LAMPIRAN A

Listing Program

Di bawah ini adalah Listing Program pada AT89C51

```
DELAY_1MS:   MOV R4,#198       ; 198us
 ULANG3: MOV R2,#05            ; 198 X 5 = 990us untuk servo gerak ke arah kiri berlawanan jarum jam
 ULANG2:  DJNZ R3,ULANG2
       DJNZ R4,ULANG3
       RET

DELAY_17MS:   MOV R4,#170      ; 170us
 ULANG3: MOV R2,#10            ; 170 X 10 = 1700us untuk refresh rate
 ULANG2:  DJNZ R3,ULANG2
       DJNZ R4,ULANG3
       RET

DELAY_1_5_MS:   MOV R4,#198    ; 198us
 ULANG3: MOV R2,#07            ; 198 X 7 = 1386us untuk servo gerak ke arah kanan searah jarum jam
 ULANG2:  DJNZ R3,ULANG2
       DJNZ R4,ULANG3
       RET
AWAL :
      IN A,P2
      AND A,#0FH
      CJNE A, #01H, BUKAN_SATU
      CALL ATAS

BUKAN_SATU : CJNE A, #02H, BUKAN_DUA
         CALL BAWAH
BUKAN_DUA  : CJNE A, #03H, BUKAN_TIGA
         CALL KANAN
BUKAN_TIGA : CJNE A, #04H, BUKAN_EMPAT
         CALL KIRI
BUKAN_EMPAT: JMP AWAL



ATAS : SETB P1.7                ;port 1.7 menggerakkan servo kanan
     CALL DELAY_1_5_MS          ;delay 1386us servo akan berputar searah jarum jam
     SETB P1.6                  ;port 1.6 menggerakkan servo kiri
     CALL DELAY_1MS             ;delay 990us servo akan berputar berlawanan arah jarum jam
     CLR P1.6                   ;clear akan mengeset port yg dituju menjadi nol semua
     CLR P1.7
     CALL DELAY_17MS            ;delay 1700us adalah refresh rate servo
     RET

KIRI : SETB P1.7
     CALL DELAY_1_5_MS
     SETB P1.6
     CALL DELAY_1_5_MS
     CLR P1.6
     CLR P1.7
     CALL DELAY_17MS
     RET
```

```
KANAN :SETB P1.7
      CALL DELAY_1MS
      SETB P1.6
      CALL DELAY_1MS
      CLR P1.6
      CLR P1.7
      CALL DELAY_17MS
      RET

BAWAH :SETB P1.7
      CALL DELAY_1MS
      SETB P1.6
      CALL DELAY_1_5_MS
      CLR P1.6
      CLR P1.7
      CALL DELAY_17MS
      RET
```

Di bawah ini adalah ActionScript untuk inisialisasi pada saat program dieksekusi. Ditulis pada frame pertama, Time Line utama.

```
fscommand("fullscreen", true);
mySound = new Sound();
start = 0;
inu = 1;
cover._visible = 1;
cove = 1;
q = new Array();
f = new Array();
g = new Array();
h = new Array();
str = new Array();
knob = 0;
fopenbaru = new Array();
output = new Array();
index = new Array();
sandi = 1;
pernah = new Array();
closepernah = new Array();
backtrack = 1;
xin = 2;
stop();
```

Di bawah ini adalah ActionScript untuk setiap tombol virtual yang ada pada area pencarian.

Tombol Sel k

```
on (release) {
        if (start == 0) {
                if (inu == 1) {
```

```
                    inuk._visible = 1;
                    st1._visible = 1;
                    mySound.attachsound("start");
                    mySound.start(0, 1);
                    start = 1;
                    startbaris = new Array();
                    startbaris[m] = 1;
                    startkolom = new Array();
                    startkolom[n] = 1;
                    str = new Array();
                    str[k] = 1;
                }
        } else if (start == 1) {
                if (inu == 1) {
                    inuk._visible = 1;
                    fi1._visible = 1;
                    mySound.attachsound("finish");
                    mySound.start(0, 1);
                    start = 2;
                    finishbaris = new Array();
                    finishbaris[m] = 1;
                    finishkolom = new Array();
                    finishkolom[n] = 1;
                    fnsh = new Array();
                    fnsh[k] = 1;
                }
        }
        if (batu == 1) {
                batuk._visible = 1;
                q[k] = 100;
                mySound.attachsound("drop");
                mySound.start(0, 1);
        }
}
```

