# Grafické operátorské panely Weintek řady iP



# Grafické operátorské panely Weintek řady iP

Copyright © TECON spol. s r. o., Vrchlabí, Česká republika

Tato publikace prošla jen částečnou jazykovou korekturou. Tato publikace vznikla na základě informací dostupných v době tisku. Veškeré změny jsou vyhrazeny. Všechny ochranné známky uvedené v této knize jsou majetkem jejich vlastníků

# Žádná část této publikace nesmí být publikována a šířena žádným způsobem a v žádné podobě bez výslovného svolení firmy TECON s.r.o.

# TECON spol. s r. o.

Komenského 63 Vrchlabí 543 01 Česká republika

Tel.: 499 429 100, 499 429 117 Fax: 499 422 508 e-mail: info@TECON.cz http://www.TECON.cz

# Obsah

Technické parametry	4
Instalace programu EasyBuilder Pro	5
Vytvoření nového projektu	6
Bitový přepínač s lampičkou	9
Zadávání a zobrazení numerických dat	14
Inkrementace a dekrementace	17
Alarmy (Eventy)	20
Logování dat	26
Data transfer	35
Více oken na panelu	37
Receptury	41
Makra	48
Security	52
Ostatní aktivní prvky	56

# Technické parametry

Nové grafické **operátorské panely firmy WEINTEK řady iP** s dotykovou obrazovkou jsou ekonomicky velice výhodným řešením pro vizualizaci a monitoring technologických procesů řízených programovatelnými automaty PLC.

Dotykové panely WEINTEK řady iP jsou aktuálně nabízeny v provedení s úhlopříčkou 4,3 palce, 7 palců a 10 palců TFT v širokoúhlém provedení.

Weintek panely řady iP, jsou hardwarově podobné jako předchozí panely řady iP. Rozdíl zde je v mnoha ohledech. Nové panely řady iP umožňují oproti předchozím panelům rychlejším Boot Time a to až o 10 vteřin. Dále je zde použit nový typ displeje, který disponuje až 16.2M barev oproti dosavadním 65536. Je zde také použita lepší a rychlejší komprese obrázků a textů. Panely řady iP disponují procesorem Cortex A8 32Bit 600 MHz, pamětí RAM 128MB, pamětí FLASH 256MB a rozsahem provozních teplot 0°C až 50°C.

Sofistikovaný konfigurační software EasyBuilder PRO je k dispozici zcela ZDARMA.

	MT8052iP	MT8072iP	MT8106iP			
DISPLEJ	4,3" TFT	7" TFT	10.1" TFT			
JAS	400	300	350			
BARVY	16.7M	16.7M	16.7M			
ROZLIŠENÍ	480x272	800x480	1024x600			
DOTYKOVÝ PANEL		Rezistivní 4 vodičový				
SÉRIOVÉ PORTY	COM1: RS-232/RS-485 2W/4W, COM3: RS-485 2W	COM1: RS-232 COM3: RS-485 2W/4W	COM1: RS-232 4W COM2: RS-485 2W/4W, COM3: RS-232 2W/RS- 485 2W/4W			
ETHERNET		10/100 Base-T x 1				
PROCESOR	Dual-	core RISC Cortex A8 32Bit				
PAMĚŤ		256 MB				
RAM		128 MB				
NAPÁJENÍ		24±20% VDC				
PROVOZNÍ TEPLOTA	0° ~ 50°C					
SOFTWARE	Easy I Weinc	Builder PRO verze 6 cloud Easy Access 2.0 (volite	lné)			

Srovnání všech řad panelů Weintek z pohledu funkcí a možností naleznete zde.

## Instalace programu EasyBuilder Pro

**Spusťte Windows,** jestliže jste tak ještě neučinili, spusťte operační systém **Microsoft Windows**. Jestliže jste doposud nepracovali se systémem Windows, poraďte se s manuálem Windows jak nakonfigurovat systém na optimální výkon a jak spouštět aplikace. Konfigurační software **EasyBuilder Pro** je určen pro 64 bitové operační systémy **Windows od verze 10 výše**.

**Rozbalte archiv** Instalační archiv **EBproV60902.ZIP** (nebo novější) rozbalte na disk a spusťte soubor **SETUP.EXE** 

Nejprve se naučíme jak spustit vývojové prostředí EB Pro a zkusíme **napsat** jednoduchou aplikaci, kterou **otestujeme** pomocí **offline simulace** na obrazovce PC. Po té projekt uložíme do panelu a vyzkoušíme jeho funkčnost.

Spusťte **Utility Manager**, který naleznete na ploše. Zobrazí se následující okno. V tomto okně nalezneme vývojové prostředí včetně další užitečných funkcí.



# Vytvoření nového projektu

Kliknutím na **EasyBuilder Pro** spustíme vývojové prostředí. Zobrazí se okno s možností založení nového projektu, otevření projektu z adresáře nebo výběr z posledních použitých projektů.

Zvolíme nový projekt. Následně se zobrazí okno s výběrem typu panelu, pro který chceme projekt vytvářet. Pro účely příručky bude stačit panel MT8072iP (800x480)



Po výběru typu panelu se otevře jako první nastavení systémových parametrů.

tem Parame	eter setting								
Cellular D	ata Network		Printer/Backu	ID Server	Time S	vnc./DST		e-Mail	
Device	Model	Gene	ral Syste	em Rem	ote Se	curity	Extend	ded Memo	ory
evice list:								Nhat's my	/ TP2
evice liber	Namo	Location	Deuice Tures			Interfe		Protocol	
Local HM	T Local HMT	Local	MT9071D / MT	007100 / MT00	1700 / 900 v 49	0) -	ice i/i r	1010001	0
٢									>
Settings ma roject descri	ide in this tab	New	v Device/Server.	 cancel)	Delete	S	Settings/Se	ecurity	
Settings ma roject descri CADA softw. nd enable [N	ide in this tal iption: are can acce	New o will be sav	v Device/Server. ved directly (no o data via MODBUS	 cancel) S TCP/IP Server	Delete on HMI. (Enab	le MODBU:	Settings/So S TCP/IP S	ecurity	st
Settings ma roject descri CADA softw nd enable [N	ide in this tab iption: are can acce IODBUS TCP	New o will be sav	y Device/Server. ved directly (no o data via MODBUS by])	 cancel) S TCP/IP Server Address Mo	Delete on HMI. (Enab	le MODBUS	Settings/Si	ecurity	st
Settings ma roject descri	ide in this tab iption: are can acce MODBUS TCP	New o will be sav	ata via MODBUS	 cancel) S TCP/IP Server Address Mc	Delete	ie MODBUS	Settings/S	ecurity	st

V tomto okně (**System Parametr Settings**) lze nastavit komunikaci s připojeným zařízením a základní vlastnosti projektu. Tato různá nastavení si ukážeme dále. Klineme na OK. Dostáváme se tak na prázdnou plochu, která symbolizuje obrazovku panelu.

💽 EasyBuilder Pro	V6.10.01.243 : E	BProject1 - [1	0 - WINDOW_	/_010 ]			- 🗆 ×
File 🔡 🖂	🔨 🧈 👳 Hon	ne Project	Object	Data/History IloT/Energy View Tool Weind	loud		- 8 × ^
Paste Cut Clipboard	System Parameters	Select	Find/Replace Multi. copy Window copy Editing	e Addr @ : # 🛗 📛 🔶 🔟 📇 🖶 🖬 ↔ IIII IIII / ~~	일 후 의 판 라 네 왕 한 아 응 않? 당 해 왕 2011년 월 카 월 - 군 - 구 - 위 Arrange	$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	80 S1 S2 S3 0 ÷ L1 L2 L3 L4 1 ÷ State/Language
Windows Tree		<b>▼</b> ×	4 10 - W	WINDOW_010 ×		\$	Picture Library 💌 🗙
Enter Legword to fill + 4 : Common - 5 : Device R - 6 : HMI Con - 7 : P assword - 8 : Storage S - 9 : Backup - 10 : Wkup - 11 : - 11 - 12 - 13 - 14 - 15 - 16 - 16 - 17 - 18 - 10 - 20 - 21 - 22 - 22 - 24 - 25 - 26 - 27 - 28 - 30 - 31 - 33 - 33 - 33 - 33 - 33 - 38 - 37 - 38 - 37 - 38 - 37 - 38 - 38 38 - 38 - 38	er windows ction window sponse nection I Restriction pace Insufficien W_010	t	and the second of the second o				[Project]
- 39 - 40 - 41							Remove Picture
42		¥	<			>	Shape Library Picture Library
MT8071iP / MT8071i	P2 / MT8072iP (8	800 x 480)			X =	351 Y = 184 🛛 CAP NUM S	SCRL 100 %

V názvu okna vlevo nahoře vidíte vedle jména konfiguračního software také jméno projektu, se kterým právě pracujete.

Pod ním se nachází **Menu**, ze kterého je přes jednotlivé položky možné dosáhnout na všechny funkce software.

V levém podokně se nachází seznam vytvořených oken. V našem případě WINDOW\_010. Toto okno je tedy zatím prázdná plocha, dále si ukážeme, jak vytvořit první jednoduchý projekt.

## Bitový přepínač s lampičkou

Pro vytvoření první aplikace vložíme na černou plochu nějaký prvek. Jako první projekt vytvoříme bitový přepínač, který nám bude rozsvěcet a zhasínat bitovou lampičku.

Vložíme bitový přepínač. Pomocí hlavního menu v záložce Objects -> Set Bit

EasyBuilder Pro : EBProject3 - [10 - WINDOW_010 ]								
File 🔚 🔜 🐟 🌧 🗸 Home	e Project	Object	Data/History	lloT/Energy	View Tool			
✓       ✓       ✓       ✓       Picture         🕒       🗠       ✓       ✓       ✓       ✓         ✓       🗠       ✓       ✓       ✓       ✓       ✓         ✓       🗠       ✓       ✓       ✓       ✓       ✓       ✓         ✓       ✓       ✓       ✓       ✓       ✓       ✓       ✓       ✓       ✓         ✓ <td>Bit Wa Lamp La</td> <td>ord Set mp Bit</td> <td>t Set En Word</td> <td>Toggle Switch Multi-State Switc Function Key Button</td> <td>Combo Button h - Slider</td>	Bit Wa Lamp La	ord Set mp Bit	t Set En Word	Toggle Switch Multi-State Switc Function Key Button	Combo Button h - Slider			
Windows Tree	• • •	د م	10 - WINDOW_	010 ×				
-       3: Fast Selection         -       4: Common Window         -       5: Device Response         -       6: HMI Connection         -       7: Password Restriction	^		۹,,,,,,,,					

V následujícím okně poté přiřadíme požadované vlastnosti tomuto prvku. V první záložce **General** nastavíme, na jakou adresu se bude bitová hodnota zapisovat (**Write address**).

