```
#NAME 7400,SN7400

#TEXT
Quad 2-input NAND Gate


This component contains
four NAND-Gates each with
2 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,ALS,AS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7401,SN7401

#TEXT
Quad 2-input NAND Gate
(O.C.)

This component contains
four NAND-Gates each with
2 inputs; the outputs
are OPEN COLLECTOR.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Output    Gate 2
```

```
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Input   1 Gate 4
12 : Input   2 Gate 4
13 : Output    Gate 4
14 : +5V

#FAMILY Std,ALS,H,L,LS

#PROGRAM

BEGIN
PIN[2,3,5,6,8,9,11,12] : INPUT;
PIN[1,4,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[11]&PIN[12]:=I;
   LOADMODEON;
   PIN[1,4,10,13] : LOAD LOW;
   IF PIN[1]<>LOW THEN ERROR(1);
   IF PIN[4]<>LOW THEN ERROR(1);
   IF PIN[10]<>LOW THEN ERROR(1);
   IF PIN[13]<>LOW THEN ERROR(1);
   PIN[1,4,10,13] : LOAD HIGH;
   IF PIN[1]<>(PIN[2] NAND PIN[3]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NAND PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NAND PIN[9]) THEN ERROR(1);
   IF PIN[13]<>(PIN[11] NAND PIN[12]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7402,SN7402

#TEXT
Quad 2-input NOR Gate


This component contains
four NOR-Gates each with
2 inputs.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Input   1 Gate 4
12 : Input   2 Gate 4
```

```
13 : Output    Gate 4
14 : +5V

#FAMILY Std,ALS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[2,3,5,6,8,9,11,12] : INPUT;
PIN[1,4,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[11]&PIN[12]:=I;
   IF PIN[1]<>(PIN[2] NOR PIN[3]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NOR PIN[9]) THEN ERROR(1);
   IF PIN[13]<>(PIN[11] NOR PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7403,SN7403

#TEXT
Quad 2-input NAND Gate
(O.C.)

This component contains
four NAND Gates each with
2 inputs; the outputs are
OPEN COLLECTOR.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,ALS,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
```

```
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[3,6,8,11] : LOAD HIGH;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   PIN[3,6,8,11] : LOAD LOW;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   IF PIN[11]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7404,SN7404

#TEXT
Hex Inverter

This component contains
six separate inverters.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY Std,ALS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.
```

```
#NAME 7405,SN7405

#TEXT
Hex Inverter (O.C.)

This component contains
six separate inverters;
the outputs are OPEN
COLLECTOR.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY Std,ALS,AS,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   LOADMODEON;
   PIN[2,4,6,8,10,12] : LOAD LOW;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>0 THEN ERROR(1);
   PIN[2,4,6,8,10,12] : LOAD HIGH;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 7406,SN7406

#TEXT
Hex Inverting Driver
(O.C.,30V)

This component contains
six separate inverters;
the outputs are OPEN
COLLECTOR with a maximum
output voltage of +30V.


#PIN 14
```

```
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V...+30V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   LOADMODEON;
   PIN[2,4,6,8,10,12] : LOAD LOW;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>0 THEN ERROR(1);
   PIN[2,4,6,8,10,12] : LOAD HIGH;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 7407,SN7407

#TEXT
Hex Driver (O.C.,30V)

This component contains
six separate drivers;
the outputs are OPEN
COLLECTOR with a maximum
output voltage of +30V.


#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
```

```
13 : Input     Gate 6
14 : +5V...+30V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   LOADMODEON;
   PIN[2,4,6,8,10,12] : LOAD LOW;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>0 THEN ERROR(1);
   PIN[2,4,6,8,10,12] : LOAD HIGH;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 7408,SN7408

#TEXT
Quad 2-Input AND Gate


This component contains
four AND gates each with
2 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,ALS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
```

```
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] AND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] AND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7409,SN7409

#TEXT
Quad 2-Input AND Gate
(O.C.)

This component contains
four AND gates each with
2 inputs; the outputs
are OPEN COLLECTOR.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,ALS,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[3,6,8,11] : LOAD LOW;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   IF PIN[11]<>LOW THEN ERROR(1);
   PIN[3,6,8,11] : LOAD HIGH;
   IF PIN[3]<>(PIN[1] AND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] AND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] AND PIN[13]) THEN ERROR(1);
```

```
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7410,SN7410

#TEXT
Triple 3-Input NAND-Gate


This component contains
three NAND gates each with
3 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Input   1 Gate 2
 4 : Input   2 Gate 2
 5 : Input   3 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Input   3 Gate 3
12 : Output    Gate 1
13 : Input   3 Gate 1
14 : +5V

#FAMILY Std,ALS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,13] : INPUT;
PIN[6,8,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[13]:=I;
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[12]<>NOT(PIN[1] AND PIN[2] AND PIN[13]) THEN ERROR(1);
   IF PIN[6]<>NOT(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7411,SN7411

#TEXT
Triple 3-Input AND Gate


This component contains
three AND gates each with
3 inputs.

#PIN 14
 1 : Input   1 Gate 1
```

```
 2 : Input   2 Gate 1
 3 : Input   1 Gate 2
 4 : Input   2 Gate 2
 5 : Input   3 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Input   3 Gate 3
12 : Output    Gate 1
13 : Input   3 Gate 1
14 : +5V

#FAMILY Std,ALS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,13] : INPUT;
PIN[6,8,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[13]:=I;
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[12]<>(PIN[1] AND PIN[2] AND PIN[13]) THEN ERROR(1);
   IF PIN[6]<>(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7412,SN7412

#TEXT
Triple 3-Input NAND Gate
(O.C.)

This component contains
three NAND gates each with
3 inputs; the outputs
are OPEN COLLECTOR.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Input   1 Gate 2
 4 : Input   2 Gate 2
 5 : Input   3 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Input   3 Gate 3
12 : Output    Gate 1
13 : Input   3 Gate 1
14 : +5V

#FAMILY Std,ALS,LS,S

#PROGRAM
```

```
BEGIN
PIN[1,2,3,4,5,9,10,11,13] : INPUT;
PIN[6,8,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[13]:=I;
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]:=I;
   LOADMODEON;
   PIN[12,6,8] : LOAD LOW;
   IF PIN[12]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   PIN[12,6,8] : LOAD HIGH;
   IF PIN[12]<>NOT(PIN[1] AND PIN[2] AND PIN[13]) THEN ERROR(1);
   IF PIN[6]<>NOT(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7413,SN7413

#TEXT
Dual 4-Input NAND Schmitt
Trigger

This component contains
two NAND Schmitt Trigger
gates each with 4 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : N.C.
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : N.C.
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
```

```
   IF PIN[6]<>NOT(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7414,SN7414

#TEXT
Hex Inverting
Schmitt-Trigger

This component contains
six separate inverting
Schmitt-Triggers.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 7415,SN7415

#TEXT
Triple 3-Input AND Gate
(O.C.)

This component contains
three AND gates each with
3 inputs; the outputs
are OPEN COLLECTOR.

#PIN 14
```

```
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Input   1 Gate 2
 4 : Input   2 Gate 2
 5 : Input   3 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Input   3 Gate 3
12 : Output    Gate 1
13 : Input   3 Gate 1
14 : +5V

#FAMILY AS,H,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,13] : INPUT;
PIN[6,8,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[13]:=I;
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]:=I;
   LOADMODEON;
   PIN[12,6,8] : LOAD LOW;
   IF PIN[12]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   PIN[12,6,8] : LOAD HIGH;
   IF PIN[12]<>(PIN[1] AND PIN[2] AND PIN[13]) THEN ERROR(1);
   IF PIN[6]<>(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7416,SN7416

#TEXT
Hex Inverting Driver
(O.C.,15V)

This component contains
six separate inverting
drivers; the outputs
are OPEN-COLLECTOR with
a maximum output voltage
of +15V.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
```

```
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   LOADMODEON;
   PIN[2,4,6,8,10,12] : LOAD LOW;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>0 THEN ERROR(1);
   PIN[2,4,6,8,10,12] : LOAD HIGH;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 7417,SN7417

#TEXT
Hex Driver (O.C.,15V)

This component contains
six separate drivers;
The outputs are OPEN
COLLECTOR with a maximum
output voltage of +15V.


#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V...+30V

#FAMILY Std

#PROGRAM

BEGIN
```

```
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   LOADMODEON;
   PIN[2,4,6,8,10,12] : LOAD LOW;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>0 THEN ERROR(1);
   PIN[2,4,6,8,10,12] : LOAD HIGH;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 7418,SN7418

#TEXT
Dual 4-Input NAND Schmitt
Trigger

This component contains
two NAND Schmitt Trigger
gates each with 4 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : N.C.
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : N.C.
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.
```

```
#NAME 7419,SN7419

#TEXT
Hex Inverting
Schmitt-Trigger

This component contains
six separate inverting
Schmitt Triggers.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 7420,SN7420

#TEXT
Dual 4-Input NAND-Gate


This component contains
two NAND gates each with
4 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : N.C.
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
```

```
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : N.C.
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V

#FAMILY Std,ALS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7421,SN7421

#TEXT
Dual 4-Input AND Gate

This component contains
two AND gates each with
4 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : N.C.
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : N.C.
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V

#FAMILY Std,ALS,F,H,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
```

```
   IF PIN[6]<>(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=0;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   IF PIN[8]<>(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=0;
   END;

ERROR(0);
END.

#NAME 7422,SN7422

#TEXT
Dual 4-Input NAND-Gate
(O.C.)

This component contains
two NAND gates each with
4 inputs; the outputs
are OPEN-COLLECTOR with
a maximum output voltage
of +5.5V.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : N.C.
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : N.C.
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V

#FAMILY Std,ALS,H,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[6,8] : LOAD LOW;
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   PIN[6,8] : LOAD HIGH;
   IF PIN[6]<>NOT(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7423,SN7423
```

```
#TEXT
Dual 4-Input NOR-Gate
with Strobe

This component contains
two NOR gates each with
4 inputs and strobe input.
In addition one gate has
two extension pins.



#PIN 16
 1 : Expand  2 Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Strobe    Gate 1
 5 : Input   3 Gate 1
 6 : Input   4 Gate 1
 7 : Output    Gate 1
 8 : GND
 9 : Output    Gate 2
10 : Input   1 Gate 2
11 : Input   2 Gate 2
12 : Strobe    Gate 2
13 : Input   3 Gate 2
14 : Input   4 Gate 2
15 : -Expand 1 Gate 1
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,4,5,6,10,11,12,13,14] : INPUT;
PIN[7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[5]&PIN[6]:=I;
   PIN[10]&PIN[11]&PIN[13]&PIN[14]:=I;
   PIN[4,12]:=HIGH;
   IF (PIN[7]<>NOT(PIN[2] OR PIN[3] OR PIN[5] OR PIN[6])) THEN ERROR(1);
   IF PIN[9]<>NOT(PIN[10] OR PIN[11] OR PIN[13] OR PIN[14]) THEN ERROR(1);
   PIN[4,12]:=LOW;
   IF PIN[7]<>HIGH THEN ERROR(1);
   IF PIN[9]<>HIGH THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7424,SN7424

#TEXT
Quad 2-Input NAND Schmitt
Trigger

This component contains
four NAND Schmitt Triggers
each with 2 inputs.

#PIN 14
```

```
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7425,SN7425

#TEXT
Dual 4-Input NOR-Gate
with Strobe

This component contains
two NOR gates each with
4 inputs plus strobe
input.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Strobe    Gate 1
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : Strobe    Gate 2
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V
```

```
#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   PIN[3,11]:=HIGH;
   IF PIN[6]<>NOT(PIN[1] OR PIN[2] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] OR PIN[10] OR PIN[12] OR PIN[13]) THEN ERROR(1);
   PIN[3,11]:=LOW;
   IF PIN[6]<>HIGH THEN ERROR(1);
   IF PIN[8]<>HIGH THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7426,SN7426

#TEXT
Quad 2-Input NAND-Gate
(O.C.,15V)

This component contains
four NAND gates each with
2 outputs; the outputs
are OPEN COLLECTOR with
a maximum output voltage
of +15V.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
```

```
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[3,6,8,11] : LOAD LOW;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   IF PIN[11]<>LOW THEN ERROR(1);
   PIN[3,6,8,11] : LOAD HIGH;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7427,SN7427

#TEXT
Triple 3-Input NOR-Gate


This component contains
three NAND gates each with
3 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Input   1 Gate 2
 4 : Input   2 Gate 2
 5 : Input   3 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Input   3 Gate 3
12 : Output    Gate 1
13 : Input   3 Gate 1
14 : +5V

#FAMILY Std,ALS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,13] : INPUT;
PIN[6,8,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[13]:=I;
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[12]<>NOT(PIN[1] OR PIN[2] OR PIN[13]) THEN ERROR(1);
   IF PIN[6]<>NOT(PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] OR PIN[10] OR PIN[11]) THEN ERROR(1);
   END;

ERROR(0);
```

```
END.

#NAME 7428,SN7428

#TEXT
Quad 2-Input NOR-
Power Gate

This component contains
four NOR gates each with
2 inputs and high
output fanouts.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Input   1 Gate 4
12 : Input   2 Gate 4
13 : Output    Gate 4
14 : +5V

#FAMILY Std,ALS,LS

#PROGRAM

BEGIN
PIN[2,3,5,6,8,9,11,12] : INPUT;
PIN[1,4,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[11]&PIN[12]:=I;
   IF PIN[1]<>(PIN[2] NOR PIN[3]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NOR PIN[9]) THEN ERROR(1);
   IF PIN[13]<>(PIN[11] NOR PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7430,SN7430

#TEXT
8-Input NAND-Gate

This component contains
a NAND gate with 8
inputs.

#PIN 14
 1 : Input
 2 : Input
 3 : Input
```

```
 4 : Input
 5 : Input
 6 : Output
 7 : GND
 8 : Output
 9 : N.C.
10 : N.C.
11 : Input
12 : Input
13 : N.C.
14 : +5V

#FAMILY Std,ALS,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,11,12] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[1,2,3,4,5,6,11,12]:=LOW;
IF PIN[8]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[11]&PIN[12]:=1 SHL I;
   IF PIN[8]<>HIGH THEN ERROR(1);
   END;

PIN[1,2,3,4,5,6,11,12]:=HIGH;
IF PIN[8]<>LOW THEN ERROR(1);

ERROR(0);
END.

#NAME 7432,SN7432

#TEXT
Quad 2-Input OR Gate


This component contains
four OR gates each with
2 inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,ALS,F,L,LS,S

#PROGRAM
```

```
BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] OR PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] OR PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] OR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7433,SN7433

#TEXT
Quad 2-Input NOR Power
Gate (O.C.)

This component contains
four NOR gates each with
2 inputs. The outputs
are OPEN-COLLECTOR and
have high output fanouts.


#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Input   1 Gate 4
12 : Input   2 Gate 4
13 : Output    Gate 4
14 : +5V

#FAMILY Std,ALS,LS

#PROGRAM

BEGIN
PIN[2,3,5,6,8,9,11,12] : INPUT;
PIN[1,4,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[11]&PIN[12]:=I;
   LOADMODEON;
```

```
   PIN[1,4,10,13] : LOAD LOW;
   IF PIN[1]<>LOW THEN ERROR(1);
   IF PIN[4]<>LOW THEN ERROR(1);
   IF PIN[10]<>LOW THEN ERROR(1);
   IF PIN[13]<>LOW THEN ERROR(1);
   PIN[1,4,10,13] : LOAD HIGH;
   IF PIN[1]<>(PIN[2] NOR PIN[3]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NOR PIN[9]) THEN ERROR(1);
   IF PIN[13]<>(PIN[11] NOR PIN[12]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7437,SN7437

#TEXT
Quad 2-Input NAND
Power Gate

This component contains
four NAND gates each with
2 inputs and high output
fan out.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,ALS,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.
```

```
#NAME 7438,SN7438

#TEXT
Quad 2-Input NAND Power
Gate (O.C.)

This component contains
four NAND gates each with
2 inputs. The outputs
are OPEN-COLLECTOR and
have high output fan-
out.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY Std,ALS,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[3,6,8,11] : LOAD LOW;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   IF PIN[11]<>LOW THEN ERROR(1);
   PIN[3,6,8,11] : LOAD HIGH;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7439,SN7439

#TEXT
Quad 2-Input NAND Power
```

```
Gate (O.C.)

This component contains
four NAND gates each with
2 inputs. The outputs
are OPEN-COLLECTOR and
have high output fan-
outs.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Input   1 Gate 4
12 : Input   2 Gate 4
13 : Output    Gate 4
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,5,6,8,9,11,12] : INPUT;
PIN[1,4,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[11]&PIN[12]:=I;
   LOADMODEON;
   PIN[1,4,10,13] : LOAD LOW;
   IF PIN[1]<>LOW THEN ERROR(1);
   IF PIN[4]<>LOW THEN ERROR(1);
   IF PIN[10]<>LOW THEN ERROR(1);
   IF PIN[13]<>LOW THEN ERROR(1);
   PIN[1,4,10,13] : LOAD HIGH;
   IF PIN[1]<>(PIN[2] NAND PIN[3]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NAND PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NAND PIN[9]) THEN ERROR(1);
   IF PIN[13]<>(PIN[11] NAND PIN[12]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7440,SN7440

#TEXT
Dual 4-Input NAND
Power Gate

This component contains
two NAND gates each with
4 inputs and high
```

```
output fanouts.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : N.C.
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : N.C.
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V

#FAMILY Std,ALS,H,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7441,SN7441

#TEXT
BCD-to-Decimal-Decoder
(70V, O.C.)

This component converts
a Standard-BCD-Code
with 4 Bits to a decimal
number from 0 to 9.
The outputs are OPEN-
COLLECTOR with a maximum
output voltage of 70V.


#PIN 16
 1 : Output  8
 2 : Output  9
 3 : Input   A
 4 : Input   D
 5 : +5V
 6 : Input   B
 7 : Input   C
 8 : Output  2
 9 : Output  3
10 : Output  7
11 : Output  6
12 : GND
```

```
13 : Output  4
14 : Output  5
15 : Output  1
16 : Output  0

#FAMILY Std

#PROGRAM

BEGIN
PIN[3,4,6,7] : INPUT;
PIN[1,2,8,9,10,11,13,14,15,16] : OUTPUT;
PIN[12] : GND;
PIN [5] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[7]&PIN[6]&PIN[3]:=I;
   LOADMODEON;
   PIN[1,2,8,9,10,11,13,14,15,16] : LOAD LOW;
   IF (PIN[2]&PIN[1]&PIN[10]&PIN[11]&PIN[14]
      &PIN[13]&PIN[9]&PIN[8]&PIN[15]&PIN[16])
      <>0 THEN ERROR(1);
   PIN[1,2,8,9,10,11,13,14,15,16] : LOAD HIGH;
   IF (NOT(PIN[2])&NOT(PIN[1])&NOT(PIN[10])&NOT(PIN[11])&NOT(PIN[14])
      &NOT(PIN[13])&NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[15])&NOT(PIN[16]))
      <>(1 SHL I) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7442,SN7442

#TEXT
BCD-to-Decimal-Decoder

This component converts
a Standard-BCD-Code to
a decimal number from
0 to 9.

#PIN 16
 1 : Output  0
 2 : Output  1
 3 : Output  2
 4 : Output  3
 5 : Output  4
 6 : Output  5
 7 : Output  6
 8 : GND
 9 : Output  7
10 : Output  8
11 : Output  9
12 : Input   D
13 : Input   C
14 : Input   B
15 : Input   A
16 : +5V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[12,13,14,15] : INPUT;
```

```
PIN[1,2,3,4,5,6,7,9,10,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]:=I;
   IF (NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[6])&
      NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2])&NOT(PIN[1]))
      <>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7443,SN7443

#TEXT
Excess 3-to-Decimal-
Decoder

This component
converts an Excess-3-
Code to a decimal number
between 0 and 9.

#PIN 16
 1 : Output  0
 2 : Output  1
 3 : Output  2
 4 : Output  3
 5 : Output  4
 6 : Output  5
 7 : Output  6
 8 : GND
 9 : Output  7
10 : Output  8
11 : Output  9
12 : Input   D
13 : Input   C
14 : Input   B
15 : Input   A
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[12,13,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9,10,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]:=I+3;
   IF (NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[6])&
      NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2])&NOT(PIN[1]))
      <>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.
•
```

```
#NAME 7444,SN7444

#TEXT
Excess-3-Gray-to-
Decimal Decoder

This component
converts an Excess-3-
Gray-Code to a decimal
number from 0 to 9.

#PIN 16
 1 : Output  0
 2 : Output  1
 3 : Output  2
 4 : Output  3
 5 : Output  4
 6 : Output  5
 7 : Output  6
 8 : GND
 9 : Output  7
10 : Output  8
11 : Output  9
12 : Input   D
13 : Input   C
14 : Input   B
15 : Input   A
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[12,13,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9,10,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]:=((I+0 AND 2) SHR 1)+((I+5 AND 4) SHR 1)+
                                    ((I+7 AND 8) SHR 1)+(I+3 AND 8);
   IF (NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2])&NOT(PIN[1]))
       <>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7445,SN7445

#TEXT
BCD-to-Decimal-Decoder
(O.C., 30V, 80mA)

This component converts
a Standard-BCD-Code
with 4 Bits to a decimal
number from 0 to 9.
The outputs are OPEN-
COLLECTOR with a maximum
output voltage from 30V
and a maximum output power
of 30mA.
```

```
#PIN 16
 1 : Output  0
 2 : Output  1
 3 : Output  2
 4 : Output  3
 5 : Output  4
 6 : Output  5
 7 : Output  6
 8 : GND
 9 : Output  7
10 : Output  8
11 : Output  9
12 : Input   D
13 : Input   C
14 : Input   B
15 : Input   A
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[12,13,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9,10,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]:=I;
   LOADMODEON;
   PIN[1,2,3,4,5,6,7,9,10,11] : LOAD LOW;
   IF (PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1])
       <>0 THEN ERROR(1);
   PIN[1,2,3,4,5,6,7,9,10,11] : LOAD HIGH;
   IF (NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2])&NOT(PIN[1]))
       <>(1 SHL I) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7446,SN7446

#TEXT
BCD-to-7-Segment-Decoder/
Display Driver (O.C., 30V)

This component converts
BCD-input data to control
data for 7-Segment-
displays. The outputs
are OPEN-COLLECTOR and
supply +30V maximum.

#PIN 16
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input   D
```

```
 7 : BCD-Input   A
 8 : GND
 9 : 7-Seg.-Output  -e
10 : 7-Seg.-Output  -d
11 : 7-Seg.-Output  -c
12 : 7-Seg.-Output  -b
13 : 7-Seg.-Output  -a
14 : 7-Seg.-Output  -g
15 : 7-Seg.-Output  -f
16 : +5V...+30V

#FAMILY Std,L

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   LOADMODEON;
   PIN[9,10,11,12,13,14,15] : LOAD LOW;
   IF (PIN[14]&PIN[15]&PIN[9]&PIN[10]&
       PIN[11]&PIN[12]&PIN[13])<>0 THEN ERROR(1);
   PIN[9,10,11,12,13,14,15] : LOAD HIGH;
   D:=NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[9])&NOT(PIN[10])&
      NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13]);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7447,SN7447

#TEXT
BCD-to-7-Segment-Decoder/
Display Driver (O.C., 15V)

This component converts
BCD-input data to control
data for 7-Segment-
displays. The outputs
are OPEN-COLLECTOR and
supply +15V maximum.

#PIN 16
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
```

```
 6 : BCD-Input   D
 7 : BCD-Input   A
 8 : GND
 9 : 7-Seg.-Output  -e
10 : 7-Seg.-Output  -d
11 : 7-Seg.-Output  -c
12 : 7-Seg.-Output  -b
13 : 7-Seg.-Output  -a
14 : 7-Seg.-Output  -g
15 : 7-Seg.-Output  -f
16 : +5V...+15V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   LOADMODEON;