Dibawah ini adalah ActionScript untuk instansi tombol “Iteration”

```
on (press) {
if (knob == 0 and start == 2) {
for (i=0; i<=167; i++) {
        if (finishkolom[i] == 1) {
                for (k=0; k<=167; k++) {
                        if (finishbaris[k] == 1) {
                                for (j=-14; j<=1; j++) {
                                        h[j] = 1000*(Math.abs(i-(j+14))+Math.abs(0-k));
                                }
                                for (j=0; j<=13; j++) {
                                        h[j] = 10*(Math.abs(j)+Math.abs(1-k));
                                }
                                for (j=14; j<=27; j++) {
                                        h[j] = 10*(Math.abs(i-(j-14))+Math.abs(2-k));
                                }
                                for (j=28; j<=41; j++) {
```

```
                                        h[j] = 10*(Math.abs(i-(j-28))+Math.abs(3-k));
                                }
                                for (j=42; j<=55; j++) {
                                        h[j] = 10*(Math.abs(i-(j-42))+Math.abs(4-k));
                                }
                                for (j=56; j<=69; j++) {
                                        h[j] = 10*(Math.abs(i-(j-56))+Math.abs(5-k));
                                }
                                for (j=70; j<=83; j++) {
                                        h[j] = 10*(Math.abs(i-(j-70))+Math.abs(6-k));
                                }
                                for (j=84; j<=97; j++) {
                                        h[j] = 10*(Math.abs(i-(j-84))+Math.abs(7-k));
                                }
                                for (j=98; j<=111; j++) {
                                        h[j] = 10*(Math.abs(i-(j-98))+Math.abs(8-k));
                                }
                                for (j=112; j<=125; j++) {
                                        h[j] = 10*(Math.abs(i-(j-112))+Math.abs(9-k));
                                }
                                for (j=126; j<=139; j++) {
                                        h[j] = 10*(Math.abs(i-(j-126))+Math.abs(10-k));
                                }
                                for (j=140; j<=153; j++) {
                                        h[j] = 10*(Math.abs(i-(j-140))+Math.abs(11-k));
                                }
                                for (j=154; j<=167; j++) {
                                        h[j] = 10*(Math.abs(i-(j-154))+Math.abs(12-k));
                                }
                        }
                }
        }
}
for (m=13; m<=167; m=m+14) {
        h[m] = h[m]+10000;
}
for (n=0; n<=154; n=n+14) {
        h[n] = h[n]+10000;
}
for (o=168; o<=181; o++) {
        h[o] = h[o]+10000;
}
for (p=0; p<=167; p++) {
        h[p] = h[p]+(q[p]*10);
}
for (d=0; d<=167; d++) {
        if (fnsh[d] == 1) {
                fin = d;
        }
}
knob = 1;
batu = 0;
}
h0 = h[0];
h1 = h[1];
h2 = h[2];
```

```
h3 = h[3];
h4 = h[4];
h5 = h[5];
h6 = h[6];
h7 = h[7];
h8 = h[8];

……..

h165 = h[165];
h166 = h[166];
h167 = h[167];
}
on (release) {
	if (step != fin) {
		for (j=0; j<=167; j++) {
			if (str[j] == 1) {
				mulai = j;
				pernah[mulai] = 1;
				closepernah[mulai] = 1;
				closelist = String(closelist)+" "+String(mulai);
				closecount = closecount+1;
				close = closelist.split(" ");
				if (h[j-15]<1000 and pernah[j-15] != 1) {
					openlist = String(openlist)+" "+String(j-15);
					opencount = opencount+1;
					g[j-15] = g[j]+14;
					pernah[j-15] = 1;
				}
				if (h[j-14]<1000 and pernah[j-14] != 1) {
					openlist = String(openlist)+" "+String(j-14);
					opencount = opencount+1;
					g[j-14] = g[j]+10;
					pernah[j-14] = 1;
				}
				if (h[j-13]<1000 and pernah[j-13] != 1) {
					openlist = String(openlist)+" "+String(j-13);
					opencount = opencount+1;
					g[j-13] = g[j]+14;
					pernah[j-13] = 1;
				}
				if (h[j-1]<1000 and pernah[j-1] != 1) {
					openlist = String(openlist)+" "+String(j-1);
					opencount = opencount+1;
					g[j-1] = g[j]+10;
					pernah[j-1] = 1;
				}
				if (h[j+1]<1000 and pernah[j+1] != 1) {
					openlist = String(openlist)+" "+String(j+1);
					opencount = opencount+1;
					g[j+1] = g[j]+10;
					pernah[j+1] = 1;
				}
				if (h[j+13]<1000 and pernah[j+13] != 1) {
					openlist = String(openlist)+" "+String(j+13);
					opencount = opencount+1;
```

```
                        g[j+13] = g[j]+14;
                        pernah[j+13] = 1;
