



AGH

**Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie**

PODSTAWY ROBOTYKI

Program robota piszącego podany tekst na kartce.

Autor: **Szydlowski Artur**
Nr albumu: 28 53 51
Kierunek: Automatyka i Robotyka - Gr. 07
Wydział Inżynierii Mechanicznej
i Robotyki

Prowadzący: **dr hab. inż. Andrzej Klepka**

Kraków, 2017/18r

1. Cel projektu.



Napisanie programu w języku Melfa-Basic IV dla ramienia robota przemysłowego RV-2AJ, który będzie mu umożliwiał napisanie na kartce długopisem lub pisakiem dowolnego tekstu podanego przez użytkownika.

Czcionka musi być sparametryzowana (rozmiar, odstępy między znakami) i dodawanie kolejnych znaków lub edycja już istniejących nie powinna być trudna.

Robot musi na początku chwycić narzędzie do pisania i wskazać swój obszar roboczy, a po skończonej pracy odłożyć element na swoje miejsce.

2. Struktura programu.

Program składa się z kilku części. W pierwszej użytkownik może zdefiniować tekst, jaki robot ma napisać (aktualnie tylko duże litery A-Z bez polskich znaków, '_' oznacza nową linię, a każdy inny znak to spacja), rozmiar czcionki, odległości między znakami oraz interlinię.

Program następnie oblicza dodatkowe parametry geometryczne na podstawie wartości podanych przez użytkownika, a także wektory pozycji względnych, które ułatwiają definicję i parametryzację czcionek. Zmieniając punkty siatki, możemy wpływać na charakter i wygląd pisanych znaków – np. przechylić je, zmienić proporcje, itp. (może jednak dochodzić przy tym do zbytniego zniekształcenia i błędów, zwłaszcza w rysowaniu krzywizn).

W kolejnym etapie robot przechodzi do programu głównego, w którym inicjalizowana jest praca robota, chwytka on narzędzie i wskazuje obszar roboczy, po czym uruchamiana jest pętla wywołań odpowiednich procedur pisania kolejnych znaków tekstu. Na koniec odkładany jest pisak i serwa robota zostają wyłączone.

Ponadto robot posiada serię procedur związanych z obsługą robota, służące do zmiany pozycji kursora (punkt P4), a także procedury pisania znaków (trajektoria zawsze jest względem kursora w punkcie P4).

3. Możliwości rozwoju projektu.

Od strony programistycznej można by zastąpić serię warunkowych instrukcji sprawdzania znaku [IF MCH = ... THEN GOSUB *L_CW..] instrukcją SELECT CASE, dzięki której po odnalezieniu właściwego znaku, pozostałe możliwości nie byłyby już sprawdzane. Mogłoby to nieco przyspieszyć wykonanie programu, ale zwiększyłyby objętość linii kodu ponad 3-krotnie i subiektywnie zmniejszyło jego czytelność.

Doświadczalne zbadanie programu pozwoliłoby na eliminację ewentualnych błędów, optymalizację pracy i być może zwiększenie szybkości pisania. Ponadto warto by było zdefiniować wszystkie 255 możliwych znaków ASCII, opracować system komunikacji z robotem i przesyłania mu tekstu bez konieczności kompilacji i wgrywania nowego programu, a także rozdzielić fazę chwycenia i odkładania narzędzia, konfiguracji obszaru roboczego oraz pisania na małe podprogramy, uruchamiane wedle potrzeby przez operatora.

4. Warstwa techniczna.

Robot wymagał również stworzenia odpowiedniego narzędzia, które byłoby łatwe do chwycenia oraz zapewniało właściwy docisk pisaka do powierzchni bez względu na niewielkie różnice w miejscu uchwycenia i wysokości trzymania przez ramię robota. W tym celu zastosowałem prostopadłościenne pudełko, w którym został umieszczony wkład z długopisu wraz ze sprężyną dociskową.

