
C On C Off C Direct C Reverse	On Msg: Off Msg:
Access Name: PLC1	
Item: V2	🗌 Use Tagname as Item Name

ly source. <local></local>		<u> </u>	8=8- 8-8-	J
fagname	Tag Type	Access Name	Alarm Group	
 \$OperatorDomainE \$OperatorEntered \$PasswordEntered \$PasswordEntered \$Second \$StartDdeConvers \$System \$Time \$TimeString \$VerifiedUserName 	System Message System Message System Message System Integer System Discrete System Alarm System Integer System Message System Message			
MILK VALVE	I/O Discrete	PLC1	\$System	
SYRUP_VALVE	I/O Discrete	PLC1	\$System	
				Ľ
ot Field: <none></none>	•		OK	
Filter: <none></none>	.		Cance	əl

17. Kolejno ożywiliśmy grafikę zbiornika z mlekiem. Zaznaczyliśmy rysunek zbiorniku i nacisnęliśmy enter. Wybraliśmy napełnianie się zbiornika pionowe.

Object tupe: Bound	Rect Pro	utink Novtlink	ОК
object type. Hound			Cancel
Touch Links	Line Color	Fill Color	Text Color
User Inputs	Discrete	Discrete	Discrete
Discrete	C Analog	C Analog	☐ Analog
Analog	Discrete Alarm	Discrete Alarm	Discrete Alarm
String	Analog Alarm	Analog Alarm	Analog Alarm
Sliders	Object Size	Location	Percent Fill
┌── Vertical	E Height	C Vertical	Vertical
Horizontal	□ Width	Horizontal	Horizontal
Touch Pushbuttons	Miscellaneous	Value Display	1
Discrete Value	🗖 Visibility	☐ Discrete	
Action	E Blink		
Show Window	C Orientation	□ String	
Hide Window	Disable		
	Tooltip]

18. Po wybraniu przycisku pojawiło się okno dialogowe, które uzupełniliśmy odpowiednimi wartościami. Ustawiliśmy również, aby poziom cieczy rósł w kierunku góry.

Object type: Round Rect	Prev Link Next Link Cance
V	ertical Fill -> Analog Value
xpression:	ОК
TANK_LEVEL	Canc
Properties Value at Max Fill: 1000 Value at Min Fill: 0	Max % Fill: 100
Direction	Background Color:

19. Ustawiliśmy również kolor tła zbiornika i zdefiniowaliśmy obiekt.

Object type: Round Rect	<u>P</u> rev Link	Next Lin <u>k</u>	OK Cancel
	Vertical Fill -> Analog '	Value	
xpression:			ОК
TANK_LEVEL			Cancel
Properties Tag Value at Max Fill: 101 Value at Min Fill: 0	name Undefined Define "TANK_		- Clear
Direction Owr			

20. Zmieniliśmy typ zmiennej na "I/O Integer"

Tagname D	ictionary					2
C Main	🖲 Details 🥤 Alarm	s 🔘 Details &	Alarms C	Members		
New	Restore Delete	Save	< Select	>> C	ancel Close	
Tagname:	TANK_LEVEL		Туре	Memory R	eal	
Group:	\$System		C F	Read only 📀 f	Read Write	
Comment:			-10-52-00			
Log Dat	a 🔲 Log Events		Retentive	Value 🗖 Rete	ntive Parameters	
Initial Value:	0		Min Value:	-32768	Deadband:	0
Eng Units:			Max Value:	32767	Log Deadband:	0
	Memory Discrete I/O Discrete Memory Integer Memory Integer Memory Real I/O Real I/O Real I/O Real I/O Message I/O Message Group Var Hist Trend Tag ID					
	OK	Cancel	Details	: Selec	Clear All	1

21. Jako element wpisaliśmy "L1", który jest elementem w programie komunikacyjnym, którego wartość odpowiada wartości poziomu mleka czekoladowego w zbiorniku.