   PIN[9,10,11,12,13,14,15] : LOAD LOW;
   IF (PIN[14]&PIN[15]&PIN[9]&PIN[10]&
       PIN[11]&PIN[12]&PIN[13])<>0 THEN ERROR(1);
   PIN[9,10,11,12,13,14,15] : LOAD HIGH;
   D:=NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[9])&NOT(PIN[10])&
      NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13]);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7448,SN7448

#TEXT
BCD-to-7-Segment-Decoder/
Display Driver

This component converts
BCD-input data to control
data for 7-Segment-
displays.

#PIN 16
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input   D
```

```
 7 : BCD-Input   A
 8 : GND
 9 : 7-Seg.-Output  -e
10 : 7-Seg.-Output  -d
11 : 7-Seg.-Output  -c
12 : 7-Seg.-Output  -b
13 : 7-Seg.-Output  -a
14 : 7-Seg.-Output  -g
15 : 7-Seg.-Output  -f
16 : +5V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7449,SN7449

#TEXT
BCD-to-7-Segment-Decoder/
Display Driver (O.C.,5.5V)

This component converts
BCD-input data to control
data for 7-Segment-
displays. The outputs
are OPEN-COLLECTOR and
supply +5.5V maximum.

#PIN 14
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : -BI
 4 : BCD-Input   D
 5 : BCD-Input   A
 6 : 7-Seg.-Output  -e
 7 : GND
 8 : 7-Seg.-Output  -d
 9 : 7-Seg.-Output  -c
10 : 7-Seg.-Output  -b
11 : 7-Seg.-Output  -a
12 : 7-Seg.-Output  -g
```

```
13 : 7-Seg.-Output  -f
14 : +5V...+5.5V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[6,8,9,10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[2]&PIN[1]&PIN[5]:=I;
   LOADMODEON;
   PIN[12,13,6,8,9,10,11] : LOAD LOW;
   IF (PIN[12]&PIN[13]&PIN[6]&PIN[8]&
       PIN[9]&PIN[10]&PIN[11])<>0 THEN ERROR(1);
   PIN[12,13,6,8,9,10,11] : LOAD HIGH;
   D:=PIN[12]&PIN[13]&PIN[6]&PIN[8]&PIN[9]&PIN[10]&PIN[11];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7450,SN7450

#TEXT
Dual 2x2-Input AND-NOR-
Gate

This component contains
two AND-NOR gates each
with 2x2 inputs. In
addition one gate has
extension inputs.

#PIN 14
 1 : Input    A Gate 1
 2 : Input    A Gate 2
 3 : Input    B Gate 2
 4 : Input    C Gate 2
 5 : Input    D Gate 2
 6 : Output   Q Gate 2
 7 : GND
 8 : Output   Q Gate 1
 9 : Input    C Gate 1
10 : Input    D Gate 1
11 : Expand  N2 Gate 1
12 : Expand -N1 Gate 1
13 : Input    B Gate 1
14 : +5V
```

```
#FAMILY Std,H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[11]&PIN[12]:=1;
FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[13]&PIN[10]&PIN[9]:=I;
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   IF PIN[8]<>((PIN[1] AND PIN[13]) NOR (PIN[10] AND PIN[9])) THEN ERROR(1);
   IF PIN[6]<>((PIN[2] AND PIN[3]) NOR (PIN[4] AND PIN[5])) THEN ERROR(1);
   END;
PIN[11]&PIN[12]:=0;
FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[13]&PIN[10]&PIN[9]:=I;
   IF PIN[8]<>LOW THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7451,SN7451

#TEXT
Dual 2x2-Input AND-NOR-
Gate

This component contains
two AND-NOR gates each
with 2x2 inputs.

Observe the difference
between this component
and the L- and LS-
versions!

#PIN 14
 1 : Input    A Gate 1
 2 : Input    A Gate 2
 3 : Input    B Gate 2
 4 : Input    C Gate 2
 5 : Input    D Gate 2
 6 : Output   Q Gate 2
 7 : GND
 8 : Output   Q Gate 1
 9 : Input    C Gate 1
10 : Input    D Gate 1
11 : N.C.
12 : N.C.
13 : Input    B Gate 1
14 : +5V

#FAMILY Std,H,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
```

```
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[13]&PIN[10]&PIN[9]:=I;
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   IF PIN[8]<>((PIN[1] AND PIN[13]) NOR (PIN[10] AND PIN[9])) THEN ERROR(1);
   IF PIN[6]<>((PIN[2] AND PIN[3]) NOR (PIN[4] AND PIN[5])) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74L51,SN74L51,74LS51,SN74LS51

#TEXT
2x2-Input AND/NOR-Gate
and a 2x3-Input AND/NOR-
Gate

This component contains
an AND-NOR gate with 2x2
inputs and an AND-NOR
gate with 2x3 inputs.

Observe the difference
between this component and
the standard version!

#PIN 14
 1 : Input    A Gate 1
 2 : Input    A Gate 2
 3 : Input    B Gate 2
 4 : Input    C Gate 2
 5 : Input    D Gate 2
 6 : Output   Q Gate 2
 7 : GND
 8 : Output   Q Gate 1
 9 : Input    F Gate 1
10 : Input    E Gate 1
11 : Input    D Gate 1
12 : Input    C Gate 1
13 : Input    B Gate 1
14 : +5V

#FAMILY L,LS


#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 63 DO
   BEGIN
   PIN[1]&PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9]:=I;
   IF PIN[8]<>((PIN[1] AND PIN[13] AND PIN[12])
           NOR (PIN[11] AND PIN[10] AND PIN[9])) THEN ERROR(1);
   END;
FOR I:=0 TO 15 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   IF PIN[6]<>((PIN[2] AND PIN[3]) NOR (PIN[4] AND PIN[5])) THEN ERROR(1);
   END;
```

```
ERROR(0);
END.

#NAME 7452,SN7452

#TEXT
3x2 and 1x3 expandable
AND/OR Gate


This component contains
an extendable OR gate
with 4 inputs, which are
internally connected to
three AND gates each with
3 inputs.

#PIN 14
 1 : Input   A
 2 : Input   B
 3 : Input   C
 4 : Input   D
 5 : Input   E
 6 : N.C.
 7 : GND
 8 : Output  Q
 9 : Expand  N
10 : Input   I
11 : Input   H
12 : Input   G
13 : Input   F
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,10,11,12,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,5,10,11,12,13]:=LOW;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   IF PIN[8]<>(PIN[1] AND PIN[2]) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   IF PIN[8]<>(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[10]&PIN[11]:=I;
   IF PIN[8]<>(PIN[10] AND PIN[11]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[12]&PIN[13]:=I;
   IF PIN[8]<>(PIN[12] AND PIN[13]) THEN ERROR(1);
```

```
   END;

ERROR(0);
END.

#NAME 7453,SN7453

#TEXT
4x2 or 3x2 and 1x3-Input
Expandable AND/NOR Gate

This component contains
an expandable NOR
gate with 4 inputs,
internally connected
to four AND gates each
having 2 inputs.

Observe the difference
between this component
and the H- version!

#PIN 14
 1 : Input   A
 2 : Input   C
 3 : Input   D
 4 : Input   E
 5 : Input   F
 6 : N.C.
 7 : GND
 8 : Output  Q
 9 : Input   G
10 : Input   H
11 : Expand  N
12 : Expand -N
13 : Input   B
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,5,9,10,13]:=LOW;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[13]:=I;
   IF PIN[8]<>NOT(PIN[1] AND PIN[13]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   IF PIN[8]<>NOT(PIN[2] AND PIN[3]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[4]&PIN[5]:=I;
   IF PIN[8]<>NOT(PIN[4] AND PIN[5]) THEN ERROR(1);
   END;
```

```
FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   IF PIN[8]<>NOT(PIN[9] AND PIN[10]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74H53,SN74H53

#TEXT
4x2 or 3x2 and 1x3
expandable AND/NOR Gate

This component contains
an extendable NOR gate
with 4 inputs, internally
connected to four AND
gates each having 2
inputs.

Observe the difference
between this component
and the standard version!

#PIN 14
 1 : Input   A
 2 : Input   C
 3 : Input   D
 4 : Input   E
 5 : Input   F
 6 : Input   G
 7 : GND
 8 : Output  Q
 9 : Input   H
10 : Input   I
11 : Expand  N
12 : Expand -N
13 : Input   B
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,9,10,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,5,9,10,13]:=LOW;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[13]:=I;
   IF PIN[8]<>NOT(PIN[1] AND PIN[13]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   IF PIN[8]<>NOT(PIN[2] AND PIN[3]) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
```

```
   PIN[4]&PIN[5]&PIN[6]:=I;
   IF PIN[8]<>NOT(PIN[4] AND PIN[5] AND PIN[6]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   IF PIN[8]<>NOT(PIN[9] AND PIN[10]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7454,SN7454,7454/1,SN7454/1

#TEXT
4x2-Input AND/NOR-Gate


This component contains
a NOR gate with 4
inputs, internally
connected to four AND
gates each having 2
inputs.

Observe the difference
between this version and
the L-, LS- and H-
versions!

#PIN 14
 1 : Input   A
 2 : Input   C
 3 : Input   D
 4 : Input   E
 5 : Input   F
 6 : N.C.
 7 : GND
 8 : Output  Q
 9 : Input   G
10 : Input   H
11 : N.C.
12 : N.C.
13 : Input   B
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,5,9,10,13]:=LOW;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[13]:=I;
   IF PIN[8]<>NOT(PIN[1] AND PIN[13]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
```

```
   IF PIN[8]<>NOT(PIN[2] AND PIN[3]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[4]&PIN[5]:=I;
   IF PIN[8]<>NOT(PIN[4] AND PIN[5]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   IF PIN[8]<>NOT(PIN[9] AND PIN[10]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74L54,SN74L54,74L54/1,SN74L54/1,74LS54,SN74LS54,74LS54/1,SN74LS54/1

#TEXT
2x2 and 2x3-Input
AND/NOR Gate

This component contains
a NOR gate with 4 inputs
internally connected to
four AND gates. Three of
the AND gates each have
2 inputs, and the fourth
has three.

Observe the difference
between this component
and the Std-, L-  and
LS- versions!

#PIN 14
 1 : Input   A
 2 : Input   B
 3 : Input   C
 4 : Input   D
 5 : Input   E
 6 : Output  Q
 7 : GND
 8 : N.C.
 9 : Input   F
10 : Input   G
11 : Input   H
12 : Input   I
13 : Input   J
14 : +5V

#FAMILY L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,5,9,10,11,12,13]:=LOW;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
```

```
   IF PIN[6]<>NOT(PIN[1] AND PIN[2]) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   IF PIN[6]<>NOT(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[6]<>NOT(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7454/2,SN7454/2,74H54,SN74H54,74H54/2,SN74H54/1

#TEXT
4x2 or 2x2 and 2x3-Input
AND/NOR Gate

This component contains
a NOR gate with 4 inputs
internally connected to
four AND gates. Two of
the AND gates each have
two inputs, the other
two have three.

Observe the difference
between this component
and the Std- version!

#PIN 14
 1 : Input   A
 2 : Input   C
 3 : Input   D
 4 : Input   E
 5 : Input   F
 6 : Input   G
 7 : GND
 8 : Output  Q
 9 : Input   H
10 : Input   I
11 : N.C.
12 : N.C.
13 : Input   B
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,9,10,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
```

```
PIN[1,2,3,4,5,6,9,10,13]:=LOW;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[13]:=I;
   IF PIN[8]<>NOT(PIN[1] AND PIN[13]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   IF PIN[8]<>NOT(PIN[2] AND PIN[3]) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[4]&PIN[5]&PIN[6]:=I;
   IF PIN[8]<>NOT(PIN[4] AND PIN[5] AND PIN[6]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   IF PIN[8]<>NOT(PIN[9] AND PIN[10]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7455,SN7455

#TEXT
2x4-Input AND/NOR-Gate

This component contains
a NOR gate with 2 inputs
internally connected to
two AND gates each having
four inputs.


#PIN 14
 1 : Input   A
 2 : Input   B
 3 : Input   C
 4 : Input   D
 5 : N.C.
 6 : N.C.
 7 : GND
 8 : Output  Q
 9 : N.C.
10 : Input   E
11 : Input   F
12 : Input   G
13 : Input   H
14 : +5V

#FAMILY H,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,10,11,12,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,10,11,12,13]:=LOW;
```

```
FOR I:=15 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]:=I;
   IF PIN[8]<>NOT(PIN[1] AND PIN[2] AND PIN[3] AND PIN[4]) THEN ERROR(1);
   END;

FOR I:=15 DOWNTO 0 DO
   BEGIN
   PIN[10]&PIN[11]&PIN[12]&PIN[13]:=I;
   IF PIN[8]<>NOT(PIN[10] AND PIN[11] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7456,SN7456

#TEXT
Frequency Divider 50:1

This component contains
a frequency divider 2:1
and two dividers 5:1.

#PIN 8
 1 : Input   1 Clock
 2 : +5V
 3 : Output  2   5:1
 4 : GND
 5 : Input   2 Clock
 6 : Clear
 7 : Output  1   5:1
 8 : Output  1  10:1

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,5,6] : INPUT;
PIN[3,7,8] : OUTPUT;
PIN[4] : GND;
PIN[2] : +5V;
PIN[1,5]:=LOW;

PIN[6]:=LOW;PIN[6]:=HIGH;
FOR I:=1 TO 5 DO
   BEGIN
   IF (PIN[7]<>LOW) OR (PIN[8]<>LOW) THEN ERROR(1);
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;
IF PIN[7]<>HIGH THEN ERROR(1);

FOR I:=1 TO 5 DO
   BEGIN
   IF PIN[8]<>LOW THEN ERROR(1);
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;
IF PIN[8]<>HIGH THEN ERROR(1);

FOR I:=1 TO 5 DO
   BEGIN
   IF PIN[3]<>LOW THEN ERROR(1);
   PIN[5]:=HIGH;PIN[5]:=LOW;
   END;
IF PIN[3]<>HIGH THEN ERROR(1);
```

```
ERROR(0);
END.

#NAME 7457,SN7457

#TEXT
Frequency Divider 60:1

This component contains
a frequency divider 2:1,
a divider 5:1 and a
divider 6:1.

#PIN 8
 1 : Input   1 Clock
 2 : +5V
 3 : Output  2   6:1
 4 : GND
 5 : Input   2 Clock
 6 : Clear
 7 : Output  1   5:1
 8 : Output  1  10:1

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,5,6] : INPUT;
PIN[3,7,8] : OUTPUT;
PIN[4] : GND;
PIN[2] : +5V;
PIN[1,5]:=LOW;

PIN[6]:=LOW;PIN[6]:=HIGH;
FOR I:=1 TO 5 DO
   BEGIN
   IF (PIN[7]<>LOW) OR (PIN[8]<>LOW) THEN ERROR(1);
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;
IF PIN[7]<>HIGH THEN ERROR(1);

FOR I:=1 TO 5 DO
   BEGIN
   IF PIN[8]<>LOW THEN ERROR(1);
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;
IF PIN[8]<>HIGH THEN ERROR(1);

FOR I:=1 TO 6 DO
   BEGIN
   IF PIN[3]<>LOW THEN ERROR(1);
   PIN[5]:=HIGH;PIN[5]:=LOW;
   END;
IF PIN[3]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 7460,SN7460

#TEXT
Dual 4-Input Expander

This component contains
two separate expanders
```

```
each with 4 inputs.

#PIN 14
 1 : Input    2 Gate 1
 2 : Input    3 Gate 1
 3 : Input    4 Gate 1
 4 : Input    1 Gate 2
 5 : Input    2 Gate 2
 6 : Input    3 Gate 2
 7 : GND
 8 : Input    4 Gate 2
 9 : Expand  -X Gate 2
10 : Output   X Gate 2
11 : Output   X Gate 1
12 : Expand  -X Gate 1
13 : Input    1 Gate 1
14 : +5V

#FAMILY Std,H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,8,9,12,13] : INPUT;
PIN[10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 31 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[1]&PIN[2]&PIN[3]:=I;
   PIN[9]&PIN[4]&PIN[5]&PIN[6]&PIN[8]:=I;
   LOADMODEON;
   PIN[10,11] : LOAD LOW;
   IF PIN[11]<>(PIN[12]
               AND PIN[13] AND PIN[1] AND PIN[2] AND PIN[3]) THEN ERROR(1);
   IF PIN[10]<>(PIN[9]
               AND PIN[4] AND PIN[5] AND PIN[6] AND PIN[8]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7461,SN7461

#TEXT
Triple 3-Input Expander

This component contains
three separate expanders
each with 3 inputs.

#PIN 14
 1 : Input    1 Gate 1
 2 : Input    2 Gate 1
 3 : Input    3 Gate 1
 4 : Input    1 Gate 2
 5 : Input    2 Gate 2
 6 : Input    3 Gate 2
 7 : GND
 8 : Output   X Gate 2
 9 : Output   X Gate 1
10 : Output   X Gate 3
11 : Input    1 Gate 3
12 : Input    2 Gate 3
13 : Input    3 Gate 3
```

```
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,11,12,13] : INPUT;
PIN[8,9,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]:=I;
   PIN[4]&PIN[5]&PIN[6]:=I;
   PIN[11]&PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[8,9,10] : LOAD LOW;
   IF PIN[9]<>(PIN[1] AND PIN[2] AND PIN[3]) THEN ERROR(1);
   IF PIN[8]<>(PIN[4] AND PIN[5] AND PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[11] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7462,SN7462

#TEXT
2x2 and 2x3-Input
AND/OR Gate

This component contains
an OR expander with 4
inputs, internally
connected to four AND
gates. Two of the AND
gates each have two
inputs and the remaining
two have three.

#PIN 14
 1 : Input   A
 2 : Input   B
 3 : Input   C
 4 : Input   D
 5 : Input   E
 6 : Expand -X
 7 : GND
 8 : Output  X
 9 : Input   F
10 : Input   G
11 : Input   H
12 : Input   I
13 : Input   J
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
```

```
PIN[14] : +5V;
PIN[1,2,3,4,5,9,10,11,12,13]:=LOW;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   LOADMODEON;
   PIN[8] : LOAD LOW;
   IF PIN[8]<>(PIN[1] AND PIN[2]) THEN ERROR(1);
   LOADMODEOFF;
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   LOADMODEON;
   PIN[8] : LOAD LOW;
   PIN[3]&PIN[4]&PIN[5]:=I;
   IF PIN[8]<>(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   LOADMODEOFF;
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   LOADMODEON;
   PIN[8] : LOAD LOW;
   PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[8]<>(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   LOADMODEOFF;
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   LOADMODEON;
   PIN[8] : LOAD LOW;
   PIN[12]&PIN[13]:=I;
   IF PIN[8]<>(PIN[12] AND PIN[13]) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7463,SN7463

#TEXT
Hex Current Sensor

This component contains 6
separate current sensors.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Output    Gate 2
 4 : Input     Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Input     Gate 5
11 : Output    Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY LS
```

```
#PROGRAM

BEGIN
PIN[2,3,6,8,11,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[1,4,5,9,10,13] : OUTPUT;
IF (PIN[2]<>LOW) OR (PIN[3]<>LOW) OR (PIN[6]<>LOW)
   OR (PIN[8]<>LOW) OR (PIN[11]<>LOW) OR (PIN[12]<>LOW) THEN ERROR(1);

PIN[1,4,5,9,10,13] : INPUT;
PIN[1,4,5,9,10,13]:=LOW;
IF (PIN[2]<>HIGH) OR (PIN[3]<>HIGH) OR (PIN[6]<>HIGH)
   OR (PIN[8]<>HIGH) OR (PIN[11]<>HIGH) OR (PIN[12]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 7464,SN7464

#TEXT
2x2, 1x3 and 1x4-Input
AND/NOR Gate

This component contains
a NOR gate with 4 inputs,
internally connected to
two AND gates each with 2
inputs, an AND gate with 3
inputs and an AND gate
with 4 inputs.

#PIN 14
 1 : Input   A
 2 : Input   E
 3 : Input   F
 4 : Input   G
 5 : Input   H
 6 : Input   I
 7 : GND
 8 : Output  Q
 9 : Expand  J
10 : Input   K
11 : Input   B
12 : Input   C
13 : Input   D
14 : +5V

#FAMILY H,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,9,10,11,12,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,5,6,9,10,11,12,13]:=LOW;

FOR I:=15 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[11]&PIN[12]&PIN[13]:=I;
   IF PIN[8]<>NOT(PIN[1] AND PIN[11] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;
```

```
FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[2]&PIN[3:=I;
   IF PIN[8]<>NOT(PIN[2] AND PIN[3]) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[4]&PIN[5]&PIN[6]:=I;
   IF PIN[8]<>NOT(PIN[4] AND PIN[5] AND PIN[6]) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   IF PIN[8]<>NOT(PIN[9] AND PIN[10]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7465,SN7465

#TEXT
2x2, 1x3 and 1x4-Input
AND/NOR Gate (O.C.)

This component contains
a NOR gate with 4 inputs
internally connected to
two AND gates each with
2 inputs, an AND gate
with 3 inputs and an AND
gate with 4 inputs.
The output of the NOR
gate is OPEN COLLECTOR.

#PIN 14
 1 : Input   A
 2 : Input   E
 3 : Input   F
 4 : Input   G
 5 : Input   H
 6 : Input   I
 7 : GND
 8 : Output  Q
 9 : Expand  J
10 : Input   K
11 : Input   B
12 : Input   C
13 : Input   D
14 : +5V

#FAMILY H,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,9,10,11,12,13] : INPUT;
PIN[8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,5,6,9,10,11,12,13]:=LOW;


FOR I:=15 DOWNTO 0 DO
   BEGIN
```

```
   PIN[1]&PIN[11]&PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[8] : LOAD LOW;
   IF PIN[8]<>LOW THEN ERROR(1);
   PIN[8] : LOAD HIGH;
   IF PIN[8]<>NOT(PIN[1] AND PIN[11] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   LOADMODEOFF;
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   LOADMODEON;
   PIN[8] : LOAD HIGH;
   PIN[2]&PIN[3:=I;
   IF PIN[8]<>NOT(PIN[2] AND PIN[3]) THEN ERROR(1);
   PIN[8] : LOAD LOW;
   IF PIN[8]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   LOADMODEON;
   PIN[8] : LOAD HIGH;
   PIN[4]&PIN[5]&PIN[6]:=I;
   IF PIN[8]<>NOT(PIN[4] AND PIN[5] AND PIN[6]) THEN ERROR(1);
   PIN[8] : LOAD LOW;
   IF PIN[8]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   LOADMODEON;
   PIN[8] : LOAD HIGH;
   PIN[9]&PIN[10]:=I;
   IF PIN[8]<>NOT(PIN[9] AND PIN[10]) THEN ERROR(1);
   PIN[8] : LOAD LOW;
   IF PIN[8]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 7470,SN7470

#TEXT
3-Input J-K Flip-Flop
with Preset and Clear

This component contains
an edge-triggered J-K
flip-flop, direct preset
and clear inputs, multiple
J- and K- inputs as well
as true and complementary
outputs.

#PIN 14
 1 : N.C.
 2 : -Reset
 3 : Input   J1
 4 : Input   J2
 5 : Input   -J
 6 : Output  -Q
 7 : GND
```

```
 8 : Output   Q
 9 : Input   -K
10 : Input   K1
11 : Input   K2
12 : Clock
13 : -Preset
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,10,11,12]:=LOW;
PIN[2,5,9,13]:=HIGH;

PIN[2]:=LOW;PIN[2]:=HIGH;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[11]&PIN[10]&PIN[9]:=I EXOR %1;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I EXOR %1;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

PIN[3,4]:=HIGH;PIN[5]:=LOW;
PIN[11,10]:=HIGH;PIN[9]:=LOW;
PIN[12]:=HIGH;PIN[12]:=LOW;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 7471,SN7471,74H71,SN74H71

#TEXT
2x2-Input J-K Master/
Slave Flip-Flop with
direct Set Input

This component contains
a pulse-triggered J-K-
Master-Slave flip-flop,
AND/OR input plus a
direct set input.


Observe the L- version!


#PIN 14
 1 : Input   J1A
```

```
 2 : Input   J1B
 3 : Input   J2A
 4 : Input   J2B
 5 : -Preset
 6 : Output  Q
 7 : GND
 8 : Output  -Q
 9 : Input   K1A
10 : Input   K1B
11 : Input   K2A
12 : Input   K2B
13 : Clock
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,9,10,11,12]:=LOW;
PIN[13]:=HIGH;

PIN[5]:=LOW;PIN[5]:=HIGH;
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[11]&PIN[12]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74L71,SN74L71

#TEXT
3-Input R-S Master/Slave
```

```
Flip-Flop with Preset
and Clear

This component contains
a pulse-triggered RS-
Master/Slave flip-flop
with 3 AND inputs,
preset and clear.

Observe the difference
between this component
and the H- version!

#PIN 14
 1 : N.C.
 2 : -Reset
 3 : Input   S1
 4 : Input   S2
 5 : Input   S3
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : Input   R1
10 : Input   R2
11 : Input   R3
12 : Clock
13 : -Preset
14 : +5V

#FAMILY L

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,5,9,10,11]:=LOW;
PIN[2,12,13]:=HIGH;

PIN[2]:=LOW;PIN[2]:=HIGH;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[11]:=I;
   PIN[12]:=LOW;PIN[12]:=HIGH;
   IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[12]:=LOW;PIN[12]:=HIGH;
   IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7472,SN7472

#TEXT
```

```
3-Input J-K Master/Slave
Flip-Flop with Preset
and Clear

This component contains
a pulse-triggered J-K
Master/Slave flip-flop
with 3 AND inputs,
preset and clear.

#PIN 14
 1 : N.C.
 2 : -Reset
 3 : Input   J1
 4 : Input   J2
 5 : Input   J3
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : Input   K1
10 : Input   K2
11 : Input   K3
12 : Clock
13 : -Preset
14 : +5V

#FAMILY Std,H,LS

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,5,9,10,11]:=LOW;
PIN[2,12,13]:=HIGH;

PIN[2]:=LOW;PIN[2]:=HIGH;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[11]:=I;
   PIN[12]:=LOW;PIN[12]:=HIGH;
   IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[12]:=LOW;PIN[12]:=HIGH;
   IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

PIN[3,4,5,9,10,11]:=HIGH;
PIN[12]:=LOW;PIN[12]:=HIGH;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 7473,SN7473
```

```
#TEXT
Dual J-K Flip-Flops with
Clear

This component contains
two separate J-K flip-
flops with independent
clock and clear inputs.

#PIN 14
 1 : Clock       FF 1
 2 : -Reset      FF 1
 3 : Input    K  FF 1
 4 : +5V
 5 : Clock       FF 2
 6 : -Reset      FF 2
 7 : Input      J  FF 2
 8 : Output  -Q  FF 2
 9 : Output   Q  FF 2
10 : Input     K  FF 2
11 : GND
12 : Input     Q  FF 1
13 : Input    -Q  FF 1
14 : Input    J  FF 1

#FAMILY Std,AS,H,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,5,6,7,10,14] : INPUT;
PIN[8,9,12,13] : OUTPUT;
PIN[11] : GND;
PIN[4] : +5V;
PIN[3,7,10,14]:=LOW;
PIN[1,2,5,6]:=HIGH;

PIN[2,6]:=LOW;PIN[2,6]:=HIGH;
IF (PIN[12]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[14,7]:=I;
   PIN[1,5]:=LOW;PIN[1,5]:=HIGH;
   IF (PIN[12]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[3,10]:=I;
   PIN[1,5]:=LOW;PIN[1,5]:=HIGH;
   IF (PIN[12]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

PIN[14,3,7,10]:=HIGH;
PIN[1,5]:=LOW;PIN[1,5]:=HIGH;
IF (PIN[12]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

ERROR(0);
END.

#NAME 7474,SN7474
```

```
#TEXT
Dual D Flip-Flops with
Preset and Clear

This component contains
two separate D flip-flops
with triggering on the
positive edge of the clock
and independent set and
reset inputs.


#PIN 14
 1 : -Reset       FF 1
 2 : Input    D   FF 1
 3 : Clock        FF 1
 4 : -Preset      FF 1
 5 : Output   Q   FF 1
 6 : Output  -Q   FF 1
 7 : GND
 8 : Output  -Q   FF 2
 9 : Output   Q   FF 2
10 : -Preset      FF 2
11 : Clock        FF 2
12 : Output   D   FF 2
13 : -Reset       FF 2
14 : +5V

#FAMILY Std,ALS,F,H,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,10,11,12,13] : INPUT;
PIN[5,6,8,9] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,11]:=LOW;
PIN[1,4,10,13]:=HIGH;

PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

PIN[4,10]:=LOW;PIN[4,10]:=HIGH;
IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[2,12]:=I;
   PIN[3,11]:=HIGH;PIN[3,11]:=LOW;
   IF (PIN[5]<>I) OR NOT(PIN[6]<>I) THEN ERROR(1);
