

# FESTO MOTION TERMINAL VTEM CONTROLLING UNIVERSAL PRESSING FACILITY

*Ladislav Červinka*

*České vysoké učení technické v Praze, Fakulta strojní, ladislav.cervinka@fs.cvut.cz*

*Abstrakt: Tato práce se zabývá programováním lisovacího přípravku pomocí ventilů VEVM - Festo Motion Terminálu (FMT) VTEM. Cílem je ukázka možného způsobu programování tohoto fenoménu v prostředí CoDeSys, a to pomocí jazyku ST (structured text). Řešen je tak CoDeSys projekt řízení lisovacího přípravku s využitím instrumentace pomocí technologie Festo Motion Terminal s možností zadávat různé výrobní sekvence, různé parametry výrobních sekvencí, různé výrobní dávky.*

*Klíčová slova: FTM, FESTO, MOTION, TERMINAL, CPX, VTEM, MECHATRONIKA, PNEUMATICKÉ SYSTÉMY, PRŮMYSL 4.0, MOTION APPS*

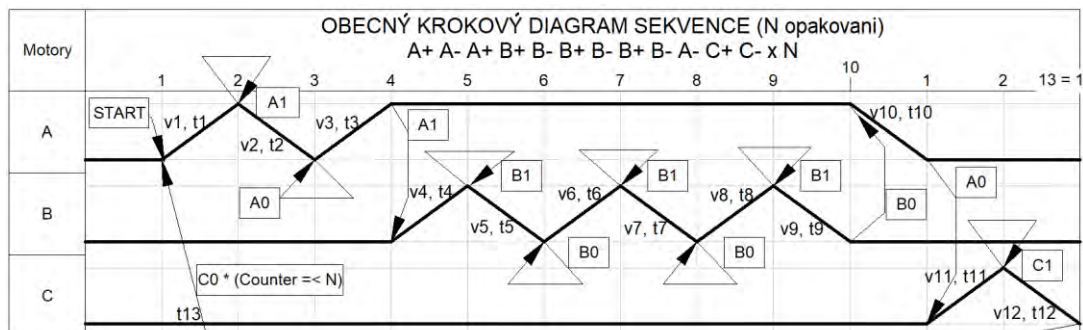
## 1. Úvod

V dnešní době jsou pneumatické systémy nedílnou součástí průmyslu, ale i běžného života člověka. Díky rychlému vývoji elektroniky a dalších nových technologií se moderní pneumatika pojí s výpočetní technikou. Vznikl tak koncept digitální pneumatiky, který spadá pod Průmysl 4.0. Firma Festo si pojmy jako mechatronika, průmysl 4.0, IoT (internet of things, resp. internet věcí) vzala za své a představila Festo Motion Terminal VTEM (FMT) a ventil VEVM. Jednotlivé ventily tohoto terminálu jsou řízeny aplikacemi funkcí (Motion Apps) a lze tak jediným ventilem nahradit funkci až 50 standardních výrobků. V této práci je věnována pozornost programování tohoto fenoménu v prostředí CoDeSys.