                }
                if (h[j+14]<1000 and pernah[j+14] != 1) {
                        openlist = String(openlist)+" "+String(j+14);
                        opencount = opencount+1;
                        g[j+14] = g[j]+10;
                        pernah[j+14] = 1;
                }
                if (h[j+15]<1000 and pernah[j+15] != 1) {
                        openlist = String(openlist)+" "+String(j+15);
                        opencount = opencount+1;
                        g[j+15] = g[j]+14;
                        pernah[j+15] = 1;
                }
        }
        str[j] = 0;
}
open = openlist.split(" ");
delete (openbarulist);
delete (openbarucount);

a = 1;
while (a<=opencount) {
        for (b=1; b<=closecount; b++) {
                opena = open[a];
                closeb = close[b];
                if (open[a] != close[b]) {
                        sandi = sandi*1;
                } else {
                        sandi = sandi*0;
                }
        }
        if (sandi == 1) {
                openbarulist = String(openbarulist)+" "+String(open[a]);
                openbarucount = openbarucount+1;
        }
        a = a+1;
        sandi = 1;
}
openbaru = openbarulist.split(" ");
for (k=0; k<=167; k++) {
        f[k] = g[k]+h[k];
}
for (c=1; c<=openbarucount; c++) {
        fopenbaru[c] = f[openbaru[c]];
        output[c] = fopenbaru[c];
}

//sorting
r = 1;
while (r<=(openbarucount-1)) {
        for (t=r+1; t<=openbarucount; t++) {
                if (fopenbaru[t]<fopenbaru[r]) {
                        output[r] = fopenbaru[t];
                        output[t] = fopenbaru[r];
```

```
                                for (u=1; u<=openbarucount; u++) {
                                        fopenbaru[u] = output[u];
                                }
                        }
                }
                r = r+1;
}
terkecil = output[1];
for (i=openbarucount; i>=1; i--) {
        if (f[openbaru[i]] == terkecil) {
                index = i;
        }
}
owner = openbaru[index];
gfin = g[fin];
g0 = g[0];
g1 = g[1];
g2 = g[2];
g3 = g[3];
g4 = g[4];
g5 = g[5];
g6 = g[6];
g7 = g[7];
g8 = g[8];
g9 = g[9];
g10 = g[10];
g11 = g[11];
g12 = g[12];
g13 = g[13];
g14 = g[14];
g15 = g[15];
g16 = g[16];
g17 = g[17];
g18 = g[18];
g19 = g[19];
g21 = g[21];
g22 = g[22];
g23 = g[23];
g24 = g[24];
g25 = g[25];
……
g165 = g[165];
g166 = g[166];
g167 = g[167];


f0 = f[0];
f1 = f[1];
f2 = f[2];
f3 = f[3];
f4 = f[4];
f5 = f[5];
f6 = f[6];
f7 = f[7];
f8 = f[8];
f9 = f[9];
```

```
f10 = f[10];
f11 = f[11];
f12 = f[12];
f13 = f[13];
f14 = f[14];
f15 = f[15];
f16 = f[16];
f17 = f[17];
f18 = f[18];
f19 = f[19];
f20 = f[20];
f21 = f[21];
f22 = f[22];
f23 = f[23];
f24 = f[24];
f25 = f[25];
f26 = f[26];
f27 = f[27];
f28 = f[28];
f29 = f[29];
f30 = f[30];
f31 = f[31];
f32 = f[32];
f33 = f[33];
f34 = f[34];
f35 = f[35];
f36 = f[36];
f37 = f[37];
f38 = f[38];
f39 = f[39];
f40 = f[40];
f41 = f[41];
f42 = f[42];
f43 = f[43];
f44 = f[44];
f45 = f[45];
f46 = f[46];
f47 = f[47];
f48 = f[48];
f49 = f[49];
f50 = f[50];
f51 = f[51];
f52 = f[52];

……..

f165 = f[165];
f166 = f[166];
f167 = f[167];


y = owner;
step = y;
fstep = f[step];
if (step == fin) {
        mySound.attachsound("found");
```

```
            mySound.start(0, 1);
}
if (y == 1) {
            close1._visible = 1;
            str[1] = 1;
}
if (y == 2) {
            close2._visible = 1;
            str[2] = 1;
}
if (y == 3) {
            close3._visible = 1;
            str[3] = 1;
}
if (y == 4) {
            close4._visible = 1;
            str[4] = 1;
}
if (y == 5) {
            close5._visible = 1;
            str[5] = 1;
}
if (y == 6) {
            close6._visible = 1;
            str[6] = 1;
}
if (y == 7) {
            close7._visible = 1;
            str[7] = 1;
}
if (y == 8) {
            close8._visible = 1;
            str[8] = 1;
}
if (y == 9) {
            close9._visible = 1;
            str[9] = 1;
}
if (y == 10) {
            close10._visible = 1;
            str[10] = 1;
}
if (y == 11) {
            close11._visible = 1;
            str[11] = 1;
}
if (y == 12) {
            close12._visible = 1;
            str[12] = 1;
}

……..

if (y == 165) {
            close165._visible = 1;
            str[165] = 1;
```

```
                }
                if (y == 166) {
                        close166._visible = 1;
                        str[166] = 1;
                }
                mySound.attachsound("pop");
                mySound.start(0, 1);
        }
}
```