   IF (PIN[9]<>I) OR NOT(PIN[8]<>I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7475,SN7475

#TEXT
Dual 2-Bit-D-Buffer
with Release

This component contains
four bistable memory
```

```
elements.

#PIN 16
 1 : Output  -Q SE 1
 2 : Input    D SE 1
 3 : Input    D SE 2
 4 : Enable     SE 3+4
 5 : +5V
 6 : Input    D SE 3
 7 : Input    D SE 4
 8 : Output  -Q SE 4
 9 : Output   Q SE 4
10 : Output   Q SE 3
11 : Output  -Q SE 3
12 : GND
13 : Enable     SE 1+2
14 : Output  -Q SE 2
15 : Output   Q SE 2
16 : Output   Q SE 1

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[2,3,4,6,7,13] : INPUT;
PIN[1,8,9,10,11,14,15,16] : OUTPUT;
PIN[12] : GND;
PIN[5] : +5V;
PIN[4,13]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7]&PIN[6]&PIN[3]&PIN[2]:=D;
   PIN[4,13]:=HIGH;PIN[4,13]:=LOW;
   PIN[7]&PIN[6]&PIN[3]&PIN[2]:=0;
   IF (PIN[9]&PIN[10]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   IF (NOT(PIN[8])&NOT(PIN[11])&NOT(PIN[14])&NOT(PIN[1]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 7476,SN7476

#TEXT
Dual J-K Flip-Flops with
Preset and Clear

This component contains
two independent J-K flip-
flops with preset and
clear.

#PIN 16
 1 : Clock       FF 1
 2 : -Preset     FF 1
 3 : -Reset      FF 1
 4 : Input    J  FF 1
 5 : +5V
 6 : Clock       FF 2
 7 : -Preset     FF 2
 8 : -Reset      FF 2
 9 : Input    J  FF 2
10 : Output  -Q  FF 2
```

```
11 : Output   Q  FF 2
12 : Input    K  FF 2
13 : GND
14 : Output  -Q  FF 1
15 : Output   Q  FF 1
16 : Input    K  FF 1

#FAMILY Std,H,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,6,7,8,9,12,16] : INPUT;
PIN[10,11,14,15] : OUTPUT;
PIN[13] : GND;
PIN[5] : +5V;
PIN[4,9,12,16]:=LOW;
PIN[1,2,3,6,7,8]:=HIGH;

PIN[3,8]:=LOW;PIN[3,8]:=HIGH;
IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
IF (PIN[11]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);

PIN[2,7]:=LOW;PIN[2,7]:=HIGH;
IF (PIN[15]<>HIGH) OR (PIN[14]<>LOW) THEN ERROR(1);
IF (PIN[11]<>HIGH) OR (PIN[10]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[16,12]:=I;
   PIN[1,6]:=LOW;PIN[1,6]:=HIGH;
   IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
   IF (PIN[11]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[4,9]:=I;
   PIN[1,6]:=LOW;PIN[1,6]:=HIGH;
   IF (PIN[15]<>HIGH) OR (PIN[14]<>LOW) THEN ERROR(1);
   IF (PIN[11]<>HIGH) OR (PIN[10]<>LOW) THEN ERROR(1);
   END;

PIN[4,16,9,12]:=HIGH;
PIN[1,6]:=LOW;PIN[1,6]:=HIGH;
IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
IF (PIN[11]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 7477,SN7477

#TEXT
Dual 2-Bit-D-Buffer
and Release

This component contains
four bistable memory
elements.

#PIN 14
 1 : Input    D SE 1
 2 : Input    D SE 2
 3 : Enable     SE 3+4
 4 : +5V
 5 : Input    D SE 3
```

```
 6 : Input    D SE 4
 7 : N.C.
 8 : Output   Q SE 4
 9 : Output   Q SE 3
10 : N.C.
11 : GND
12 : Enable     SE 1+2
13 : Output   Q SE 2
14 : Output   Q SE 1

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,5,6,12] : INPUT;
PIN[8,9,13,14] : OUTPUT;
PIN[11] : GND;
PIN[4] : +5V;
PIN[3,12]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[2]&PIN[1]:=D;
   PIN[12,3]:=HIGH;PIN[12,3]:=LOW;
   PIN[6]&PIN[5]&PIN[2]&PIN[1]:=0;
   IF (PIN[8]&PIN[9]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.
•
```

```
#NAME 7478,SN7478,74H78,SN74H78

#TEXT
Dual J-K Flip-Flops with
Preset, Combined Clear
and Combined Clock.


This component contains
two separate J-K flip-
flops with combined
clock, combined clear
and separate inputs for
the preset.

Observe the difference
between this component
and the L- and LS-
versions!

#PIN 14
 1 : Input    K  FF 1
 2 : Output   Q  FF 1
 3 : Output  -Q  FF 1
 4 : Input    J  FF 1
 5 : Output  -Q  FF 2
 6 : Output   Q  FF 2
 7 : GND
 8 : Input    K  FF 2
 9 : Clock
10 : -Preset     FF 2
11 : Input    J  FF 2
12 : -Reset
13 : -Preset     FF 1
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,4,8,9,10,11,12,13] : INPUT;
PIN[2,3,5,6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,8,11]:=LOW;
PIN[9,10,12,13]:=HIGH;

PIN[12]:=LOW;PIN[12]:=HIGH;
IF (PIN[2]<>LOW) OR (PIN[3]<>HIGH) THEN ERROR(1);
IF (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);

PIN[13,10]:=LOW;PIN[13,10]:=HIGH;
IF (PIN[2]<>HIGH) OR (PIN[3]<>LOW) THEN ERROR(1);
IF (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[1,8]:=I;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   IF (PIN[2]<>LOW) OR (PIN[3]<>HIGH) THEN ERROR(1);
   IF (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[4,11]:=I;
```

```
   PIN[9]:=LOW;PIN[9]:=HIGH;
   IF (PIN[2]<>HIGH) OR (PIN[3]<>LOW) THEN ERROR(1);
   IF (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
   END;

PIN[1,4,8,11]:=HIGH;
PIN[9]:=LOW;PIN[9]:=HIGH;
IF (PIN[2]<>LOW) OR (PIN[3]<>HIGH) THEN ERROR(1);
IF (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74L78,SN74L78,74LS78,SN74LS78

#TEXT
Dual J-K Flip-Flops with
Preset, Combined Clear
and Combined Clock


This component contains
two separate J-K flip-
flops with combined clock
and clear functions and
separate inputs for the
preset.

Observe the difference
between this component
and the H- version!

#PIN 14
 1 : Clock
 2 : -Preset     FF 1
 3 : Input    J  FF 1
 4 : +5V
 5 : -Reset
 6 : -Preset     FF 2
 7 : Input    K  FF 2
 8 : Output   Q  FF 2
 9 : Output  -Q  FF 2
10 : Input    J  FF 2
11 : GND
12 : Output  -Q  FF 1
13 : Output   Q  FF 1
14 : Input    K  FF 1

#FAMILY L,LS

#PROGRAM

BEGIN
PIN[1,2,3,5,6,7,10,14] : INPUT;
PIN[8,9,12,13] : OUTPUT;
PIN[11] : GND;
PIN[4] : +5V;
PIN[3,7,10,14]:=LOW;
PIN[1,2,5,6]:=HIGH;

PIN[5]:=LOW;PIN[5]:=HIGH;
IF (PIN[13]<>LOW) OR (PIN[12]<>HIGH) THEN ERROR(1);
IF (PIN[8]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);

PIN[2,6]:=LOW;PIN[2,6]:=HIGH;
IF (PIN[13]<>HIGH) OR (PIN[12]<>LOW) THEN ERROR(1);
IF (PIN[8]<>HIGH) OR (PIN[9]<>LOW) THEN ERROR(1);
```

```
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[14,7]:=I;
   PIN[1]:=LOW;PIN[1]:=HIGH;
   IF (PIN[13]<>LOW) OR (PIN[12]<>HIGH) THEN ERROR(1);
   IF (PIN[8]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[3,10]:=I;
   PIN[1]:=LOW;PIN[1]:=HIGH;
   IF (PIN[13]<>HIGH) OR (PIN[12]<>LOW) THEN ERROR(1);
   IF (PIN[8]<>HIGH) OR (PIN[9]<>LOW) THEN ERROR(1);
   END;

PIN[3,14,10,7]:=HIGH;
PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[13]<>LOW) OR (PIN[12]<>HIGH) THEN ERROR(1);
IF (PIN[8]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 7480,SN7480

#TEXT
1-Bit Full Adder

This component contains
a 1-bit full adder with
complementary inputs
and outputs.

#PIN 14
 1 : Input   B*
 2 : Input   Bc
 3 : Input   Cn
 4 : Output  -Cn+1
 5 : Output  ä
 6 : Output  -ä
 7 : GND
 8 : Input   A1
 9 : Input   A2
10 : Input   A*
11 : Input   Ac
12 : Input   B1
13 : Input   B2
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,8,9,11,12,13] : INPUT;
PIN[4,5,6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[1,10] : OUTPUT;
FOR C:=1 DOWNTO 0 DO
   BEGIN
   PIN[3]:=C;
   FOR I:=7 DOWNTO 0 DO
      BEGIN
```

```
      PIN[8]&PIN[9]&PIN[11]:=I;
      FOR J:=7 DOWNTO 0 DO
         BEGIN
         PIN[12]&PIN[13]&PIN[2]:=J;
         A:=PIN[8] NAND PIN[9] NAND PIN[11];
         B:=PIN[12] NAND PIN[13] NAND PIN[2];
         IF PIN[5]<>(A EXOR B EXOR C) THEN ERROR(1);
         IF NOT(PIN[6]<>(A EXOR B EXOR C)) THEN ERROR(1);
         END;
      END;
   END;

PIN[1,10] : INPUT;
FOR C:=1 DOWNTO 0 DO
   BEGIN
   PIN[3]:=C;
   FOR I:=3 DOWNTO 0 DO
      BEGIN
      PIN[10]&PIN[11]:=I;
      FOR J:=3 DOWNTO 0 DO
         BEGIN
         PIN[1]&PIN[2]:=J;
         A:=PIN[10] NAND PIN[11];
         B:=PIN[1] NAND PIN[2];
         IF PIN[5]<>(A EXOR B EXOR C) THEN ERROR(1);
         IF NOT(PIN[6]<>(A EXOR B EXOR C)) THEN ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 7481,SN7481

#TEXT
16-Bit-RAM

This component contains
a read/write memory
(RAM) with 16-bit,
organised as memory 16x1.
The outputs are OPEN
COLLECTOR.

#PIN 14
 1 : Address X3
 2 : Address X2
 3 : Address X1
 4 : +5V
 5 : Address Y1
 6 : Address Y2
 7 : Address Y3
 8 : Address Y4
 9 : Input   WL
10 : GND
11 : Output  QL
12 : Output  QH
13 : Input   WH
14 : Address X4

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,5,6,7,8,9,13,14] : INPUT;
```

```
PIN[11,12] : OUTPUT;
PIN[10] : GND;
PIN[4] : +5V;

FOR I:=0 TO 1 DO
   BEGIN
   FOR X:=0 TO 3 DO
      BEGIN
      PIN[14]&PIN[1]&PIN[2]&PIN[3]:=1 SHL X;
      FOR Y:=0 TO 3 DO
         BEGIN
         PIN[8]&PIN[7]&PIN[6]&PIN[5]:=1 SHL Y;
         IF ((I EXOR X EXOR Y) AND 1)=0
            THEN BEGIN
                 PIN[9]:=LOW;PIN[13]:=HIGH;
                 END
            ELSE BEGIN
                 PIN[13]:=LOW;PIN[9]:=HIGH;
                 END;
         END;
      END;

   PIN[13]:=LOW;PIN[9]:=LOW;
   FOR X:=0 TO 3 DO
      BEGIN
      PIN[14]&PIN[1]&PIN[2]&PIN[3]:=1 SHL X;
      FOR Y:=0 TO 3 DO
         BEGIN
         PIN[8]&PIN[7]&PIN[6]&PIN[5]:=1 SHL Y;
         LOADMODEON;
         PIN[11,12] : LOAD LOW;
         IF (PIN[11]<>LOW) OR (PIN[12]<>LOW) THEN ERROR(1);
         PIN[11,12] : LOAD HIGH;
         IF ((I EXOR X EXOR Y) AND 1)=0
         THEN BEGIN
              IF (PIN[11]<>HIGH) OR (PIN[12]<>LOW) THEN ERROR(1);
              END
         ELSE BEGIN
              IF (PIN[11]<>LOW) OR (PIN[12]<>HIGH) THEN ERROR(1);
              END;
         LOADMODEOFF;
         END;
      END;
   END;

ERROR(0);
END.

#NAME 7482,SN7482

#TEXT
2-Bit Full Adder

This component contains
a full adder for 2x2
bits, with carry provision
for the second bit.

#PIN 14
 1 : Output  ä1
 2 : Input   A1
 3 : Input   B1
 4 : +5V
 5 : Input   Cin
 6 : N.C.
 7 : N.C.
 8 : N.C.
```

```
 9 : N.C.
10 : Output  C2
11 : GND
12 : Output  ä2
13 : Input   B2
14 : Input   A2

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,5,13,14] : INPUT;
PIN[1,10,12] : OUTPUT;
PIN[11] : GND;
PIN[4] : +5V;

FOR C:=0 TO 1 DO
   BEGIN
   PIN[5]:=C;
   FOR A:=0 TO 3 DO
      BEGIN
      PIN[14]&PIN[2]:=A;
      FOR B:=0 TO 3 DO
         BEGIN
         PIN[13]&PIN[3]:=B;
         IF (PIN[10]&PIN[12]&PIN[1])<>(C+A+B) THEN ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 7483,SN7483

#TEXT
4-Bit Full Adder

This component contains
a full adder, which
provides the sum of two-
figure 4-bit binary
numbers with carry-over.

#PIN 16
 1 : Input   A4
 2 : Output  ä3
 3 : Input   A3
 4 : Input   B3
 5 : +5V
 6 : Output  ä2
 7 : Input   B2
 8 : Input   A2
 9 : Output  ä1
10 : Input   A1
11 : Input   B1
12 : GND
13 : Input   C0
14 : Output  C4
15 : Output  ä4
16 : Input   B4

#FAMILY Std,LS

#PROGRAM
```

```
BEGIN
PIN[1,3,4,7,8,10,11,13,16] : INPUT;
PIN[2,6,9,14,15] : OUTPUT;
PIN[12] : GND;
PIN[5] : +5V;

FOR C:=0 TO 1 DO
   BEGIN
   PIN[13]:=C;
   FOR A:=0 TO 15 DO
      BEGIN
      PIN[1]&PIN[3]&PIN[8]&PIN[10]:=A;
      FOR B:=A MOD 3 TO 15 BY 3 DO
         BEGIN
         PIN[16]&PIN[4]&PIN[7]&PIN[11]:=B;
         IF (PIN[14]&PIN[15]&PIN[2]&PIN[6]&PIN[9])<>(C+A+B) THEN ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 7484,SN7484

#TEXT
16-Bit-RAM with two
additional Write-
Inputs

This component contains
a write/read memory
(RAM) with 16 bit,
organised as 16x1 memory,
with two additional write
inputs and OPEN-COLLECTOR
outputs.

#PIN 16
 1 : Address X4
 2 : Address X3
 3 : Address X2
 4 : Address X1
 5 : +5V
 6 : Address Y1
 7 : Address Y2
 8 : Address Y3
 9 : Address Y4
10 : Input   WL2
11 : Input   WL1
12 : GND
13 : Output  QL
14 : Output  QH
15 : Input   WH2
16 : Input   WH1

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,6,7,8,9,10,11,15,16] : INPUT;
PIN[13,14] : OUTPUT;
PIN[12] : GND;
PIN[5] : +5V;

FOR I:=0 TO 1 DO
```

```
   BEGIN
   FOR X:=0 TO 3 DO
      BEGIN
      PIN[1]&PIN[2]&PIN[3]&PIN[4]:=1 SHL X;
      FOR Y:=0 TO 3 DO
         BEGIN
         PIN[9]&PIN[8]&PIN[7]&PIN[6]:=1 SHL Y;
         IF ((I EXOR X EXOR Y) AND 1)=0
            THEN BEGIN
                 PIN[10]:=LOW;PIN[11]:=LOW;PIN[15]:=HIGH;PIN[16]:=HIGH;
                 END
            ELSE BEGIN
                 PIN[15]:=LOW;PIN[16]:=LOW;PIN[10]:=HIGH;PIN[11]:=HIGH;
                 END;
         END;
      END;

   PIN[10]:=LOW;PIN[11]:=LOW;PIN[15]:=LOW;PIN[16]:=LOW;
   FOR X:=0 TO 3 DO
      BEGIN
      PIN[1]&PIN[2]&PIN[3]&PIN[4]:=1 SHL X;
      FOR Y:=0 TO 3 DO
         BEGIN
         PIN[9]&PIN[8]&PIN[7]&PIN[6]:=1 SHL Y;
         LOADMODEON;
         PIN[13,14] : LOAD LOW;
         IF (PIN[13]<>LOW) OR (PIN[14]<>LOW) THEN ERROR(1);
         PIN[13,14] : LOAD HIGH;
         IF ((I EXOR X EXOR Y) AND 1)=0
         THEN BEGIN
              IF (PIN[13]<>HIGH) OR (PIN[14]<>LOW) THEN ERROR(1);
              END
         ELSE BEGIN
              IF (PIN[13]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
              END;
         LOADMODEOFF;
         END;
      END;

   END;

ERROR(0);
END.

#NAME 7485,SN7485

#TEXT
4-Bit Comparator

This component contains
a 2 x 4 bit comparator.
It indicates whether one
of the four bit words is
"greater than" or "equal
to" the other word.

Observe the difference
between this component
and the L- version!

#PIN 16
 1 : Input   B3
 2 : Carry Input    A<B
 3 : Carry Input    A=B
 4 : Carry Input    A>B
 5 : Output  A>B
 6 : Output  A=B
```

```
 7 : Output  A<B
 8 : GND
 9 : Input   B0
10 : Input   A0
11 : Input   B1
12 : Input   A1
13 : Input   A2
14 : Input   B2
15 : Input   A3
16 : +5V

#FAMILY Std,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,9,10,11,12,13,14,15] : INPUT;
PIN[5,6,7] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[2]:=HIGH;PIN[3]:=LOW;PIN[4]:=LOW;
PIN[15]&PIN[13]&PIN[12]&PIN[10]:=0;
PIN[1]&PIN[14]&PIN[11]&PIN[9]:=0;
IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[2]:=LOW;PIN[3]:=HIGH;PIN[4]:=LOW;
PIN[15]&PIN[13]&PIN[12]&PIN[10]:=0;
PIN[1]&PIN[14]&PIN[11]&PIN[9]:=0;
IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[2]:=LOW;PIN[3]:=LOW;PIN[4]:=HIGH;
PIN[15]&PIN[13]&PIN[12]&PIN[10]:=0;
PIN[1]&PIN[14]&PIN[11]&PIN[9]:=0;
IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);

C:=0;
```

```
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   C:=C+1;
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74L85,SN74L85

#TEXT
This component contains
a 2 x 4-Bit Comparator.
It indicates whether
one of the four bit words
is "greater than" or
"equal to" the other word.

Observe the difference
between this component
and the Std-, LS- and
S- versions!

#PIN 16
 1 : Input   B2
 2 : Input   A2
 3 : Output  A=B
 4 : Carry Input    A>B
 5 : Carry Input    A=B
 6 : Carry Input    A<B
 7 : Input   A1
 8 : GND
 9 : Input   B1
10 : Input   A0
11 : Input   B0
12 : Output  A<B
13 : Output  A>B
14 : Input   B3
15 : Input   A3
16 : +5V

#FAMILY L

#PROGRAM

BEGIN
PIN[1,2,4,5,6,7,9,10,11,14,15] : INPUT;
PIN[3,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[6]:=HIGH;PIN[5]:=LOW;PIN[4]:=LOW;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
```

```
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[6]:=LOW;PIN[5]:=HIGH;PIN[4]:=LOW;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[6]:=LOW;PIN[5]:=LOW;PIN[4]:=HIGH;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 7486,SN7486

#TEXT
Quad 2-Input Exclusive
OR Gate

This component contains
four EXOR gates each with
2 inputs.

Observe the difference
```

```
between this component
and the L- version!

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Output    Gate   1
 4 : Input   1 Gate   2
 5 : Input   2 Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input   1 Gate   3
10 : Input   2 Gate   3
11 : Output    Gate   4
12 : Input   1 Gate   4
13 : Input   2 Gate   4
14 : +5V

#FAMILY Std,ALS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] EXOR PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] EXOR PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] EXOR PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] EXOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74L86,SN74L86

#TEXT
Quad 2-Input Exclusive
OR Gate

This component contains
four EXOR gates each with
2 inputs.

Observe the difference
between this component
and the Std-, etc.-
versions!

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Output    Gate   1
 4 : Output    Gate   2
 5 : Input   1 Gate   2
 6 : Input   2 Gate   2
 7 : GND
```

```
 8 : Input   1 Gate   3
 9 : Input   2 Gate   3
10 : Output    Gate   3
11 : Output    Gate   4
12 : Input   1 Gate   4
13 : Input   2 Gate   4
14 : +5V

#FAMILY L

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] EXOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] EXOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] EXOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] EXOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7487,SN7487

#TEXT
4-Bit-Complementer

With this component
entered 4-bit words
are reproduced either
unchanged or comple-
mented.

#PIN 14
 1 : Input   C
 2 : Input   A1
 3 : Output  Q1
 4 : N.C.
 5 : Input   A2
 6 : Output  Q2
 7 : GND
 8 : Input   B
 9 : Output  Q3
10 : Input   A3
11 : N.C.
12 : Output  Q4
13 : Input   A4
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,5,8,10,13] : INPUT;
PIN[3,6,9,12] : OUTPUT;
```

```
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]:=LOW;
   PIN[1]:=I XOR 1;
   FOR J:=0 TO 1 DO
      BEGIN
      PIN[2]:=J;
      PIN[5]:=J;
      PIN[10]:=J;
      PIN[13]:=J;
      IF PIN[3]<>(PIN[2] EXOR I) THEN ERROR(1);
      IF PIN[6]<>(PIN[5] EXOR I) THEN ERROR(1);
      IF PIN[9]<>(PIN[10] EXOR I) THEN ERROR(1);
      IF PIN[12]<>(PIN[13] EXOR I) THEN ERROR(1);
      END;
   PIN[8]:=HIGH;
   PIN[1]:=I XOR 1;
   FOR J:=0 TO 1 DO
      BEGIN
      PIN[2]:=J;
      PIN[5]:=J;
      PIN[10]:=J;
      PIN[13]:=J;
      IF PIN[3]<>I THEN ERROR(1);
      IF PIN[6]<>I THEN ERROR(1);
      IF PIN[9]<>I THEN ERROR(1);
      IF PIN[12]<>I THEN ERROR(1);
      END;
   END;

ERROR(0);
END.

#NAME 7489,SN7489

#TEXT
16x4-Bit RAM

This component contains
a write/read memory (RAM)
organised in 16x4 bits.
The outputs are OPEN
COLLECTOR.


#PIN 16
 1 : Address A0
 2 : -CS
 3 : -WE
 4 : Input   D1
 5 : Output  Q1
 6 : Input   D2
 7 : Output  Q2
 8 : GND
 9 : Output  Q3
10 : Input   D3
11 : Output  Q4
12 : Input   D4
13 : Address A3
14 : Address A2
15 : Address A0
16 : +5V

#FAMILY Std,LS,S
```

```
#PROGRAM

BEGIN
PIN[1,2,3,4,6,10,12,13,14,15] : INPUT;
PIN[5,7,9,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[3]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      PIN[12]&PIN[10]&PIN[6]&PIN[4]:=D;
      PIN[3]:=LOW;PIN[3]:=HIGH;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      LOADMODEON;
      PIN[11,9,7,5] : LOAD LOW;
      IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>0 THEN ERROR(1);
      PIN[11,9,7,5] : LOAD HIGH;
      IF (NOT(PIN[11])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[5]))<>D THEN ERROR(1);
      LOADMODEOFF;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 7490,SN7490

#TEXT
Decimal Counter

This component is
composed of a divide-
by-two and a divide-
by-five counter.

#PIN 14
 1 : Clock B
 2 : Input   R0(1)
 3 : Input   R0(2)
 4 : N.C.
 5 : +5V
 6 : Input   R9(1)
 7 : Input   R9(2)
 8 : Output  QC
 9 : Output  QB
10 : GND
11 : Output  QD
12 : Output  QA
13 : N.C.
14 : Clock A

#FAMILY Std,L,LS

#PROGRAM
```

```
BEGIN
PIN[1,2,3,6,7,14] : INPUT;
PIN[8,9,11,12] : OUTPUT;
PIN[10] : GND;
PIN[5] : +5V;
PIN[2,3,6,7]:=LOW;
PIN[1,14]:=HIGH;

PIN[6,7]:=HIGH;
PIN[6,7]:=LOW;
IF (PIN[11]&PIN[8]&PIN[9]&PIN[12])<>9 THEN ERROR(1);

PIN[2,3]:=HIGH;
PIN[2,3]:=LOW;
IF (PIN[11]&PIN[8]&PIN[9]&PIN[12])<>0 THEN ERROR(1);

PIN[14]:=LOW;PIN[14]:=HIGH;
IF PIN[12]<>HIGH THEN ERROR(1);

FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[11]&PIN[8]&PIN[9])<>I THEN ERROR(1);
   PIN[1]:=LOW;PIN[1]:=HIGH;
   END;

ERROR(0);
END.

#NAME 7491,SN7491

#TEXT
8-Bit Shift Register

This component contains
an 8-stage shift register.
Data is serially shifted
in and out of the
register.


#PIN 14
 1 : N.C.
 2 : N.C.
 3 : N.C.
 4 : N.C.
 5 : +5V
 6 : N.C.
 7 : N.C.
 8 : N.C.
 9 : Clock
10 : GND
11 : Input    B
12 : Input    A
13 : Output   Q
14 : Output  -Q

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[9,11,12] : INPUT;
PIN[13,14] : OUTPUT;
PIN[10] : GND;
PIN[5] : +5V;
PIN[9]:=LOW;
```

```
D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[12,11]:=X AND 1;
      X:=X SHR 1;
      PIN[9]:=HIGH;PIN[9]:=LOW;
      END;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[13] SHL 7);
      Y:=(Y SHR 1) OR ((PIN[14] EXOR %1) SHL 7);
      PIN[9]:=HIGH;PIN[9]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 7492,SN7492

#TEXT
12-Stage Counter

This component contains
a divide-by-two and a
divide-by-six counter.

#PIN 14
 1 : Clock B
 2 : N.C.
 3 : N.C.
 4 : N.C.
 5 : +5V
 6 : Input   R0(1)
 7 : Input   R0(2)
 8 : Output  QD
 9 : Output  QC
10 : GND
11 : Output  QB
12 : Output  QA
13 : N.C.
14 : Clock A

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,6,7,14] : INPUT;
PIN[8,9,11,12] : OUTPUT;
PIN[10] : GND;
PIN[5] : +5V;
PIN[1,14]:=HIGH;

PIN[6,7]:=HIGH;PIN[6,7]:=LOW;
IF (PIN[8]&PIN[9]&PIN[11]&PIN[12])<>0 THEN ERROR(1);

PIN[14]:=LOW;PIN[14]:=HIGH;
IF PIN[12]<>HIGH THEN ERROR(1);
```

```
FOR I:=0 TO 2 DO
   BEGIN
   IF (PIN[8]&PIN[9]&PIN[11])<>I THEN ERROR(1);
   PIN[1]:=LOW;PIN[1]:=HIGH;
   END;
FOR I:=4 TO 6 DO
   BEGIN
   IF (PIN[8]&PIN[9]&PIN[11])<>I THEN ERROR(1);
   PIN[1]:=LOW;PIN[1]:=HIGH;
   END;

ERROR(0);
END.

#NAME 7493,SN7493

#TEXT
4-Bit Binary Counter

This component contains
a divide-by-two and a
divide-by-eight counter.

Observe the difference
between this component
and the L- version!

#PIN 14
 1 : Clock B
 2 : Input   R0(1)
 3 : Input   R0(2)
 4 : N.C.
 5 : +5V
 6 : N.C.
 7 : N.C.
 8 : Output  QC
 9 : Output  QB
10 : GND
11 : Output  QD
12 : Output  QA
13 : N.C.
14 : Clock A

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,14] : INPUT;
PIN[8,9,11,12] : OUTPUT;
PIN[10] : GND;
PIN[5] : +5V;
PIN[1,14]:=HIGH;

PIN[2,3]:=HIGH;PIN[2,3]:=LOW;
IF (PIN[11]&PIN[8]&PIN[9]&PIN[12])<>0 THEN ERROR(1);

PIN[14]:=LOW;PIN[14]:=HIGH;
IF PIN[12]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 DO
   BEGIN
   IF (PIN[11]&PIN[8]&PIN[9])<>I THEN ERROR(1);
   PIN[1]:=LOW;PIN[1]:=HIGH;
   END;

ERROR(0);
```

```
END.

#NAME 74L93,SN74L93

#TEXT
4-Bit Binary Counter

This component contains
a divide-by-two and a
divide-by-eight counter.

Observe the difference-
between this component
and the Std- abd LS-
versions!

#PIN 14
 1 : Input   R0(1)
 2 : Input   R0(2)
 3 : N.C.
 4 : +5V
 5 : N.C.
 6 : N.C.
 7 : N.C.
 8 : Clock B
 9 : Output  QB
10 : Output  QC
11 : GND
12 : Output  QD
13 : Output  QA
14 : Clock A

#FAMILY L

#PROGRAM

BEGIN
PIN[1,2,8,14] : INPUT;
PIN[9,10,12,13] : OUTPUT;
PIN[11] : GND;
PIN[4] : +5V;
PIN[8,14]:=HIGH;

PIN[1,2]:=HIGH;PIN[1,2]:=LOW;
IF (PIN[12]&PIN[10]&PIN[9]&PIN[13])<>0 THEN ERROR(1);

PIN[14]:=LOW;PIN[14]:=HIGH;
IF PIN[13]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 DO
   BEGIN
   IF (PIN[12]&PIN[10]&PIN[9])<>I THEN ERROR(1);
   PIN[8]:=LOW;PIN[8]:=HIGH;
   END;

ERROR(0);
END.

#NAME 7494,SN7494

#TEXT
4-Bit Parallel In/Clear
Shift Register

This component contains
a four-stage shift
register with serial
```

```
and parallel data input
and serial output.


#PIN 16
 1 : Input   P1A
 2 : Input   P1B
 3 : Input   P1C
 4 : Input   P1D
 5 : +5V
 6 : Input   PL1
 7 : Input   DS
 8 : Clock
 9 : Output  Q
10 : Reset
11 : Input   P2D
12 : GND
13 : Input   P2C
14 : Input   P2B
15 : Input   PL2
16 : Input   P2A

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,6,7,8,10,11,13,14,15,16] : INPUT;
PIN[9] : OUTPUT;
PIN[12] : GND;
PIN[5] : +5V;
PIN[8]:=LOW;

D:=%0101;
PIN[6,15]:=LOW;
PIN[10]:=HIGH;PIN[10]:=LOW;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[7]:=X AND 1;
      X:=X SHR 1;
      PIN[8]:=HIGH;PIN[8]:=LOW;
      END;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[9] SHL 3);
      PIN[8]:=HIGH;PIN[8]:=LOW;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[10]:=HIGH;PIN[10]:=LOW;
   PIN[1]&PIN[2]&PIN[3]&PIN[4]:=D;
   PIN[6]:=HIGH;PIN[6]:=LOW;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[9] SHL 3);
      PIN[8]:=HIGH;PIN[8]:=LOW;
      END;
```

```
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[10]:=HIGH;PIN[10]:=LOW;
   PIN[16]&PIN[14]&PIN[13]&PIN[11]:=D;
   PIN[15]:=HIGH;PIN[15]:=LOW;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[9] SHL 3);
      PIN[8]:=HIGH;PIN[8]:=LOW;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 7495,SN7495

#TEXT
4-Bit Parallel In/
Parallel Out/Right-/
Left Shift Register

This component contains
a 4-bit shift register.
Parallel input and output
is possible and right-
shift or left-shift
operations can be
performed.

Observe the difference
between this component
and the L- version!

#PIN 14
 1 : Input   Ds
 2 : Input   A
 3 : Input   B
 4 : Input   C
 5 : Input   D
 6 : Mode
 7 : GND
 8 : -SL
 9 : -SR
10 : Output  QD
11 : Output  QC
12 : Output  QB
13 : Output  QA
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,8,9] : INPUT;
PIN[10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
```

```
D:=%0101;
PIN[6]:=LOW;
PIN[8,9]:=HIGH;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[1]:=X AND 1;
      X:=X SHR 1;
      PIN[9]:=LOW;PIN[9]:=HIGH;
      END;
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   PIN[6]:=HIGH;
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=X;
   PIN[8]:=LOW;PIN[8]:=HIGH;
   PIN[6]:=LOW;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>X THEN ERROR(1);