## 2. Lisovací přípravek

### 2.1 Popis úlohy

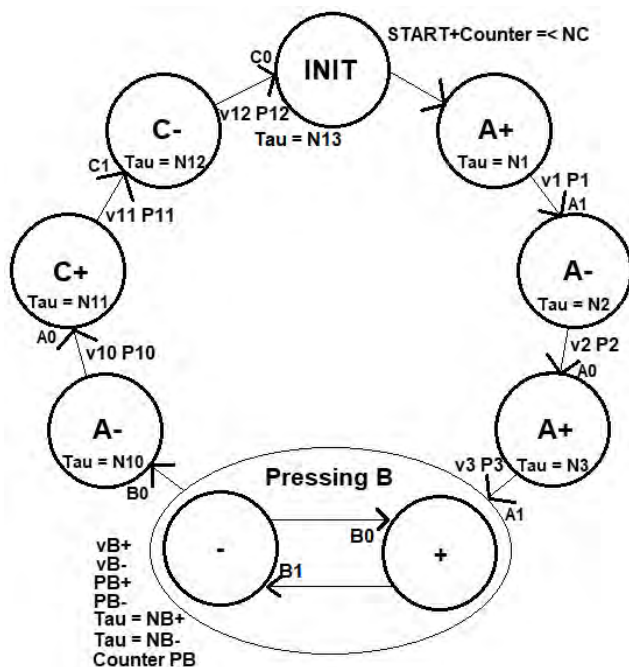
Podle zadání má být po stisknutí tlačítka START postupně zalisováno N obrobků. Raznice upevněná na pneumotoru A jedním dlouhým a krátkým úderem provede zalisování polotovaru. Následně, kdy zůstane motor A ve vysunuté poloze, nalisuje pneumotor B z boku pojistný kolík do obrobku. Nalisování pneumotorem B je provedeno jedním pomalým a dvěma rychlými údery. Nakonec pneumotor C (vyhazovač) vysune hotový kus z lisovacího ukotvení a výroba může pokračovat do dalšího cyklu [1].



Obr. 1 – Krokový diagram úlohy

## 2.2 Návrh projektu FMT

Celý projekt byl zpracován bez fyzického přístupu k FMT. Proto není možné program odzkoušet a ladit. Algoritmus a jednotlivé kroky byly navrženy pomocí větvení CASE - OF a podmínek IF - THEN. Základní sekvence byla navržena podle obecného krokového diagramu na obr. 1, a to s tím rozdílem, že bylo přidáno cyklování lisování pneumotorem B a díky funkcím FMT bylo přidáno ovládání nejen rychlostí pneumotorů a časových prodlev, ale také tlaků [3]. Stavový diagram je pak v podobě následujícího obr. 2.