5. Kod programu.

10 CLR 0 <i>'wyczyszczenie pamieci</i>	550 ' _____ PROCEDURY _____
20 ' _____ USTAWIENIA_UZYTEKOWNIKA _____	560 *L_INIT <i>'procedura inicjalizacji, wziecie pisaka</i>
30 CTXT\$ = "ARTUR_SZYDLOWSKI" <i>'tekst</i>	570 MOV P2+UP*10
40 MFS = 25 <i>'rozmiar czcionki</i>	580 HOPEN 1
50 MFK = 5 <i>'kerning</i>	590 DLY 4
60 MFI = 10 <i>'interlinia</i>	600 SPD 30
70 ' _____ INICJALIZACJA_STALYCH _____	610 MOV P2
80 MFX = MFS / 2 <i>'szerokosc liter</i>	620 HCLOSE 1 <i>'zamkniecie chwybaka</i>
90 MFY = MFS <i>'wysokosc liter</i>	630 DLY 2
100 DEF POS UP <i>'definicja podniesienia pisaka</i>	640 MOV P2+UP*20
110 UP.z = 10	650 MOV P3+UP*10 <i>'wskazanie dolnego naroznika ob. robocz.</i>
120 P4 = P1 <i>'kursor</i>	660 MVS P3+UP*2
130 DEF POS P11 <i>'definicja punktow siatki</i>	670 DLY 5
140 DEF POS P12	680 MVS P1+UP*2 <i>'wskazanie lewego naroznika (poczatku)</i>
150 P12.y = -MFX/2	690 DLY 5
160 DEF POS P13	700 RETURN
170 P13.y = -MFX	710 *L_END <i>'procedura odlozenia pisaka</i>
180 DEF POS P21	720 MOV P2+UP*10
190 P21.x = -MFY/4	730 MVS P2
200 DEF POS P23	740 HOPEN 1
210 P23.x = -MFY/4	750 DLY 1
220 P23.y = -MFX	760 MVS P2+UP*10
230 DEF POS P31	770 RETURN
240 P31.x = -MFY/2	780 *L_CPR <i>'cursor print, procedura pizaca pierwszy znak tekstu</i>
250 DEF POS P32	790 MCH = ASC (CTXT\$) <i>'pobranie wartosci pierwszego znaku</i>
260 P32.x = -MFY/2	800 M3 = LEN(CTXT\$) - 1 <i>'obliczenie nowej dlugosci wyrazu</i>
270 P32.y = -MFX/2	810 CTXT\$ = MIRROR\$ (CTXT\$) <i>'odbicie lustrzane tekstu,</i>
280 DEF POS P33	<i>z powodu bladnego dzialania RIGHT</i>
290 P33.x = -MFY/2	820 CTXT\$ = LEFT\$ (CTXT\$, M3) <i>'obciecie znaku</i>
300 P33.y = -MFX	830 CTXT\$ = MIRROR\$ (CTXT\$) <i>'powrot tekstu do pierwotnej</i>