Zde jsou možnosti LB, RW\_Bit, RW\_A\_Bit, LW\_bit, RBI. Použijeme LB 0. Dále nastavíme, jakým způsobem se bude tlačítko chovat (Attribute). Lze vybrat mnoho způsobů chování (Set ON, Set OFF, Toggle, momentary atd...) Jelikož mi chceme vytvořit přepínač, zvolíme Toggle.

New Set Bit Object
General Security Shape Label
Comment :
Write address
PLC : Local HMI    Settings
Address : LB 🗸 0
Write after button is released
Attribute Set style : Toggle
Macro Execute macro
OK Storno Nápověda

V záložce **Shape** můžeme nastavit vzhled daného tlačítka. Pro tuto možnost jsou zde před vytvořeny knihovny s tvary. Klikneme tedy na **Shape Library**.

	State : 0 -
	Picture :
0 1 Picture	
	Picture Library Use picture Set to original dimensions
Shape	Shape Library
	☑ Inner
	✓ Frame
	Interior pattern : Pattern Style
	Duplicate these attributes to every state

Zde si můžeme zvolit z různých tvarů. Jsou zde vytvořeny nejen tlačítka ale i lampy (signálky) a další užitečné prvky. Zvolíme tedy takovýto tvar pro náš přepínač. (v knihovně **button2**). Tyto knihovny lze rozšiřovat o vlastní knihovny.

oiect	Library						-	694		
-,		System	Switch - Cr	ystal	•	0 0	3			
Systen	n Switch - Cr	ystal							-	
0	- 0	Switch - Crystal States : 2 Objects : 1			Switch - Crystal States : 2 Objects : 0		<u>,</u>			
2		Switch - Crystal States : 2 Objects : 0			Switch - Crystal States : 2 Objects : 0	-	0	ixport	Modify SYSTEM 62x87 13642 bytes	
4		Switch - Crystal		5	Switch - Crystal States : 2		1		SYSTEM 62x87 13612 bytes	[
6		Objects : 0		7	Objects : 0					
		pwitch - Crystal		Bad	ground :		1	New	Delete	Clean

Na ploše obrazovky se nám objevil první prvek - bitový přepínač. Tento přepínač nám bude měnit hodnotu na adrese LB-0 vždy z 1 do 0 a z 0do 1. Tuto funkci můžeme nechat zobrazit pomocí bitové lampičky, kterou si také vložíme do projektu.



Bitovou lampičku můžeme vybrat v menu Object-> Bit Lamp.

EasyBuilder Pro : EBProject3 - [10 - WINDOW_010 ]								
File 🛛 🚼 💌 🐟 🥕 📮 Home	e Project C	bject Data/Hist	tory lloT/Energy					
✓        ✓        ✓        ✓ <td>Bit Word Lamp</td> <td>Set Set Bit Word</td> <td>✤ Toggle Switch ☑ Multi-State Switch ☑ Function Key</td>	Bit Word Lamp	Set Set Bit Word	✤ Toggle Switch ☑ Multi-State Switch ☑ Function Key					
Draw	Lamp		Button					
Windows Tree	<b>▼</b> X	4 10 - WINDO	)W_010 X					
-      3 : Fast Selection     -      4 : Common Window     5 : Device Response     6 : HMI Connection	^		1.1.1.1.1.1.1.1.1					

Abychom docílili spárování s předchozím bitovým přepínačem, musíme nastavit Read address na stejnou adresu jako Write address u bitového přepínače, tedy na LB-0. Dále můžeme opět v záložce Shape pomocí Shape Library přiřadit lampičce určitý tvar.

New Bit La	amp/Tog	gle Switch Object			<b>E</b>
General	Security	Shape Label			
	Comment	:			
		O Bit Lamp	() T	oggle Switch	
⊂Read a	ddress				
1	PLC : Loc	al HMI		•	Settinas
Addr	ess : LB		• 0		
		Invert signal			
Blinking	,				
	Mode	None			•
E H	ide picture	/shape if no corresp	onding picture fo	r current state	
		OK	Storno	<b>)</b>	Nápověda

EasyBuilder Pro : EBProject3 - [10 - WINDOW_010 ]		- 🗆 ×
File 🗄 🖂 🐟 🌧 🗸 Home Project Object Da	a/History IIoT/Energy View Tool	- 8 × *
✓     ✓✓     ✓     ✓     ✓	23 🔮 Toggle Switch 🖻 Combo Button Set Data Switch - O Slider Mutri-State Switch - O Slider Mutri State Switch - O Slide	Action-related
Windows Tree 👻 🗸 10-	VINDOW_010 ×	Picture Library 👻 🗙
3: Fat Selection         A           4: Common Window         A           5: Device Response         A           6: HM Connection         B           7: Password Restriction         B           9: Backup         B           10: WINDOW_010         B           11:         B           12:         B           13:         B           14:         B           15:         B           16:         B           17:         B           18:         B           19:         B           20:         C           21:         C           22:         C           23:         C           24:         C           25:         C           20:         C           21:         C           22:         C           23:         C           30:         S           31:         S           32:         S           33:         S           34:         S		System Switch - Crystal  System Switch - Crystal System Switch - Crystal
Windows Tree Address	>	Shape Lib Picture Li Sound Lib
MT8071iE/MT8101iE (800 x 480)	X = 250 Y = 503 CAP NUM SC	CRL 100 % - + H

Výsledný projekt potom může vypadat následovně:

Nyní je čas na uložení a kompilaci projektu. Pro uložení projektu můžete použít, stejně jako u Windows, ikonku diskety.

Kompilovat projekt můžete v menu sekci Project, zde naleznete i spuštění Online či Offline simulace.

EasyBuilder Pro : EBProject3 - [10 - WINDOW_010 ]								
File	8 🖂 🔸 🦂	⇒ Home	Project	Object	Data/History	lloT/Energy V		
P	R	Ŕ		×		SD [		
System Message	Language & Font	Compile	Online Simulation	Offline Simulation	Download (PC->HMI)	Build Download R Files		
Set	ting			В	uild			
Windows Tr	Windows Tree 🔹 🗙 4 10 - WINDOW_010 🗙							
3: 4: 5:	Fast Selection Common Wii Device Respo	ndow nse			۹			

Zobrazí se okno simulující obrazovku panelu. Po kliknutí na přepínač se lampička rozsvítí.



Aplikace ke stažení zde

# Zadávání a zobrazení numerických dat

V minulém projektu jsme si ukázali, jak pracovat s bitovou hodnotou. Jak ji měnit a jak ji nechat zobrazovat pomocí bitové lampičky. Nyní si ukážeme, jak pracovat s numerickými hodnotami. Jak je zapisovat a jak je číst a zobrazovat.

Založíme si nový projekt (File -> New). Pro zobrazení numerických dat použijeme připravený prvek, který naleznete pomocí menu Objects -> Numeric.



Objeví se opět dialogové okno pro zadání parametrů. V záložce General zvolíme zobrazované místo v paměti. Lze zvolit LW, RW, RW\_A, RWI. Pro možnost i zápisu do dané paměti je nutné zaškrtnout Allow input. U tohoto objektu je také možnost čtení a zápisu do různých části paměti.

Vew Numeric Object
General Data Entry Format Security Shape Font
Description :
I Allow input
Read Address PLC : Local HMI
Address : LW
Notification
Notification on invalid input
OK Storno Nápověda

Dále pak v záložce Format můžete nastavit požadovaný formát vstupních dat. Na výběr jsou kombinace 16 a 32 bitové BCD,HEX Unsigned, Signed nebo Float. Dále lze nastavit počet čísel před a za desetinou čárkou. Dále lze nastavit limity pro danou hodnotu včetně alarmového ukazatele. Vše potvrdíme a na obrazovce nám vznikne nový objekt:

eneral Data Entry Format	Security   Shape   Font				
Data format : 16-bit Uns	signed 👻 🕅 Mask				
Left of decimal Pt. : 4	Right of decima	il Pt. : 0			
Display format			51		
Enable					
Use star "*" to represent [Truncated digits]	t each digits. Number of "*" = [Lef	t of decimal Pt.] -			
Use star "*" to represent [Truncated digits] Scaling	t each digits. Number of *** = [Lef	t of decimal Pt.] -			
Use star *** to represent [Truncated digits] Scaling Method : Nor	t each digits. Number of *** = [Lef	t of decimal Pt.] -		M	IE_0 (LW-200)
Use star *** to represent [Truncated digits] Scaling Method : Nor	t each digits. Number of *** = [Lef	t of decimal Pt.] -		M	IE_0 (LW-200) #######
Use star *** to represent [[runcated digits] Scaling Method : Nor mits	t each digits. Number of *** = [Lef	t of decimal Pt.] -			IE_0 (LW-200) ######
Use star *** to represent [Truncated digits] Scaling Method : Nor mits Orrect O Dy	ie each digits. Number of *** = [Lef	t of decimal Pt.] -			IE_0 (LW-200) #######
Use star *** to represent [Truncated digits] Scaling Method : Nor mits	t each digits. Number of *** = [Lef	t of decimal Pt.] -			IE_0 (LW-200) ######
Use star *** to represent [Truncated digits] Scaling mits @ Direct @ Dy PLC low : 0 Input low : 0	reach digits. Number of *** = [Lef	9999 9999			IE_0 (LW-200) #######
Use star *** to represent [Truncated digits] Scaling Method : Nor mits © Direct © Dy PLC low : 0 Input low : 0 □ Use ala	t each digits. Number of *** = [Lef	9999 9999			IE_0 (LW-200) ######

Pro nastavení hodnoty numerické proměnné zvolíme prvek Set Word. Použijime objekt Set Word, který naleznete přes menu Objects -> Set word.



V nově otevřeném dialogovém okně postupujeme analogicky jako u ostatních prvků. V záložce General nastavíme adresu, kam chceme zapsat danou hodnotu. Dále nastavíme, jak chceme hodnotu zapsat. Zda chceme zapsat konstantu, inkrementovat či dekrementovat stávající hodnotu atd.