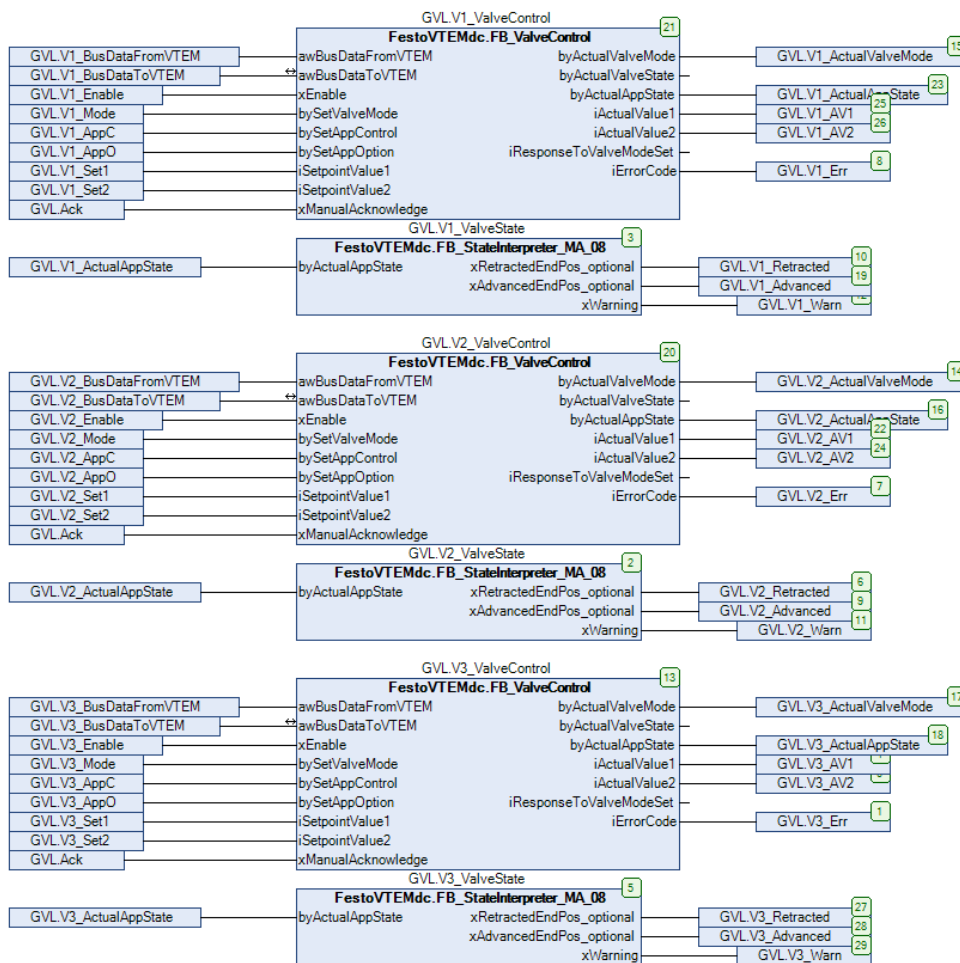
Obr. 2 – Stavový diagram úlohy

Tab. 1 – Popis označení krokového a stavového diagramu

OZNAČENÍ	POPIS
INIT	Inicializace
A+, B+, C+	Akce pneumotorů vysunout
A-, B-, C-	Akce pneumotorů zasunout
vX	Rychlost pístu
PX	Tlak vzduchu
Tau, tx	Časová prodleva
NX	Délka časové prodlevy
Counter	Čítač
N, NC	Limit čítače
START	Startovací tlačítko

Nejprve bylo zapotřebí v rámci projektu zvolit typ řídicího modulu CPX-CEC a programovací jazyk, ve kterém bude program pro PLC založen. Zařízeno bylo zvoleno CPX-CEC-S1-V3 a programovací jazyk strukturovaného textu. Do správce knihoven byla přidána knihovna „Festo\_VTEM\_DevCon“, která je nutná pro návrh celého projektu [2].

Aby bylo možné začít s programováním, bylo nejdřív nutné vytvořit programovou organizační jednotku (POU) a v ní založit funkční bloky (FB) pro MotionApp č.8, pomocí kterých se řídí všechny 3 pneumotory, resp. ventily [2]. Tyto bloky je možné vytvořit pomocí funkce „Input Asistant“, a to díky přidání zmíněné knihovně. V následujícím obr. 3 jsou funkční bloky „FB\_ValveControl“ a „FB\_StateInterpreter\_MA\_08“. Vstupy a výstupy těchto bloků jsou navázány na globální proměnné, se kterými funkční bloky pracují.



Obr. 3 – Zapojení funkčních bloků

## 2.3 Návrh programu FMT

Hlavní část programu je složená z větvení pomocí CASE – OF a IF – THEN, tedy skoku mezi jednotlivými číslovanými kroky a vykonávání akcí po splnění zadaných podmínek. Na začátku programu se nachází deklarace časovačů a čítačů. Na začátku programu, tj. před částí, která je vytvořená pomocí CASE – OF, se také nachází ovládání tlačítka ON ve vizualizaci programu, stejně jako krátké větvení pro ovládání prvku „spin control“. Typicky se v praxi program pro HMI nebo např. také pro alarmy nachází v jiné části projektu, tedy mimo hlavní sekvencér, ovšem v tomto případě je program pro HMI krátký, a proto bylo souzruzeno, že celý program bude v jediném celku.

Program začíná v „IDLE“ stavu, ve kterém čeká na přepnutí BOOL vstup „ON\_OFF“ z HMI. Po splnění této podmínky se v HMI zobrazí zbylé ovládací prvky HMI.