310 DEF POS P41	<i>postaci, ale bez obcietego znaku</i>
320 P41.x = -MFY*3/4	840 <i>'WYWOLANIE ODPOWIEDNIEJ PROCEDURY PISZACEJ</i>
330 DEF POS P43	850 IF MCH = 65 THEN GOSUB *L_CWA 'A
340 P43.x = -MFY*3/4	860 IF MCH = 66 THEN GOSUB *L_CWB 'B
350 P43.y = -MFX	870 IF MCH = 67 THEN GOSUB *L_CWC 'C
360 DEF POS P51	880 IF MCH = 68 THEN GOSUB *L_CWD 'D
370 P51.x = -MFY	890 IF MCH = 69 THEN GOSUB *L_CWE 'E
380 DEF POS P52	900 IF MCH = 70 THEN GOSUB *L_CWF 'F
390 P52.x = -MFY	910 IF MCH = 71 THEN GOSUB *L_CWG 'G
400 P52.y = -MFX/2	920 IF MCH = 72 THEN GOSUB *L_CWH 'H
410 DEF POS P53	930 IF MCH = 73 THEN GOSUB *L_CWI 'I
420 P53.x = -MFY	940 IF MCH = 74 THEN GOSUB *L_CWJ 'J
430 P53.y = -MFX	950 IF MCH = 75 THEN GOSUB *L_CWK 'K
440 ' _____ PROGRAM_GLOWNY _____	960 IF MCH = 76 THEN GOSUB *L_CWL 'L
450 SERVO ON <i>'uruchomienie serwonapedow</i>	970 IF MCH = 77 THEN GOSUB *L_CWM 'M
460 GOSUB *L_INIT <i>'inicjalizacja, wziecie pisaka</i>	980 IF MCH = 78 THEN GOSUB *L_CWN 'N
470 GOSUB *L_CHM <i>'ustawienie kursora</i>	990 IF MCH = 79 THEN GOSUB *L_CWO 'O
480 M2 = LEN(CTXT\$) <i>'petla for pizaca kolejne znaki</i>	1000 IF MCH = 80 THEN GOSUB *L_CWP 'P
490 FOR M1=1 TO M2	1010 IF MCH = 82 THEN GOSUB *L_CWR 'R
500 GOSUB *L_CPR	1020 IF MCH = 83 THEN GOSUB *L_CWS 'S
510 NEXT M1	1030 IF MCH = 84 THEN GOSUB *L_CWT 'T
520 GOSUB *L_END <i>'odlozenie pisaka</i>	1040 IF MCH = 85 THEN GOSUB *L_CWU 'U
530 SERVO OFF	1050 IF MCH = 87 THEN GOSUB *L_CWW 'W
540 END <i>'koniec programu glownego</i>	1060 IF MCH = 89 THEN GOSUB *L_CWY 'Y
	1070 IF MCH = 90 THEN GOSUB *L_CWZ 'Z
	1080 IF MCH = 95 THEN GOSUB *L_CNL ' <i>_ NOWA LINIA</i>
	1090 GOSUB *L_CNC <i>'przesuniecie kursora</i>
	1100 RETURN
	1110 *L_CHM <i>'cursor home, ustawienie kursora na pocztku</i>
	1120 P4 = P1
	1130 RETURN