Danému tlačítku můžeme opět přiřadit libovolný tvar ze Shape library. Pokud bychom chtěli mít tlačítko s popisem, umožní to záložka Label, kde můžeme danému tlačítku přiřadit text s libovolnou barvou, fontem či zarovnáním.

New Set Word Object	×
General Security Shape Label	
Comment :	
Write address	
PLC : Local HMI	ettings
Address : LW V 0 16-b	pit Unsigned
Write after button is released	
Notification Enable	
Attribute	
Set Style : Write constant value	•
Set value : 0	
OK Storno	Nápověda

Aplikace ke stažení zde

#### Inkrementace a dekrementace

Dále si ukážeme, jak můžeme **danou hodnotu ovlivnit tlačítkem**. Vytvoříme si projekt, kde budeme pomocí dvou tlačítek upravovat hodnotu v LW-200 tak, že první tlačítko bude tuto hodnotu **zvyšovat o 5** a druhé ji bude **snižovat o 5**.

Použijeme opět objekt Set Word, naleznete ho buď jako ikonu nebo pomocí horního menu Objects -> Button -> **Set word**. V nově otevřeném dialogovém okně postupujeme analogicky jako u ostatních prvků. V záložce General nastavíme adresu, kam chceme zapsat danou hodnotu. Dále nastavíme, jak chceme hodnotu zapsat. V našem případě tedy zvolíme **Increment value (JOG +)**. Poté se nám pod tímto výběrem objeví dvě textová pole. První Inc. Value: určuje jaká hodnota se bude přičítat k hodnotě v LW-200. Druhá, Upper limit: nastaví maximální hodnotu do, které bude tlačítko inkrementovat. Dosáhne-li hodnota v LW-200 této hodnoty, tlačítko nebude dále inkrementovat.

Set Word Object's Properties
General Security Shape Label Profile
Comment:
Write address
PLC : Local HMI     Settings
Address : LW 👻 200 16-bit Unsigned
Write after button is released
Notification
Attribute
Set Style : Increment value (JOG+)
Inc. value : 5 Upper limit : 100
Dynamic limits
OK Storno Nápověda

Poté co jsme nastavili funkčnost tlačítka, můžeme nastavit jeho tvar nebo popis. Tvar necháme zachovalí a nastavíme si popisek daného tlačítka v záložce Label. Nejprve nahoře zaškrtneme Use Label, poté se nám zobrazí veškeré možnosti textu. Lze tu nastavit font, barva a velikost písma, dále jeho zarovnání nebo efekt v podobě běžícího textu. Ve spodní části se nachází textové pole pro samotný popis tlačítka, v našem případě zadáme Zvyš o 5.

eneral	Security	Shape	Label	Profile				
Use	e <mark>label</mark> e label libra	iry				L	abel Library	
Co	nvert label	s to bitm	ap image	es (Use bitr	nap font)			
Attribu	State	: 0	•	••	0 1			
	Font	: Arial						-
	Color	:				Size :	16	•
	Align	: Cente	r	•		Blink :	None	•
	_	Itali	c		Underline			
		Dupli	- ate the	se attribut	es to			
		E	ery st	ate				
Moven	nent Direction	: No mo	vement	•				
Conten	t		<b>V</b>	Preview wi	ith actual fo	nt size		
Zvyš	05							
Trad	cing				Dup	licate this lab	el to every	state

Vše potvrdíme a tlačítko pro inkrementaci máme vytvořeno. Bude vypadat následovně:



Nyní vytvoříme totožné tlačítko pouze s tím rozdílem, že nepoužijeme vlastnost Increment value (JOG +), ale **Decrement value(JOG -)**. U této vlastnosti nastavíme hodnotu, o kterou se bude hodnota v LW-200 snižovat opět na 5 a oproti vlastnosti Icrement nevolíme maximální hodnotu, ale minimální. Nastavíme ji tedy na 0. V záložce Label nastavíme popisek Sniž o 5. Výsledné tlačítko bude vypadat následovně:



Teď si vytvoříme Numeric objekt pro zobrazení dané hodnoty. Postupujeme stejně jako v předešlé kapitole. Adresu, ze které budeme číst, nastavíme na LW-200. Objekt vytvoříme a umístíme na obrazovku. Výsledný projekt zkompilujeme a spustíme Off-line simulaci. Výsledný projekt může vypadat takto.



Aplikace ke stažení zde

# Alarmy (Eventy)

V dalším příkladu si ukážeme, jak vytvořit výpis hlášení informující obsluhu o důležitých událostech nebo chybách technologie – Alarmů.

V konfiguračním software EB Pro se tato funkce jmenuje **Event( Alarm) Log**. Zde si nadefinujeme, jakou událost chceme sledovat. Tuto nastalou událost poté můžeme zobrazovat pomocí tří různých typů objektů a to: **Alarm Bar**, **Alarm Display** a **Event Display**.

Vytvoříme si tedy projekt ve, kterém budeme chybové události "simulovat" pomocí dvou binárních a jedné numerické proměnné.

Definujeme si tyto chyby:

**Porucha PR1** – sepnut bit LB-10 **Porucha PR2** – sepnut bit LB-11

**Max. mez L1** – kritická mez nádrže, numerická hodnota v LW-200, kterou budeme ovládat pomocí již vytvořených inkrementačních tlačítek, přesáhne číslo 20, bude ve výpisu vyhlášen alarm s touto chybou.

Do předešlého projektu si vytvoříme funkci Event (Alarm) Log. Zde si nadefinujeme proměnné, které chceme sledovat a při jakých podmínkách se mají spustit chyby. Tuto funkci naleznete v menu **Data/History -> Event (Alarm) log**.

EasyBuilder Pro : EBProject3 - [10 - WINDOW_010 ]								
File 🗄 🖾 🐟 🌧 🗸 Home	Project Object	Data/History Ilo						
Data Sampling	Event (Alarm) Log	Alarm Bar 📃 Eve Alarm Display Event Display						
Windows Tree		10 - WINDOW_010						

V nově otevřeném dialogovém okně klikneme na tlačítko New a vytvoříme tak novou událost (alarm).

Event (Alarm) Lo	g							
Category : 🛕	I [0]	~						×
No. Category	Text Mode	Condition	Read address	Notificatio	n address	Buzzer	e-Mail	Save to histor
<								>
🗹 Enable ba	ck light when a	larm occurs						
History files								
Save to H	MI memory	Save to	USB disk 1	Save to	o USB disk 2			
New	Insert	D	elete Set	ttings		Export.		Import
Сору	Paste		Paste (Add Mod	e)				Exit

Nejprve nastavíme bitovou poruchu PR1. V záložce General nastavíme jako **Address type Bit**, nastavíme **Read address** na LB-10, jako **Condition** (podmínku) nastavíme ON (lze zvolit mezi ON,OFF,ON->OFF, OFF->ON). Tímto jsme nastavili novou událost, která se uloží v případě kdy se bit LB-10 sepne do stavu ON. Tato událost bude zobrazována, dokud nedojde k uvolnění LB-10 do stavu OFF. Nyní si nastavíme znění zprávy.

Event (Alarm) Log
General Message Occurrence
Category : 0 Priority level : Low
Address type :      Bit      Word      Delay time for event monitoring when HMI resets : 1 second(s)
Read address
PLC : Local HMI   Settings
Address : LB 🔹 10
Notification
Condition Trigger : ON
OK Storno Nánověda

V záložce **Message** si nadefinujeme znění samotné chybové hlášky. V textovém poli **Content:** napíšeme znění chyby. V mém případě Chyba PR1. Dále lze nastavit font a barva písma dané chyby. Potvrdíme dané nastavení a zopakujeme ho pro poruchu PR2. Nastavíme **Read address** na LB-11 a textu zprávy napíšeme Chyba PR2.

Content : Cr	nyba PR 1	A 7
Use label library		Label Library
Use string table		String Table
Font : Ar	ial	•
Color :		
Acknowledge val	ue: 11	
Enable	Sound Library	
Enable	Play	

Pro alarm, který se bude spouštět na základě hodnoty v LW-200 musíme zvolit **Address** type Word. V **Read address** nastavíme LW-200. Dále v jako podmínku zvolíme > 20 podle zadání. Poté v záložce **Message** nastavíme text zprávy na Zvýšená hladina.

Event (Alarm) Log
General Message Occurrence
Category : 0 Priority level : Low
Address type :      Bit     Word     Delay time for event monitoring when HMI resets : 1second(s)
Read address PLC : Local HMI  ▼ Settings
Address : LW 👻 200 16-bit Unsigned
Notification Enable
Condition Trigger if value is : > 20 Dynamic condition value
OK Storno Nápověda

Nyní musíme na obrazovku vložit veškeré prvky pro tento projekt. Nejprve vložíme dva bitové přepínače, které nám budou simulovat poruchy PR1 a PR2. Tyto objekty vložte stejně jako v kapitole výše. Nastavme si u nich, ale Label s popiskem PR1 a PR2. Výsledná tlačítka mohou vypadat následovně.



Teď máme na obrazovce veškeré aktivní prvky pro vyvolání vytvořených událostí. Nyní si vytvoříme objekty, které nám dané události (alarmy) zobrazí.

Jako první zvolíme **Alarm Display**. Naleznete ho v horním menu Data/History -> Alarm Display.