      PIN[1]:=LOW;
      PIN[9]:=LOW;PIN[9]:=HIGH;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74L95,SN74L95

#TEXT
4-Bit Parallel In/
Parallel Out/Right-
Left Shift Register

This component contains
a 4-bit shift register.
Parallel input and output
of data is possible and
right-shift or left-shift
operations can be
performed.

Observe the difference
between this component
and the Std- and LS-
versions!

#PIN 14
 1 : Input   Ds
 2 : Input   B
 3 : Input   C
 4 : +5V
 5 : Input   D
 6 : Mode
 7 : -SR
 8 : -SL
 9 : Output  QD
```

```
10 : Output  QC
11 : GND
12 : Output  QB
13 : Output  QA
14 : Input   A

#FAMILY L

#PROGRAM

BEGIN
PIN[1,2,3,5,6,7,8,14] : INPUT;
PIN[9,10,12,13] : OUTPUT;
PIN[11] : GND;
PIN[4] : +5V;

D:=%0101;
PIN[6]:=LOW;
PIN[7,8]:=HIGH;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[1]:=X AND 1;
      X:=X SHR 1;
      PIN[7]:=LOW;PIN[7]:=HIGH;
      END;
   IF (PIN[13]&PIN[12]&PIN[10]&PIN[9])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   PIN[6]:=HIGH;
   PIN[14]&PIN[2]&PIN[3]&PIN[5]:=X;
   PIN[8]:=LOW;PIN[8]:=HIGH;
   PIN[6]:=LOW;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      IF (PIN[13]&PIN[12]&PIN[10]&PIN[9])<>X THEN ERROR(1);
      PIN[1]:=LOW;
      PIN[7]:=LOW;PIN[7]:=HIGH;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 7496,SN7496

#TEXT
5-Bit Parallel In/
Parallel Out Shift
Register with Clear

This component contains
a 5-stage right-shift
register. The register
features serial and
parallel inputs and
serial and parallel
outputs.
```

```
#PIN 16
 1 : Clock
 2 : Input   A
 3 : Input   B
 4 : Input   C
 5 : +5V
 6 : Input   D
 7 : Input   E
 8 : Load
 9 : Input   DS
10 : Output  QE
11 : Output  QD
12 : GND
13 : Output  QC
14 : Output  QB
15 : Output  QA
16 : -Reset

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,6,7,8,9,16] : INPUT;
PIN[10,11,13,14,15] : OUTPUT;
PIN[12] : GND;
PIN[5] : +5V;
PIN[1]:=LOW;

D:=%01010;
PIN[16]:=LOW;PIN[16]:=HIGH;
PIN[8]:=LOW;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 4 DO
      BEGIN
      PIN[9]:=X AND 1;
      X:=X SHR 1;
      PIN[1]:=HIGH;PIN[1]:=LOW;
      END;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[11]&PIN[10])<>D THEN ERROR(1);
   D:=D EXOR %11111;
   END;

PIN[16]:=LOW;PIN[16]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   PIN[2]&PIN[3]&PIN[4]&PIN[6]&PIN[7]:=X;
   PIN[8]:=HIGH;PIN[8]:=LOW;
   FOR J:=4 DOWNTO 0 DO
      BEGIN
      IF (PIN[15]&PIN[14]&PIN[13]&PIN[11]&PIN[10])<>X THEN ERROR(1);
      PIN[9]:=LOW;
      PIN[1]:=HIGH;PIN[1]:=LOW;
      X:=X SHR 1;
      END;
   D:=D EXOR %11111;
   END;

ERROR(0);
END.

#NAME 7497,SN7497
```

```
#TEXT
Synchronous Pre-
Programmable 6-Bit
Binary Frequency Divider

This component contains
a programmable 6-bit
frequency divider,
often called Bit-Rate-
Multiplier (Modulo 64).


#PIN 16
 1 : Input   B
 2 : Input   E
 3 : Input   F
 4 : Input   A
 5 : Output  -Q
 6 : Cascade Out
 7 : Enable Out
 8 : GND
 9 : Clock
10 : Strobe
11 : Enable
12 : Cascade
13 : Clear
14 : Input   C
15 : Input   D
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,9,10,11,12,13,14,15] : INPUT;
PIN[5,6,7] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9,10,11,12,13]:=LOW;

PIN[12]:=LOW;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 5 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]&PIN[1]&PIN[4]:=1 SHL I;
   Q:=0;C:=0;E:=0;
   FOR J:=0 TO 63 DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      E:=E+PIN[7];
      END;
   IF Q<>(64-(1 SHL I)) THEN ERROR(1);
   IF C<>64 THEN ERROR(1);
   IF E<>63 THEN ERROR(1);
   END;

PIN[12]:=HIGH;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]&PIN[1]&PIN[4]:=1 SHL I;
   Q:=0;C:=0;
   FOR J:=0 TO 63 DO
```

```
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      END;
   IF Q<>(64-(1 SHL I)) THEN ERROR(1);
   IF C<>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 7498,SN7498

#TEXT
Quad 2-to-1-Data Selector/
Multiplexer with Memory

With this component it is
possible to choose between
two 4-bit data sources and
store the chosen data.

#PIN 16
 1 : Input   A2
 2 : Input   A1
 3 : Input   B1
 4 : Input   B2
 5 : Input   C1
 6 : Input   C2
 7 : Input   D2
 8 : GND
 9 : Word Select
10 : Clock
11 : Output  QD
12 : Input   D1
13 : Output  QC
14 : Output  QB
15 : Output  QA
16 : +5V

#FAMILY L

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10,12] : INPUT;
PIN[11,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10]:=HIGH;

PIN[1,2,3,4,5,6,7,12]:=LOW;
PIN[9]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[12]&PIN[5]&PIN[3]&PIN[2]:=D;
   PIN[10]:=LOW;PIN[10]:=HIGH;
   PIN[12]&PIN[5]&PIN[3]&PIN[2]:=0;
   IF (PIN[11]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,2,3,4,5,6,7,12]:=LOW;
PIN[9]:=HIGH;
D:=%0101;
```

```
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7]&PIN[6]&PIN[4]&PIN[1]:=D;
   PIN[10]:=LOW;PIN[10]:=HIGH;
   PIN[7]&PIN[6]&PIN[4]&PIN[1]:=0;
   IF (PIN[11]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

##NAME 7499,SN7499

#TEXT
4-Bit Parallel-In/
Parallel-Out Shift
Register

This component contains a
4-Bit shift register. Data
can be entered and read
out and right-shift and
left-shift operations are
possible.

#PIN 16
 1 : Input   A
 2 : J
 3 : Input   B
 4 : Input   C
 5 : +5V
 6 : Input   D
 7 : Mode Control
 8 : Clock 1
 9 : Clock 2
10 : Output  QD
11 : Output  -QD
12 : Output  QC
13 : GND
14 : Output  QB
15 : Output  QA
16 : -K

#FAMILY L

#PROGRAM

BEGIN
END.

#NAME 74101,SN74101

#TEXT
J-K Flip-Flop with 2x2
AND/OR-Inputs and Preset

This component contains
a fast edge-triggered
J-K flip flop with 2x2
AND/OR inputs and a
preset input.


#PIN 14
 1 : Input   J1A
 2 : Input   J1B
```

```
 3 : Input   J2A
 4 : Input   J2B
 5 : -Preset
 6 : Output  Q
 7 : GND
 8 : Output  -Q
 9 : Input   K1A
10 : Input   K1B
11 : Input   K2A
12 : Input   K2B
13 : Clock
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,2,3,4,9,10,11,12]:=LOW;
PIN[5,13]:=HIGH;

PIN[5]:=LOW;PIN[5]:=HIGH;
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[11]&PIN[12]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=3 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74102,SN74102

#TEXT
J-K Flip-Flop with 3 AND-
Inputs, Preset and Clear
```

```
This component contains
a fast edge-triggered
J-K flip-flop with 3
AND inputs, preset and
clear.

#PIN 14
 1 : N.C.
 2 : -Clear
 3 : Input   J1
 4 : Input   J2
 5 : Input   J3
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : Input   K1
10 : Input   K2
11 : Input   K3
12 : Clock
13 : -Preset
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,5,9,10,11]:=LOW;
PIN[2,12,13]:=HIGH;

PIN[2]:=LOW;PIN[2]:=HIGH;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[11]:=I;
   PIN[12]:=LOW;PIN[12]:=HIGH;
   IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[12]:=LOW;PIN[12]:=HIGH;
   IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

PIN[3,4,5,9,10,11]:=HIGH;
PIN[12]:=LOW;PIN[12]:=HIGH;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74103,SN74103

#TEXT
Dual J-K Flip-Flops
```

```
with Clear

This component contains
two separate J-K flip-
flops with separate clock
and clear inputs.

#PIN 14
 1 : Clock       FF 1
 2 : -Clear      FF 1
 3 : Input    K  FF 1
 4 : +5V
 5 : Clock       FF 2
 6 : -Clear      FF 2
 7 : Input    J  FF 2
 8 : Output  -Q  FF 2
 9 : Output   Q  FF 2
10 : Input    K  FF 2
11 : GND
12 : Input    Q  FF 1
13 : Input   -Q  FF 1
14 : Input    J  FF 1

#FAMILY H,L

#PROGRAM

BEGIN
PIN[1,2,3,5,6,7,10,14] : INPUT;
PIN[8,9,12,13] : OUTPUT;
PIN[11] : GND;
PIN[4] : +5V;
PIN[3,7,10,14]:=LOW;
PIN[1,2,5,6]:=HIGH;

PIN[2,6]:=LOW;PIN[2,6]:=HIGH;
IF (PIN[12]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[14,7]:=I;
   PIN[1,5]:=LOW;PIN[1,5]:=HIGH;
   IF (PIN[12]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[3,10]:=I;
   PIN[1,5]:=LOW;PIN[1,5]:=HIGH;
   IF (PIN[12]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

PIN[14,3,7,10]:=HIGH;
PIN[1,5]:=LOW;PIN[1,5]:=HIGH;
IF (PIN[12]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

ERROR(0);
END.

#NAME 74104,SN74104

#TEXT
J-K Master/Slave Flip-Flop
```

```
with 3 AND-Inputs, Preset
and Clear

This component contains
a J-K Master/Slave flip-
flop with 3 AND inputs,
an additional J-K input,
plus inputs for preset
and clear.


#PIN 14
 1 : Input   JK
 2 : -Preset
 3 : Input   K1
 4 : Input   J1
 5 : Input   J2
 6 : Output   Q
 7 : GND
 8 : Output  -Q
 9 : Clock
10 : Input   K2
11 : Input   K3
12 : Input   J2
13 : -Clear
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,5,9,10,11,12]:=LOW;
PIN[1,2,13]:=HIGH;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

PIN[2]:=LOW;PIN[2]:=HIGH;
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[10]&PIN[11]:=I;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[4]&PIN[5]&PIN[12]:=I;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

PIN[3,4,5,10,11,12]:=HIGH;
PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

PIN[1]:=LOW;
PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
PIN[9]:=HIGH;PIN[9]:=LOW;
```

```
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

ERROR(0);
END.
•
```

```
#NAME 74105,SN74105

#TEXT
J-K Master/Slave Flip-Flop
with 3 AND-Inputs, Preset
and Clear

This component contains
a J-K Master/Slave flip-
flop with 3 AND inputs
(-J2 and -J3 inverted),
an addtional J-K input
JK-Input and preset
and clear.

#PIN 14
 1 : JK
 2 : -Preset
 3 : Input    K1
 4 : Input    J1
 5 : Input   -J2
 6 : Output   Q
 7 : GND
 8 : Output  -Q
 9 : Clock
10 : Input   -K2
11 : Input    K3
12 : Input    J3
13 : -Clear
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,5,9,10,11,12]:=LOW;
PIN[1,2,13]:=HIGH;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

PIN[2]:=LOW;PIN[2]:=HIGH;
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[10]&PIN[11]:=I EXOR %10;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[4]&PIN[5]&PIN[12]:=I EXOR %10;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

PIN[3,4,11,12]:=HIGH;PIN[5,10]:=LOW;
PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[6]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
```

```
PIN[1]:=LOW;
PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[6]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

ERROR(0);
END.

#NAME 74106,SN74106

#TEXT
Dual J-K Flip-Flop with
Preset and Clear

This component contains
two independent J-K flip-
flops with preset and
clear.

#PIN 16
 1 : Clock       FF 1
 2 : -Preset     FF 1
 3 : -Clear      FF 1
 4 : Input    J  FF 1
 5 : +5V
 6 : Clock       FF 2
 7 : -Preset     FF 2
 8 : -Clear      FF 2
 9 : Input    J  FF 2
10 : Output  -Q  FF 2
11 : Output   Q  FF 2
12 : Input    K  FF 2
13 : GND
14 : Output  -Q  FF 1
15 : Output   Q  FF 1
16 : Input    K  FF 1

#FAMILY H

#PROGRAM

BEGIN
PIN[1,2,3,4,6,7,8,9,12,16] : INPUT;
PIN[10,11,14,15] : OUTPUT;
PIN[13] : GND;
PIN[5] : +5V;
PIN[4,9,12,16]:=LOW;
PIN[1,2,3,6,7,8]:=HIGH;

PIN[3,8]:=LOW;PIN[3,8]:=HIGH;
IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
IF (PIN[11]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);

PIN[2,7]:=LOW;PIN[2,7]:=HIGH;
IF (PIN[15]<>HIGH) OR (PIN[14]<>LOW) THEN ERROR(1);
IF (PIN[11]<>HIGH) OR (PIN[10]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[16,12]:=I;
   PIN[1,6]:=LOW;PIN[1,6]:=HIGH;
   IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
   IF (PIN[11]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
```

```
   BEGIN
   PIN[4,9]:=I;
   PIN[1,6]:=LOW;PIN[1,6]:=HIGH;
   IF (PIN[15]<>HIGH) OR (PIN[14]<>LOW) THEN ERROR(1);
   IF (PIN[11]<>HIGH) OR (PIN[10]<>LOW) THEN ERROR(1);
   END;

PIN[4,16,9,12]:=HIGH;
PIN[1,6]:=LOW;PIN[1,6]:=HIGH;
IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
IF (PIN[11]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74107,SN74107

#TEXT
Dual J-K Flip-Flops
with Clear

This component contains
two separate J-K flip-
flops with independent
clock and clear inputs.

#PIN 14
 1 : Input    J  FF 1
 2 : Output  -Q  FF 1
 3 : Output   Q  FF 1
 4 : Input    K  FF 1
 5 : Output   Q  FF 2
 6 : Output  -Q  FF 2
 7 : GND
 8 : Input    J  FF 2
 9 : Clock       FF 2
10 : -Clear      FF 2
11 : Input    K  FF 2
12 : Clock       FF 1
13 : -Clear      FF 1
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,4,8,9,10,11,12,13] : INPUT;
PIN[2,3,5,6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,8,9,11,12]:=LOW;
PIN[10,13]:=HIGH;

PIN[13,10]:=LOW;PIN[13,10]:=HIGH;
IF (PIN[3]<>LOW) OR (PIN[2]<>HIGH) THEN ERROR(1);
IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[1,8]:=I;
   PIN[12,9]:=HIGH;PIN[12,9]:=LOW;
   IF (PIN[3]<>HIGH) OR (PIN[2]<>LOW) THEN ERROR(1);
   IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
```

```
   BEGIN
   PIN[4,11]:=I;
   PIN[12,9]:=HIGH;PIN[12,9]:=LOW;
   IF (PIN[3]<>LOW) OR (PIN[2]<>HIGH) THEN ERROR(1);
   IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;

PIN[1,4,8,11]:=HIGH;
PIN[12,9]:=HIGH;PIN[12,9]:=LOW;
IF (PIN[3]<>HIGH) OR (PIN[2]<>LOW) THEN ERROR(1);
IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

ERROR(0);
END.

#NAME 74108,SN74108

#TEXT
Dual J-K Flip-Flops
with Preset, Combined
Clear and Combined Clock


This component contains
two separate, fast J-K
flip-flops with combined
clock and combined clear
inputs. The preset inputs
are independent.


#PIN 14
 1 : Input    K  FF 1
 2 : Output   Q  FF 1
 3 : Output  -Q  FF 1
 4 : Input    J  FF 1
 5 : Output  -Q  FF 2
 6 : Output   Q  FF 2
 7 : GND
 8 : Input    K  FF 2
 9 : Clock
10 : -Preset    FF 2
11 : Input    J  FF 2
12 : -Clear
13 : -Preset    FF 1
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[1,4,8,9,10,11,12,13] : INPUT;
PIN[2,3,5,6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,8,11]:=LOW;
PIN[9,10,12,13]:=HIGH;

PIN[12]:=LOW;PIN[12]:=HIGH;
IF (PIN[2]<>LOW) OR (PIN[3]<>HIGH) THEN ERROR(1);
IF (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);

PIN[13,10]:=LOW;PIN[13,10]:=HIGH;
IF (PIN[2]<>HIGH) OR (PIN[3]<>LOW) THEN ERROR(1);
IF (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
```

```
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[1,8]:=I;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   IF (PIN[2]<>LOW) OR (PIN[3]<>HIGH) THEN ERROR(1);
   IF (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[4,11]:=I;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   IF (PIN[2]<>HIGH) OR (PIN[3]<>LOW) THEN ERROR(1);
   IF (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
   END;

PIN[1,4,8,11]:=HIGH;
PIN[9]:=LOW;PIN[9]:=HIGH;
IF (PIN[2]<>HIGH) OR (PIN[3]<>LOW) THEN ERROR(1);
IF (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);

ERROR(0);
END.

#NAME 74109,SN74109

#TEXT
Dual J-K Flip-Flops
with Preset and Clear

This component contains
two separate J-K flip-
flops with preset, clear
and inverted K- inputs.


#PIN 16
 1 : Clear       FF 1
 2 : Input    J  FF 1
 3 : Input   -K  FF 1
 4 : Clock       FF 1
 5 : -Preset     FF 1
 6 : Output   Q  FF 1
 7 : Output  -Q  FF 1
 8 : GND
 9 : Output  -Q  FF 2
10 : Output   Q  FF 2
11 : -Reset      FF 2
12 : Clock       FF 2
13 : Input   -K  FF 2
14 : Input    J  FF 2
15 : -Clear      FF 2
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,11,12,13,14,15] : INPUT;
PIN[6,7,9,10] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,4,12,14]:=LOW;
PIN[1,3,5,11,13,15]:=HIGH;

PIN[1,15]:=LOW;PIN[1,15]:=HIGH;
```

```
IF (PIN[6]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);
IF (PIN[10]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);

PIN[5,11]:=LOW;PIN[5,11]:=HIGH;
IF (PIN[6]<>HIGH) OR (PIN[7]<>LOW) THEN ERROR(1);
IF (PIN[10]<>HIGH) OR (PIN[9]<>LOW) THEN ERROR(1);

FOR I:=0 TO 1 DO
   BEGIN
   PIN[3,13]:=I;
   PIN[4,12]:=HIGH;PIN[4,12]:=LOW;
   IF (PIN[6]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);
   IF (PIN[10]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[2,14]:=I;
   PIN[4,12]:=HIGH;PIN[4,12]:=LOW;
   IF (PIN[6]<>HIGH) OR (PIN[7]<>LOW) THEN ERROR(1);
   IF (PIN[10]<>HIGH) OR (PIN[9]<>LOW) THEN ERROR(1);
   END;

PIN[2,14]:=HIGH;PIN[3,13]:=LOW;
PIN[4,12]:=HIGH;PIN[4,12]:=LOW;
IF (PIN[6]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);
IF (PIN[10]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74110,SN74110

#TEXT
J-K Master/Slave Flip-
Flop with 3 AND Inputs,
Preset and Clear

This component contains
a J-K master/slave flip-
flop with 3 AND-inputs,
preset, clear and input
latch.

#PIN 14
 1 : N.C.
 2 : -Clear
 3 : Input   J1
 4 : Input   J2
 5 : Input   J3
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : Input   K1
10 : Input   K2
11 : Input   K3
12 : Clock
13 : -Preset
14 : +5V

#FAMILY H

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
```

```
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,5,9,10,11,12]:=LOW;
PIN[2,13]:=HIGH;

PIN[2]:=LOW;PIN[2]:=HIGH;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[11]:=I;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;
FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

PIN[3,4,5,9,10,11]:=HIGH;
PIN[12]:=HIGH;PIN[12]:=LOW;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74111,SN74111

#TEXT
Dual J-K Flip-Flops with
Preset and Clear

This component contains
two separate J-K flip-
flops with preset and
clear.

#PIN 16
 1 : Input    K  FF 1
 2 : -Preset     FF 1
 3 : -Clear      FF 1
 4 : Input    J  FF 1
 5 : Clock       FF 1
 6 : Output  -Q  FF 1
 7 : Output   Q  FF 1
 8 : GND
 9 : Output   Q  FF 2
10 : Output  -Q  FF 2
11 : Clock       FF 2
12 : Input    J  FF 2
13 : -Clear      FF 2
14 : -Preset     FF 2
15 : Input    K  FF 2
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,5,11,12,13,14,15] : INPUT;
PIN[6,7,9,10] : OUTPUT;
```

```
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,4,5,11,12,15]:=LOW;
PIN[2,3,13,14]:=HIGH;

PIN[3,13]:=LOW;PIN[3,13]:=HIGH;
IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);

PIN[2,14]:=LOW;PIN[2,14]:=HIGH;
IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
IF (PIN[9]<>HIGH) OR (PIN[10]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[1,15]:=I;
   PIN[5,11]:=HIGH;PIN[5,11]:=LOW;
   IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   IF (PIN[9]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[4,12]:=I;
   PIN[5,11]:=HIGH;PIN[5,11]:=LOW;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   IF (PIN[9]<>HIGH) OR (PIN[10]<>LOW) THEN ERROR(1);
   END;

PIN[1,4,12,15]:=HIGH;
PIN[5,11]:=HIGH;PIN[5,11]:=LOW;
IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[10]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74112,SN74112

#TEXT
Dual J-K Flip-Flops with
Preset and Clear

This component contains
two separate J-K flip-
flops with preset and
clear. Triggering occurs
on the negative edge
of the clock.

#PIN 16
 1 : Clock       FF 1
 2 : Input    K  FF 1
 3 : Input    J  FF 1
 4 : -Preset     FF 1
 5 : Output   Q  FF 1
 6 : Output  -Q  FF 1
 7 : Output   Q  FF 2
 8 : GND
 9 : Output   Q  FF 2
10 : -Preset     FF 2
11 : Input    J  FF 2
12 : Input    K  FF 2
13 : Clock       FF 2
14 : -Clear      FF 2
15 : -Clear      FF 1
16 : +5V
```

```
#FAMILY ALS,AS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,10,11,12,13,14,15] : INPUT;
PIN[5,6,7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,3,11,12]:=LOW;
PIN[1,4,10,13,14,15]:=HIGH;

PIN[15,14]:=LOW;PIN[15,14]:=HIGH;
IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);

PIN[4,10]:=LOW;PIN[4,10]:=HIGH;
IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
IF (PIN[9]<>HIGH) OR (PIN[7]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[2,12]:=I;
   PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
   IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   IF (PIN[9]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[3,11]:=I;
   PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
   IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   IF (PIN[9]<>HIGH) OR (PIN[7]<>LOW) THEN ERROR(1);
   END;

PIN[3,2,11,12]:=HIGH;
PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74113,SN74113

#TEXT
Dual J-K Flip-Flops
with Preset

This component contains
two separate J-K flip-
flops with independent
clock and preset inputs.
Triggering occurs on the
negative edge of the
clock pulse.

#PIN 14
 1 : Clock       FF 1
 2 : Input    K  FF 1
 3 : Input    J  FF 1
 4 : -Reset      FF 1
 5 : Output   Q  FF 1
 6 : Output  -Q  FF 1
 7 : GND
```

```
 8 : Output  -Q  FF 2
 9 : Output   Q  FF 2
10 : -Preset     FF 2
11 : Input    J  FF 2
12 : Input    K  FF 2
13 : Clock       FF 2
14 : +5V

#FAMILY ALS,AS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,10,11,12,13] : INPUT;
PIN[5,6,8,9] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[2,3,11,12]:=LOW;
PIN[1,4,10,13]:=HIGH;

PIN[4,10]:=LOW;PIN[4,10]:=HIGH;
IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[2,12]:=I;
   PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
   IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[3,11]:=I;
   PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
   IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

PIN[2,3,11,12]:=HIGH;
PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74114,SN74114

#TEXT
Dual J-K Flip-Flops
with Preset, Combined
Clear and Combined
Clock

This component contains
two J-K flip-flops with
preset, combined clear
and combined clock.
Triggering occurs on the
negative edge of the
clock pulse.


#PIN 14
 1 : -Clear
```

```
 2 : Input    K  FF 1
 3 : Input    J  FF 1
 4 : -Reset      FF 1
 5 : Output   Q  FF 1
 6 : Output  -Q  FF 1
 7 : GND
 8 : Output  -Q  FF 2
 9 : Output   Q  FF 2
10 : -Preset     FF 2
11 : Input    J  FF 2
12 : Input    K  FF 2
13 : Clock
14 : +5V

#FAMILY ALS,AS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,10,11,12,13] : INPUT;
PIN[5,6,8,9] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[2,3,11,12]:=LOW;
PIN[1,4,10,13]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

PIN[4,10]:=LOW;PIN[4,10]:=HIGH;
IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[2,12]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[3,11]:=I;
   PIN[13]:=LOW;PIN[13]:=HIGH;
   IF (PIN[5]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   IF (PIN[9]<>HIGH) OR (PIN[8]<>LOW) THEN ERROR(1);
   END;

PIN[2,3,11,12]:=HIGH;
PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
IF (PIN[9]<>LOW) OR (PIN[8]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 74115,SN74115

#TEXT
Dual J-K Master/Slave
Flip-Flops with Clear

This component contains
two separate J-K flip-
flops with independent
```

```
clock and clear inputs.
Triggering occurs on the
positive edge of the
clock.

#PIN 14
 1 : Input    K  FF 1
 2 : -Clear      FF 1
 3 : Input    J  FF 1
 4 : Clock       FF 1
 5 : Output  -Q  FF 1
 6 : Output   Q  FF 1
 7 : GND
 8 : Output   Q  FF 2
 9 : Output  -Q  FF 2
10 : Clock       FF 2
11 : Input    J  FF 2
12 : -Clear      FF 2
13 : Input    K  FF 2
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,10,11,12,13] : INPUT;
PIN[5,6,8,9] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,3,4,10,11,13]:=LOW;
PIN[2,12]:=HIGH;

PIN[2,12]:=LOW;PIN[2,12]:=HIGH;
IF (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
IF (PIN[8]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[3,11]:=I;
   PIN[4,10]:=HIGH;PIN[4,10]:=LOW;
   IF (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
   IF (PIN[8]<>HIGH) OR (PIN[9]<>LOW) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[1,13]:=I;
   PIN[4,10]:=HIGH;PIN[4,10]:=LOW;
   IF (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   IF (PIN[8]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);
   END;

PIN[1,3,11,13]:=HIGH;
PIN[4,10]:=HIGH;PIN[4,10]:=LOW;
IF (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
IF (PIN[8]<>HIGH) OR (PIN[9]<>LOW) THEN ERROR(1);

ERROR(0);
END.

#NAME 74118,SN74118

#TEXT
Hex RS-Buffer with
Combined Reset
```

```
This component contains
six RS buffers (latches)
with combined reset.


#PIN 16
 1 : Input   -S1
 2 : Output   Q1
 3 : Output   Q2
 4 : Input   -S2
 5 : Output   Q3
 6 : Input   -S3
 7 : N.C.
 8 : GND
 9 : Input   -R
10 : Input   -S4
11 : Output   Q4
12 : Input   -S5
13 : Output   Q5
14 : Output   Q6
15 : Input   -S6
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,4,6,9,10,12,15] : INPUT;
PIN[2,3,5,11,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=HIGH;

D:=%101010;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[1,4,6,10,12,15]:=HIGH;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   IF (PIN[14]&PIN[13]&PIN[11]&
       PIN[5]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[15]&PIN[12]&PIN[10]&PIN[6]&PIN[4]&PIN[1]:=D;
   IF (NOT(PIN[14])&NOT(PIN[13])&NOT(PIN[11])&
       NOT(PIN[5])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 74120,SN74120

#TEXT
Dual Pulsesynchronizer/
Driver

This component contains
two separate pulse-
synchronizers.

#PIN 16
 1 : Input    M  Sychr.1
 2 : Input   -S1 Sychr.1
 3 : Input   -S2 Sychr.1
 4 : Input   -R  Sychr.1
```

```
 5 : Input    C  Sychr.1
 6 : Output   Q  Sychr.1
 7 : Output  -Q  Sychr.1
 8 : GND
 9 : Output  -Q  Sychr.2
10 : Output   Q  Sychr.2
11 : Input    C  Sychr.2
12 : Input   -R  Sychr.2
13 : Input   -S1 Sychr.2
14 : Input   -S2 Sychr.2
15 : Input    M  Sychr.2
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,5,11,12,13,14,15] : INPUT;
PIN[6,7,9,10] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,3,5,11,13,14]:=LOW;