1140 *L_CNC ' <i>cursor next char, procedura przesunięcia kursora</i> 1150 P4.y = P4.y - MFX - MFK 1160 IF P4.y < P3.y THEN GOSUB *L_CNL 1170 RETURN 1180 *L_CNL ' <i>cursor next line, procedura następnej linii</i> 1190 P4.y = P1.y 1200 P4.x = P4.x - MFX - MFK 1210 IF P4.x < P3.x THEN GOSUB *L_CHM 1220 RETURN 1230 *L_CPL ' <i>cursor previous line, powrót do poprzedniej linii</i> 1240 P4.y = P1.y 1250 P4.x = P4.x + MFX + MFK 1260 IF P4.x > P1.x THEN GOSUB *L_CHM 1270 RETURN 1280 ' ____ DEFINICJE_LITER ____ 1290 *L_CWA ' <i>procedura pisanía znaku A</i> 1300 MOV P4+P51+UP 1310 MVS P4+P51 1320 MVS P4+P12 1330 MVS P4+P53 1340 MOV P4+P53+UP 1350 MOV P4+P31+UP 1360 MVS P4+(P31+P32)/2 1370 MVS P4+(P32+P33)/2 1380 MOV P4+P33+UP 1390 RETURN 1400 *L_CWB ' <i>procedura pisanía znaku B</i> 1410 MOV P4+P11+UP 1420 MVS P4+P11 1430 MVS P4+P51 1440 MVS P4+P52 1450 MVR P4+P52,P4+P43,P4+P32 1460 MVS P4+P31 1470 MVS P4+P32 1480 MVR P4+P32,P4+P23,P4+P12 1490 MVS P4+P11 1500 MVS P4+P11+UP 1510 RETURN 1520 *L_CWC ' <i>procedura pisanía znaku C</i> 1530 MOV P4+P13+UP 1540 MVS P4+P13 1550 MVR P4+P13,P4+P31,P4+P53 1560 MVS P4+P53+UP 1570 RETURN 1580 *L_CWD ' <i>procedura pisanía znaku D</i> 1590 MOV P4+P11+UP 1600 MVS P4+P11 1610 MVS P4+P51 1620 MVR P4+P51,P4+P33,P4+P11 1630 MVS P4+P11+UP 1640 RETURN 1650 *L_CWE ' <i>procedura pisanía znaku E</i> 1660 MOV P4+P13+UP 1670 MVS P4+P13 1680 MVS P4+P11 1690 MVS P4+P51 1700 MVS P4+P53	1710 MVS P4+P53+UP 1720 MOV P4+P31+UP 1730 MVS P4+P31 1740 MVS P4+P32 1750 MVS P4+P33+UP 1760 RETURN 1770 *L_CWF ' <i>procedura pisanía znaku F</i> 1780 MOV P4+P13+UP 1790 MVS P4+P13 1800 MVS P4+P11 1810 MVS P4+P51 1820 MVS P4+P51+UP 1830 MOV P4+P31+UP 1840 MVS P4+P31 1850 MVS P4+P32 1860 MVS P4+P33+UP 1870 RETURN 1880 *L_CWG ' <i>procedura pisanía znaku G</i> 1890 MOV P4+P13+UP 1900 MVS P4+P13 1910 MVR P4+P13,P4+P31,P4+P53 1920 MVS P4+P33 1930 MVS P4+P32 1940 MVS P4+P32+UP 1950 RETURN 1960 *L_CWH ' <i>procedura pisanía znaku H</i> 1970 MOV P4+P11+UP 1980 MVS P4+P11 1990 MVS P4+P51 2000 MVS P4+P51+UP 2010 MOV P4+P31+UP 2020 MVS P4+P31 2030 MVS P4+P33 2040 MVS P4+P33+UP 2050 MOV P4+P13+UP 2060 MVS P4+P13 2070 MVS P4+P53 2080 MVS P4+P53+UP 2090 RETURN 2100 *L_CWI ' <i>procedura pisanía znaku I</i> 2110 MOV P4+P52+UP 2120 MVS P4+P52 2130 MVS P4+P12 2140 MVS P4+P12+UP 2150 RETURN 2160 *L_CWJ ' <i>procedura pisanía znaku J</i> 2170 MOV P4+P13+UP 2180 MVS P4+P13 2190 MVS P4+P33 2200 MVR P4+P43,P4+P52,P4+P41 2210 MVS P4+P41+UP 2220 RETURN 2230 *L_CWK ' <i>procedura pisanía znaku K</i> 2240 MOV P4+P11+UP 2250 MVS P4+P11 2260 MVS P4+P51 2270 MVS P4+P31 2280 MVS P4+P13 2290 MVS P4+P13+UP
--	--