V nově otevřeném dialogovém okně nás bude zajímat především záložka Alarm. Zde si nahoře můžeme vybrat, které kategorie alarmů chceme zobrazovat. Tyto kategorie se nastavují u vytváření samotných událostí. Lze tak každé události přiřadit jiná kategorie.

eneral Al	larm Security Shap	e Font Profile	1	
Include	e categories ; 🛐	thru 255 (see	Event (Al	arm) Log object}
color —	Trans	sparent		
	Frame :	-	Backg	round :
Grid —				
	🗹 Enab	le		Color :
format -				
Sort				
() Tin	ne ascending	Time descending		
Order & C	Characters			
	Display items	Display chars		Display order
1	Event trigger date	0		Event message
V	Event trigger time	0		Occurrence count
V	Event message	20		Event trigger time
1	Occurrence count	0		
			<b>IM</b>	
_				
If "Displa	y chars" is 0, it means	that the system wi	l display a	l of characters.
Date :	MM/DD/YY	Time : HH:M	M:SS	•
	(			
		1000		

Dále lze nastavit vzhled tohoto objektu. Objekt je zobrazen jako tabulka, lze tedy vybrat barvu pozadí a mřížky a v záložce **Font** i font písma. Další možností je způsob řazení, **Time ascending** nebo **Time descending**, tedy pokud chcete řadit vzestupně či sestupně. Další možností je samotné zobrazení alarmu. Zde můžete zaškrtnout, co vše chcete v tabulce zobrazovat. Je zde na výběr datum, čas, zpráva alarmu nebo počet kolikrát už k alarmu došlo. V pravém sloupci je potom možnost uspořádání sloupců tabulky. Můžete si tedy určit, jak bude daná tabulka vypadat. Vše potvrdíme a na obrazovce se nám vytvoří tabulka. Nyní projekt kompilujeme a spustíme **Off-line** simulaci. Spuštěný projekt tak bude vypadat asi následovně:

			-	×	
Chyba PR1 Zvýšení hladiny	12:23:38 12:23:42	10/20/20 10/20/20			
PR1 PR2	Zvyš o 5 Sniž o 5	27			

Po kliknutí na PR1 nebo PR2 se v tabulce objeví nový řádek s danou chybou, datem, časem a počtem, kolikrát už k chybě došlo. Tato chyba zde bude zobrazena dokud, nedojde k jejímu odstranění. Tedy znovu nestiskneme PR1 nebo PR2. To samé platí o události na hodnotu v LW-200. Dokud bude v LW-200 hodnota vyšší než 20 chyba nezmizí.

Nyní si přidáme do projektu **Alarm Bar**. Naleznete ho pod ikonkou **m** nebo opět v horním menu Data/History -> Alarm Bar.



Tento objekt zobrazuje aktuální alarmy v podobě běžícího textu. V horní části lze nastavit, které kategorie alarmů se zde budou zobrazovat. Dále lze nastavit rychlost běhu textu. Dále lze jako u objektu **Alarm display** nastavit řazení vzniklých událostí podle času vzestupně a sestupně.

Opět můžeme vybrat, které informace chceme zobrazovat. Ve spodní části pak můžeme nastavit formát data a času. Vše potvrdíme, zkompilujeme a spustíme **Off-line** simulaci.

w Alarm Ba	ir Object				le l
larm Secu	rity Shape	Font			
Include So	categories : [ roll speed : [	thru 255 Speed 5	{see	Event (Al	larm) Log object}
Color —			<i>.</i>		
		Transparent			
	Frame :	•		Backg	ground :
Format Sort Tim	e ascending	Time desc	ending		
Order & C	naracters				
<ul><li>✓</li></ul>	Display ite Event trigg Event trigg Event mes	ms er date er time sage			Display order Event trigger time Event message
Date :	MM/DD/YY	▼ Time	: [HH:M	IM:SS	•
			Ctor	-	Négouže

Aplikace bude vypadat následovně:

			_	×
	12:	26:00 Chyba PR1		
Chyba PR1	12:26:00	10/20/20		
PR1	Zvyš o 5			
PR2	Sniž o 5			

Aplikace ke stažení zde

# Logování dat

V dalším příkladu si ukážeme, jak vytvořit tzv. **logování dat**. Tedy ukládání načtených dat do paměti panelu nebo na externí zařízení (USB disk).

Založíme si nový projekt. Abychom mohli logovat data, potřebujeme vytvořit tzv. Data **Sampling Object**. Nalezneme ho v menu Data/History -> Data Sampling.



V nově zobrazeném dialogovém okně zvolíme možnost New....



Data Sampling Object	
Comment :	PLC : Local HMI 🔹
Sampling mode	Clear real-time data address
High priority (this may reduce refresh rate of screen components.)	Enable
Time-based     Trigger-based     Sampling time interval : 1 second(c)	
	Hold address
	Enable
Read address	
PLC : Local HMI    Settings	History files
Address : LW	Save to HMI memory
$\ast$ In prior to display or store the data log, you can use the conversion tag to check and modify the data log.	Save to USB disk
* When the Data record is converted by the user-defined conversion tag, the GetCnvTagArraryIndex() function of [Read conversion] subroutine can get the relative array index.	
Data Record	
Max. data records (real-time mode) : 1000	
Data Format Data length: 0 word(s)	
	OK Cancel

Nejprve si popíšeme nově otevřené okno. V levé části okna vybíráme, kdy a jaká data se budou logovat. V pravé části můžeme nastavit kontrolní adresy, které mohou ovládat logování dat, a také kam a jak se budou data ukládat. Začneme tedy s levou části, kde specifikujeme, jaká data chceme ukládat.

V levé horní části vybereme způsob, kterým data budeme načítat. Jsou zde dvě možnosti **Time-based** a **Trigger-based**, tedy na základě času nebo změny libovolné proměnné. Možnost **Time-based** bude každý nastavený interval načítat a ukládat vybraná data. Tento interval lze nastavit v rozsahu 0.1 sekundy – 120 minut.

Možnost **Trigger-based** nám umožní zvolit proměnou, která bude řídit čtení dat. Tato proměnná může být v panelu nebo na straně připojeného zařízení. Pro tuto proměnnou lze zvolit mód změny: OFF->ON, ON->OFF, ON<->OFF.

	○ Time-based
Mode :	OFF->ON   Set OFF after triggered
PLC :	Local HMI    Settings
Address :	LB • 0
Read address	s
PLC :	Local HMI   Settings
Address :	LW

Nyní se dostáváme k samotným datům, které chceme logovat. U možnosti **Read** address zvolíme počáteční adresu, odkud budeme data načítat. Ve spodní části **Data record**, pak zvolíme kolik dat v tzv. **Real-time modu** chceme ukládat. Jedná se o počet dat, které nebudou ukládány do paměti panelu, ale pouze do mezi paměti a po překročení tohoto limitu se budou nejstarší data mazat a nahrazovat novými tak, aby byl zachovám maximální počet dat. Nyní musíme zvolit, jaká data se budou načítat. Klikneme na **Data Format**.

Data Format		
Comment : 1	16-bit Unsigned	
Read Address		
Data type : 1	16-bit Unsigned 🔹 👻	
1	.6-bit BCD 32-bit BCD	1
1	6-bit Unsigned	
Save to default	.6-bit Signed 32-bit Unsigned 32-bit Signed 32-bit Float	OK Cancel
ime mor	String	

V nově zobrazeném dialogovém okně budeme přidávat formáty daných proměnných, které se budou načítat. Vždy klikneme na **New** a vybereme, jakého datového formátu dané hodnoty budou. Každou hodnotu si můžeme okomentovat a přiřadit jí tak název.

Poté co jsme hodnoty přidali, může jejich výpis vypadat třeba následovně:

Data Format	
1. "Hodnota1"       16-bit Unsigned         2. "Hodnota2"       16-bit Unsigned         3. "32-bit Unsigned"       32-bit Unsigned         4. "32-bit Float"       32-bit Float	▲ ▼
New         Delete         Settings           * When the [Read address] uses an user-defined conversion tag, it is suggested t each item of the Data Record selects the same data length even in different dat types. (i.e. 16-bit BCD, 16-bit Signed, 16-bit Unsigned)	hat a
Exit	

Zavřeme toto okno a ujistíme se, zda se v sekci **Data record** objevila velikost dat.

Data Record	
Max. data records (real-time mode) : 1000	Auto. stop
Data Format Data length : 6 word(s)	

V mém případě je **Data length** 6 slov. Budu tedy načítat přesně 6 16-bitových slov od počáteční adresy, kterou jsem nastavil výše. Maximální počet dat je nastavený na 1000. Data se tedy budou ukládat do mezipaměti a po dosažení maximálního počtu (1000) se nejstarší data budou nahrazovat nejnovějšími.

Nyní se dostáváme k pravé části dialogového okna objektu **Data Sampling**. V pravé části můžeme nastavit bity pro jednoduché ovládání dat (mazaní real-time dat, pozdržení ukládání), dále můžeme vybrat možnost ukládání dat. Jak jsem zmínil výše, zatím jsme nastavili logování dat do mezi paměti, tak že se data po překročení maximálního počtu, budou přepisovat novějšími. Zde tedy můžeme zvolit ukládání do paměti panelu nebo na USB disk. Nelze obojí.

ata Sampling Object	
Comment :	PLC : Local HMI
Sampling mode	Clear real-time data address
High priority (this may reduce refresh rate of screen components.)	Enable
Sampling time interval : 1 second(s)	
	Hold address
	Enable
Read address	
Address : Luw - 0	History files
	Save to HMI memory
In prior to display or store the data log, you can use the conversion tag to check and modify the data log.	Save to USB disk
* When the Data record is converted by the user-defined conversion tag, the	
GetCnvTagArraryIndex() function of [Read conversion] subroutine can get the relative array index.	
Data Record	
Max. data records (real-time mode): 1000 🔲 Auto. stop	
Data Format Data length : 6 word(s)	
	OK Cancel

Začneme první částí v pravé horní polovině. Zde je možnost PLC:. Zde si vybíráme kdo bude ovládat daný data sampling (mazaní real-time dat, pozdržení logování). Na výběr je vždy Local HMI, tedy vnitřní paměť panelu, dále bude k dispozici jakékoli PLC, které je připojené. Pod touto možností je volba mazaní real.time dat.

F	PLC : Local HMI	•
Clear real-time	e data address	
	🔽 Enable	Mode : OFF->ON 🔻
PLC :	Local HMI	Settings
Address :	LB	▼ 200

Zaškrtneme-li **Enable** zobrazí se možnost vybrat ovládací bit z paměti panelu. Následně i mód při, kterém se data smažou (ON -> OFF, OFF -> ON). Pokud tedy dojde ke změně, v mém případě bitu LB 200, z OFF -> ON, smažou se veškerá nalogovaná real-time data.

-Hold address	🔽 Enable	Mode	: ON	•
PLC :	Local HMI			✓ Settings
Address :	LB	▼ 201		

Další možností je **Hold address**. Pokud zaškrtneme možnost **Enable**, zobrazí se možnost vybrat ovládací bit z paměti panelu. Následně i mód při, kterém se data smažou (ON, OFF). Pokud nastavíme LB 201 na ON data se přestanou logovat, tedy ukládat do paměti i mezi paměti. Pokud v LB 201 bude OFF, data se budou standardně logovat.