PIN[1]:=HIGH;PIN[4]:=LOW;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>I) OR NOT(PIN[7]<>I) THEN ERROR(1);
   END;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);
   END;
PIN[1]:=LOW;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>I) OR NOT(PIN[7]<>I) THEN ERROR(1);
   END;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>I) OR NOT(PIN[7]<>I) THEN ERROR(1);
   END;
PIN[2,3]:=HIGH;PIN[4]:=HIGH;
BEGIN
PIN[2]:=LOW;PIN[2]:=HIGH;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>I) OR NOT(PIN[7]<>I) THEN ERROR(1);
   END;
PIN[4]:=LOW;PIN[4]:=HIGH;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);
   END;
PIN[3]:=LOW;PIN[3]:=HIGH;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>I) OR NOT(PIN[7]<>I) THEN ERROR(1);
   END;
PIN[4]:=LOW;PIN[4]:=HIGH;
```

```
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5]:=I;
   IF (PIN[6]<>LOW) OR (PIN[7]<>HIGH) THEN ERROR(1);
   END;

PIN[15]:=HIGH;PIN[12]:=LOW;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>I) OR NOT(PIN[9]<>I) THEN ERROR(1);
   END;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);
   END;
PIN[15]:=LOW;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>I) OR NOT(PIN[9]<>I) THEN ERROR(1);
   END;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>I) OR NOT(PIN[9]<>I) THEN ERROR(1);
   END;
PIN[13,14]:=HIGH;PIN[12]:=HIGH;
PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>I) OR NOT(PIN[9]<>I) THEN ERROR(1);
   END;
PIN[12]:=LOW;PIN[12]:=HIGH;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);
   END;
PIN[14]:=LOW;PIN[14]:=HIGH;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>I) OR NOT(PIN[9]<>I) THEN ERROR(1);
   END;
PIN[12]:=LOW;PIN[12]:=HIGH;
FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[11]:=I;
   IF (PIN[10]<>LOW) OR (PIN[9]<>HIGH) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74121,SN74121

#TEXT
Monostable Multivibrator
with Schmitt Trigger
Input

This device contains
a non-retriggerable
one-shot with comple-
```

```
mentary outputs.

#PIN 14
 1 : Output  -Q
 2 : N.C.
 3 : Input   A1
 4 : Input   A2
 5 : Input   B
 6 : Output  Q
 7 : GND
 8 : N.C.
 9 : R int.
10 : C
11 : C/R
12 : N.C.
13 : N.C.
14 : +5V

#FAMILY Std,L

#PROGRAM

BEGIN
PIN[3,4,5] : INPUT;
PIN[1,6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[14,11,10] : RC;

PIN[3,4,5]:=LOW;
IF PIN[6]<>LOW THEN ERROR(1);
C:=256;
PIN[5]:=HIGH;
IF PIN[6]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[6]<>LOW) DO
   C:=C-1;
IF PIN[6]<>LOW THEN ERROR(1);

PIN[3,4,5]:=HIGH;
IF PIN[1]<>HIGH THEN ERROR(1);
C:=256;
PIN[3]:=LOW;
IF PIN[1]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[1]<>HIGH) DO
   C:=C-1;
IF PIN[1]<>HIGH THEN ERROR(1);

PIN[14] : +5V OFF;
PIN[7] : GND OFF;

FOR C:=0 TO 512 DO ;

ERROR(0);
END.

#NAME 74122,SN74122

#TEXT
Dual Retriggerable Mono-
stable Multivibrator
with Clear

This device contains
two retriggerable one-
shots with complementary
outputs and a clear
```

```
input.

#PIN 14
 1 : Input   A1
 2 : Input   A2
 3 : Input   B1
 4 : Input   B2
 5 : -Clear
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : R int.
10 : N.C.
11 : C
12 : N.C.
13 : C/R
14 : +5V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[14,13,11] : RC;

PIN[1,2,3,4]:=LOW;
PIN[5]:=LOW;
PIN[5]:=HIGH;
IF PIN[8]<>LOW THEN ERROR(1);
C:=256;
PIN[3,4]:=HIGH;
IF PIN[8]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[8]<>LOW) DO
   C:=C-1;
IF PIN[8]<>LOW THEN ERROR(1);

PIN[1,2,3,4]:=HIGH;
PIN[5]:=LOW;
PIN[5]:=HIGH;
IF PIN[6]<>HIGH THEN ERROR(1);
C:=256;
PIN[1]:=LOW;
IF PIN[6]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[6]<>HIGH) DO
   C:=C-1;
IF PIN[6]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 74123,SN74123

#TEXT
Dual Retriggerable Mono-
stable Multivibrator
with Clear

This device contains
two retriggerable one-
shots with complementary
outputs and a clear
input.
```

```
#PIN 16
 1 : Input    A MF 1
 2 : Input    B MF 1
 3 : -Clear     MF 1
 4 : Output  -Q MF 1
 5 : Output   Q MF 2
 6 : C ext.     MF 2
 7 : C/R        MF 2
 8 : GND
 9 : Input    A MF 2
10 : Input    B MF 2
11 : -Clear     MF 2
12 : Output  -Q MF 2
13 : Output   Q MF 1
14 : C ext.     MF 1
15 : C/R        MF 1
16 : +5V

#FAMILY Std,ALS,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,9,10,11] : INPUT;
PIN[4,5,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[16,15,14] : RC;

PIN[1,2]:=LOW;
PIN[3]:=LOW;
PIN[3]:=HIGH;
IF PIN[13]<>LOW THEN ERROR(1);
C:=256;
PIN[2]:=HIGH;
IF PIN[13]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[13]<>LOW) DO
   C:=C-1;
IF PIN[13]<>LOW THEN ERROR(1);

PIN[1,2]:=HIGH;
PIN[3]:=LOW;
PIN[3]:=HIGH;
IF PIN[4]<>HIGH THEN ERROR(1);
C:=256;
PIN[1]:=LOW;
IF PIN[4]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[4]<>HIGH) DO
   C:=C-1;
IF PIN[4]<>HIGH THEN ERROR(1);

PIN[16,15,14] : RC OFF;
PIN[16,7,6] : RC;

PIN[9,10]:=LOW;
PIN[11]:=LOW;
PIN[11]:=HIGH;
IF PIN[5]<>LOW THEN ERROR(1);
C:=256;
PIN[10]:=HIGH;
IF PIN[5]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[5]<>LOW) DO
   C:=C-1;
IF PIN[5]<>LOW THEN ERROR(1);
```

```
PIN[9,10]:=HIGH;
PIN[11]:=LOW;
PIN[11]:=HIGH;
IF PIN[12]<>HIGH THEN ERROR(1);
C:=256;
PIN[9]:=LOW;
IF PIN[12]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[12]<>HIGH) DO
   C:=C-1;
IF PIN[12]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 74125,SN74125

#TEXT
TRI-STATE Quad Buffers

This device contains
four separate non-
inverting buffers with
TRI-STATE outputs.

#PIN 14
 1 : -Control  Gate   1
 2 : Input     Gate   1
 3 : Output    Gate   1
 4 : -Control  Gate   2
 5 : Input     Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input     Gate   3
10 : -Control  Gate   3
11 : Output    Gate   4
12 : Input     Gate   4
13 : -Control  Gate   4
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,10,13]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[5]&PIN[9]&PIN[12]:=D;
   PIN[1,4,10,13]:=LOW;
   IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>D THEN ERROR(1);
   PIN[1,4,10,13]:=HIGH;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[3,6,8,11] : LOAD LOW;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,6,8,11] : LOAD HIGH;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;
```

```
ERROR(0);
END.

#NAME 74126,SN74126

#TEXT
TRI-STATE Quad Buffers

This device contains
four separate non-
inverting buffers with
TRI-STATE outputs.

#PIN 14
 1 : Control   Gate   1
 2 : Input     Gate   1
 3 : Output    Gate   1
 4 : Control   Gate   2
 5 : Input     Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input     Gate   3
10 : Control   Gate   3
11 : Output    Gate   4
12 : Input     Gate   4
13 : Control   Gate   4
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,10,13]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[5]&PIN[9]&PIN[12]:=D;
   PIN[1,4,10,13]:=HIGH;
   IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>D THEN ERROR(1);
   PIN[1,4,10,13]:=LOW;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[3,6,8,11] : LOAD LOW;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,6,8,11] : LOAD HIGH;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74128,SN74128

#TEXT
Quad 2-Input 50-ê-NOR
Buffer
```

```
This device contains
four separate 50-ê-NOR
buffers each with two
inputs.

#PIN 14
 1 : Output    Gate   1
 2 : Input   1 Gate   1
 3 : Input   2 Gate   1
 4 : Output    Gate   2
 5 : Input   1 Gate   2
 6 : Input   2 Gate   2
 7 : GND
 8 : Input   1 Gate   3
 9 : Input   2 Gate   3
10 : Output    Gate   3
11 : Input   1 Gate   4
12 : Input   2 Gate   4
13 : Output    Gate   4
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,5,6,8,9,11,12] : INPUT;
PIN[1,4,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[11]&PIN[12]:=I;
   IF PIN[1]<>(PIN[2] NOR PIN[3]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NOR PIN[9]) THEN ERROR(1);
   IF PIN[13]<>(PIN[11] NOR PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74131,SN74131

#TEXT
3-Bit-Binary Decoder/
Demultiplexer (3-to-8)
with Address Latches

This device contains
a high-speed 3-to-8-
decoder/demultiplexer with
address latches.

#PIN 16
 1 : Input    A0
 2 : Input    A1
 3 : Input    A2
 4 : Clock
 5 : -E2
 6 : E1
 7 : Output  -Q7
 8 : GND
```

```
 9 : Output  -Q6
10 : Output  -Q5
11 : Output  -Q4
12 : Output  -Q3
13 : Output  -Q2
14 : Output  -Q1
15 : Output  -Q0
16 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6] : INPUT;
PIN[7,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[5]:=LOW;
PIN[6]:=HIGH;
FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[1]:=I;
   PIN[4]:=HIGH;
   PIN[4]:=LOW;
   PIN[3]&PIN[2]&PIN[1]:=%000;
   IF (NOT(PIN[7])&NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11])
      &NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15]))
      <>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74132,SN74132

#TEXT
Quad 2-Input NAND
Schmitt Trigger

This device contains
four NAND gates each
with 2 inputs and
Schmitt Trigger func-
tion.

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Output    Gate   1
 4 : Input   1 Gate   2
 5 : Input   2 Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input   1 Gate   3
10 : Input   2 Gate   3
11 : Output    Gate   4
12 : Input   1 Gate   4
13 : Input   2 Gate   4
14 : +5V

#FAMILY Std,LS,S

#PROGRAM
```

```
BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74133,SN74133

#TEXT
13-Input NAND Gate

This device contains
a NAND gate with 13
inputs.

#PIN 16
 1 : Input    1
 2 : Input    2
 3 : Input    3
 4 : Input    4
 5 : Input    5
 6 : Input    6
 7 : Input    7
 8 : GND
 9 : Output
10 : Input    8
11 : Input    9
12 : Input   10
13 : Input   11
14 : Input   12
15 : Input   13
16 : +5V

#FAMILY ALS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,10,11,12,13,14,15] : INPUT;
PIN[9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[1,2,3,4,5,6,7,10,11,12,13,14,15]:=LOW;
IF PIN[9]<>HIGH THEN ERROR(1);

FOR I:=0 TO 12 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]
   &PIN[10]&PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]:=1 SHL I;
   IF PIN[9]<>HIGH THEN ERROR(1);
   END;
```

```
PIN[1,2,3,4,5,6,7,10,11,12,13,14,15]:=HIGH;
IF PIN[9]<>LOW THEN ERROR(1);

ERROR(0);
END.

#NAME 74134,SN74134

#TEXT
TRI-STATE 12-Input
NAND Gate

This device contains
a NAND-Gate with 12
inputs and a TRI-STATE
output.

#PIN 16
 1 : Input    1
 2 : Input    2
 3 : Input    3
 4 : Input    4
 5 : Input    5
 6 : Input    6
 7 : Input    7
 8 : GND
 9 : Output
10 : Input    8
11 : Input    9
12 : Input   10
13 : Input   11
14 : Input   12
15 : -E
16 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,10,11,12,13,14,15] : INPUT;
PIN[9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=HIGH;

PIN[1,2,3,4,5,6,7,10,11,12,13,14]:=LOW;
PIN[15]:=LOW;
IF PIN[9]<>HIGH THEN ERROR(1);
PIN[15]:=HIGH;

FOR I:=0 TO 11 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]
   &PIN[10]&PIN[11]&PIN[12]&PIN[13]&PIN[14]:=1 SHL I;
   PIN[15]:=LOW;
   IF PIN[9]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;
   END;

PIN[1,2,3,4,5,6,7,10,11,12,13,14]:=HIGH;
PIN[15]:=LOW;
IF PIN[9]<>LOW THEN ERROR(1);
PIN[15]:=HIGH;

LOADMODEON;
```

```
PIN[9] : LOAD LOW;
IF PIN[9]<>LOW THEN ERROR(1);
PIN[9] : LOAD HIGH;
IF PIN[9]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74135,SN74135

#TEXT
Quad 2-Input Exclusive-
OR/NOR Gate

This device contains
four exclusive OR gates,
which can be switched
to an exclusive NOR-
gate via an additional
gate.

#PIN 16
 1 : Input   A Gate   1
 2 : Input   B Gate   1
 3 : Output  Q Gate   1
 4 : Input   C Gate   1+2
 5 : Input   A Gate   2
 6 : Input   B Gate   2
 7 : Output  Q Gate   2
 8 : GND
 9 : Output  Q Gate   3
10 : Input   A Gate   3
11 : Input   B Gate   3
12 : Input   C Gate   3+4
13 : Output  Q Gate   4
14 : Input   A Gate   4
15 : Input   B Gate   4
16 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,4,5,6,10,11,12,14,15] : INPUT;
PIN[3,7,9,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[4,12]:=I;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[1]&PIN[2]:=J;
      PIN[5]&PIN[6]:=J;
      PIN[10]&PIN[11]:=J;
      PIN[14]&PIN[15]:=J;
      IF PIN[3]<>(PIN[1] EXOR PIN[2] EXOR PIN[4]) THEN
ERROR(1);
      IF PIN[7]<>(PIN[5] EXOR PIN[6] EXOR PIN[4]) THEN
ERROR(1);
      IF PIN[9]<>(PIN[10] EXOR PIN[11] EXOR PIN[12]) THEN
ERROR(1);
      IF PIN[13]<>(PIN[14] EXOR PIN[15] EXOR PIN[12]) THEN
ERROR(1);
```

```
      END;
   END;

ERROR(0);
END.

#NAME 74136,SN74136

#TEXT
Quad 2-Input Exclusive-
OR Gate (O.C.)

This device contains
four exclusive OR gates
each with 2 inputs; the
outputs are OPEN
COLLECTOR.

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Output    Gate   1
 4 : Input   1 Gate   2
 5 : Input   2 Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input   1 Gate   3
10 : Input   2 Gate   3
11 : Output    Gate   4
12 : Input   1 Gate   4
13 : Input   2 Gate   4
14 : +5V

#FAMILY Std,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[3,6,8,11] : LOAD HIGH;
   IF PIN[3]<>(PIN[1] EXOR PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] EXOR PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] EXOR PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] EXOR PIN[13]) THEN ERROR(1);
   PIN[3,6,8,11] : LOAD LOW;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   IF PIN[11]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.
```

```
#NAME 74137,SN74137

#TEXT
3-to-8 Line Decoder
with Address Latches

This device contains
a high-speed 3-to-8-
decoder/demultiplexer
with address latches.

#PIN 16
 1 : Input    A0
 2 : Input    A1
 3 : Input    A2
 4 : -Latch Enable
 5 : -E1
 6 : E2
 7 : Output  -Q7
 8 : GND
 9 : Output  -Q6
10 : Output  -Q5
11 : Output  -Q4
12 : Output  -Q3
13 : Output  -Q2
14 : Output  -Q1
15 : Output  -Q0
16 : +5V

#FAMILY ALS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6] : INPUT;
PIN[7,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[5]:=LOW;
PIN[6]:=HIGH;
FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[1]:=I;
   PIN[4]:=LOW;
   PIN[4]:=HIGH;
   PIN[3]&PIN[2]&PIN[1]:=%000;
   IF (NOT(PIN[7])&NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11])
      &NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15]))
      <>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74138,SN74138

#TEXT
3-to-8 Line Decoder

This device contains
a high speed 3-zu-8-
decoder/demultiplexer
with 3 enable inputs.

#PIN 16
 1 : Input    A0
```

```
 2 : Input    A1
 3 : Input    A2
 4 : -E1
 5 : -E2
 6 : E3
 7 : Output  -Q7
 8 : GND
 9 : Output  -Q6
10 : Output  -Q5
11 : Output  -Q4
12 : Output  -Q3
13 : Output  -Q2
14 : Output  -Q1
15 : Output  -Q0
16 : +5V

#FAMILY ALS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6] : INPUT;
PIN[7,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[4,5]:=LOW;
PIN[6]:=HIGH;
FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[1]:=I;
   IF (NOT(PIN[7])&NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11])
      &NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15]))
      <>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74139,SN74139

#TEXT
Dual 2-to-4 Line Decoder

This device contains
two independent one-of-
four (or 2-to-4)
decoders which can be
used either as decoders
or distributors. With
an external inverter
they can also be used
as a 1-to-8 decoder or
distributor.

#PIN 16
 1 : -Enable     DMP A
 2 : Input    A  DMP A
 3 : Input    B  DMP A
 4 : Output  -Q0 DMP A
 5 : Output  -Q1 DMP A
 6 : Output  -Q2 DMP A
 7 : Output  -Q3 DMP A
 8 : GND
 9 : Output  -Q3 DMP B
10 : Output  -Q2 DMP B
11 : Output  -Q1 DMP B
```

```
12 : Output  -Q0 DMP B
13 : Input    B  DMP B
14 : Input    A  DMP B
15 : -Enable     DMP B
16 : +5V

#FAMILY F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]:=I;
   PIN[13]&PIN[14]:=I;
   PIN[1,15]:=LOW;
   IF (NOT(PIN[7])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[4]))
      <>(1 SHL I) THEN ERROR(1);
   IF (NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11])&NOT(PIN[12]))
      <>(1 SHL I) THEN ERROR(1);
   PIN[1,15]:=HIGH;
   IF (NOT(PIN[7])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[4]))<>0
THEN ERROR(1);
   IF (NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11])&NOT(PIN[12]))<>0
THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74140,SN74140

#TEXT
Dual 4-Input 50-ê-NAND
Gate

This device contains
two independent NAND
gates each with 4
inputs and high fan-
out.

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : N.C.
 4 : Input   3 Gate   1
 5 : Input   4 Gate   1
 6 : Output    Gate   1
 7 : GND
 8 : Output    Gate   2
 9 : Input   1 Gate   2
10 : Input   2 Gate   2
11 : N.C.
12 : Input   3 Gate   2
13 : Input   4 Gate   2
14 : +5V

#FAMILY S

#PROGRAM
```

```
BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5])
THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13])
THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74141,SN74141

#TEXT
BCD-to-Decimal Decoder/
Driver (60V)

This device converts a
standard BCD code with
4 bits into a decimal
number from 0 to 9.
The outputs have a
maximum output voltage
of 60V.

#PIN 16
 1 : Output  8
 2 : Output  9
 3 : Input   A
 4 : Input   D
 5 : +5V...+60V
 6 : Input   B
 7 : Input   C
 8 : Output  2
 9 : Output  3
10 : Output  7
11 : Output  6
12 : GND
13 : Output  4
14 : Output  5
15 : Output  1
16 : Output  0

#FAMILY Std

#PROGRAM

BEGIN
PIN[3,4,6,7] : INPUT;
PIN[1,2,8,9,10,11,13,14,15,16] : OUTPUT;
PIN[12] : GND;
PIN[5] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[7]&PIN[6]&PIN[3]:=I;
   LOADMODEON;
   PIN[2,1,10,11,14,13,9,8,15,16] : LOAD HIGH;
   IF
```

```
(NOT(PIN[2])&NOT(PIN[1])&NOT(PIN[10])&NOT(PIN[11])&NOT(PIN[1-
4])

&NOT(PIN[13])&NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[15])&NOT(PIN[1-
6]))
      <>(1 SHL I) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

##NAME 74142,SN74142

#TEXT
Decade Counter/Buffer/
Decoder/Display Driver
(60V)

This device contains a
decade counter, a buffer,
and a decoder with display
driver for numerical
display tubes.

#PIN 16
 1 : -Clear
 2 : Output  7
 3 : Output  6
 4 : Output  4
 5 : Output  5
 6 : Output  3
 7 : Output  2
 8 : GND
 9 : Output  1
10 : Output  0
11 : Output  8
12 : Output  9
13 : Strobe
14 : Output  -QD
15 : Clock
16 : +5V...+60V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,13,15] : INPUT;
PIN[2,3,4,5,6,7,9,10,11,12,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[13,15]:=LOW;

PIN[1]:=LOW;
PIN[1]:=HIGH;
FOR I:=0 TO 9 DO
   BEGIN
   PIN[15]:=HIGH;
   PIN[15]:=LOW;
   LOADMODEON;
   PIN[12,11,2,3,5,4,6,7,9,10] : LOAD HIGH;
   IF (NOT(PIN[12])&NOT(PIN[11])&NOT(PIN[2])&NOT(PIN[3])&NOT(PIN[5])
      &NOT(PIN[4])&NOT(PIN[6])&NOT(PIN[7])&NOT(PIN[9])&NOT(PIN[10]))
      <>(1 SHL I) THEN ERROR(1);
   LOADMODEOFF;
   END;
```

```
ERROR(0);
END.
•
```

```
#NAME 74145,SN74145

#TEXT
BCD-to-Decimal-Decoder/
Display Driver
(O.C., 15V, 80mA)

This device converts a
standard BCD code with
4 bits into a decimal
number from 0 to 9.
The outputs are OPEN-
COLLECTOR with a maximum
output voltage of 15V
and maximum output
power of 80mA.

#PIN 16
 1 : Output 0
 2 : Output 1
 3 : Output 2
 4 : Output 3
 5 : Output 4
 6 : Output 5
 7 : Output 6
 8 : GND
 9 : Output 7
10 : Output 8
11 : Output 9
12 : Input  D
13 : Input  C
14 : Input  B
15 : Input  A
16 : +5V...+15V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[12,13,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9,10,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]:=I;
   LOADMODEON;
   PIN[1,2,3,4,5,6,7,9,10,11] : LOAD LOW;
   IF (PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1])
       <>0 THEN ERROR(1);
   PIN[11,10,9,7,6,5,4,3,2,1] : LOAD HIGH;
   IF (NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2])&NOT(PIN[1]))
       <>(1 SHL I) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74147,SN74147

#TEXT
Decimal-to-BCD Priority
```

```
Encoder

This device arranges the
9 input signals in order
of importance.
The highest-numbered
input is converted
into a BCD number.

#PIN 16
 1 : Input  4
 2 : Input  2
 3 : Input  6
 4 : Input  7
 5 : Input  8
 6 : Output C
 7 : Output B
 8 : GND
 9 : Output A
10 : Input  9
11 : Input  1
12 : Input  2
13 : Input  3
14 : Output D
15 : N.C.
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,10,11,12,13] : INPUT;
PIN[6,7,9,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

D:=%111111111;
FOR I:=0 TO 9 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[1]&
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[10]:=D;
   IF (NOT(PIN[14])&NOT(PIN[6])&NOT(PIN[7])&NOT(PIN[9]))<>I THEN ERROR(1);
   D:=D SHR 1;
   END;

ERROR(0);
END.

#NAME 74148,SN74148

#TEXT
Binary 8-to-3
Priority Encoder

This device arranges the
8 input signals in order
of importance.
The highest-numbered
input is converted into
a BCD number.
Several devices can be
cascaded.

#PIN 16
 1 : Input  4
 2 : Input  5
```

```
 3 : Input  6
 4 : Input  7
 5 : -Enable In
 6 : Output C
 7 : Output B
 8 : GND
 9 : Output A
10 : Input  0
11 : Input  1
12 : Input  2
13 : Input  3
14 : -GS
15 : -Enable Out
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,10,11,12,13] : INPUT;
PIN[6,7,9,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[5]:=LOW;

D:=%11111111;
PIN[10]&PIN[11]&PIN[12]&PIN[13]&PIN[1]&
PIN[2]&PIN[3]&PIN[4]:=D;
IF (PIN[14]<>HIGH) OR (PIN[15]<>LOW) THEN ERROR(1);
D:=D SHR 1;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[10]&PIN[11]&PIN[12]&PIN[13]&PIN[1]&
   PIN[2]&PIN[3]&PIN[4]:=D;
   IF (NOT(PIN[6])&NOT(PIN[7])&NOT(PIN[9]))<>I THEN ERROR(1);
   IF (PIN[14]<>LOW) OR (PIN[15]<>HIGH) THEN ERROR(1);
   D:=D SHR 1;
   END;

ERROR(0);
END.

#NAME 74151,SN74151

#TEXT
8-Input Multiplexer

This device selects
one of the eight data
input sources using a
3-bit binary code.

#PIN 16
 1 : Input   3
 2 : Input   2
 3 : Input   1
 4 : Input   0
 5 : Output  Q
 6 : Output -Q
 7 : -Strobe
 8 : GND
 9 : Input   C
10 : Input   B
11 : Input   A
12 : Input   7
```

```
13 : Input   6
14 : Input   5
15 : Input   4
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,7,9,10,11,12,13,14,15] : INPUT;
PIN[5,6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7]:=HIGH;

D:=%10100101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[1]&PIN[2]&PIN[3]&PIN[4]:=D;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[9]&PIN[10]&PIN[11]:=J;
      PIN[7]:=LOW;
      X:=(X SHL 1) OR PIN[5];
      Y:=(Y SHL 1) OR NOT(PIN[6]);
      PIN[7]:=HIGH;
      IF (PIN[5]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74152,SN74152

#TEXT
8-Input Multiplexer
with Inverted Input

In this device a 3-bit
binary code is used to
select one of the 8
input signals.
The appropriate signal
is inverted at the out-
put.

#PIN 14
 1 : Input   4
 2 : Input   3
 3 : Input   2
 4 : Input   1
 5 : Input   0
 6 : Output -Q
 7 : GND
 8 : Input   C
 9 : Input   B
10 : Input   A
11 : Input   7
12 : Input   6
13 : Input   5
14 : +5V
```

```
#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,8,9,10,11,12,13] : INPUT;
PIN[6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%10100101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]:=D;
   X:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[8]&PIN[9]&PIN[10]:=J;
      X:=(X SHL 1) OR NOT(PIN[6]);
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74153,SN74153

#TEXT
Dual 4-Input Multiplexer

This device contains
two 4-to-1 line data
selectors with common
address inputs and
independent strobe
inputs.

#PIN 16
 1 : Strobe   MP 1
 2 : Input    B
 3 : Input  3 MP 1
 4 : Input  2 MP 1
 5 : Input  1 MP 1
 6 : Input  0 MP 1
 7 : Output Q MP 1
 8 : GND
 9 : Output Q MP 2
10 : Input  0 MP 2
11 : Input  1 MP 2
12 : Input  2 MP 2
13 : Input  3 MP 2
14 : Input    A
15 : Strobe   MP 2
16 : +5V

#FAMILY Std,ALS,AS,F,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,10,11,12,13,14,15] : INPUT;
PIN[7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;
```

```
D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[1]:=LOW;
      X:=(X SHL 1) OR PIN[7];
      PIN[1]:=HIGH;
      IF PIN[7]<>LOW THEN ERROR(1);
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[11]&PIN[10]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[15]:=LOW;
      X:=(X SHL 1) OR PIN[9];
      PIN[15]:=HIGH;
      IF PIN[9]<>LOW THEN ERROR(1);
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74155,SN74155

#TEXT
Dual 2-Bit Binary Decoder/
Demultiplexer

This device contains two
2-bit data distributors
with common address
inputs.

#PIN 16
 1 : Input      DMP 1
 2 : -Strobe    DMP 1
 3 : Address    B
 4 : Output -Q3 DMP 1
 5 : Output -Q2 DMP 1
 6 : Output -Q1 DMP 1
 7 : Output -Q0 DMP 1
 8 : GND
 9 : Output -Q0 DMP 2
10 : Output -Q1 DMP 2
11 : Output -Q2 DMP 2
12 : Output -Q3 DMP 2
13 : Address    A
14 : -Strobe    DMP 2
15 : Input      DMP 2
16 : +5V
```

```
#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,14]:=HIGH;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[13]:=I;
   FOR J:=0 TO 1 DO
      BEGIN
      PIN[1]:=J;
      PIN[15]:=NOT(J);
      PIN[2,14]:=LOW;
      IF (NOT(PIN[4])&NOT(PIN[5])&NOT(PIN[6])&NOT(PIN[7]))
         <>(J SHL I) THEN ERROR(1);
      IF (NOT(PIN[12])&NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9]))
         <>(J SHL I) THEN ERROR(1);
      PIN[2,14]:=HIGH;
      END;
   IF (NOT(PIN[4])&NOT(PIN[5])&NOT(PIN[6])&NOT(PIN[7]))<>0 THEN ERROR(1);
