LDL06F02A

LDLABO

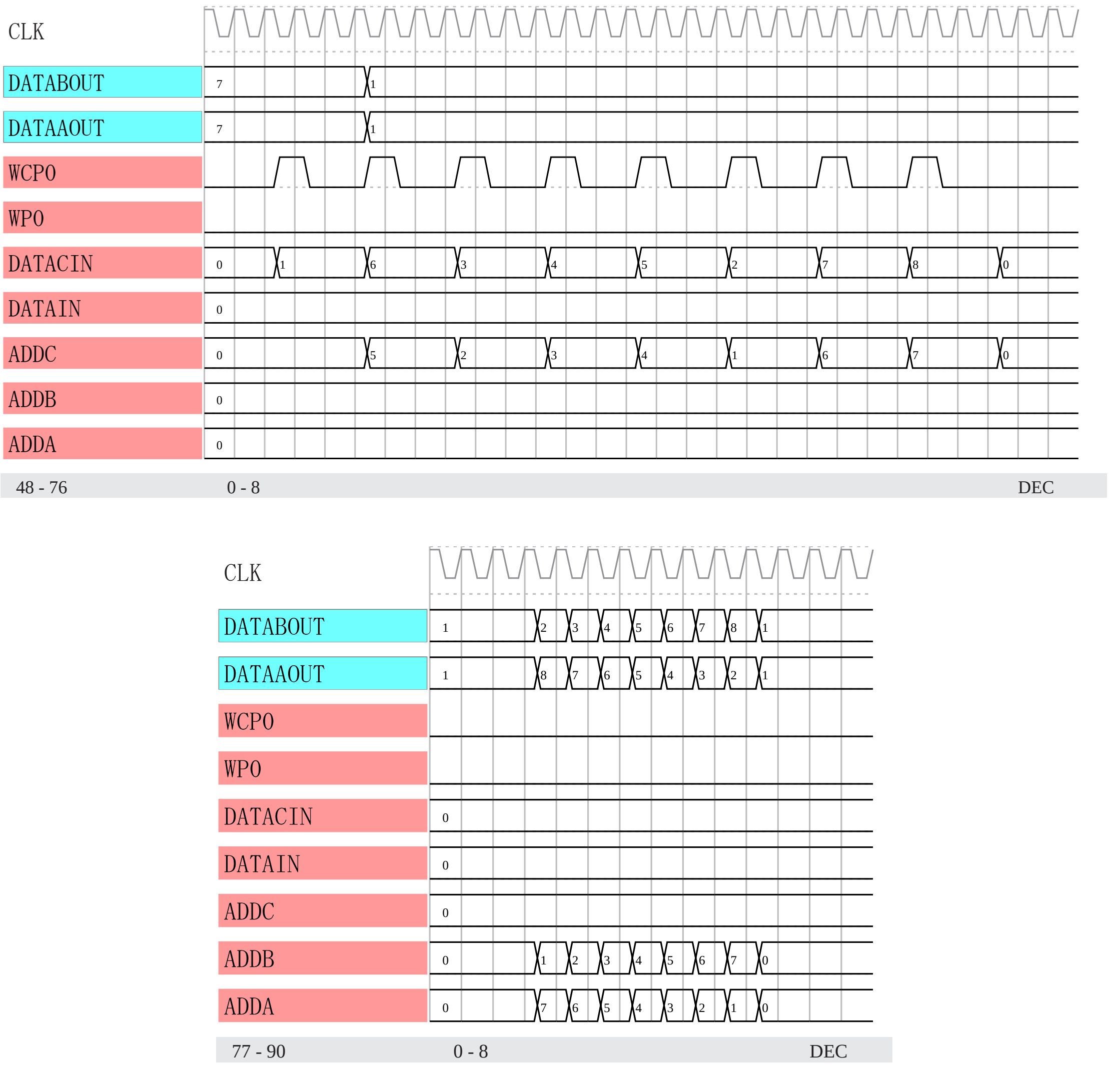
2006年6月2日

1　論理譜

```
logicname LDL06F02A

procedure MEMORY
input  RESET;
input  CS;
input  WP;
input  DATAIN[4];
output DATAOUT[4];

bitr memory[4];

   DATAOUT=memory;

   if (RESET)
      memory=0;
   else
      if (WP)
         if (CS)
            memory=DATAIN;
         else
            memory=memory;
         endif
      else
         memory=memory;
      endif
   endif
endp

entity main
input RESET;
input ADDA[3];
input ADDB[3];
input ADDC[3];
input DATAIN[4];
input DATACIN[4];
input WP;
input WCP;
output DATAAOUT[4];
output DATABOUT[4];

bitn memory0[4];
bitn memory1[4];
bitn memory2[4];
bitn memory3[4];
bitn memory4[4];
bitn memory5[4];
bitn memory6[4];
bitn memory7[4];
bitr wp[2];
bitr process[2];
bitn we;
bitn mcs[8];
bitn writedata0p[4];
bitn writedata1p[4];
bitn writedata2p[4];
bitn writedata3p[4];
bitn writedata4p[4];
bitn writedata5p[4];
bitn writedata6p[4];
bitn writedata7p[4];

   switch(ADDA)
      case 0: DATAAOUT=memory0;
      case 1: DATAAOUT=memory1;
      case 2: DATAAOUT=memory2;
      case 3: DATAAOUT=memory3;
      case 4: DATAAOUT=memory4;
      case 5: DATAAOUT=memory5;
      case 6: DATAAOUT=memory6;
      case 7: DATAAOUT=memory7;
   endswitch

   switch(ADDB)
      case 0: DATABOUT=memory0;
      case 1: DATABOUT=memory1;
      case 2: DATABOUT=memory2;
      case 3: DATABOUT=memory3;
      case 4: DATABOUT=memory4;
      case 5: DATABOUT=memory5;
      case 6: DATABOUT=memory6;
      case 7: DATABOUT=memory7;
   endswitch

   memory0=MEMORY(RESET,mcs.0,we,writedata0p);
   memory1=MEMORY(RESET,mcs.1,we,writedata1p);
   memory2=MEMORY(RESET,mcs.2,we,writedata2p);
   memory3=MEMORY(RESET,mcs.3,we,writedata3p);
   memory4=MEMORY(RESET,mcs.4,we,writedata4p);
   memory5=MEMORY(RESET,mcs.5,we,writedata5p);
   memory6=MEMORY(RESET,mcs.6,we,writedata6p);
   memory7=MEMORY(RESET,mcs.7,we,writedata7p);

   if (RESET)
      wp=0;
   else
      switch(wp)
         case 0:
            switch(WP,WCP)
               case 0,0: wp=0;
               case 1,0: wp=1;   { せり上がり書き込み }