His	tory files			
	Save to HMI memor	гу		
	Save to USB disk			
	Each file consists of Customized file han	f all records of a day Idling		
	Folder name :	Test		
	File name example :	20160712.dtl		
	Preservation limit		7	day(s)

Poslední možností je samotné ukládání dat. Jak bylo napsáno výše je zde výběr mezi **HMI memory** (paměť panelu) a USB diskem. Nelze vybrat obojí. Další možností je, zda se bude soubor s daty ukládat každý den nebo podle uživatelského nastavení. Volba **Preservation limit** umožňuje nastavit maximální počet souborů, které se budou ukládat. Pokud dojde k jeho překročení, opět se budou starší soubory nahrazovat novějšími, tak aby byl zachován tento limit.

Vše potvrdíme a první **Data Sampling** je nyní vytvořený. Bude tedy načítat hodnoty od LW-0, přesně 6 slov. Nastaven je na sbírání dat každou vteřinu. Maximální počet real-time dat je 1000. Tuto hodnotu real.time data nepřekročí. Hodnoty se budou ukládat i do paměti panelu, zde je omezení ve formě počtu souborů, které se budou vytvářet podle nastavení. V našem případě jednou za den s limitem 7 souborů (dnů). Historické data budou ukládány 7 dnů zpětně. K těmto datům lze přistupovat pomocí objektů **Trend Display, History Data Display**. Dále pak pomocí ftp nebo lze tyto data vyexportovat pomocí objektu **Backup** na externí zařízení USB disk.

Nyní si ukážeme možnost zobrazení dat pomocí grafu (objekt Trend Display). Abychom mohli použít tento objekt, musíme mít vytvořený **Data Sampling**.

Trend Display nalezneme v menu Data/History -> Trend Display.



V nově otevřeném dialogovém okně budeme nastavovat jaká data a jakým způsobem chceme zobrazovat.

Začneme záložkou **General**. Nejprve vybereme Data Sampling Object index, tedy sampling, který jsme si vytvořili. Dále zvolíme typ trendu. Zda chceme zobrazovat **Real-time** data či **History**. Jako další možností je určit jakého typu bude X-osa. Lze vybrat možnost **Pixel** nebo **Time**. Pixelová osa nám udává počet dat, které se budou zobrazovat, oproti tomu časová nám zobrazí data za určitý časový úsek. U obou typů lze zaškrtnou **Dynamic X axis range**. Lze tedy libovolně měnit rozsah zobrazovaných dat. Dále lze u typu **Real-time** zvolit **Hold control**. Jedná se o bit, který když bude sepnutý tak zastaví zobrazování dat. Po jeho uvolnění se zobrazí data, která byla pozdržena a zobrazování probíhá stejně jako před tím. Možnost **Watch Line** umožní po kliknutí na libovolné místo na grafu vyčíst dané hodnoty. Zde se nastavuje počáteční adresa. V našem případě je nutné mít prostor 6 slov.

neral	Trend	Channel	Y Scale	Security	Shape	Profile		
	Commer							
	commen		0-	te Cerreli				
			De	ita sampii	ig Object		1. Test	
					Irend	type :	Real-time	•
Na	te : if n	o. of chann	els is char	nged, you	must rese	et HMI's	data samplin	gs !!
		x	axis time	range: (	) Pixel	<u>о</u> т	ime	
		D	efault dis	tance :	100	se	cond(s)	
		Dun:	amic V avi	s time roo				
,	n c . li	U Dyna		suneran	yc			W
r Adde		ocal HMI					▼ Se	tangs
Auur	ess . [[	vv		U				
old co	ntrol —							
	_	V Enab	ole					
F	PLC :	ocal HMI					▼ Se	ttings
Addr	ess : L	В		▼ 1	.50			
atch l	ine							
		🗸 Enab	ble					
F	PLC :	ocal HMI					▼ Se	ttings
Addr	ess : 🛛	W		▼ 1	.50			
ne sta	amp out	put						
		🗸 Enab	ble					
F	LC :	ocal HMI					▼ Se	ttings
Addr	ess :	W		▼ 1	.80		32-bil	: Unsigned

Tato funkce tedy funguje tak, že pokud chceme získat hodnoty z grafu v určitém místě, klikneme na toto místo a hodnoty z grafu (podle počtu čar-kanálů) se naplní od adresy LW 150. Poslední možností je **Time stamp output**, tato funkce zapíše čas na danou adresu. Tento čas reprezentuje čas vzorku na, který bylo kliknuto. Pokud tedy používáme **Watch Line** a potřebujeme zjistit, v jakém čase byly dané vzorky uloženy, nalezneme tento čas právě v dané adrese. Je nutné počítat s tím, že čas je ukládán v 32-bitovém formátu.

Další záložkou je záložka **Trend.** V této záložce se nastavuje samotný vzhled grafu (trendu). Nejprve můžeme nastavit pozadí a ohraničení grafu, nebo můžeme zvolit možnost transparent. Dále nastavujeme mřížku v grafu. Mřížka v grafu být může, ale nemusí. Pokud ji potřebujeme lze nastavit počet vertikálních i horizontálních čar včetně barvy. Dále můžeme nastavit, pokud jsme zvolili v záložce **General** X-axis range jako Time, formát časové osy, včetně fontu, barvy a velikosti písma. Dále můžeme nastavit zobrazení aktuálního času v horní části grafu. Můžeme volit z více formátů zápisu.

eneral Trend	Channel Security Shape		
Fr	Transparent	Background :	<b> </b> ▼
Grid			
	Tenable	Color : 📕	
X-axis inte	rval : 4 second(s)	Y-axis : 4	division(s)
Time scale	C Enable		
For	mat: HH:MM 🔻	Font : Ar	ial 👻
c	olor:	Size : 8	-) -
📝 Time	HH:MM:SS     HH:MM     SSSSS (Leading zero)	© SSSSS	
🔽 Date		C DD.MM.YY	O YY/MM/DD
Color :			

Nyní se dostáváme k záložce **Channel.** V této záložce budeme nastavovat, jak se budou zobrazovat data z **Data Samplingu**. V horní části máme vypsané všechny proměnné, které jsme si nadefinovali v **Data Samplingu**, pokud je chcete zobrazit, musí být zaškrtnut **Display.** Pro každý **Channel** můžeme zvolit libovolnou barvu, styl a tloušťku čáry. Dále pak její limity. Klikneme na libovolný **Channel** a spodní část se týká právě jeho. Tedy nastavení čáry a limitů. Poslední možnost **Channel visibility Control** je společná pro všechny **kanály**. Po zaškrtnutí **Enable** se zobrazí možnost vybrat libovolné slovo (z panelu či automatu), které bude určovat, které **kanály** budou zobrazeny a které ne. Jednotlivé **kanály** odpovídají jednotlivým bitům tohoto slova v pořadí, jakém jsou uvedeny výše. Zobrazit **kanály** lze buďto při hodnotě ON nebo OFF. Čili pokud bychom chtěli zobrazit pouze **Channel** 2 a 3 zapíšeme do slova hodnotu 6 (0110) resp. 9 (1001) podle módu (ON či OFF)

eneral	Trend	Channel	Y Scale	Security	Shape	Profile	]	
ata sa	ampling o	- bject						
C	hannel	Display	Descriptio	n		Data type		Y scale
▶ 1		V	Hodnota1			16-bit Uns	igned	None
2		1	Hodnota2			16-bit Uns	igned	None
3		1	32-bit Uns	igned	1	32-bit Uns	igned	None
4		<b>V</b>	32-bit Floa	ət	1	32-bit Floa	t	None
Pen (	oroperty Col	or :				Width : [	2	•
	Mir	Dyn. : 1	namic limits	3		Max. :	100	
hanne	el visibility	y control						
		🔲 Ena	able					

Takto vytvořený graf (**Trend Object**) vložíme na obrazovku. Přidáme veškeré kontrolní objekty. Výsledná aplikace může vypadat následovně:



LB-200 jsme nastavili tak, že bude mazat real-time data. LB-201 zastaví logování dat, jakmile se uvolní, data se budou znovu logovat. Ve slovech LW-0 až LW-4 jsou data, které logujeme. V LW-150 až LW-154 budou vypsaná konkrétní data po kliknutí na libovolné místo v grafu. V LW-180 bude aktuální čas těchto dat. LB-150 pozastaví zobrazování dat v grafu, po jeho uvolnění se data začnou opět zobrazovat. Spustíme **Off-line** simulaci. Výsledná aplikace bude vypadat takto:



Aplikace ke stažení zde

TECON s.r.o.

## Data transfer

Jestliže potřebujeme posílat větší množství dat mezi panelem a připojeným zařízením, použijeme k tomu objekt **Data Transfer**. Tento objekt slouží k přenosu dat v nastaveném rozsahu. Tento objekt se spouští dvěma způsoby a to, na základě stisku nebo periodicky podle nastaveného času. Data, která chceme přenést, musí být vždy seřazena v paměti za sebou. Data se zde nastavují podle počáteční adresy a množství dat. Nyní si ukážeme jak takový objekt nastavit.

Data Transfer (Per page) naleznete v menu Objects -> Data Transfer (Per Page).

Object Data/History	lloT/Energy View Tool					
ord Set Set mp Bit Word	Toggle Switch 🛛 Combo Bu Multi-State Switch 🖵 Slider Function Key 🔲 Option List Button	ton 999 ABC Numeric ASCII Input	Chart Animation	PLC Control	Embed Window .	Data Transfer 🗸
	New Data Transfer (Trigger-base         General       Security       Shape       Lab         Comment :       Source address       PLC :       Local HMI         Address :       LW       LW         Destination address       PLC :       Local HMI         Address :       LW         Address :       LW         Address :       LW         Address :       LW         Mode :       S         Mode :       Touch trigg	d) Object	Settings.     Settings.			
		OK Storno	Ná	ápověda		

V nově otevřeném okně nastavujeme zdrojovou a cílovou adresu. Dále pak počet slov, které chceme přenést. Nastavíme zdrojovou (Source) adresu na LW-0 a cílovou (Destination) adresu na LW-100. Počet slov nastavíme na 5. Pod nastavením počtu slov nastavujeme mód. Tento mód může být **Touch trigger** nebo **External trigger.** V případě **Touch trigger** se vytvoří tlačítko, které po stisku přenese data. Pokud zvolíme možnost **External trigger**, můžeme zvolit libovolný bit na straně panelu nebo připojeného zařízení, který po nastavené změně (ON -> OFF, OFF -> ON, ON <-> OFF) také přenese data. Pokud zvolíme Touch trigger nastavíme tvar a popis tlačítka v záložce Shape resp. Label.