   IF (NOT(PIN[12])&NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9]))<>0 THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74156,SN74156

#TEXT
Dual 2-to-4 Line Decoder/
Demultiplexer (O.C.)

This device contains
two 2-bit data distri-
butors with common
address inputs. Outputs
are open collector.

#PIN 16
 1 : Input      DMP 1
 2 : -Strobe    DMP 1
 3 : Address    B
 4 : Output -Q3 DMP 1
 5 : Output -Q2 DMP 1
 6 : Output -Q1 DMP 1
 7 : Output -Q0 DMP 1
 8 : GND
 9 : Output -Q0 DMP 2
10 : Output -Q1 DMP 2
11 : Output -Q2 DMP 2
12 : Output -Q3 DMP 2
13 : Address    A
14 : -Strobe    DMP 2
15 : Input      DMP 2
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
```

```
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,14]:=HIGH;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[13]:=I;
   FOR J:=0 TO 1 DO
      BEGIN
      PIN[1]:=J;
      PIN[15]:=NOT(J);
      PIN[2,14]:=LOW;
      LOADMODEON;
      PIN[4,5,6,7,12,11,10,9] : LOAD HIGH;
      IF (NOT(PIN[4])&NOT(PIN[5])&NOT(PIN[6])&NOT(PIN[7]))
         <>(J SHL I) THEN ERROR(1);
      IF (NOT(PIN[12])&NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9]))
         <>(J SHL I) THEN ERROR(1);
      PIN[4,5,6,7,12,11,10,9] : LOAD LOW;
      IF (PIN[4]&PIN[5]&PIN[6]&PIN[7])<>0 THEN ERROR(1);
      IF (PIN[12]&PIN[11]&PIN[10]&PIN[9])<>0 THEN ERROR(1);
      LOADMODEOFF;
      PIN[2,14]:=HIGH;
      END;
   LOADMODEON;
   PIN[4,5,6,7,12,11,10,9] : LOAD HIGH;
   IF (NOT(PIN[4])&NOT(PIN[5])&NOT(PIN[6])&NOT(PIN[7]))<>0 THEN ERROR(1);
   IF (NOT(PIN[12])&NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9]))<>0 THEN ERROR(1);
   PIN[4,5,6,7,12,11,10,9] : LOAD LOW;
   IF (PIN[4]&PIN[5]&PIN[6]&PIN[7])<>0 THEN ERROR(1);
   IF (PIN[12]&PIN[11]&PIN[10]&PIN[9])<>0 THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74157,SN74157

#TEXT
Quad 2-Input Multiplexer

This device contains
four 2-to-1 line data
selectors.

#PIN 16
 1 : Select
 2 : Input  A MP 1
 3 : Input  B MP 1
 4 : Output Q MP 1
 5 : Input  A MP 2
 6 : Input  B MP 2
 7 : Output Q MP 2
 8 : GND
 9 : Output Q MP 3
10 : Input  B MP 3
11 : Input  A MP 3
12 : Output Q MP 4
13 : Input  B MP 4
14 : Input  A MP 4
15 : -Enable
16 : +5V

#FAMILY Std,ALS,F,L,LS,S
```

```
#PROGRAM

BEGIN
PIN[1,2,3,5,6,10,11,13,14,15] : INPUT;
PIN[4,7,9,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=HIGH;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[11]&PIN[5]&PIN[2]:=D;
   PIN[15]:=LOW;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>0 THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[10]&PIN[6]&PIN[3]:=D;
   PIN[15]:=LOW;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>0 THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74158,SN74158

#TEXT
Quad 2-Input Multiplexer
with Inverted Outputs

This device contains
four 2-to-1 line data
selectors/multiplexers
with inverted outputs.

#PIN 16
 1 : Select
 2 : Input  A MP 1
 3 : Input  B MP 1
 4 : Output Q MP 1
 5 : Input  A MP 2
 6 : Input  B MP 2
 7 : Output Q MP 2
 8 : GND
 9 : Output Q MP 3
10 : Input  B MP 3
11 : Input  A MP 3
12 : Output Q MP 4
13 : Input  B MP 4
14 : Input  A MP 4
15 : -Enable
16 : +5V
```

```
#FAMILY Std,ALS,F,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,5,6,10,11,13,14,15] : INPUT;
PIN[4,7,9,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=HIGH;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[11]&PIN[5]&PIN[2]:=D;
   PIN[15]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>0 THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[10]&PIN[6]&PIN[3]:=D;
   PIN[15]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>0 THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74160,SN74160

#TEXT
Sychronous Presettable
Decade Counter with
Asynchronous Clear

This device contains a
presettable, synchronous
decade counter which
counts up in BCD code
and is asynchronously
reset.

#PIN 16
 1 : -R asyn
 2 : Clock
 3 : Input  P0
 4 : Input  P1
 5 : Input  P2
 6 : Input  P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output Q3
```

```
12 : Output Q2
13 : Output Q1
14 : Output Q0
15 : Carry Out
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74161,SN74161

#TEXT
Sychronous presettable
4-Bit Binary Counter
with Asynchronous Clear

This device contains
a presettable, synchronous
4-bit binary counter
which counts upwards in
binary code and is cleared
asynchronously.

#PIN 16
 1 : -R asyn
 2 : Clock
 3 : Input  P0
 4 : Input  P1
 5 : Input  P2
 6 : Input  P3
 7 : PE
 8 : GND
 9 : -Load
```

```
10 : TE
11 : Output Q3
12 : Output Q2
13 : Output Q1
14 : Output Q0
15 : Carry Out
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74162,SN74162

#TEXT
Synchronous Decade
Counter with Synchronous
Clear

This device contains a
presettable, synchronous
decade counter which
counts up in BCD
code and is cleared
synchronously.

#PIN 16
 1 : -R syn
 2 : Clock
 3 : Input  P0
 4 : Input  P1
 5 : Input  P2
 6 : Input  P3
 7 : PE
 8 : GND
 9 : -Load
```

```
10 : TE
11 : Output Q3
12 : Output Q2
13 : Output Q1
14 : Output Q0
15 : Carry Out
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[1]:=HIGH;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74163,SN74163

#TEXT
Sychronous 4-Bit Binary
Counter with Synchronous
Clear

This device contains
a presettable, synchronous
4-bit binary counter,
which counts upwards in
binary code and is cleared
synchronously.

#PIN 16
 1 : -R syn
 2 : Clock
 3 : Input  P0
 4 : Input  P1
 5 : Input  P2
 6 : Input  P3
 7 : PE
```

```
 8 : GND
 9 : -Load
10 : TE
11 : Output Q3
12 : Output Q2
13 : Output Q1
14 : Output Q0
15 : Carry Out
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[1]:=HIGH;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74164,SN74164

#TEXT
8-Bit Serial-In/Parallel-
Out Shift Register

This device contains
a high-speed 8-bit
shift register with
serial input and parallel
or serial output, plus
a clear input.

#PIN 14
 1 : Input  S1
 2 : Input  S2
 3 : Output Q1
 4 : Output Q2
 5 : Output Q3
 6 : Output Q4
```

```
 7 : GND
 8 : Clock
 9 : -Clear
10 : Output Q5
11 : Output Q6
12 : Output Q7
13 : Output Q8
14 : +5V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,8,9] : Input ;
PIN[3,4,5,6,10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[8]:=LOW;
PIN[1,2,9]:=HIGH;

D:=%01010101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[1]:=(X AND 1) EXOR 1;
      X:=X SHR 1;
      PIN[8]:=LOW;PIN[8]:=HIGH;
      END;
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[10]&
       PIN[6]&PIN[5]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[9]:=LOW;PIN[9]:=HIGH;
IF (PIN[13]&PIN[12]&PIN[11]&PIN[10]&
      PIN[6]&PIN[5]&PIN[4]&PIN[3])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74165,SN74165

#TEXT
8-Bit Parallel-In/Serial-
Out Shift Register

This device contains
an 8-Bit right shift
register with serial or
parallel input and
serial output.

#PIN 16
 1 : -Load
 2 : Clock
 3 : Input   P5
 4 : Input   P6
 5 : Input   P7
 6 : Input   P8
 7 : Output -Q8
 8 : GND
 9 : Output  Q8
10 : Serial  Input
```

```
11 : Input   P1
12 : Input   P2
13 : Input   P3
14 : Input   P4
15 : -Enable
16 : +5V

#FAMILY Std,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,10,11,12,13,14,15] : INPUT;
PIN[7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,15]:=LOW;
PIN[1]:=HIGH;

D:=%01010101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[10]:=X AND 1;
      X:=X SHR 1;
      PIN[2]:=HIGH;PIN[2]:=LOW;
      END;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[9] SHL 7);
      Y:=(Y SHR 1) OR (NOT(PIN[7]) SHL 7);
      PIN[2]:=HIGH;PIN[2]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=LOW;
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[3]&PIN[4]&PIN[5]&PIN[6]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[1]:=HIGH;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[9] SHL 7);
      Y:=(Y SHR 1) OR (NOT(PIN[7]) SHL 7);
      PIN[2]:=HIGH;PIN[2]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74166,SN74166

#TEXT
8-Bit Parallel-In/Serial-
Out Shift Register
```

```
This device contains
an 8-Bit shift register
with serial or parallel
input and serial output
as well as clear and
clock inhibit inputs.

#PIN 16
 1 : Serial Input
 2 : Input  P1
 3 : Input  P2
 4 : Input  P3
 5 : Input  P4
 6 : Clock  Inhibit
 7 : Clock
 8 : GND
 9 : -Clear
10 : Input  P5
11 : Input  P6
12 : Input  P7
13 : Output Q8
14 : Input  P8
15 : -Load
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10,11,12,14,15] : INPUT;
PIN[13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[6,7]:=LOW;
PIN[9,15]:=HIGH;

D:=%01010101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[1]:=X AND 1;
      X:=X SHR 1;
      PIN[7]:=HIGH;PIN[7]:=LOW;
      END;
   X:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[13] SHL 7);
      PIN[7]:=HIGH;PIN[7]:=LOW;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]:=LOW;
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[10]&PIN[11]&PIN[12]&PIN[14]:=D;
   PIN[7]:=HIGH;PIN[7]:=LOW;
   PIN[15]:=HIGH;
   X:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
```

```
      X:=(X SHR 1) OR (PIN[13] SHL 7);
      PIN[7]:=HIGH;PIN[7]:=LOW;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[9]:=LOW;PIN[9]:=HIGH;
X:=0;
FOR J:=7 DOWNTO 0 DO
   BEGIN
   X:=(X SHR 1) OR (PIN[13] SHL 7);
   PIN[7]:=HIGH;PIN[7]:=LOW;
   END;
IF X<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74167,SN74167

#TEXT
Synchronous Programmable
Decimal Frequency
Divider

This component contains
a programmable decimal
frequency divider, also
known as a bit rate
multiplier.

#PIN 16
 1 : N.C.
 2 : Input  C
 3 : Input  D
 4 : Set 9
 5 : Output  Out
 6 : Cascade Out
 7 : Enable  Out
 8 : GND
 9 : Clock
10 : Strobe
11 : Enable
12 : Cascade
13 : Clear
14 : Input  A
15 : Input  B
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,4,9,10,11,12,13,14,15] : INPUT;
PIN[5,6,7] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9,10,11]:=LOW;

PIN[12]:=LOW;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]:=1 SHL I;
   Q:=0;C:=0;E:=0;
```

```
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      E:=E+PIN[7];
      END;
   IF Q<>(16-(1 SHL I)) THEN ERROR(1);
   IF C<>16 THEN ERROR(1);
   IF E<>15 THEN ERROR(1);
   END;

PIN[12]:=HIGH;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]&PIN[1]&PIN[4]:=1 SHL I;
   Q:=0;C:=0;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      END;
   IF Q<>(16-(1 SHL I)) THEN ERROR(1);
   IF C<>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74168,SN74168

#TEXT
Synchronous Programmable
Decade Up/Down Counter

This device contains
a synchronous, programm-
able decimal up/down
counter.

#PIN 16
 1 : Up/-Down
 2 : Clock
 3 : Input  D0
 4 : Input  D1
 5 : Input  D2
 6 : Input  D3
 7 : -CEP
 8 : GND
 9 : -PE
10 : -CET
11 : Output Q3
12 : Output Q2
13 : Output Q1
14 : Output Q0
15 : -TC
16 : +5V

#FAMILY ALS,AS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
```

```
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,7,10]:=LOW;
PIN[9]:=HIGH;

PIN[6]&PIN[5]&PIN[4]&PIN[3]:=0;
PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;

PIN[1]:=HIGH;
FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

PIN[1]:=LOW;
FOR I:=9 DOWNTO 1 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74169,SN74169

#TEXT
Synchronous Binary Up/
Down Counter

This device contains
a synchronous, pre-
settable binary up/
down counter.

#PIN 16
 1 : Up/-Down
 2 : Clock
 3 : Input  D0
 4 : Input  D1
 5 : Input  D2
 6 : Input  D3
 7 : -CEP
 8 : GND
 9 : -PE
10 : -CET
11 : Output Q3
12 : Output Q2
13 : Output Q1
14 : Output Q0
15 : -TC
```

```
16 : +5V

#FAMILY ALS,AS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,7,10]:=LOW;
PIN[9]:=HIGH;

PIN[9]:=LOW;
PIN[6]&PIN[5]&PIN[4]&PIN[3]:=0;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[9]:=HIGH;

PIN[1]:=HIGH;
FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

PIN[1]:=LOW;
FOR I:=15 DOWNTO 1 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74170,SN74170

#TEXT
16-Bit-RAM (4x4, O.C.)

This device contains
a write/read memory (RAM),
organised in 4x4 words and
allowing data to be
written and read simul-
taneously.

The outputs are OPEN
COLLECTOR.
```

```
#PIN 16
 1 : Input   D2
 2 : Input   D3
 3 : Input   D4
 4 : Address RB
 5 : Address RA
 6 : Output  Q4
 7 : Output  Q3
 8 : GND
 9 : Output  Q2
10 : Output  Q1
11 : -RE
12 : -WE
13 : Address WB
14 : Address WA
15 : Input   D1
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,11,12,13,14,15] : INPUT;
PIN[6,7,9,10] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11,12]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[13]&PIN[14]:=J;
      PIN[12]:=LOW;
      PIN[3]&PIN[2]&PIN[1]&PIN[15]:=D;
      PIN[12]:=HIGH;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[4]&PIN[5]:=J;
      PIN[11]:=LOW;
      LOADMODEON;
      PIN[6,7,9,10] : LOAD HIGH;
      IF (PIN[6]&PIN[7]&PIN[9]&PIN[10])<>D THEN ERROR(1);
      LOADMODEOFF;
      PIN[11]:=HIGH;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74173,SN74173

#TEXT
Quad-D-Register with
Enable and Clear
(TRI-STATE outputs)

This device contains
four D-flip-flops with
```

```
enable inputs and clear,
plus three-state outputs.


#PIN 16
 1 : Out.-Enable -OE1
 2 : Out.-Enable -OE2
 3 : Output Q1
 4 : Output Q2
 5 : Output Q3
 6 : Output Q4
 7 : Clock
 8 : GND
 9 : In.-Enable -IE1
10 : In.-Enable -IE2
11 : Input  D4
12 : Input  D3
13 : Input  D2
14 : Input  D1
15 : Clear
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,7,9,10,11,12,13,14,15] : INPUT;
PIN[3,4,5,6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7,15]:=LOW;
PIN[1,2,9,10]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9,10]:=LOW;
   PIN[11]&PIN[12]&PIN[13]&PIN[14]:=D;
   PIN[7]:=HIGH;PIN[7]:=LOW;
   PIN[11]&PIN[12]&PIN[13]&PIN[14]:=0;
   PIN[9,10]:=HIGH;
   PIN[1,2]:=LOW;
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   PIN[1,2]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[15]:=HIGH;PIN[15]:=LOW;
PIN[1,2]:=LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
PIN[1,2]:=HIGH;

LOADMODEON;
PIN[6,5,4,3] : LOAD LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
PIN[6,5,4,3] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74174,SN74174

#TEXT
Hex-D-Register with
```

```
Clear

This device contains
six bistable memory
elements.

#PIN 16
 1 : -Clear
 2 : Output Q SE 1
 3 : Input  D SE 1
 4 : Input  D SE 2
 5 : Output Q SE 2
 6 : Input  D SE 3
 7 : Output Q SE 3
 8 : GND
 9 : Clock
10 : Output Q SE 4
11 : Input  D SE 4
12 : Output Q SE 5
13 : Input  D SE 5
14 : Input  D SE 6
15 : Output Q SE 6
16 : +5V

#FAMILY Std,ALS,AS,LS,S

#PROGRAM

BEGIN
PIN[1,3,4,6,9,11,13,14] : INPUT;
PIN[2,5,7,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=LOW;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=0;
   IF (PIN[15]&PIN[12]&PIN[10]&PIN[7]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[15]&PIN[12]&PIN[10]&PIN[7]&PIN[5]&PIN[2])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74175,SN74175

#TEXT
Quad-D-Register with
Clear

This device contains
four bistable memory
elements.

#PIN 16
 1 : -Clear
 2 : Output  Q SE 1
 3 : Output -Q SE 1
 4 : Input   D SE 1
```

```
 5 : Input   D SE 2
 6 : Output -Q SE 2
 7 : Output  Q SE 2
 8 : GND
 9 : Clock
10 : Output  Q SE 3
11 : Output -Q SE 3
12 : Input   D SE 3
13 : Input   D SE 4
14 : Output -Q SE 4
15 : Output  Q SE 4
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,4,5,9,12,13] : INPUT;
PIN[2,3,6,7,10,11,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=LOW;
PIN[1]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=0;
   IF (PIN[15]&PIN[10]&PIN[7]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[14])&NOT(PIN[11])&NOT(PIN[6])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74176,SN74176

#TEXT
Programmable Decade
Counter with Clear

This device contains a
divide by 2, a divide by
five and preset and clear
input.

#PIN 14
 1 : -Load
 2 : Output QC
 3 : Input   C
 4 : Input   A
 5 : Output QA
 6 : Clock   2
 7 : GND
 8 : Clock   1
 9 : Output QB
10 : Input   B
11 : Input   D
12 : Output QD
13 : -Clear
14 : +5V

#FAMILY Std
```

```
#PROGRAM

BEGIN
PIN[1,3,4,6,8,10,11,13] : INPUT;
PIN[2,5,9,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[6,8]:=LOW;
PIN[1,13]:=HIGH;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   IF PIN[5]<>I THEN ERROR(1);
   PIN[8]:=HIGH;PIN[8]:=LOW;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[12]&PIN[2]&PIN[9])<>I THEN ERROR(1);
   PIN[6]:=HIGH;PIN[6]:=LOW;
   END;

IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=LOW;
   PIN[11]&PIN[3]&PIN[10]&PIN[4]:=D;
   PIN[1]:=HIGH;
   IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74177,SN74177

#TEXT
Programmable 4-Bit
Binary Counter with
Clear

This device contains
a divide-by-2 counter,
a divide-by-8 counter
and preset and clear
inputs.

#PIN 14
 1 : -Load
 2 : Output QC
 3 : Input  C
 4 : Input  A
 5 : Output QA
 6 : Clock  2
 7 : GND
 8 : Clock  1
 9 : Output QB
10 : Input  B
```

```
11 : Input  D
12 : Output QD
13 : -Clear
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,3,4,6,8,10,11,13] : INPUT;
PIN[2,5,9,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[6,8]:=LOW;
PIN[1,13]:=HIGH;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   IF PIN[5]<>I THEN ERROR(1);
   PIN[8]:=HIGH;PIN[8]:=LOW;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 7 DO
   BEGIN
   IF (PIN[12]&PIN[2]&PIN[9])<>I THEN ERROR(1);
   PIN[6]:=HIGH;PIN[6]:=LOW;
   END;

IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=LOW;
   PIN[11]&PIN[3]&PIN[10]&PIN[4]:=D;
   PIN[1]:=HIGH;
   IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74178,SN74178

#TEXT
4-Bit-Serial-In/Parallel-
Out Shift Register

This device contains
a 4-bit right shift
register with parallel
or serial input and
parallel output.

#PIN 14
 1 : Input  B
 2 : Input  A
 3 : Serial Input
 4 : Output QA
 5 : Clock
```

```
 6 : Output QB
 7 : GND
 8 : Output QC
 9 : Load
10 : Output QD
11 : Shift
12 : Input  D
13 : Input  C
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,5,9,11,12,13] : INPUT;
PIN[4,6,8,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[9]:=LOW;
PIN[5,11]:=HIGH;

D:=%0101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[3]:=X AND 1;
      X:=X SHR 1;
      PIN[5]:=LOW;PIN[5]:=HIGH;
      END;
   IF (PIN[4]&PIN[6]&PIN[8]&PIN[10])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]:=LOW;
   PIN[9]:=HIGH;
   PIN[2]&PIN[1]&PIN[13]&PIN[12]:=D;
   PIN[5]:=LOW;PIN[5]:=HIGH;
   PIN[9]:=LOW;
   PIN[11]:=HIGH;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      IF (PIN[4]&PIN[6]&PIN[8]&PIN[10])<>X THEN ERROR(1);
      PIN[3]:=LOW;
      PIN[5]:=LOW;PIN[5]:=HIGH;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74179,SN74179

#TEXT
4-Bit Serial-In/Parallel-
Out Shift Register with
Clear

This device contains
```

```
a 4-bit right shift
register with parallel
or serial input and
parallel output plus
clear.

#PIN 16
 1 : -Clear
 2 : Input   B
 3 : Input   A
 4 : Serial Input
 5 : Output  QA
 6 : Clock
 7 : Output  QB
 8 : GND
 9 : Output  QC
10 : Load
11 : Output  QD
12 : Output -QD
13 : Shift
14 : Input   D
15 : Input   C
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,6,10,13,14,15] : INPUT;
PIN[5,7,9,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10]:=LOW;
PIN[1,6,13]:=HIGH;

D:=%0101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[4]:=X AND 1;
      X:=X SHR 1;
      PIN[6]:=LOW;PIN[6]:=HIGH;
      END;
   IF (PIN[5]&PIN[7]&PIN[9]&PIN[11])<>D THEN ERROR(1);
   IF NOT(PIN[12])<>(D AND 1) THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]:=LOW;
   PIN[10]:=HIGH;
   PIN[3]&PIN[2]&PIN[15]&PIN[14]:=D;
   PIN[6]:=LOW;PIN[6]:=HIGH;
   PIN[10]:=LOW;
   PIN[13]:=HIGH;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      IF (PIN[5]&PIN[7]&PIN[9]&PIN[11])<>X THEN ERROR(1);
      IF NOT(PIN[12])<>(X AND 1) THEN ERROR(1);
      PIN[4]:=LOW;
      PIN[6]:=LOW;PIN[6]:=HIGH;
```

```
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[5]&PIN[7]&PIN[9]&PIN[11])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74180,SN74180

#TEXT
9-Bit Parity Generator/
8-Bit Parity Checker

This device contains
a parity generator/
checker for 9-bits
( 8 data bits plus 1
parity bit).

#PIN 14
 1 : Input  D6
 2 : Input  D7
 3 : Input  even
 4 : Input  odd
 5 : Output even
 6 : Output odd
 7 : GND
 8 : Input  D0
 9 : Input  D1
10 : Input  D2
11 : Input  D3
12 : Input  D4
13 : Input  D5
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,8,9,10,11,12,13] : INPUT;
PIN[5,6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[3]:=HIGH;PIN[4]:=LOW;
D:=0;
FOR I:=0 TO 8 DO
   BEGIN
   PIN[2]&PIN[1]&PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9]&PIN[8]:=D;
   IF (NOT(PIN[5])<>(I MOD 2)) OR (PIN[6]<>(I MOD 2)) THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;

PIN[3]:=LOW;PIN[4]:=HIGH;
D:=0;
FOR I:=0 TO 8 DO
   BEGIN
   PIN[2]&PIN[1]&PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9]&PIN[8]:=D;
   IF (PIN[5]<>(I MOD 2)) OR (NOT(PIN[6])<>(I MOD 2)) THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;
```

```
ERROR(0);
END.

#NAME 74183,SN74183

#TEXT
Dual 1-Bit Full Adder

This device contains two
independent, high speed
1-bit full adders.

#PIN 14
 1 : Input  A    FA 1
 2 : N.C.
 3 : Input  B    FA 1
 4 : Input  Cn   FA 1
 5 : Output Cn+1 FA 1
 6 : Output ä    FA 1
 7 : GND
 8 : Output ä    FA 2
 9 : N.C.
10 : Output Cn+1 FA 2
11 : Input  Cn   FA 2
12 : Input  B    FA 2
13 : Input  A    FA 2
14 : +5V

#FAMILY H,LS

#PROGRAM

BEGIN
PIN[1,3,4,11,12,13] : INPUT;
PIN[5,6,8,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR C:=0 TO 1 DO
   BEGIN
   PIN[4,11]:=C;
   FOR A:=0 TO 1 DO
      BEGIN
      PIN[1,13]:=A;
      FOR B:=0 TO 1 DO
         BEGIN
         PIN[3,12]:=B;
         IF (PIN[5]&PIN[6])<>(C+A+B) THEN ERROR(1);
         IF (PIN[10]&PIN[8])<>(C+A+B) THEN ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 74184,SN74184

#TEXT
BCD to Binary Decoder
(O.C.)

This device contains
a decoder, which converts
a 6-bit BCD code into a
binary code.
```

```
The outputs are OPEN
COLLECTOR.

#PIN 16
 1 : Output Q1
 2 : Output Q2
 3 : Output Q3
 4 : Output Q4
 5 : Output Q5
 6 : Output Q6
 7 : Output Q7
 8 : GND
 9 : Output Q8
10 : Input  A
11 : Input  B
12 : Input  C
13 : Input  D
14 : Input  E
15 : -Enable
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[10,11,12,13,14,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=LOW;

FOR I:=0 TO 39 BY 2 DO
   BEGIN
   X:=((I DIV 10) SHL 4)+(I MOD 10);
   PIN[14]&PIN[13]&PIN[12]&PIN[11]&PIN[10]:=X SHR 1;
   LOADMODEON;
   PIN[9,7,6,5,4,3,2,1] : LOAD HIGH;
   IF I<>(PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1]&(X AND 1)) THEN ERROR(1);
   PIN[9,7,6,5,4,3,2,1] : LOAD LOW;
   IF (PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1])<>0 THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74185,SN74185

#TEXT
Binary-to-BCD Decoder
(O.C.)

This device contains a
decoder which converts a
6-bit binary code into a
BCD code.

The outputs are OPEN
COLLECTOR.

#PIN 16
 1 : Output Q1
 2 : Output Q2
 3 : Output Q3
 4 : Output Q4
 5 : Output Q5
```

```
 6 : Output Q6
 7 : Output Q7
 8 : GND
 9 : Output Q8
10 : Input  A
11 : Input  B
12 : Input  C
13 : Input  D
14 : Input  E
15 : -Enable
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[10,11,12,13,14,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=LOW;

FOR I:=0 TO 63 BY 2 DO
   BEGIN
   X:=((I DIV 10) SHL 4)+(I MOD 10);
   PIN[14]&PIN[13]&PIN[12]&PIN[11]&PIN[10]:=I SHR 1;
   LOADMODEON;
   PIN[9,7,6,5,4,3,2,1] : LOAD HIGH;
   IF X<>(PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1]&(I AND 1)) THEN ERROR(1);
   PIN[9,7,6,5,4,3,2,1] : LOAD LOW;
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1])<>0 THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.
•
```

```
#NAME 74189,SN74189

#TEXT
16x4-Bit-RAM (TRI-STATE)

This device contains a
write/read memory (RAM)
arranged in 16x4 bits.
The outputs are three-
state.

#PIN 16
 1 : Address A0
 2 : -CS
 3 : -WE
 4 : Input   D1
 5 : Output -Q1
 6 : Input   D2
 7 : Output -Q2
 8 : GND
 9 : Output -Q3
10 : Input   D3
11 : Output -Q4
12 : Input   D4
13 : Address A3
14 : Address A2
15 : Address A1
16 : +5V

#FAMILY Std,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,6,10,12,13,14,15] : INPUT;
PIN[5,7,9,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,3]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      PIN[12]&PIN[10]&PIN[6]&PIN[4]:=D;
      PIN[2]:=LOW;
      PIN[3]:=LOW;PIN[3]:=HIGH;
      PIN[2]:=HIGH;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      PIN[2]:=LOW;
      IF (NOT(PIN[11])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[5]))<>D THEN ERROR(1);
      PIN[2]:=HIGH;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[11,9,7,5] : LOAD LOW;
IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>%0000 THEN ERROR(1)
PIN[11,9,7,5] : LOAD HIGH;
```

```
IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>%1111 THEN ERROR(1)
LOADMODEOFF;

ERROR(0);
END.

#NAME 74190,SN74190

#TEXT
Sychronous Decade Up/
Down Counter

This device contains
a synchronous, programm-
able decade up/down
counter.

#PIN 16
 1 : Input  P1
 2 : Output Q1
 3 : Output Q0
 4 : -CE
 5 : -Up/Down
 6 : Output Q2
 7 : Output Q3
 8 : GND
 9 : Input  P3
10 : Input  P2
11 : -Load