Pro HMI prvek „spin control“, který ovládá výběr kroku, ve kterém jsou nastavovány parametry rychlost (INT), tlak (INT) a čas (REAL), byl vytvořen „struct“. V tomto prvku je vytvořeno 8 dalších prvků, resp. čísel v poli (array [1..8]), což je počet kroků, ve kterých jsou měněny zmíněné parametry. Nastavená hodnota (INT) v HMI prvku „spin control“ (dále jako „výběr kroku“) se porovnává s aktuální hodnotou v bufferu a pokud se nerovná, hodnota z výběru kroku se do bufferu nahraje. Poté se hodnoty v proměnných zmíněného structu uloží do HMI proměnných, tak se v HMI tyto hodnoty zobrazí. Poslední větev IF zajišťuje ukládání z HMI do hodnot pole structu, a tím tak ukládá parametry všech kroků, ve kterých se prvky structu nachází.

Dále následují větvení pomocí CASE – OF [4]. V nulovém kroku na začátku dochází k resetování / odstranění chybových hlášek ventilů. Následují vstupy pro inicializaci stavů ventilů, čítačů a parametrů ventilů, dále vstupy pro

zapnutí automatického režimu a režimu manuálního ovládání ventilů. Pro zapnutí jednoho nebo druhého režimu je nutné pomocí HMI spustit inicializaci.

V algoritmu sekvenceru automatického režimu se vždy opakují dva kroky pro vykonání jedné celé akce pneumotoru (např. vysunutí pístu). První krok, po splnění IF podmínky (inicializovaný stav, nebo sepnutí koncového spínače z akce předešlého kroku a případně doběhnutí časovače), slouží v této posloupnosti k provedení: resetu časovače, nastavení parametrů rychlosti, tlaku a časové prodlevy po vykonání kroku, vykonání akce pneumotoru (vysunout píst / zasunout píst) a skok do dalšího kroku. Druhý krok slouží pro vykonání časové prodlevy po vykonání akcí z předešlého kroku. Pro lepší přehlednost jsou tyto dva kroky oddělené, avšak pro funkčnost to není nutné (počet větví IF – THEN není nijak omezený ve větvích CASE – OF)[5]. Na konci sekvenceru je čítač, pomocí kterého je řešené opakování celé sekvence. Počet cyklů se nastavuje v HMI a do výše tohoto čísla se postupně inkrementuje hodnota po každém dokončeném cyklu hlavní sekvence. Pokud je čítač ve své maximální přednastavené hodnotě, tzn. uběhl daný počet cyklů sekvence, skočí program do nulového kroku a tím je celý cyklus u konce. Akce pneumotoru B jsou stejným způsobem zacyklené od 40. až do 60. kroku. Počet lisovacích akcí pneumotorem B je možné v HMI nastavit stejným způsobem, jako počet cyklů celé sekvence. Parametry pro cyklus lisování pneumotorem B jsou také parametrizovatelné, ovšem jsou stejné pro celý cyklus.