Dibawah ini adalah ActionScript untuk instansi tombol “Find Path”

```
on (press) {
        while (xin != backtrack and step == fin and knob == 1) {
                mySound.attachsound("path");
                mySound.start(0, 1);
                for (z=0; z<=167; z++) {
                        if (fnsh[z] == 1) {
                                xin = z;
                        }
                }
                for (l=0; l<=2; l++) {
                        for (r=(xin+l-15); r<=(xin+l+15); r=r+14) {
                                if (g[r] == (g[xin]-14) or g[r] == (g[xin]-10)) {
                                        if (closepernah[r] == 1) {
                                                backtrack = r;
                                                fnsh[xin] = 0;
                                        }
                                }
                        }
                }
                if (backtrack<xin) {
                        for (z=167; z>=0; z--) {
                                if (fnsh[z] == 1) {
                                        xin = z;
                                }
                        }
                        for (l=0; l<=2; l++) {
                                for (r=(xin+l-15); r<=(xin+l+15); r=r+14) {
                                        if (g[r] == (g[xin]-14) or g[r] == (g[xin]-10)) {
                                                if (closepernah[r] == 1) {
                                                        backtrack = r;
                                                        fnsh[xin] = 0;
                                                }
                                        }
                                }
                        }
                }
                k = backtrack;
                if (k == 1) {
                        inu1._visible = 1;
                        inu1._alpha = 50;
                        fnsh[1] = 1;
                }
                if (k == 2) {
                        inu2._visible = 1;
```

```
                    inu2._alpha = 50;
                    fnsh[2] = 1;
                }

                ……

                if (k == 165) {
                    inu165._visible = 1;
                    inu165._alpha = 50;
                    fnsh[165] = 1;
                }


                if (k == 166) {
                    inu166._visible = 1;
                    inu166._alpha = 50;
                    fnsh[166] = 1;
                }
                langsung = 0;
            }
}
```