12 : Max./Min. Out
13 : -RC
14 : Clock
15 : Input  P0
16 : +5V

#FAMILY Std,ALS,F,LS

#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,14]:=LOW;
PIN[11]:=HIGH;

PIN[5]:=LOW;
PIN[9]&PIN[10]&PIN[1]&PIN[15]:=0;
PIN[11]:=LOW;PIN[11]:=HIGH;
FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>LOW THEN ERROR(1);
   PIN[14]:=HIGH;PIN[14]:=LOW;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);

PIN[5]:=HIGH;
PIN[9]&PIN[10]&PIN[1]&PIN[15]:=8;
PIN[11]:=LOW;PIN[11]:=HIGH;
FOR I:=8 DOWNTO 1 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>LOW THEN ERROR(1);
   PIN[14]:=HIGH;PIN[14]:=LOW;
```

```
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74191,SN74191

#TEXT
Synchronous Binary Up/
Down Counter

This device contains
a synchronous, programm-
able binary up/down
counter.

#PIN 16
 1 : Input  P1
 2 : Output Q1
 3 : Output Q0
 4 : -CE
 5 : -Up/Down
 6 : Output Q2
 7 : Output Q3
 8 : GND
 9 : Input  P3
10 : Input  P2
11 : -Load
12 : Max./Min. Out
13 : -RC
14 : Clock
15 : Input  P0
16 : +5V

#FAMILY Std,ALS,F,LS

#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,14]:=LOW;
PIN[11]:=HIGH;

PIN[5]:=LOW;
PIN[9]&PIN[10]&PIN[1]&PIN[15]:=0;
PIN[11]:=LOW;PIN[11]:=HIGH;
FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>LOW THEN ERROR(1);
   PIN[14]:=HIGH;PIN[14]:=LOW;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
```

```
IF PIN[12]<>HIGH THEN ERROR(1);

PIN[5]:=HIGH;
PIN[9]&PIN[10]&PIN[1]&PIN[15]:=14;
PIN[11]:=LOW;PIN[11]:=HIGH;
FOR I:=14 DOWNTO 1 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>LOW THEN ERROR(1);
   PIN[14]:=HIGH;PIN[14]:=LOW;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74192,SN74192

#TEXT
Synchronous Decade Up/
Down Counter with Clear

This device contains
a programmable, synchro-
nous decade BCD counter
with two separate clock
inputs and a clear
input.

#PIN 16
 1 : Input  P1
 2 : Output Q1
 3 : Output Q0
 4 : Down
 5 : Up
 6 : Output Q2
 7 : Output Q3
 8 : GND
 9 : Input  P3
10 : Input  P2
11 : -Load
12 : -Carry Up
13 : -Carry Down
14 : Clear
15 : Input  P0
16 : +5V

#FAMILY Std,ALS,F,L,LS

#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,5,14]:=LOW;
```

```
PIN[4,5,11]:=HIGH;

PIN[14]:=HIGH;PIN[14]:=LOW;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>HIGH THEN ERROR(1);
   PIN[5]:=LOW;PIN[5]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

FOR I:=9 DOWNTO 1 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[13]<>HIGH THEN ERROR(1);
   PIN[4]:=LOW;PIN[4]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74193,SN74193

#TEXT
Synchronous 4-Bit Binary
Up/Down Counter with
Clear

This device contains
a programmable, synchro-
nous 4-bit binary counter
which counts up and down
using two separate clock
inputs. The counter also
has a clear input.

#PIN 16
 1 : Input  P1
 2 : Output Q1
 3 : Output Q0
 4 : Down
 5 : Up
 6 : Output Q2
 7 : Output Q3
 8 : GND
 9 : Input  P3
10 : Input  P2
11 : -Load
12 : -Carry Up
13 : -Carry Down
14 : Clear
15 : Input  P0
16 : +5V

#FAMILY Std,ALS,F,L,LS
```

```
#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,5,14]:=LOW;
PIN[4,5,11]:=HIGH;

PIN[14]:=HIGH;PIN[14]:=LOW;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>HIGH THEN ERROR(1);
   PIN[5]:=LOW;PIN[5]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

FOR I:=15 DOWNTO 1 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[13]<>HIGH THEN ERROR(1);
   PIN[4]:=LOW;PIN[4]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74194,SN74194

#TEXT
4-Bit Bidirectional
Universal Shift Register

This 4-bit shift register
features parallel inputs,
parallel output, right
shift, left shift,
operating mode control
inputs and a clear input.

#PIN 16
 1 : -Clear
 2 : Data   SR
 3 : Input  P0
 4 : Input  P1
 5 : Input  P2
 6 : Input  P3
 7 : Data   SL
 8 : GND
 9 : Mode   S0
10 : Mode   S1
11 : Clock
12 : Output Q3
13 : Output Q2
```

```
14 : Output Q1
15 : Output Q0
16 : +5V

#FAMILY Std,ALS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10,11] : INPUT;
PIN[12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11]:=LOW;
PIN[1]:=HIGH;

D:=%0101;

PIN[9]:=LOW;
PIN[10]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[7]:=X AND 1;
      X:=X SHR 1;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      END;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[9]:=HIGH;
PIN[10]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2]:=X AND 1;
      X:=X SHR 1;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      END;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[9]:=HIGH;
PIN[10]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74195,SN74195

#TEXT
4-Bit Parallel Shift
Register with Clear
```

```
This device contains
a 4-bit right shift
register with parallel
or serial input,
parallel output and
clear.

#PIN 16
 1 : -Clear
 2 : Input    J
 3 : Input   -K
 4 : Input   P0
 5 : Input   P1
 6 : Input   P2
 7 : Input   P3
 8 : GND
 9 : -Load
10 : Clock
11 : Output -Q3
12 : Output  Q3
13 : Output  Q2
14 : Output  Q1
15 : Output  Q0
16 : +5V

#FAMILY Std,H,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10]:=LOW;
PIN[1,9]:=HIGH;

D:=%0101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2,3]:=X AND 1;
      X:=X SHR 1;
      PIN[10]:=LOW;PIN[10]:=HIGH;
      END;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>D THEN ERROR(1);
   IF NOT(PIN[11])<>(D AND 1) THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;PIN[2,3]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[4]&PIN[5]&PIN[6]&PIN[7]:=D;
   PIN[10]:=LOW;PIN[10]:=HIGH;
   PIN[9]:=HIGH;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>X THEN ERROR(1);
      IF NOT(PIN[11])<>(X AND 1) THEN ERROR(1);
      PIN[2,3]:=LOW;
      PIN[10]:=LOW;PIN[10]:=HIGH;
```

```
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74196,SN74196

#TEXT
Programmable Decade
Counter with Clear

This counter features
a divide by 2 and a
divide by 5 counter,
preset and clear
inputs.

#PIN 14
 1 : -Load
 2 : Output QC
 3 : Input   C
 4 : Input   A
 5 : Output QA
 6 : Clock   2
 7 : GND
 8 : Clock   1
 9 : Output QB
10 : Input   B
11 : Input   D
12 : Output QD
13 : -Clear
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,3,4,6,8,10,11,13] : INPUT;
PIN[2,5,9,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[6,8]:=LOW;
PIN[1,13]:=HIGH;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   IF PIN[5]<>I THEN ERROR(1);
   PIN[8]:=HIGH;PIN[8]:=LOW;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[12]&PIN[2]&PIN[9])<>I THEN ERROR(1);
   PIN[6]:=HIGH;PIN[6]:=LOW;
   END;

IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);
```

```
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=LOW;
   PIN[11]&PIN[3]&PIN[10]&PIN[4]:=D;
   PIN[1]:=HIGH;
   IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74197,SN74197

#TEXT
Programmable 4-Bit
Binary Counter with
Clear

This counter features
a divide by 2 and a
divide by 8 counter,
a preset and a clear
input.

#PIN 14
 1 : -Load
 2 : Output QC
 3 : Input   C
 4 : Input   A
 5 : Output QA
 6 : Clock   2
 7 : GND
 8 : Clock   1
 9 : Output QB
10 : Input   B
11 : Input   D
12 : Output QD
13 : -Clear
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,3,4,6,8,10,11,13] : INPUT;
PIN[2,5,9,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[6,8]:=LOW;
PIN[1,13]:=HIGH;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   IF PIN[5]<>I THEN ERROR(1);
   PIN[8]:=HIGH;PIN[8]:=LOW;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
FOR I:=0 TO 7 DO
   BEGIN
```

```
   IF (PIN[12]&PIN[2]&PIN[9])<>I THEN ERROR(1);
   PIN[6]:=HIGH;PIN[6]:=LOW;
   END;

IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=LOW;
   PIN[11]&PIN[3]&PIN[10]&PIN[4]:=D;
   PIN[1]:=HIGH;
   IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[13]:=LOW;PIN[13]:=HIGH;
IF (PIN[12]&PIN[2]&PIN[9]&PIN[5])<>0 THEN ERROR(1);

ERROR(0);
END.

##NAME 74200,SN74200

#TEXT
256-Bit-RAM (TRI-STATE)

This device contains
a write-read memory (RAM)
arranged as 256-bit.
The outputs are three-
state.

#PIN 16
 1 : Address A0
 2 : Address A1
 3 : -CS1
 4 : -CS2
 5 : -CS3
 6 : Output  -Q
 7 : Address A2
 8 : GND
 9 : Address A3
10 : Address A4
11 : Address A5
12 : Read/-Write
13 : Data
14 : Address A6
15 : Address A7
16 : +5V

#FAMILY Std,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,7,9,10,11,12,13,14,15] : INPUT;
PIN[6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5,12]:=HIGH;

D:=%0;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 255 DO
      BEGIN
```

```
      PIN[15]&PIN[14]&PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[2]&PIN[1]:=J;
      PIN[13]:=D;
      PIN[3,4,5]:=LOW;
      PIN[12]:=LOW;PIN[12]:=HIGH;
      PIN[3,4,5]:=HIGH;
      D:=D EXOR %1;
      END;
   FOR J:=0 TO 255 DO
      BEGIN
      PIN[15]&PIN[14]&PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[2]&PIN[1]:=J;
      PIN[3,4,5]:=LOW;
      IF NOT(PIN[6])<>D THEN ERROR(1);
      PIN[3,4,5]:=HIGH;
      D:=D EXOR %1;
      END;
   D:=D EXOR %1;
   END;

LOADMODEON;
PIN[6] : LOAD LOW;
IF PIN[6]<>LOW THEN ERROR(1);
PIN[6] : LOAD HIGH;
IF PIN[6]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

##NAME 74201,SN74201

#TEXT
256x1-Bit-RAM (TRI-STATE)

This device contains
a write-read memory (RAM)
arranged as 256-bit.
The outputs are three-state.

#PIN 16
 1 : Address A0
 2 : Address A1
 3 : -CS1
 4 : -CS2
 5 : -CS3
 6 : Output  -Q
 7 : Address A3
 8 : GND
 9 : Address A4
10 : Address A5
11 : Address A6
12 : Read/-Write
13 : Data
14 : Address A7
15 : Address A2
16 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,7,9,10,11,12,13,14,15] : INPUT;
PIN[6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5,12]:=HIGH;
```

```
D:=%0;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 255 DO
      BEGIN
      PIN[14]&PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[15]&PIN[2]&PIN[1]:=J;
      PIN[13]:=D;
      PIN[3,4,5]:=LOW;
      PIN[12]:=LOW;PIN[12]:=HIGH;
      PIN[3,4,5]:=HIGH;
      D:=D EXOR %1;
      END;
   FOR J:=0 TO 255 DO
      BEGIN
      PIN[14]&PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[15]&PIN[2]&PIN[1]:=J;
      PIN[3,4,5]:=LOW;
      IF NOT(PIN[6])<>D THEN ERROR(1);
      PIN[3,4,5]:=HIGH;
      D:=D EXOR %1;
      END;
   D:=D EXOR %1;
   END;

LOADMODEON;
PIN[6] : LOAD LOW;
IF PIN[6]<>LOW THEN ERROR(1);
PIN[6] : LOAD HIGH;
IF PIN[6]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74219,SN74219

#TEXT
64-Bit-RAM (16x4)
(TRI-STATE)

This device contains a
read/write memory (RAM)
organised in 16x4 bits.
The outputs are three-
state.

#PIN 16
 1 : Address A0
 2 : -CS
 3 : -WE
 4 : Input   D1
 5 : Output  Q1
 6 : Input   D2
 7 : Output  Q2
 8 : GND
 9 : Output  Q3
10 : Input   D3
11 : Output  Q4
12 : Input   D4
13 : Address A3
14 : Address A2
15 : Address A1
16 : +5V

#FAMILY LS

#PROGRAM
```

```
BEGIN
PIN[1,2,3,4,6,10,12,13,14,15] : INPUT;
PIN[5,7,9,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,3]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      PIN[12]&PIN[10]&PIN[6]&PIN[4]:=D;
      PIN[2]:=LOW;
      PIN[3]:=LOW;PIN[3]:=HIGH;
      PIN[2]:=HIGH;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      PIN[2]:=LOW;
      IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>D THEN ERROR(1);
      PIN[2]:=HIGH;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[11,9,7,5] : LOAD LOW;
IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>%0000 THEN ERROR(1)
PIN[11,9,7,5] : LOAD HIGH;
IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>%1111 THEN ERROR(1)
LOADMODEOFF;

ERROR(0);
END.

#NAME 74221,SN74221

#TEXT
Dual Monostable Multi-
vibrator with Schmitt
Trigger Input and Clear

This device contains
two one-shots. They can
be positive or negative-
edge triggered.

#PIN 16
 1 : Input   A MF 1
 2 : Input   B MF 1
 3 : -Clear    MF 1
 4 : Output -Q MF 1
 5 : Output  Q MF 2
 6 : C ext.    MF 2
 7 : C/R       MF 2
 8 : GND
 9 : Input   A MF 2
10 : Input   B MF 2
11 : -Clear    MF 2
12 : Output -Q MF 2
13 : Output  Q MF 1
14 : C ext.    MF 1
```

```
15 : C/R       MF 1
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,9,10,11] : INPUT;
PIN[4,5,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[16,15,14] : RC;

PIN[1,2]:=LOW;
PIN[3]:=LOW;
PIN[3]:=HIGH;
IF PIN[13]<>LOW THEN ERROR(1);
C:=256;
PIN[2]:=HIGH;
IF PIN[13]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[13]<>LOW) DO
   C:=C-1;
IF PIN[13]<>LOW THEN ERROR(1);

PIN[1,2]:=HIGH;
PIN[3]:=LOW;
PIN[3]:=HIGH;
IF PIN[4]<>HIGH THEN ERROR(1);
C:=256;
PIN[1]:=LOW;
IF PIN[4]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[4]<>HIGH) DO
   C:=C-1;
IF PIN[4]<>HIGH THEN ERROR(1);

PIN[16,15,14] : RC OFF;
PIN[16,7,6] : RC;

PIN[9,10]:=LOW;
PIN[11]:=LOW;
PIN[11]:=HIGH;
IF PIN[5]<>LOW THEN ERROR(1);
C:=256;
PIN[10]:=HIGH;
IF PIN[5]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[5]<>LOW) DO
   C:=C-1;
IF PIN[5]<>LOW THEN ERROR(1);

PIN[9,10]:=HIGH;
PIN[11]:=LOW;
PIN[11]:=HIGH;
IF PIN[12]<>HIGH THEN ERROR(1);
C:=256;
PIN[9]:=LOW;
IF PIN[12]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[12]<>HIGH) DO
   C:=C-1;
IF PIN[12]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 74240,SN74240
```

```
#TEXT
Inverting Octal TRI-
STATE Buffer

This device contains
eight inverting
buffers with three-
state outputs.

#PIN 20
 1 : -G1
 2 : Input   E0
 3 : Output -Q7
 4 : Input   E1
 5 : Output -Q6
 6 : Input   E2
 7 : Output -Q5
 8 : Input   E3
 9 : Output -Q4
10 : GND
11 : Input   E4
12 : Output -Q3
13 : Input   E5
14 : Output -Q2
15 : Input   E6
16 : Output -Q1
17 : Input   E7
18 : Output -Q0
19 : -G2
20 : +5V

#FAMILY ALS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[18])&NOT(PIN[16])&NOT(PIN[14])&NOT(PIN[12])&
       NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[5])&NOT(PIN[3]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74241,SN74241
```

```
#TEXT
Octal TRI-STATE Buffer

This device contains
eight non-inverting
buffers with three-
state outputs.

#PIN 20
 1 : -G1
 2 : Input  E0
 3 : Output Q7
 4 : Input  E1
 5 : Output Q6
 6 : Input  E2
 7 : Output Q5
 8 : Input  E3
 9 : Output Q4
10 : GND
11 : Input  E4
12 : Output Q3
13 : Input  E5
14 : Output Q2
15 : Input  E6
16 : Output Q1
17 : Input  E7
18 : Output Q0
19 : -G2
20 : +5V

#FAMILY ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=LOW;
PIN[1]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1]:=LOW;PIN[19]:=HIGH;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1]:=HIGH;PIN[19]:=LOW;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
```

```
#NAME 74242,SN74242

#TEXT
Inverting Quad TRI-
STATE Transceiver

This device contains
four inverting
bidirectional buffers
with three-state
outputs.

#PIN 14
 1 : -GA
 2 : N.C.
 3 : In-/Output A0
 4 : In-/Output A1
 5 : In-/Output A2
 6 : In-/Output A3
 7 : GND
 8 : In-/Output B3
 9 : In-/Output B2
10 : In-/Output B1
11 : In-/Output B0
12 : N.C.
13 : GB
14 : +5V

#FAMILY ALS,AS,F,L,LS,S

#PROGRAM

BEGIN
PIN[1,13] : Input ;
PIN[7] : GND;
PIN[14] : +5V;

D:=%0101;
PIN[1,13]:=LOW;
PIN[3,4,5,6] : INPUT;
PIN[8,9,10,11] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   IF (NOT(PIN[8])&NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,13]:=HIGH;
PIN[8,9,10,11] : INPUT;
PIN[3,4,5,6] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[9]&PIN[10]&PIN[11]:=D;
   IF (NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=HIGH;PIN[13]:=LOW;
PIN[3,4,5,6,8,9,10,11] : OUTPUT;
LOADMODEON;
PIN[3,4,5,6,8,9,10,11] : LOAD LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,4,5,6,8,9,10,11] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%1111 THEN ERROR(1);
```

```
LOADMODEOFF;

ERROR(0);
END.

#NAME 74243,SN74243

#TEXT
Quad TRI-STATE
Transceiver

This device contains
four non-inverting
bidirectional buffers
with three-state
outputs.

#PIN 14
 1 : -GA
 2 : N.C.
 3 : In-/Output A0
 4 : In-/Output A1
 5 : In-/Output A2
 6 : In-/Output A3
 7 : GND
 8 : In-/Output B3
 9 : In-/Output B2
10 : In-/Output B1
11 : In-/Output B0
12 : N.C.
13 : GB
14 : +5V

#FAMILY ALS,AS,F,L,LS,S

#PROGRAM

BEGIN
PIN[1,13] : INPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%0101;
PIN[1,13]:=LOW;
PIN[3,4,5,6] : INPUT;
PIN[8,9,10,11] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,13]:=HIGH;
PIN[8,9,10,11] : INPUT;
PIN[3,4,5,6] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[9]&PIN[10]&PIN[11]:=D;
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=HIGH;PIN[13]:=LOW;
PIN[3,4,5,6,8,9,10,11] : OUTPUT;
LOADMODEON;
PIN[3,4,5,6,8,9,10,11] : LOAD LOW;
```

```
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,4,5,6,8,9,10,11] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74244,SN74244

#TEXT
Octal TRI-STATE
Buffer

This device contains
eight non-inverting
buffers with TRI-STATE
outputs.

#PIN 20
 1 : -G1
 2 : Input  E0
 3 : Output Q7
 4 : Input  E1
 5 : Output Q6
 6 : Input  E2
 7 : Output Q5
 8 : Input  E3
 9 : Output Q4
10 : GND
11 : Input  E4
12 : Output Q3
13 : Input  E5
14 : Output Q2
15 : Input  E6
16 : Output Q1
17 : Input  E7
18 : Output Q0
19 : -G2
20 : +5V

#FAMILY ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
```

```
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74245,SN74245

#TEXT
Octal TRI-STATE
Transceiver

This device contains
eight non-inverting
bidirectional buffers
with three-state
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -Enable
20 : +5V

#FAMILY ALS,F,L,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
```

```
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74246,SN74246

#TEXT
BCD-to-7-Segment-Decoder/
Driver (O.C., 30V)

This device converts BCD
input data into control
data for seven segment
displays. The outputs are
OPEN-COLLECTOR and out-
put voltage is +30V
maximum.

#PIN 16
 1 : BCD-Input  B
 2 : BCD-Input  C
 3 : -LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input  D
 7 : BCD-Input  A
 8 : GND
 9 : 7-Seg.-Output -e
10 : 7-Seg.-Output -d
11 : 7-Seg.-Output -c
12 : 7-Seg.-Output -b
13 : 7-Seg.-Output -a
14 : 7-Seg.-Output -g
15 : 7-Seg.-Output -f
16 : +5V...+30V

#FAMILY Std
```

```
#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   LOADMODEON;
   PIN[9,10,11,12,13,14,15] : LOAD LOW;
   IF (PIN[14]&PIN[15]&PIN[9]&PIN[10]&
       PIN[11]&PIN[12]&PIN[13])<>0 THEN ERROR(1);
   PIN[9,10,11,12,13,14,15] : LOAD HIGH;
   D:=NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[9])&NOT(PIN[10])&
      NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13]);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74247,SN74247

#TEXT
BCD-to-7-Segment-Decoder/
Driver (O.C., 15V)

This device converts
BCD input data into
control data for seven
segment displays. The out-
puts are OPEN COLLECTOR
and maximum output voltage
+15V.

#PIN 16
 1 : BCD-Input  B
 2 : BCD-Input  C
 3 : -LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input  D
 7 : BCD-Input  A
 8 : GND
 9 : 7-Seg.-Output -e
10 : 7-Seg.-Output -d
11 : 7-Seg.-Output -c
12 : 7-Seg.-Output -b
13 : 7-Seg.-Output -a
14 : 7-Seg.-Output -g
15 : 7-Seg.-Output -f
16 : +5V...+15V
```

```
#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   LOADMODEON;
   PIN[9,10,11,12,13,14,15] : LOAD LOW;
   IF (PIN[14]&PIN[15]&PIN[9]&PIN[10]&
       PIN[11]&PIN[12]&PIN[13])<>0 THEN ERROR(1);
   PIN[9,10,11,12,13,14,15] : LOAD HIGH;
   D:=NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[9])&NOT(PIN[10])&
      NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13]);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74248,SN74248

#TEXT
BCD-to-7-Segment-Decoder/
Driver

This device converts
BCD input data into
control data for seven-
segment displays.

#PIN 16
 1 : BCD-Input  B
 2 : BCD-Input  C
 3 : -LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input  D
 7 : BCD-Input  A
 8 : GND
 9 : 7-Seg.-Output -e
10 : 7-Seg.-Output -d
11 : 7-Seg.-Output -c
12 : 7-Seg.-Output -b
13 : 7-Seg.-Output -a
14 : 7-Seg.-Output -g
15 : 7-Seg.-Output -f
16 : +5V

#FAMILY Std,LS
```

```
#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74249,SN74249

#TEXT
BCD-to-7-Segment-Decoder/
Driver (O.C.,5.5V)

This device converts
BCD input data into
control data for seven-
segment displays. The out-
puts are OPEN-COLLECTOR
with a maximum output
voltage of +5.5V.

#PIN 16
 1 : BCD-Input  B
 2 : BCD-Input  C
 3 : -LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input  D
 7 : BCD-Input  A
 8 : GND
 9 : 7-Seg.-Output -e
10 : 7-Seg.-Output -d
11 : 7-Seg.-Output -c
12 : 7-Seg.-Output -b
13 : 7-Seg.-Output -a
14 : 7-Seg.-Output -g
15 : 7-Seg.-Output -f
16 : +5V..+5,5V

#FAMILY L,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
```

```
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   LOADMODEON;
   PIN[14,15,9,10,11,12,13] : LOAD LOW;
   IF (PIN[14]&PIN[15]&PIN[9]&PIN[10]
      &PIN[11]&PIN[12]&PIN[13])<>0 THEN ERROR(1);
   PIN[14,15,9,10,11,12,13] : LOAD HIGH;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74251,SN74251

#TEXT
8-Input TRI-STATE
Multiplexer

This device contains
a data selector having
a 3-bit binary address
to select one of the
eight input signals. The
outputs are TRI-STATE.

#PIN 16
 1 : Input   3
 2 : Input   2
 3 : Input   1
 4 : Input   0
 5 : Output  Q
 6 : Output -Q
 7 : -Enable
 8 : GND
 9 : Input   C
10 : Input   B
11 : Input   A
12 : Input   7
13 : Input   6
14 : Input   5
15 : Input   4
16 : +5V

#FAMILY Std,ALS,AS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,7,9,10,11,12,13,14,15] : INPUT;
```

```
PIN[5,6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7]:=HIGH;

D:=%10100101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[1]&PIN[2]&PIN[3]&PIN[4]:=D;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[9]&PIN[10]&PIN[11]:=J;
      PIN[7]:=LOW;
      X:=(X SHL 1) OR PIN[5];
      Y:=(Y SHL 1) OR NOT(PIN[6]);
      PIN[7]:=HIGH;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[5,6] : LOAD LOW;
IF (PIN[5]<>LOW) OR (PIN[6]<>LOW) THEN ERROR(1);
PIN[5,6] : LOAD HIGH;
IF (PIN[5]<>HIGH) OR (PIN[6]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74253,SN74253

#TEXT
Dual 4-Input TRI-STATE
Multiplexer

This device contains
two 1-of-4 data selectors
with common address inputs
but separate enable
inputs.

#PIN 16
 1 : -OE      MP 1
 2 : Address  B
 3 : Input  3 MP 1
 4 : Input  2 MP 1
 5 : Input  1 MP 1
 6 : Input  0 MP 1
 7 : Output Q MP 1
 8 : GND
 9 : Output Q MP 2
10 : Input  0 MP 2
11 : Input  1 MP 2
12 : Input  2 MP 2
13 : Input  3 MP 2
14 : Address  A
15 : -OE      MP 2
16 : +5V

#FAMILY ALS,AS,F,L,LS,S

#PROGRAM

BEGIN
```

```
PIN[1,2,3,4,5,6,10,11,12,13,14,15] : INPUT;
PIN[7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
 PIN[1,15]:=HIGH;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[1]:=LOW;
      X:=(X SHL 1) OR PIN[7];
      PIN[1]:=HIGH;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[11]&PIN[10]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[15]:=LOW;
      X:=(X SHL 1) OR PIN[9];
      PIN[15]:=HIGH;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[7,9] : LOAD LOW;
IF (PIN[7]<>LOW) OR (PIN[9]<>LOW) THEN ERROR(1);
PIN[7,9] : LOAD HIGH;
IF (PIN[7]<>HIGH) OR (PIN[9]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74256,SN74256

#TEXT
Dual 4-Bit Addressable
Latch

This device contains
two addressable 4-bit
latches with common
control inputs.

#PIN 16
 1 : Address   A0
 2 : Address   A1
 3 : Input     ZS 1
 4 : Output Q0 ZS 1
 5 : Output Q1 ZS 1
 6 : Output Q2 ZS 1
 7 : Output Q3 ZS 1
```

```
 8 : GND
 9 : Output Q0 ZS 2
10 : Output Q1 ZS 2
11 : Output Q2 ZS 2
12 : Output Q3 ZS 2
13 : Input     ZS 2
14 : -Enable
15 : -Clear
16 : +5V

#FAMILY F,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[14,15]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[3]:=(D SHR J) AND 1;
      PIN[2]&PIN[1]:=J;
      PIN[14]:=LOW;PIN[14]:=HIGH;
      END;
   IF (PIN[7]&PIN[6]&PIN[5]&PIN[4])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[15]:=LOW;PIN[15]:=HIGH;
IF (PIN[7]&PIN[6]&PIN[5]&PIN[4])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[13]:=(D SHR J) AND 1;
      PIN[2]&PIN[1]:=J;
      PIN[14]:=LOW;PIN[14]:=HIGH;
      END;
   IF (PIN[12]&PIN[11]&PIN[10]&PIN[9])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[15]:=LOW;PIN[15]:=HIGH;
IF (PIN[12]&PIN[11]&PIN[10]&PIN[9])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74257,SN74257

#TEXT
Quad 2-Input TRI-STATE
Multiplexer


This device contains
four 1-of-2 data selec-
tors. The outputs are
TRI-STATE.
```

```
#PIN 16
 1 : Select
 2 : Input  A MP 1
 3 : Input  B MP 1
 4 : Output Q MP 1
 5 : Input  A MP 2
 6 : Input  B MP 2
 7 : Output Q MP 2
 8 : GND
 9 : Output Q MP 3
10 : Input  B MP 3
11 : Input  A MP 3
12 : Output Q MP 4
13 : Input  B MP 4
14 : Input  A MP 4
15 : -Enable
16 : +5V

#FAMILY ALS,F,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,5,6,10,11,13,14,15] : INPUT;
PIN[4,7,9,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=HIGH;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[11]&PIN[5]&PIN[2]:=D;
   PIN[15]:=LOW;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[10]&PIN[6]&PIN[3]:=D;
   PIN[15]:=LOW;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[4,7,9,12] : LOAD LOW;
IF (PIN[4]<>LOW) OR (PIN[7]<>LOW) OR
   (PIN[9]<>LOW) OR (PIN[12]<>LOW) THEN ERROR(1);
PIN[4,7,9,12] : LOAD HIGH;
IF (PIN[4]<>HIGH) OR (PIN[7]<>HIGH) OR
   (PIN[9]<>HIGH) OR (PIN[12]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```