2300 MOV P4+P31+UP 2310 MVS P4+P31 2320 MVS P4+P53 2330 MVS P4+P53+UP 2340 RETURN 2350 *L_CWL <i>'procedura pisanian znaku L</i> 2360 MOV P4+P11+UP 2370 MVS P4+P11 2380 MVS P4+P51 2390 MVS P4+P53 2400 MVS P4+P53+UP 2410 RETURN 2420 *L_CWM <i>'procedura pisanian znaku M</i> 2430 IF P4.y-MFX-MFK < P3.y THEN GOSUB *L_CNC 2440 MOV P4+P51+UP 2450 MVS P4+P51 2460 MVS P4+P12 2470 P5 = P4+P43 2480 P5.y = P5.y - MFK/2 2490 MVS P5 2500 GOSUB *L_CNC 2510 MVS P4+P12 2520 MVS P4+P53 2530 MVS P4+P53+UP 2540 RETURN 2550 *L_CWN <i>'procedura pisanian znaku N</i> 2560 MOV P4+P51+UP 2570 MVS P4+P51 2580 MVS P4+P11 2590 MVS P4+P53 2600 MVS P4+P13 2610 MVS P4+P13+UP 2620 RETURN 2630 *L_CWO <i>'procedura pisanian znaku O</i> 2640 MOV P4+P21+UP 2650 MVS P4+P21 2660 MVR P4+P21,P4+P12,P4+P23 2670 MVS P4+P43 2680 MVR P4+P43,P4+P52,P4+P41 2690 MVS P4+P21 2700 MVS P4+P21+UP 2710 RETURN 2720 *L_CWP <i>'procedura pisanian znaku P</i> 2730 MOV P4+P51+UP 2740 MVS P4+P51 2750 MVS P4+P11 2760 MVS P4+P12 2770 MVR P4+P12,P4+P23,P4+P32 2780 MVS P4+P31 2790 MVS P4+P31+UP 2800 RETURN 2810 *L_CWR <i>'procedura pisanian znaku R</i> 2820 MOV P4+P51+UP 2830 MVS P4+P51 2840 MVS P4+P11 2850 MVS P4+P12 2860 MVR P4+P12,P4+P23,P4+P32 2870 MVS P4+P31 2880 MVS P4+P53 2890 MVS P4+P53+UP 2900 RETURN	2910 *L_CWS <i>'procedura pisanian znaku S</i> 2920 MOV P4+P13+UP 2930 MVS P4+P13 2940 MVS P4+P12 2950 MVR P4+P12,P4+P21,P4+P32 2960 MVR P4+P32,P4+P43,P4+P52 2970 MVS P4+P51 2980 MVS P4+P51+UP 2990 RETURN 3000 *L_CWT <i>'procedura pisanian znaku T</i> 3010 MOV P4+P52+UP 3020 MVS P4+P52 3030 MVS P4+P12 3040 MVS P4+P11+UP 3050 MVS P4+P11 3060 MVS P4+P13 3070 MVS P4+P13+UP 3080 RETURN 3090 *L_CWU <i>'procedura pisanian znaku U</i> 3100 MOV P4+P11+UP 3110 MVS P4+P11 3120 MVS P4+P41 3130 MVR P4+P41,P4+P52,P4+P43 3140 MVS P4+P13 3150 MVS P4+P13+UP 3160 RETURN 3170 *L_CWW <i>'procedura pisanian znaku W</i> 3180 IF P4.y-MFX-MFK < P3.y THEN GOSUB *L_CNC 3190 MOV P4+P11+UP 3200 MVS P4+P11 3210 MVS P4+P52 3220 P5 = P4+P23 3230 P5.y = P5.y - MFK/2 3240 MVS P5 3250 GOSUB *L_CNC 3260 MVS P4+P52 3270 MVS P4+P13 3280 MVS P4+P13+UP 3290 RETURN 3300 *L_CWY <i>'procedura pisanian znaku Y</i> 3310 MOV P4+P11+UP 3320 MVS P4+P11 3330 MVS P4+P32 3340 MVS P4+P52 3350 MVS P4+P52+UP 3360 MOV P4+P32+UP 3370 MVS P4+P32 3380 MVS P4+P13 3390 MVS P4+P13+UP 3400 RETURN 3410 *L_CWZ <i>'procedura pisanian znaku Z</i> 3420 MOV P4+P11+UP 3430 MVS P4+P11 3440 MVS P4+P13 3450 MVS P4+P51 3460 MVS P4+P53 3470 MVS P4+P53+UP 3480 RETURN
---	---