               case 0,1: wp=2;   { 番地指定書き込み }
               case 1,1: wp=2;   {    〃    }
            endswitch
         case 1: wp=3;   { せり上がり書き込み行程選択 }
         case 2: wp=3;   { 番地指定書き込み行程選択 }
         case 3: wp=0;   { 書き込み実行 }
      endswitch
   endif

   we=wp==3;

   if (RESET)
      process=0;
   else
      switch(wp)
         case 0:  process=0;
         case 1:  process=1;
         case 2:  process=2;
         default: process=process;
      endswitch
   endif

   switch(process)
      case 1:
         mcs=0xff;
      case 2:
         switch(ADDC)
            case 0: mcs.0=1;
            case 1: mcs.1=1;
            case 2: mcs.2=1;
            case 3: mcs.3=1;
            case 4: mcs.4=1;
            case 5: mcs.5=1;
            case 6: mcs.6=1;
            case 7: mcs.7=1;
         endswitch
   endswitch

   switch(process)
      case 1:
         writedata0p=DATAIN;
         writedata1p=memory0;
         writedata2p=memory1;
         writedata3p=memory2;
         writedata4p=memory3;
         writedata5p=memory4;
         writedata6p=memory5;
         writedata7p=memory6;
      case 2:
         writedata0p=DATACIN;
         writedata1p=DATACIN;
         writedata2p=DATACIN;
         writedata3p=DATACIN;
         writedata4p=DATACIN;
         writedata5p=DATACIN;
         writedata6p=DATACIN;
         writedata7p=DATACIN;
   endswitch

ende

entity sim
output RESET;
output ADDA[3];
output ADDB[3];
output ADDC[3];
output DATAIN[4];
output DATACIN[4];
output WP;
output WCP;
output DATAAOUT[4];
output DATABOUT[4];

bitr tc[8];

   part main(RESET,ADDA,ADDB,ADDC,DATAIN,DATACIN,WP,WCP
            ,DATAAOUT,DATABOUT)

   tc=tc+1;

   if (tc<5) RESET=1; endif

   switch(tc)
      case 10: DATAIN=10; WP=1;
      case 11: DATAIN=10; WP=0;
      case 12: DATAIN=10; WP=0;

      case 13: DATAIN=11; WP=1;
      case 14: DATAIN=11; WP=0;
      case 15: DATAIN=11; WP=0;

      case 16: DATAIN=12; WP=1;
      case 17: DATAIN=12; WP=0;
      case 18: DATAIN=12; WP=0;

      case 19: DATAIN=14; WP=1;
      case 20: DATAIN=14; WP=0;
      case 21: DATAIN=14; WP=0;

      case 22: DATAIN=15; WP=1;