Tagname Dictionary			x
C Main 💽 Details C Alarms C Details & A	alarms C Membr	ers	
New Restore Delete Save <<	Select >	Cancel C	lose
Tagname: TANK_LEVEL	Туре:	I/O Integer	
Group: \$System	C Read o	nly 💿 Read Write	9
Comment:			
🗖 Log Data 🗖 Log Events 🛛 🗖	Retentive Value	Retentive Para	meters
Initial Value: 0	Min EU:	-32768	Max EU: 32767
Deadband: 0	Min Raw:	-32768	Max Raw: 32767
Eng Units:	Log Deadband:	0	Conversion • Linear C Square Root
Access Name: PLC1			- -
Item: L1			Use Tagname as Item Name

22. Ożywiliśmy również "#" pod zbiornikiem, aby wyświetlał wartość ilości substancji w zbiorniku.

W tym celu zaznaczyliśmy obiekt "#" i wcisnęliśmy enter, w oknie dialogowym ustawiliśmy parametry tak, aby wyświetlana była analogowa wartość. Jako źródło podaliśmy TANK_LEVEL – czyli ilość substancji w zbiorniku.

23. Ożywiony został także przycisk "Start". Po zaznaczeniu go i wciśnięciu enter ukazało się następujące okno dialogowe

Object type: Button	Prev Link Next Link	OK Cancel
(Output -> Discrete Expression	
Expression:		OK
Start		Cancel
On Message:	Off Message:	Clear

24. Zmieniliśmy ustawienia w oknie bibliotek na przycisk dyskretny.

			Cancel
Touch Links	Line Color	Fill Color	Text Color
User Inputs	C Discrete	C Discrete	
Discrete			
Analog	Discrete Alarm	Discrete Alarm	Discrete Alarm
□ String □	C Analog Alarm	Analog Alarm	Analog Alarm
Sliders	Object Size	Location	Percent Fill
└── Vertical	E Height	Vertical	
Horizontal	□ Width	Horizontal	F Horizontal
Touch Pushbuttons	Miscellaneous	Value Display	
Discrete Value	Uisibility	Discrete	
Action	E Blink	Analog	
Show Window	C Orientation	String	
Hide Window	Disable		
	Tooltip		

25. Ustawiliśmy, aby przycisk zmieniał swój stan za każdym naciśnięciem na przeciwny. Ustawiliśmy typ na "IO Discrete". Nazwę dostępu ustawiliśmy na PLC1. Wpisaliśmy wartości wysyłane włączonego i wyłączonego przycisku. Użyliśmy również nazwy zmiennej jako nazwy elementu. Zapisaliśmy projekt.

Object type: Button	Prev Link Next Link	OK Cancel
(Jutput -> Discrete Expression	
xpression: Start		OK
Start		Cancel
On Message: Running	Off Message: Stopped	Clear



26. Po wykonaniu powyższych poleceń Window Maker wyglądał następująco:

27. Rozpoczęliśmy konfigurowanie programu Simulate, wiedząc, ze wszystkie nazwy dostępowe ustawiliśmy jako "PLC1". Zmienne uzupełniliśmy odpowiednimi danymi.

28. Zapisaliśmy plik konfiguracyjny.

SIMULATE	Topic Definition		-OX
Configure Help		Done New Modify Delete	

 Wróciliśmy do InTouch`a. Uruchomiliśmy aplikację za pomocą przycisku "Uruchom". Efekty jej działania są przedstawione na poniższych grafikach. Na początku oba zawory są zamknięte.

chocolate milk	
Stopped Milk Valve	



następuje otwarcie zaworu mleka



Kolejno zamyka się zawór mleka i otwiera zawór syropu

Zbiornik dopełnia się do końca, zawór syropu się zamyka, następuje opróżnienie zbiornika.

chocolate milk			
Running	Milk Valve		
	Syrup Valve		
		200	

4. Wnioski

Przy pomocy systemu InTouch można w prosty i intuicyjny sposób zwizualizować różnorodne procesy przemysłowe i produkcyjne.

System InTouch daje możliwość wizualizacji i symulacji procesu. Dzięki temu można przed przystąpieniem do realizacji fizycznej instalacji, zasymulować ją w oprogramowaniu i sprawdzić poprawność działania.