Vše potvrdíme a vložíme vytvořené tlačítko na obrazovku. Vytvoříme k na obrazovce 10 Numeric objektů LW0 – LW4 a LW100 – LW105. Uložíme a spustíme Off-line simulaci.

Výsledná aplikace může vypadat následovně:



Data Transfer (Global) naleznete na stejném místě jako Data Transfer (Per page).

r	Data Transfer (Time-based)	
Data Transfer	Canaral	
Time-based	General Comment : Attribute Address type : Word  No. of word : 1 Active only when designated window opened	
	Source address PLC : Local HMI Address : LW  Destination address	
New	PLC : Local HMI   Address : LW   O	Exit
	OK Storno Nápověda	

Zde se nastavuje, jaký typ dat budeme přenášet, zda Bity nebo Wordy. Dále pak časový interval po kterém se data vždy přenesou. Data se přenáší stejně jako u trigger-base. Tedy od zdrojové adresy určený počet slov. Nastavíme tedy zdrojovou a cílovou adresu. Tímto je objekt nastaven a vše potvrdíme. Daný objekt nám bude přenášet data ze zdrojové adresy na cílovou v určitém časovém intervalu.

Aplikace ke stažení zde

### Více oken na panelu

Pokud děláme větší aplikaci a nestačí nám pro zobrazení všech prvků jedna obrazovka, je zapotřebí vytvořit obrazovek více a přepínat mezi nimi. Nyní si ukážeme jak vytvořit nové obrazovky a jak mezi nimi přepínat.

Založíme si nový projekt. Poté si na úvodní obrazovku umístíme nápis "Hlavní Obrazovka". Použijeme objekt **Text,** naleznete ho v menu Object -> Text.



Nyní si vytvoříme novou obrazovku. V levé části vedle hlavní obrazovky se nachází výpis obrazovek. Hlavní obrazovka má název **WINDOW\_010**, klikneme pravým tlačítkem myši na 11 a zvolíme **New.** 

File 🖪 🖂	🔸 🥕 👳 🛛 Hor	me Project	Object Data/	/History	lloT/Energy	View	Tool
Paste X Cut Paste	System Parameters	Select	d/Replace Addr Ilti. copy ndow copy	<ul> <li>♀ :∎</li> <li>Ⅲ :■</li> <li>∴ A</li> </ul>	≝ 23 � ∕ ∾ ふ M □ Ⅲ	<ul> <li>□</li> <li>○</li> <li>○</li> <li>□</li> <li>□</li></ul>	
Clipboard		Edi	ting		O	bject	
Windows Tree		<b>▼</b> ×	4 🖊 10 - WI	NDOW_010	) X		
	ection In Window Response nnection d Restriction Space Insufficier	nt		100	9 <mark></mark>		
- 12 Ne	w		<u>8</u>				
	ben						
	ose						
- 16 De	lete						
- 16 Se	ttings		8-				
- 19 - 20 Ad	ld laver		- -				

Name •	Strang 2
Nume .	Suana_2
Window no. :	11
Size	
Width :	800 Height: 480
Frame	
Width :	0  Color :
Background	
Color :	
	Filled
Underlay window	
Bottom :	None
Middle :	None 👻
Top :	None 🔹
Popup window	
Start position	
X: 100	Y: 100 Monopoly

V nově otevřeném dialogovém okně si obrazovku pojmenujeme Strana\_2. Zde lze nastavit i velikost dané obrazovky, její pozadí atd.. Vše potvrdíme OK.

Rozklikneme novou obrazovku Štrana\_2. Zobrazí se stejná černá plocha jako u hlavní stránky. Použijeme znovu objekt Text a napíšeme na tuto obrazovku Strana 2. Nyní máme vytvořenou druhou obrazovku. Klikněte zpět na hlavní obrazovku. Teď si ukážeme, jak mezi těmito obrazovkami listovat. K tomuto účelu slouží objekt **Function Key** nalezneteho v menu Objects -> Function Key.

ł	lome	Proje	ct	Obj	ect	1	Data/Hi	story	lloT/Energy	View
:t a	ure pe		=			머	123	<b>4</b>	Toggle Switch Multi-State Switc	H → SI
b	le	Lamp	La	mp	Bi	t	Word	En	Function Key	0 🗐
		Lar	mp						Button	

Otevře se dialogové okno.

ew Function	Key Object		2
General Sec	urity Shape Label		
Comm	ent:		
	Activate after button is	released	
Change	full-screen window	Change common window	
O Display	oopup window	0	
Window	no. : 11. Strana_2		
🔘 Return t	o previous window	Close window	
ASCII/Unicod	de mode		
[Enter]	[Backspace]	[Clear]	
[Delete]	🔘 [Left]	[Right]	
[ASCII]	/ [Unicode]		
Execute	macro		
O Window	title bar		
Hard copy so	reen to USB disk, SD card or prir	iter	
) Screen H	hardcopy		
) Import u	iser data/Use [USB Security Key]		
Notification	Enable		
	ОК	Storno	ověda

Nás bude zajímat především možnost **Change full-screen window**. Poté stačí vybrat obrazovku, na kterou chceme přejít. Zvolíme tedy 11. Strana\_2. V záložce **Label** si nadefinujeme text "Strana 2". Umístíme tento objekt na hlavní obrazovku.

Nyní provedeme tu samou operaci na obrazovce Strana\_2. Zvolíme **Function Key** a jako obrazovku, na kterou chceme přejít, zvolíme WINDOW\_010. Opět použijeme záložku **Label** a nadepíšeme si ji Hlavní obr. .

<sup>⊧ĸ_₀</sup> Strana 2
-----------------------------

Výsledný projekt bude vypadat následovně:



Projekt zkompilujeme a pustíme v **Off-line** simulaci. Obrazovky se budou měnit po stisku daného tlačítka. Tímto způsobem je možno vytvořit několik obrazovek a listovat mezi nimi.

Aplikace ke stažení zde

### Receptury

**Receptury** slouží k přiřazování opakovatelných hodnot. Představme si to na příkladu. Vezmeme si pivovar, který vaří různé druhy piva. Každý druh piva se skládá z určitých složek, které se liší pouze množstvím. Pokud by tedy řídicí systém řídil tuto výrobu, bylo by pro něho snazší, kdyby měl tyto informace o hodnotách jednotlivých surovin uložené v určité databázi. Mohl by je tak načíst všechny najednou. K tomu lze použít právě receptury, které vyberou **dané hodnoty z databáze** a naplní jimi určité systémové proměnné, které řídí výrobu. Ukážeme si na **příkladu automatu na nápoje**, jak pracovat s recepturami. Založíme si nový projekt. První, co v novém projektu vytvoříme, bude právě databáze receptur. Seznam receptur nalezneme v menu Data/History -> Recipe Database



V nově otevřeném dialogovém okně vybereme záložku Recipes. Zde budeme vytvářet seznam receptur.

Klikneme na ikonku **I**, abychom vytvořili **novou recepturu**. Nazveme ji Napoje. Dále budeme vytvářet jednotlivé složky dané receptury. V našem případě složky nápojů. Klikneme na tlačítko New a vložíme tyto položky Položka, Kava, Caj, Cokolada, Cukr, Voda. Nejedná se o samotné nápoje, ale pouze o složky, z nichž se nápoj bude skládat. U Polozka zvolíme datový typ ASCII s velikostí 20, jelikož to bude text. Ostatní necháme jako 16-bit Unsigned.

ecipes 🔥 🗙	1	Item name	Data type	Size	Display width	Decimal Pt.	Alignment	1
Nanoje	1	Polozka	ASCII	20	21	0	Left	
, hapoje		Kava	16-bit BCD	1	8	0	Left	
		Caj	16-bit BCD	1	8	0	Left	-
		Cokolada	16-bit BCD	1	8	0	Left	
		Cukr	16-bit BCD	1	8	0	Left	-
	•	Voda	16-bit BCD	1	8	0	Left	
Import Export					New	Delete	Settings	

Nyní máme vytvořené jednotlivé **položky receptury** a nyní je můžeme naplnit daty. K tomu slouží záložka Data. V závorce je počet záznamů této receptury. Klikneme na Add a budeme vkládat jednotlivé nápoje a jejich složení. Např. Kava 25ml složení 5 dílků kávy, 2 cukru a 10 vody, Caj 25ml 5 dílků čaje, 2 cukru a 10 vody atd. Těchto nápojů můžete vytvořit několik.

Recipe Database						×
Definition Data System Regist	ers					
Recipes List :						
Napoje (6)		Polozka	Kava	Caj	Cokolada	Cukr
	1	Kava 25ml	5	0	0	2
	2	Caj 25ml	0	5	0	2
	3	Turecka kava 25ml	10	0	0	2
	4	Turecka kava 50ml	20	0	0	4
	5	Cokolada 25ml	0	0	5	2
	6	Cokolada 50ml	0	0	10	4
	<					>
	Data type = Decimal Pt.	= '16-bit BCD' = '0'		[	New D	elete
				ОК	Zrušit	Nápověda

Nyní si ukážeme, jak s těmito daty pracovat. Vytvoříme si na obrazovce několik objektů, které nám k tomu poslouží. Začneme výpisem položek receptury. Zvolíme objekt **Recipe View**, který nalezneme v menu Data/History -> Recipe View

Project	Object	Data/History	lloT/Energy	View	Tool	
		Alarm Bar 📃	Event Bar Char	t 🗖		👗 Recipe View
Event (Al	arm) 🐥	Alarm Display		Re	cipe	
Log	J,	Event Display		Data	abase	
		Event/Alarm			F	Recipe

V nově otevřeném okně vybereme, jakou recepturu chceme zobrazit. Tedy Napoje. Dále můžeme vybrat, jak bude tabulka vypadat, tedy barvu pozadí, barvu vybraného řádku, způsob řazení dat v tabulce atd.