      case 23: DATAIN=15; WP=0;
      case 24: DATAIN=15; WP=0;

      case 25: DATAIN=9; WP=1;
      case 26: DATAIN=9; WP=0;
      case 27: DATAIN=9; WP=0;

      case 28: DATAIN=8; WP=1;
      case 29: DATAIN=8; WP=0;
      case 30: DATAIN=8; WP=0;

      case 31: DATAIN=7; WP=1;
      case 32: DATAIN=7; WP=0;
      case 33: DATAIN=7; WP=0;

      case 35: ADDB=1; ADDA=7;
      case 36: ADDB=2; ADDA=6;
      case 37: ADDB=3; ADDA=5;
      case 38: ADDB=4; ADDA=4;
      case 39: ADDB=5; ADDA=3;
      case 40: ADDB=6; ADDA=2;
      case 41: ADDB=7; ADDA=1;

      case 50: DATACIN=1; ADDC=0; WCP=1;
      case 51: DATACIN=1; ADDC=0; WCP=0;
      case 52: DATACIN=1; ADDC=0; WCP=0;

      case 53: DATACIN=6; ADDC=5; WCP=1;
      case 54: DATACIN=6; ADDC=5; WCP=0;
      case 55: DATACIN=6; ADDC=5; WCP=0;

      case 56: DATACIN=3; ADDC=2; WCP=1;
      case 57: DATACIN=3; ADDC=2; WCP=0;
      case 58: DATACIN=3; ADDC=2; WCP=0;

      case 59: DATACIN=4; ADDC=3; WCP=1;
      case 60: DATACIN=4; ADDC=3; WCP=0;
      case 61: DATACIN=4; ADDC=3; WCP=0;

      case 62: DATACIN=5; ADDC=4; WCP=1;
      case 63: DATACIN=5; ADDC=4; WCP=0;
      case 64: DATACIN=5; ADDC=4; WCP=0;

      case 65: DATACIN=2; ADDC=1; WCP=1;
      case 66: DATACIN=2; ADDC=1; WCP=0;
      case 67: DATACIN=2; ADDC=1; WCP=0;

      case 68: DATACIN=7; ADDC=6; WCP=1;
      case 69: DATACIN=7; ADDC=6; WCP=0;
      case 70: DATACIN=7; ADDC=6; WCP=0;

      case 71: DATACIN=8; ADDC=7; WCP=1;
      case 72: DATACIN=8; ADDC=7; WCP=0;
      case 73: DATACIN=8; ADDC=7; WCP=0;

      case 80: ADDB=1; ADDA=7;
      case 81: ADDB=2; ADDA=6;
      case 82: ADDB=3; ADDA=5;
      case 83: ADDB=4; ADDA=4;
      case 84: ADDB=5; ADDA=3;
      case 85: ADDB=6; ADDA=2;
      case 86: ADDB=7; ADDA=1;

   endswitch

ende

endlogic
```

2 動作

図１　動作


CLK
DATABOUT 7 1
DATAAOUT 7 1
WCP0
WP0
DATACIN 0 1 6 3 4 5 2 7 8 0
DATAIN 0
ADDC 0 5 2 3 4 1 6 7 0
ADDB 0
ADDA 0
48 - 76
0 - 8
DEC
CLK
DATABOUT 1 2 3 4 5 6 7 8 1
DATAAOUT 1 8 7 6 5 4 3 2 1
WCP0
WP0
DATACIN 0
DATAIN 0
ADDC 0
ADDB 0 1 2 3 4 5 6 7 0
ADDA 0 7 6 5 4 3 2 1 0
77 - 90
0 - 8
DEC