```

1  POU( );
2  // časovače, čítače a default stavy
3  Casovac (IN:= Cas_ON, PT:= REAL_TO_TIME(Cas_kroku*1000),
4  Q=> Cas_ubehl , ET=> );
5  Citac_Cyklu(
6  CU:= (Cyklus_sekvence),
7  RESET:=(Reset_citac_sekvence) ,
8  PV:= Cyklus_CNT ,
9  Q => Sekvence_vykonana ,
10 CV=> HMI_CNT_Cyklu_aktualni);
11 Citac_B(
12 CU:= (Cyklus_b_dokoncen),
13 RESET:=(Reset_citac_b) ,
14 PV:= B_POCETCYKLU_CNT ,
15 Q => B_vykonano ,
16 CV=> HMI_CNT_B_aktualni);
17
18 IF spinindex <> spinbuffer THEN
19 load :=1;
20 spinbuffer := spinindex;
21 END_IF;
22
23 IF load = 1 THEN
24 GVL.HMI_RYCHLOST := GVL.HMI_CONTROL[SpinIndex].Rychlost;
25 GVL.HMI_CAS := GVL.HMI_CONTROL[SpinIndex].Cas;
26 GVL.HMI_TLAK := GVL.HMI_CONTROL[SpinIndex].Tlak;
27 load := 0;
28 END_IF;
29
30 IF hmi_save = 1 THEN
31 GVL.HMI_CONTROL[SpinIndex].Tlak := GVL.HMI_TLAK;
32 GVL.HMI_CONTROL[SpinIndex].Cas := GVL.HMI_CAS;
33 GVL.HMI_CONTROL[SpinIndex].Rychlost := GVL.HMI_RYCHLOST;
34 END_IF;
35
36 IF ON_OFF THEN
37 Viditelnost_HMI := TRUE;
38 GVL.V1_Enable := TRUE;
39 GVL.V2_Enable := TRUE;
40 GVL.V3_Enable := TRUE;
41 ELSE
42 GVL.V1_Enable := FALSE ;
43 GVL.V2_Enable := FALSE ;
44 GVL.V3_Enable := FALSE ;
45 Viditelnost_HMI := FALSE;
46 INIT_DONE := 0;
47 MANUAL := FALSE;
48 AutoMod := FALSE;
49 END_IF
50
51 IF (GVL.V1_ActualValveMode = 61 OR GVL.V2_ActualValveMode = 61
52 OR GVL.V3_ActualValveMode = 61) THEN
53 GVL.xMan := TRUE;
54 END_IF
55
56 IF INIT_SET THEN
57 Cyklus_sekvence := 0 ; //Nulování vstupu CU na čítači
58 Cyklus_b_dokoncen := 0 ; //Nulování vstupu CU na čítači
59 GVL.V1_Mode := 8; //MotionApp 8
60 GVL.V2_Mode := 8;
61 GVL.V3_Mode := 8;
62 GVL.V1_AppC := 2; //Zasunutí motoru
63 GVL.V2_AppC := 2;
64 GVL.V3_AppC := 2;
65 Reset_citac_sekvence := 1;
66 Reset_citac_b := 1;
67 GVL.V1_Set1 := 10000 ;
68 GVL.V1_Set2 := 6000 ;
69 GVL.V2_Set1 := 10000 ;
70 GVL.V2_Set2 := 6000 ;
71 GVL.V3_Set1 := 10000 ;
72 GVL.V3_Set2 := 6000 ;
73 INIT_DONE := 1;
74
75 Cyklus_CNT := HMI_pocet_cyklu_sekvence;
76 B_POCETCYKLU_CNT := HMI_pocet_cyklu_b;
77 Krok := 0;
78 END_IF
79
80 IF AutoMod THEN
81 MANUAL := 0;
82 AutoMod_STATE := 1;
83 Krok := 0;
84 END_IF
85
86 IF AutoMod_STATE AND Start AND GVL.V1_Retracted AND GVL.V2_Retracted
87 AND GVL.V3_Retracted AND INIT_DONE THEN
88 Reset_citac_sekvence := 0;
89 Reset_citac_b := 0;
90 Krok := 10 ;
91 END_IF