Recipe View Object's Properties
General Security Shape Font Profile
Comment :
Refresh data automaticaliy
Recipe table
Recipe Name : Napoje
Title
Color:
Profile
Grid
Color :
Selection control
Color :
Default sort method
C Enable
Sort by : Polozka 🔻
Ascending     O     Descending
OK Storno Nápověda

Potvrdíme a vložíme tabulku na obrazovku. Nyní si ukážeme jak tyto údaje vytvořit editovat nebo mazat. Nejprve si vložíme do projektu jeden ASCII objekt a pět Numeric objektů. ASCII objekt bude sloužit pro zobrazení názvů nápojů, nastavíme takto:

	cer s r ropen	iles						
General	Data Entry	Security	Shape	Font	Profile			
De	escription :							
	Allow inp	ut						
	Mask		Use U	nicode				
	Reverse	high/low by	yte					
Read a	address							
F	PLC : Local H	IMI				•	Settings	
							<u></u>	_
Addr	ess : RECIP	E		•	Polozka	•	[	2
Addr	ess : RECIP	E		-	Polozka Napoje	•	✓ Polozka	_
Addr	ress : RECIP	E		•	Polozka Napoje	•	✓ Polozka	_
Addr	ress : RECIP	E		• •	Polozka Napoje	•	✓ Polozka	
Addr	ress : RECIP	E		<ul> <li>✓</li> </ul>	Polozka Napoje	•	✓ Polozka	
Addr	ress : RECIP	E		•	Polozka Napoje		✓ Polozka	
Addr Notifica	ation	E		• •	Polozka Napoje		✓ Polozka	
Addr - Notifica	ation	E		•	Polozka Napoje	•	✓ Polozka	

Jako adresu zvolíme RECIPE dále pak Napoje -> Polozka. Nyní se vytvoří ASCII objekt, do kterého se bude vkládat název nápoje. Takto přiřadíme ostatní položky do Numeric objektu.

eneral	Data	Entry For	mat Secur	ity Shap	e Font	Profile		
D	escripti	ion :						
	V Al	ow input						
	Re	ad Mrite u	se different	addresse	e.			
Deed -			oc unici cint	000-0000				
Kead a	PLC :	i Local HMI						Settinas
Addr	ress :	RECIPE		•	Kav	a 🔻		
					/ Napo	je ▶		Selection
								Count
								Command
								Result
Matific	ation						$\checkmark$	Kava
Nounc		able						Caj
		able						Cokolada
								Cukr
								Voda
Notifica	ation o	n invalid inp	ut					
	🗐 En	able						

Nyní máme vytvořeno 6 objektů, do kterých se budou zapisovat hodnoty z tabulky receptur. **Vždy se do těchto objektů zapíše vybraný řádek.** Nadepíšeme si všechny tyto objekty, abychom věděli, který co reprezentuje a spustíme Off-line simulaci. Výsledná aplikace může vypadat následovně:

							-		
Po	lozka		Kava	Caj	Cokolada	Cukr	Vod		
Tur	ecka kava 5	50ml	20	0	0	4	20		
Tur	ecka kava 2	25ml	10	0	0	2	10		
Ka	va 25ml		5	0	0	2	10		
Co	kolada 50ml		0	0	10	4	20		
00	kalada 25ml		0	0	E	2	10	▼	
<u> </u>							•		
Položka:			Cokolada 50ml						
	Káva	Čaj	Čok	oláda	Cukr		Voda	a	
	0	0		10	4		20		

Nyní když máme objekty, které jsou svázány s danou recepturou, můžeme vytvořit funkci, která nám vytvoří nový záznam v tabulce. K tomuto účelu se používá numerická hodnota, která se zapíše do proměnné dané receptury a to do proměnné **Command**. Tato hodnota určí, co se s danými daty stane.

- 1 Zapiš data do tabulky, jako nový záznam
- 2 Uprav data ve vybraném řádku tabulky
- 3 Smaže vybraný řádek v tabulce
- 4 Smaže celou tabulku a veškerá data

Vytvoříme tedy nový objekt a to **Set Word**, který naleznete v menu Objects -> Set Word. Zde nastavíme adresu na RECIPE -> Napoje -> Command. Jako atribut zvolíme **Write constant value** a nastavíme zde 1. Tlačítko si nadepíšeme textem "Nový". Toto opakujeme i pro tlačítko "Uprav" a "Smaž", jako hodnotu zde, ale zvolíme 2 respektive 3. Vše potvrdíme a zkompilujeme. Projekt spustíme v Off-line simulaci.

New Set Word (	Object	×
General Secur	rity Shape Label	
Comme	ent:	
Write address	S	
Device :	Local HMI 🗸 🗸	4
Address :	RECIPE V Command V 16-bit Uns	aigned
	Write after button is released	
Notification	Enable	
Attribute		
Set Sty	yle: Write constant value	$\sim$
Set val	lue: 1	
	OK Zrušit Ná	ipověda

Výsledná aplikace může vypadat takto:



Někdy lze těžko prstem rolovat či dokonce vybrat určitý záznam. Proto si ukážeme, jak lze **listovat v tabulce** pomocí dvou tlačítek.

Tento způsob lze použít efektivně pouze v případě, že necháme záznamy v tabulce seřazeny tak, jak byly vytvořeny. Každý záznam má totiž své číslo a princip listování pomocí dvou tlačítek je založen na inkrementaci a dekrementaci této hodnoty. Pokud si tedy data v tabulce seřadíte podle jiného kritéria, nebude listování pomocí tlačítek fungovat efektivně.

Vložíme do projektu dva nové objekty a opět Set Word. Tlačítka nastavíme takto:

Set Word Object's Properties	Set Word Object's Properties
General Security Shape Label Profile	General Security Shape Label Profile
Comment :	Comment : Write address PLC : Local HMI Settings
Write after button is released Notification	Address : RECIPE
Attribute	Enable
Set Style : [Increment value (JOG+)  Inc. value : 1	Attribute Set Style : Decrement value (JOG-)
Image: PLC :     Local HMI     Settings       Address :     RECIPE     Count	Dec. value : 1 Bottom limit : 0
OK Storno Nápověda	OK Storno Nápověda

Okno vlevo je pro tlačítko dolu. Je zde použit inkrement o hodnotu 1 a jako horní hranice je zde použit počet receptur, tak abychom zbytečně neinkrementovali. U tlačítka nahoru je použita dekrementace o 1 s limitem 0.

Při práci s recepturami je důležitá funkce zápisu jednotlivých dat receptur do připojeného zařízení. Je tedy nezbytné, abychom byli schopni data z receptury přenést do zařízení podle daného výběru. K tomuto účelu zvolíme objekt **Data Transfer (Trigger-based).** Tyto objekty Budou dva. Jeden pro zápis do zařízení a druhý pro načtení ze zařízení. Zařízení nám bude simulovat paměťový prostor LW-100... . **Data Transfer (Trigger-based)** objekt nastavíme následovně:

Data Transfer (Trigger-based) Object's Properties	Data Transfer (Trigger-based) Object's Properties
General Security Shape Label Profile	General Security Shape Label Profile
Comment :	Comment :
Source address	Source address
PLC : Local HMI    Settings	PLC : Local HMI   Settings
Address : RECIPE   Polozka	Address : LW 🗸 100
Destination address PLC: Local HMI	Destination address PLC : Local HMI  Address : RECIPE  Polozka
Attribute No. of word : 25 Mode : Touch trigger	Attribute No. of word : 25 Mode : Touch trigger
OK Storno Nápověda	OK Storno Nápověda

Pro zápis do zařízen

Následovně vložíme na obrazovku **Numeric** objekty a jeden **ASCII** objekt pro každou hodnotu receptury. Spustíme Off-line simulaci. Výsledný projekt může vypadat následovně:

							1
F	Polozka	Kava	Caj	Cokolada	Cukr	Voda	
	Furecka kava 50ml	20	0	0	4	20	
T T	Furecka kava 25ml	10	0	0	2	10	
	NOVY NAPOJ	0	20	0	5	10	
	Kava 25ml	5	0	0	2	10	
	Cokolada 50ml	0	0	10	4	20	
		^	<u>^</u>	-	-	1.0	<u> </u>
Položka	vžka: Kava 25ml						
	Káva Čaj	Čok	oláda	Cukr		Voda	
Download	5 0		0	2		10	
Upload Nový Uprav Smaž							

Aplikace ke stažení zde

TECON s.r.o.

#### Pro načtení ze zařízení

 $\times$ 

## Makra

Pokud bychom potřebovali využít určité funkce, které nejsou předvytvořeny jako objekty, můžeme použít **Makro.** Makro je spustitelný kód, který se vykoná na základě určené podmínky. Lze jej spustit stiskem tlačítka, změnou hodnoty nebo podle časového intervalu. Samotné **Makro** nalezneme v menu Project -> Macro.

Hom	e Project	Object	Data/History	lloT/Energy	View	Tool					
×				SD Ruild Developed	Pahaat	<b>Shane</b>	Disture		⇒ <mark>abC</mark> XXX		
mplie	Simulation	Simulation	(PC->HMI)	Files	HMI	- snape	Picture	Label	Sunng	IVIdCIO	Address
	Build								Library	r	

V nově otevřeném dialogovém okně zvolíme možnost New a otevře se nám následující okno:

Macro		
Macro list		
[ID:000]	macro_0 New	
	Work Space	
	Macro ID: 1 Macro name : macro_1	Security  Use execution condition
	Periodical execution	Execute one time when HMI starts
	□ Ω Ω Å 🛍 🛍 🔺 🕻 🕻 🌾 📯	
	1 2 macro_command main() 3 4 5 end macro command	ŕ
*I : Execute one ti		
*P : Periodical exec		
Macro under develo		E
Set password		-
Address variables address format in	Click the right mouse button to display edit menu.	•
	GET/SET FN Save & Compile Off-line Simulation	On-line Simulation Exit Help

V horní části si makro můžeme pojmenovat a nastavit podmínky jeho spuštění. Pokud zaškrtneme **Use execution condition** a stiskneme **Settings**, tak můžeme nastavit podmínku pro spouštění, resp. nespuštění. Zde můžeme nastavit nadřazenou podmínku, která deaktivuje dané makro. Pokud budeme makro spouštět tlačítkem nebo časovým intervalem, nebude spuštěn, pokud bude tato podmínka splněna.

	Security     Image: Security     Image: Security     Image: Security     Settings
	Execute one time when HMI starts
Sec	rity
	Disable when Bit is ON     Disable when Bit is OFF
	PLC : Local HMI
	OK Cancel

Dále lze nastavit časové spouštění makra. V horní části zaškrtneme možnost **Periodical execution**. Poté stačí nastavit časový interval pro spouštění makra.

			-
Periodical execution	Time interval (0 ~ 864000) :	20	x 100ms

Samotný kód makra je nutné vložit mezi **macro\_command main()** a **end macro\_command.** Kód makra je tvořen před vytvořenými funkcemi. Tyto funkce můžete vložit pomocí tlačítka **GET/SET FN... GET/SET FN...**. V nově otevřeném dialogovém okně volíme, kterou funkci chceme použít. Popis všech funkcí, jejich použití a zápis, jsou popsány v manuálu EB Pro.

🗹 Build-in 🔲 Library	
Function name : ACOS	
BIN2BCD	
Buzzer	
COS	
[Description]	<u> </u>
CSC	
[Usage] CUBERT	E
ACOS(source, result DEC2ASCII	
float source = 0.5 rs FindDataSamplingDate	
FindDataSamplingIndex	-
FindEventLogDate	Þ
Variable 1 FI OAT2ASCII	
Variable type : GETBIT	
GetCnvTagArrayIndex	
Variable : GetCTS	
GetData	
GetError	
Variable type : HEX2ASCII	
Variable : LINKOPD	
INPORT	
INPORT2	
INPORT3	
INVBIT	
LOG	
ОК Са	ncel

Založíme si nový projekt. Tento projekt bude načítat hodnotu z paměti LW-100. Tato hodnota bude v rozsahu 0-100. Řekněme, že bude simulovat vstup z teplotního čidla. Pomocí makra budeme tuto hodnotu přepočítávat na hodnotu v rozsahu -20° až 20°. Vložíme na obrazovku dva **Numeric** objekty, LW-100 a LW-50 do které budeme zapisovat výslednou hodnotu ve stupních. Pro hodnotu v LW-50 zvolíme datový typ 16-bit Signed. Otevřeme makro v menu Project -> Macro a zvolíme možnost New.

Abychom docílili požadovaných vlastností, zvolíme funkce **GetData()** a **SetData()**. Funkce **GetData()** načte hodnotu z LW-100 a uloží ji do proměnné vstup. Poté se vykoná matematická operace pro přepočet na stupně. Dále použijeme funkci **SetData()** pomocí, které zapíšeme proměnou vystup do paměti LW-50. Makro bude vypadat následovně.

Work Space	
Macro ID : 0 Macro name : macro_0	Security Use execution condition
Periodical execution	Execute one time when HMI starts
2 C X B B / % % % %	
<pre>1 macro_command main() 2 int vstup 3 GetData(vstup, "Local HMI", LW, 100, 1) 4 int vystup 5 vystup = (vstup * 40 / 100) - 20 6 SetData(vystup, "Local HMI", LW, 50, 1) 7 8 end macro_command </pre>	E
<	• • • • • • • • • • • • • • • • • • •
* Click the right mouse button to display edit menu. 0 error(s)	
GET/SET FN Save & Compile Off-line Simulation	On-line Simulation Exit Help

Zkompilujeme a uložíme makro pomocí tlačítka **Save & Compile.** Ve spodním okně se zobrazí, zda proběhla kompilace v pořádku bez chyb. Zavřeme okno.

Dané makro budeme spouštět pomocí bitového tlačítka. Přidáme na obrazovku **Toggle Switch**, který naleznete v menu Objects -> Toggle Switch. V nově otevřeném dialogovém okně, nastavíme vlastnosti daného tlačítka. Důležitá je funkce **Macro** umístěna ve spodní části. Zaškrtneme **Execute macro** a zvolíme makro z uložených. Dále pak mód pro spuštění makra. Vložíme tlačítko na obrazovku.

Toggle Switch/Bit Lamp Object's Properties
General Security Shape Label Profile
Comment :
Bit Lamp     O     Toggle Switch
Read/Write use different addresses
PLC : Local HMI
Address : LB 🗸 0
Invert signal
Write address :
Write when button is released
Attribute Switch style : Toggle
Macro
✓ Execute macro     Macro : [ID:000] macro_0     Trigger mode : OFF<->ON     ✓
OK Storno Nápověda

Projekt zkompilujeme a spustíme **Off-line** simulaci. Výsledný projekt může vypadat následovně:

	-	$\times$
Spusť makro	Načtená hodnota rozsah 0 - 100 35	
	Přepočítaná hodnota rozsah -20° až 20° <u>-6</u>	

Aplikace ke stažení zde

# Security

Mezi další vlastnosti, které lze využít na panelech Weintek, patří možnost využití uživatelů. Lze tak jednotlivým prvkům **nastavit úroveň zabezpečení**. Stejné je to i s uživateli. Lze tak rozdělit uživatele a objekty do různých tříd a udělat tak **různé úrovně přístupů**. Samotné přihlašování a správa uživatelů v projektu je již před vytvořena na obrazovkách 70,71,72,73,74.

**Správu uživatelů** naleznete v menu Home -> System parameters



V nově otevřeném okně přejdeme na záložku Security. Zde si vytvoříme tři nové uživatele. Obsluha, Technik a Administrator. Nastavíme jim třídy přístupu. Obsluze pouze Class A, Technik bude mít nastaveno Class A a Class B. Administrator dostane Class A,B,C.

	Cellular Da	ata Network	Print	ter/Backu	p Server	Time	Sync./DST		6	e-Mail
Dev	/ice	Model	General	Syste	m Remot	e S	ecurity		Extended	Memo
С	) General I	mode	Enhanced security mode		LDAP		Editable			
Sele	ect operat Use exi below.	ole classes for o sting user acco	each user ounts and admir	nistrator s	ettings on HMI fi	rst (if exist	ed). Other	wise,	use settir	ngs
	Enable	Secret user	User r	name		Passwor	d	_	Class A	Clas:
1	$\checkmark$		Obsluha		•••••		weak	۲		$\checkmark$
2	$\checkmark$		Technik		•••••		weak	۲	$\checkmark$	$\checkmark$
3	$\checkmark$		Administrato	or	•••••		weak	۲		
4			user4		•		weak	۲		
5			user5		•		weak	۲		
6			user6		•		weak	۲		
7			user7		•		weak	۲		
Ac	lministrato ] Secret u	or ser	User name :	admin					7	
	-		Password :	•••••	•		wea	ak 🤇		
Co	ontrol add	ress								
		Device : Loca	al HMI					<	÷	
	4	Address : LW		$\sim$	8950		16-bit Un	signe	d <u>Usa</u>	iqe
	file pass	word								
MTF	Chocker	assword when	open MTP file				Settings			
MTF [										
MTF [ Exe	cute auto	.login/logout w	/hen insert an U	ISB key in	to HMI					
MTF [ Exe	cute auto	.login/logout w	hen insert an U	ISB key in	to HMI					
MTF Exe	cute auto Enable	.login/logout w	/hen insert an L	ISB key in	to HMI					
MTF [ Exe	Cute auto Enable	.login/logout w	/hen insert an U	ISB key in	to HMI					
MTF [ Exe [	Check ; cute auto Enable	.login/logout w	/hen insert an U	ISB key in	to HMI	voiration *	ne of LICP	kav		

Obsluha bude mít nastavené heslo na "111111", Technik bude mít heslo "222222" a Administrator bude mit heslo "admin""

Nyní si vytvoříme několik objektů, kterým nastavíme určité třídy zabezpečení. Vytvoříme si tři bitové lampičky, které nastavíme jako toggle. Každou z nich však nastavíme na jinou úroveň ochrany. Přejdeme do záložky Security a nastavíme pro první Class A, pro druhou Class B atd...

Toggle Switch/Bit Lamp Object's Properties	×
General Security Shape Label Profile	
Safety control	
Min. press time (sec) : U	
Display confirmation request	
Enable/Disable	
Use register status/value	
User restriction Object class : Class : A Object class	
Sound	
OK Zrušit Použít Nápověda	3

Dále si vytvoříme tři Numeric objekty. Opět v záložce Security nastavíme pro každý jinou Class zabezpečení. Zde ale použijeme jednu z možností pod výběrem úrovně, a to **Display warning message if access denied**.

Numeric Object's Properties	×
General Data Entry Format Security Shape Font Profile	
Safety control           Display confirmation request	
Enable/Disable	
User restriction Object dass : Class : A	
Disable protection permanently after initial activation	
Display warning message if access denied	
Make invisible while protected	
* If the user tries to operate on an object without authorization, LB-12056 will be set to ON.	
Sound Enable	
OK Zrušit Použít Nápověd	a

Dále vložíme několik Function Key objektů k vyvolání oken. Nejprve okno pro přidání uživatele. Nastavíme ho tak, aby zobrazil popup okno a poté ze seznamu vybereme okno 71. Add account.

-		
seneral	Security Shape Label Profile	
	Comment :	
_		
Ac	tivate after button is released	
•	Display popup window	~
d	ose this popup window when parent window is closed	
_	Style : Marsh Hills have	
vv	71. Add account	~
Keyboa	ard input	
0	[Enter] V	
- 1	к <i>э</i>	
	ecute macro	
00		
Ow	indow title bar	
0		
-Hard o	opy screen to USB disk, SD card or printer	
() Sc	reen hardcopy	
	port user data/Use [USB Security Key]	
Notifica	ation	
F	able	

Takto si vytvoříme několik tlačítek na vyvolání i oken 72 pro mazání uživatelů, 73 pro správu práv uživatelů a 74 pro správu hesel. Opět je můžeme rozdělit do různých úrovní přístupu.

 $\times$ Uživatel: Obsluha Class A Class B Class C Odhlásit Přihlásit 0 25 0 Přidat + Přidat + Password Protected! Access Denied!! Close and the Smazat Smazat

Výsledný projekt může vypadat následovně:

# Ostatní aktivní prvky

Na předchozích příkladech jsme si ukázali způsob vytváření obrazovek, definování aktivních prvků, práci s daty, receptury, psaní maker apod. Všechny ostatní prvky se definují velmi obdobným způsobem.

Prvky jako sloupcový graf, ručkový ukazatel, více stavový přepínač, animace, zadávání textů atd. mají velmi intuitivní rozhraní a jejich fungování je z dialogových oken zřejmé.

Pokud jste si prošli všechny uvedené příklady, nebude pro vás jistě problém vytvořit plně funkční aplikaci pro libovolnou řízenou technologii. Narazíte-li přesto na nějakou nejasnost, nahlédněte **do originálního manuálu k software EasyBuilder Pro**.

**Ukázkové projekty**, tak jak jsme je vytvářeli v této příručce, nevyžadují připojení programovatelného automatu a **můžete jejich funkčnost na panelu vyzkoušet** stejně tak, jako jsme je testovali v off-line simulaci na PC. Všechny projekty naleznete <u>zde</u>