```

```

91 10:
92     IF GVL.V1_Retracted THEN
93         Cas_ON := 0 ; //Reset časovače
94         Reset_citac_buchar := FALSE ; //Shození resetu čítače
95         GVL.V1_Set1 := GVL.HMI_CONTROL[1].Rychlost*100 ;
96 //Nastavení rychlosti pneumotoru V1 z HMI pro krok 1
97         GVL.V1_Set2 := GVL.HMI_CONTROL[1].Tlak ;
98 //Nastavení tlaku pro krok1
99         Cas_kroku := GVL.HMI_CONTROL[1].Cas ;
100 //Nastavení časovače z HMI po kroku 1
101         GVL.V1_AppC := 1 ; //Pneumotor 1 vysunout
102         Krok := 15 ;
103     END_IF
104 15:
105     IF GVL.V1_Advanced THEN
106         Cas_ON := 1 ;
107         Krok := 20 ;
108     END_IF
109 20:
110     IF (GVL.V1_Advanced AND Cas_ubehl) THEN
111         Cas_ON := 0 ; //Reset časovače
112         GVL.V1_Set1 := GVL.HMI_CONTROL[2].Rychlost*100
113 ; //Nastavení rychlosti pneumotoru V1 z HMI pro krok 2
114         GVL.V1_Set2 := GVL.HMI_CONTROL[2].Tlak ;
115 //Nastavení tlaku pro krok2
116         Cas_kroku := GVL.HMI_CONTROL[2].Cas ;
117 //Nastavení časovače z HMI po kroku 2
118         GVL.V1_AppC := 4 ; //Pneumotor 1 zasunout
119         Krok := 25 ;
120     END_IF
121 25:
122     IF GVL.V1_Retracted THEN
123         Cas_ON := 1 ;
124         Krok := 30 ;
125     END_IF
126 30:
127     IF (GVL.V1_Retracted AND Cas_ubehl) THEN
128         Cas_ON := 0 ; //Reset časovače
129         GVL.V1_Set1 := GVL.HMI_CONTROL[3].Rychlost*100 ;
130 //Nastavení rychlosti pneumotoru V1 z HMI pro krok 3
131         GVL.V1_Set2 := GVL.HMI_CONTROL[3].Tlak ;
132 //Nastavení tlaku pro krok3
133         Cas_kroku := GVL.HMI_CONTROL[3].Cas ;
134 //Nastavení časovače z HMI po kroku 3
135         GVL.V1_AppC := 1 ; //Pneumotor 1 vysunout
136         Krok := 35 ;
137     END_IF
138 35:
139     IF GVL.V1_Advanced THEN
140         Cas_ON := 1 ;
141         Krok := 40 ;
142     END_IF
143 40:
144     IF GVL.V2_Retracted AND Cas_ubehl THEN
145         Cas_ON := 0 ; //Reset časovače
146         Cyklus_b_dokoncen := 0 ; //Reset vstupu čítače b
147         GVL.V2_Set1 := GVL.HMI_CONTROL[4].Rychlost*100 ;
148 //Nastavení rychlosti pneumotoru V2 z HMI pro aktuální krok
149         GVL.V2_Set2 := GVL.HMI_CONTROL[4].Tlak ;
150 //Nastavení tlaku pro krok
151         Cas_kroku := GVL.HMI_CONTROL[4].Cas ;
152 //Nastavení časovače z HMI po kroku
153         GVL.V2_AppC := 1 ;
154         Krok := 45 ;
155     END_IF
156 45:
157     IF GVL.V2_Advanced THEN
158         Cas_ON := 1 ;
159         Krok := 50 ;
160     END_IF
161 50:
162     IF GVL.V2_Advanced AND Cas_ubehl THEN
163         Cas_ON := 0 ;
164         GVL.V2_Set1 := GVL.HMI_CONTROL[5].Rychlost*100 ;
165 //Nastavení rychlosti pneumotoru V2 z HMI pro aktuální krok
166         GVL.V2_Set2 := GVL.HMI_CONTROL[5].Tlak ;
167 //Nastavení tlaku pro krok
168         Cas_kroku := GVL.HMI_CONTROL[5].Cas ;
169 //Nastavení časovače z HMI po kroku
170         GVL.V2_AppC := 4 ;
171         Krok := 55 ;
172     END_IF
173 55:
174     IF GVL.V2_Retracted THEN
175         Cas_ON := 1 ;
176         Cyklus_b_dokoncen := 1 ;
177         Krok := 60 ;
178     END_IF
179 60:
180     IF GVL.V2_Retracted AND Cas_ubehl AND B_vykonano = 0 THEN
181         Krok := 40 ;
182     END_IF
183     IF GVL.V2_Retracted AND Cas_ubehl AND B_vykonano = 1 THEN
184 // 10. krok dle obecného krokového diagramu
185         Cas_ON := 0 ;
186         Reset_citac_b := TRUE ;
187         GVL.V1_Set1 := GVL.HMI_CONTROL[6].Rychlost*100 ;
188 //Nastavení rychlosti pneumotoru V1 z HMI pro aktuální krok
189         GVL.V1_Set2 := GVL.HMI_CONTROL[6].Tlak ;
190 //Nastavení tlaku pro aktuální krok
191         Cas_kroku := GVL.HMI_CONTROL[6].Cas ;
192 //Nastavení časovače z HMI po aktuálním kroku
193         GVL.V1_AppC := 4 ;
194         Krok := 105 ;
195     END_IF
196 105:
197     IF GVL.V1_Retracted THEN
198         Cas_ON := 1 ;
199         Krok := 110 ;
200     END_IF
201 110:
202     IF GVL.V1_Retracted AND Cas_ubehl THEN
203         Cas_ON := 0 ;
204         GVL.V3_Set1 := GVL.HMI_CONTROL[7].Rychlost*100 ;
205 //Nastavení rychlosti pneumotoru V3 z HMI pro aktuální krok
206         GVL.V3_Set2 := GVL.HMI_CONTROL[7].Tlak ;
207 //Nastavení tlaku pro aktuální krok
208         Cas_kroku := GVL.HMI_CONTROL[7].Cas ;
209 //Nastavení časovače z HMI po aktuálním kroku
210         GVL.V3_AppC := 1 ;
211         Krok := 115 ;
212     END_IF
213 115:
214     IF GVL.V3_Advanced THEN
215         Cas_ON := 1 ;
216         Krok := 120 ;
217     END_IF
218 120:
219     IF GVL.V3_Advanced AND Cas_ubehl THEN
220         Cas_ON := 0 ;
221         GVL.V3_Set1 := GVL.HMI_CONTROL[8].Rychlost*100 ;
222 //Nastavení rychlosti pneumotoru V3 z HMI pro aktuální krok
223         GVL.V3_Set2 := GVL.HMI_CONTROL[8].Tlak ;
224 //Nastavení tlaku pro aktuální krok
225         Cas_kroku := GVL.HMI_CONTROL[8].Cas ;
226 //Nastavení časovače z HMI po aktuálním kroku
227         GVL.V3_AppC := 1 ;
228         Krok := 125 ;
229     END_IF
230 125:
231     IF GVL.V3_Retracted THEN
232         Cas_ON := 1 ;
233         Cyklus_sekvence := 0 ;
234         Cyklus_sekvence := 1 ;
235         Krok := 130 ;
236     END_IF

```



```

237 130:
238 IF GVL.V3_Retracted AND Cas_ubehl AND Sekvence_vykonana = 0 THEN
239     Krok := 10 ;
240 END_IF
241 IF GVL.V3_Retracted AND Cas_ubehl AND Sekvence_vykonana = 1 THEN
242     HMI_sekvence_dokoncena := Sekvence_vykonana;
243     Cas_ON := 0 ;
244     Reset_citac_sekvence := TRUE ;
245     Krok := 0 ;
246 END_IF
247 END_CASE

```

Obr. 6 – Třetí část ST programu

### 3. Závěr

Byl vytvořen projekt v prostředí CoDeSys pro ovládání 3 pneumotorů pomocí Festo Motion Terminálu VTEM - VEVN ventilů zahrnující: volbu aplikace funkcí „Motion Apps“ č.8 s názvem „volitelná úroveň tlaku“, která dovoluje parametrizovat úroveň tlaku v milibarech a otevření větrání ventilu v procentech - pro řízení rychlosti pístů pneumotorů. Dále je v projektu vybrán řídicí modul CPC-CEC-S1-V3 a programovací jazyk ST (strukturovaný text). Byla přidána knihovna „Festo\_VTEM\_DevCon“ pro následné programování zmíněného hardwaru.

V rámci projektu byla vytvořena programová organizační jednotka, ve které se nachází funkční bloky přidané ze zmíněné knihovny. Pro ovládání jednoho VEVN ventilu, v rámci aplikace funkcí č.8, byly přidány vždy dva funkční bloky (tzn. celkově 6 bloků), pracující s globálními proměnnými, které jsou následně přímo ovlivňovány programem a ovládáním z HMI. Byl vytvořen návrh řídicího algoritmu VEVN ventilů pro ovládání 3 pneumotorů. Toho bylo docíleno pomocí větvení CASE – OF a podmínek IF – THEN. Jednotlivé kroky sekvence lze parametrizovat (rychlost pístu pneumotoru, tlak a časová prodleva po vykonání kroku). Celou sekvenci je možné cyklit pomocí čítače. Akce pneumotorem B je možné uvnitř každého jednoho cyklu také cyklit pomocí druhého čítače. Všechny tyto parametry, tedy rychlost, tlak, čas a oba čítače, lze měnit ve vizualizaci HMI.

Byl tak vytvořen projekt řízení lisovacího přípravku s využitím instrumentace pomocí technologie Festo Motion Terminal s možností zadávat různé výrobní sekvence, různé parametry výrobních sekvencí, různé výrobní dávky

Na tento projekt navazuje vytvoření vizualizace. Pomocí vizualizace HMI by bylo možné ovládat inicializaci, start sekvence a měnit zmíněné parametry v jednotlivých krocích. V takové vizualizaci by také bylo možné sledovat a ukládat provozní data pro jejich vyhodnocení.

### ZDROJE

- [1] *ZADANI\_RA\_E\_V2.pdf* [online]. [vid. 2021-07-26]. Dostupné z: [https://moodle-vyuka.cvut.cz/pluginfile.php/153425/mod\\_resource/content/1/ZADANI\\_RA\\_E\\_V2.pdf](https://moodle-vyuka.cvut.cz/pluginfile.php/153425/mod_resource/content/1/ZADANI_RA_E_V2.pdf)
- [2] *Koubová - DIDAKTICKÉ ÚLOHY PRO DIGITÁLNÍ PNEUMATIKU S VYUŽITÍM.pdf* [online]. [vid. 2021-07-11]. Dostupné z: [https://dspace.cvut.cz/bitstream/handle/10467/90334/F2-DP-2020-Koubova-Jana-DP\\_Koubova\\_Jana\\_2020.pdf?sequence=-1&isAllowed=y](https://dspace.cvut.cz/bitstream/handle/10467/90334/F2-DP-2020-Koubova-Jana-DP_Koubova_Jana_2020.pdf?sequence=-1&isAllowed=y)
- [3] *Digitalizovaný pneumatický systém | Festo Motion Terminal* [online]. [vid. 2020-12-27]. Dostupné z: <https://www.festo.com/vtem/cs/cms/10169.htm>
- [4] *F3-BP-2017-Hanak-Miroslav-Advanced PLC Programming Methods.pdf* [online]. [vid. 2021-08-16]. Dostupné z: [https://dspace.cvut.cz/bitstream/handle/10467/68419/F3-BP-2017-Hanak-Miroslav-Advanced\\_PLC\\_Programming\\_Methods.pdf](https://dspace.cvut.cz/bitstream/handle/10467/68419/F3-BP-2017-Hanak-Miroslav-Advanced_PLC_Programming_Methods.pdf)
- [5] *Structured Text - an overview | ScienceDirect Topics* [online]. [vid. 2021-08-16]. Dostupné z: <https://www.sciencedirect.com/topics/computer-science/structured-text>



**Selected article from**

**Tento dokument byl publikován ve sborníku**

**Nové metody a postupy v oblasti přístrojové  
techniky, automatického řízení a informatiky 2021  
New Methods and Practices in the Instrumentation,  
Automatic Control and Informatics 2021  
15. 9. – 17. 9. 2021, Žatec**

**ISBN 978-80-01-06889-2**

Web page of the original document:

<http://iat.fs.cvut.cz/nmp/2021.pdf>

Obsah čísla/individual articles:

<http://iat.fs.cvut.cz/nmp/2021/>

Ústav přístrojové a řídicí techniky, FS ČVUT v Praze, Technická 4, Praha 6