```
#NAME 74258,SN74258

#TEXT
Quad 2-Input TRI-STATE
Multiplexer (Inverted
Output)

This device contains
four 1-of-2 data selec-
tors/multiplexers with
inverting TRI-STATE
outputs.

#PIN 16
 1 : Select
 2 : Input   A MP 1
 3 : Input   B MP 1
 4 : Output  Q MP 1
 5 : Input   A MP 2
 6 : Input   B MP 2
 7 : Output  Q MP 2
 8 : GND
 9 : Output  Q MP 3
10 : Input   B MP 3
11 : Input   A MP 3
12 : Output  Q MP 4
13 : Input   B MP 4
14 : Input   A MP 4
15 : -Enable
16 : +5V

#FAMILY ALS,F,L,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,5,6,10,11,13,14,15] : INPUT;
PIN[4,7,9,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=HIGH;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[11]&PIN[5]&PIN[2]:=D;
   PIN[15]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[10]&PIN[6]&PIN[3]:=D;
   PIN[15]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

LOADMODEON;
```

```
PIN[4,7,9,12] : LOAD LOW;
IF (PIN[4]<>LOW) OR (PIN[7]<>LOW) OR
   (PIN[9]<>LOW) OR (PIN[12]<>LOW) THEN ERROR(1);
PIN[4,7,9,12] : LOAD HIGH;
IF (PIN[4]<>HIGH) OR (PIN[7]<>HIGH) OR
   (PIN[9]<>HIGH) OR (PIN[12]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74259,SN74259

#TEXT
8-Bit Addressable
Latch

This device contains
an addressable 8-bit
latch with enable and
clear.

#PIN 16
 1 : Address A0
 2 : Address A1
 3 : Address A2
 4 : Output  Q0
 5 : Output  Q1
 6 : Output  Q2
 7 : Output  Q3
 8 : GND
 9 : Output  Q4
10 : Output  Q5
11 : Output  Q6
12 : Output  Q7
13 : Input
14 : -Enable
15 : -Clear
16 : +5V

#FAMILY Std,ALS,F,L,LS

#PROGRAM

BEGIN
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[14,15]:=HIGH;

D:=%01011010;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[13]:=(D SHR J) AND 1;
      PIN[3]&PIN[2]&PIN[1]:=J;
      PIN[14]:=LOW;PIN[14]:=HIGH;
      END;
   IF (PIN[12]&PIN[11]&PIN[10]&PIN[9]&
       PIN[7]&PIN[6]&PIN[5]&PIN[4])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[15]:=LOW;PIN[15]:=HIGH;
IF (PIN[12]&PIN[11]&PIN[10]&PIN[9]&
```

```
    PIN[7]&PIN[6]&PIN[5]&PIN[4])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74260,SN74260

#TEXT
Dual 5-Input NOR Gate

This device contains
two NOR gates each
with 5 inputs.

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Input   3 Gate   1
 4 : Input   1 Gate   2
 5 : Output    Gate   1
 6 : Output    Gate   2
 7 : GND
 8 : Input   2 Gate   2
 9 : Input   3 Gate   2
10 : Input   4 Gate   2
11 : Input   5 Gate   2
12 : Input   4 Gate   1
13 : Input   5 Gate   1
14 : +5V

#FAMILY LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,8,9,10,11,12,13] : INPUT;
PIN[6,5] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 31 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[12]&PIN[13]:=I;
   PIN[4]&PIN[8]&PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[5]<>NOT(PIN[1] OR PIN[2] OR PIN[3] OR
                  PIN[12] OR PIN[13]) THEN ERROR(1);
   IF PIN[6]<>NOT(PIN[4] OR PIN[8] OR PIN[9] OR
                  PIN[10] OR PIN[11]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74265,SN74265

#TEXT
Dual Inverter and Dual
NAND Gate with
Complementary Outputs

This device contains
two independent inverters
and two independent NAND
gates with complementary
outputs.

#PIN 16
```

```
 1 : Input     Inv. 1
 2 : -Output   Inv. 1
 3 : Output    Inv. 1
 4 : Input   1 NAND 1
 5 : Input   2 NAND 1
 6 : -Output   NAND 1
 7 : Output    NAND 1
 8 : GND
 9 : Output    NAND 2
10 : -Output   NAND 2
11 : Input   1 NAND 2
12 : Input   2 NAND 2
13 : Output    Inv. 2
14 : -Output   Inv. 2
15 : Input     Inv. 2
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,4,5,11,12,15] : INPUT;
PIN[2,3,6,7,9,10,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=I;
   PIN[15]:=I;
   IF (PIN[2]<>I) OR (NOT(PIN[3])<>I) THEN ERROR(1);
   IF (PIN[14]<>I) OR (NOT(PIN[13])<>I) THEN ERROR(1);
   END;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[4]&PIN[5]:=I;
   PIN[11]&PIN[12]:=I;
   IF (PIN[6]<>(PIN[4] AND PIN[5])) OR
      (PIN[7]<>(PIN[4] NAND PIN[5])) THEN ERROR(1);
   IF (PIN[10]<>(PIN[11] AND PIN[12])) OR
      (PIN[9]<>(PIN[11] NAND PIN[12])) THEN ERROR(1);

ERROR(0);
END.

#NAME 74266,SN74266

#TEXT
Quad 2-Input XNOR Gate
(O.C.)

This device contains
four XNOR gates each with
2 inputs. The outputs are
OPEN COLLECTOR.

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Output    Gate   1
 4 : Output    Gate   2
 5 : Input   1 Gate   2
 6 : Input   2 Gate   2
 7 : GND
 8 : Input   1 Gate   3
```

```
 9 : Input   2 Gate   3
10 : Output    Gate   3
11 : Output    Gate   4
12 : Input   1 Gate   4
13 : Input   2 Gate   4
14 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[3,4,10,11] : LOAD HIGH;
   IF PIN[3]<>(PIN[1] NEXOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NEXOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NEXOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NEXOR PIN[13]) THEN ERROR(1);
   PIN[3,4,10,11] : LOAD LOW;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[4]<>LOW THEN ERROR(1);
   IF PIN[10]<>LOW THEN ERROR(1);
   IF PIN[11]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74273,SN74273

#TEXT
8-Bit-D-Register with
Clear

This device contains
eight bistable memory
elements.

#PIN 20
 1 : -Clear
 2 : Output  Q SE 1
 3 : Input   D SE 1
 4 : Input   D SE 2
 5 : Output  Q SE 2
 6 : Output  Q SE 3
 7 : Input   D SE 3
 8 : Input   D SE 4
 9 : Output  Q SE 4
10 : GND
11 : Clock
12 : Output  Q SE 5
13 : Input   D SE 5
14 : Input   D SE 6
15 : Output  Q SE 6
16 : Output  Q SE 7
```

```
17 : Input   D SE 7
18 : Input   D SE 8
19 : Output  Q SE 8
20 : +5V

#FAMILY Std,ALS,F,LS,S

#PROGRAM

BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[11]:=LOW;
PIN[1]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
   IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
   IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74276,SN74276

#TEXT
Quad J-K Flip-Flop
with Common Preset
and Clear

This device contains
four separate J-K flip-
flops with preset and
clear.

#PIN 20
 1 : -Clear
 2 : Input    J FF 1
 3 : Clock      FF 1
 4 : Input   -K FF 1
 5 : Output   Q FF 1
 6 : Output   Q FF 2
 7 : Input   -K FF 2
 8 : -Clock     FF 2
 9 : Input    J FF 2
10 : GND
11 : -Preset
12 : Input    J FF 3
13 : -Clock     FF 3
14 : Input   -K FF 3
15 : Output   Q FF 3
16 : Output   Q FF 4
17 : Input   -K FF 4
18 : -Clock     FF 4
19 : Input    J FF 4
```

```
20 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,7,8,9,11,12,13,14,17,18,19] : INPUT;
PIN[5,6,15,16] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[2,9,12,19]:=LOW;
PIN[1,3,4,7,8,11,13,14,17,18]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF PIN[5]<>LOW THEN ERROR(1);
IF PIN[6]<>LOW THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);
IF PIN[16]<>LOW THEN ERROR(1);

PIN[11]:=LOW;PIN[11]:=HIGH;
IF PIN[5]<>HIGH THEN ERROR(1);
IF PIN[6]<>HIGH THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);
IF PIN[16]<>HIGH THEN ERROR(1);

FOR I:=0 TO 1 DO
   BEGIN
   PIN[4,7,14,17]:=I;
   PIN[3,8,13,18]:=LOW;PIN[3,8,13,18]:=HIGH;
   IF PIN[5]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   IF PIN[16]<>LOW THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[2,9,12,19]:=I;
   PIN[3,8,13,18]:=LOW;PIN[3,8,13,18]:=HIGH;
   IF PIN[5]<>HIGH THEN ERROR(1);
   IF PIN[6]<>HIGH THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   IF PIN[16]<>HIGH THEN ERROR(1);
   END;

PIN[2,9,12,19]:=HIGH;
PIN[4,7,14,17]:=LOW;
PIN[3,8,13,18]:=LOW;PIN[3,8,13,18]:=HIGH;
IF PIN[5]<>LOW THEN ERROR(1);
IF PIN[6]<>LOW THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);
IF PIN[16]<>LOW THEN ERROR(1);

ERROR(0);
END.

#NAME 74278,SN74278

#TEXT
Cascadable 4-Bit
Priority Latch

This device accepts
data via 5 inputs
and arranges it in
order of priority.
```

```
#PIN 14
 1 : Strobe
 2 : Input   D3
 3 : Input   D4
 4 : Input   P0
 5 : Output  P1
 6 : Output  Q4
 7 : GND
 8 : Output  Q3
 9 : Output  Q2
10 : Output  Q1
11 : N.C.
12 : Input   D1
13 : Input   D2
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,12,13] : INPUT;
PIN[5,6,8,9,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1]:=HIGH;

PIN[12]&PIN[13]&PIN[2]&PIN[3]:=0;
PIN[4]:=LOW;
IF PIN[5]<>LOW THEN ERROR(1);

D:=0;
FOR I:=0 TO 4 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[2]&PIN[3]:=D;
   PIN[4]:=LOW;
   IF (PIN[10]&PIN[9]&PIN[8]&PIN[6])<>((1 SHL I) SHR 1) THEN ERROR(1);
   PIN[4]:=HIGH;
   IF (PIN[10]&PIN[9]&PIN[8]&PIN[6])<>0 THEN ERROR(1);
   IF PIN[5]<>HIGH THEN ERROR(1);
   D:=D EXOR (1 SHL I);
   END;

ERROR(0);
END.

#NAME 74279,SN74279

#TEXT
Quad R/S Latch

This device contains
four R/S latches. Two
of these each have
2 S-inputs.

#PIN 16
 1 : Input   -R  RS 1
 2 : Input   -S1 RS 1
 3 : Input   -S2 RS 1
 4 : Output   Q  RS 1
 5 : Input   -R  RS 2
 6 : Input   -S  RS 2
 7 : Output   Q  RS 2
 8 : GND
 9 : Output   Q  RS 3
```

```
10 : Input   -R  RS 3
11 : Input   -S1 RS 3
12 : Input   -S2 RS 3
13 : Output   Q  RS 4
14 : Input   -R  RS 4
15 : Input   -S  RS 4
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,5,6,10,11,12,14,15] : INPUT;
PIN[4,7,9,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

D:=%0101;
PIN[11,2]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[10]&PIN[5]&PIN[1]:=D;
   PIN[15]&PIN[12]&PIN[6]&PIN[3]:=D EXOR %1111;
   IF (PIN[13]&PIN[9]&PIN[7]&PIN[1])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

D:=%0101;
PIN[12,3]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[10]&PIN[5]&PIN[1]:=D;
   PIN[15]&PIN[11]&PIN[6]&PIN[2]:=D EXOR %1111;
   IF (PIN[13]&PIN[9]&PIN[7]&PIN[1])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74280,SN74280

#TEXT
9-Bit Parity Generator/
Checker

This device contains
a parity generator/
checker for 9 bits
( 8 data bits plus 1
parity bit).

#PIN 14
 1 : Input   D6
 2 : Input   D7
 3 : N.C.
 4 : Input   D8
 5 : Output  even
 6 : Output  odd
 7 : GND
 8 : Input   D0
 9 : Input   D1
10 : Input   D2
11 : Input   D3
12 : Input   D4
13 : Input   D5
```

```
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,8,9,10,11,12,13] : INPUT;
PIN[5,6] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=0;
FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[2]&PIN[1]&PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9]&PIN[8]:=D;
   IF (NOT(PIN[5])<>(I MOD 2)) OR (PIN[6]<>(I MOD 2)) THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;

ERROR(0);
END.

#NAME 74283,SN74283

#TEXT
4-Bit Full Adder

This device contains
a full adder which
provides the sum of two
4-bit binary numbers.
The adder features full-
carry look ahead across
the four bits.

#PIN 16
 1 : Output  ä2
 2 : Input   B2
 3 : Input   A2
 4 : Output  ä1
 5 : Input   A1
 6 : Input   B1
 7 : Input   C0
 8 : GND
 9 : Output  C4
10 : Output  ä4
11 : Input   B4
12 : Input   A4
13 : Output  ä3
14 : Input   A3
15 : Input   B3
16 : +5V

#FAMILY Std,F,LS,S

#PROGRAM

BEGIN
PIN[2,3,5,6,7,11,12,14,15] : INPUT;
PIN[1,4,9,10,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR C:=0 TO 1 DO
   BEGIN
   PIN[7]:=C;
```

```
   FOR A:=0 TO 15 DO
      BEGIN
      PIN[12]&PIN[14]&PIN[3]&PIN[5]:=A;
      FOR B:=A MOD 3 TO 15 BY 3 DO
         BEGIN
         PIN[11]&PIN[15]&PIN[2]&PIN[6]:=B;
         IF (PIN[9]&PIN[10]&PIN[13]&PIN[1]&PIN[4])<>(C+A+B) THEN ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 74284,SN74284

#TEXT
4-Bit x 4-Bit
Multiplier (40 æA)

This device together
with the 74285 operates
as a 4 x 4-bit
multiplier.

#PIN 16
 1 : Input    C  Word 2
 2 : Input    B  Word 2
 3 : Input    A  Word 2
 4 : Input    D  Word 1
 5 : Input    A  Word 1
 6 : Input    B  Word 1
 7 : Input    C  Word 1
 8 : GND
 9 : Output   Q7
10 : Output   Q6
11 : Output   Q5
12 : Output   Q4
13 : Enable  -GB
14 : Enable  -GA
15 : Input    D  Word 2
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,13,14,15] : INPUT;
PIN[9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[13,14]:=LOW;

FOR A:=0 TO 15 DO
   BEGIN
   PIN[4]&PIN[7]&PIN[6]&PIN[5]:=A;
   FOR B:=A MOD 3 TO 15 BY 3 DO
      BEGIN
      PIN[15]&PIN[1]&PIN[2]&PIN[3]:=B;
      LOADMODEON;
      PIN[9,10,11,12] : LOAD HIGH;
      IF (PIN[9]&PIN[10]&PIN[11]&PIN[12])<>((A*B) SHR 4) THEN ERROR(1);
      LOADMODEOFF;
      END;
   END;
```

```
ERROR(0);
END.

#NAME 74285,SN74285

#TEXT
4-Bit x 4-Bit
Multiplier (40 æA)

This device together
with the 74284 operates
as a 4 x 4-bit
multiplier

#PIN 16
 1 : Input    C  Word 2
 2 : Input    B  Word 2
 3 : Input    A  Word 2
 4 : Input    D  Word 1
 5 : Input    A  Word 1
 6 : Input    B  Word 1
 7 : Input    C  Word 1
 8 : GND
 9 : Output   Q3
10 : Output   Q2
11 : Output   Q1
12 : Output   Q0
13 : Enable  -GB
14 : Enable  -GA
15 : Input    D  Word 2
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,13,14,15] : INPUT;
PIN[9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[13,14]:=LOW;

FOR A:=0 TO 15 DO
   BEGIN
   PIN[4]&PIN[7]&PIN[6]&PIN[5]:=A;
   FOR B:=A MOD 3 TO 15 BY 3 DO
      BEGIN
      PIN[15]&PIN[1]&PIN[2]&PIN[3]:=B;
      LOADMODEON;
      PIN[9,10,11,12] : LOAD HIGH;
      IF (PIN[9]&PIN[10]&PIN[11]&PIN[12])<>((A*B) AND 15) THEN ERROR(1);
      LOADMODEOFF;
      END;
   END;

ERROR(0);
END.

##NAME 74289,SN74289

#TEXT
64-Bit RAM (O.C.)

This device contains a
read/write memory (RAM)
arranged in 16x4 bits.
```

```
The outputs are OPEN-
COLLECTOR.

#PIN 16
 1 : Address   A0
 2 : -CS
 3 : -WE
 4 : Input     D1
 5 : Output   -Q1
 6 : Input     D2
 7 : Output   -Q2
 8 : GND
 9 : Output   -Q3
10 : Input     D3
11 : Output   -Q4
12 : Input     D4
13 : Address   A3
14 : Address   A2
15 : Address   A1
16 : +5V

#FAMILY F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,6,10,12,13,14,15] : INPUT;
PIN[5,7,9,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[3]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      PIN[12]&PIN[10]&PIN[6]&PIN[4]:=D;
      PIN[3]:=LOW;PIN[3]:=HIGH;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      LOADMODEON;
      PIN[11,9,7,5] : LOAD LOW;
      IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>0 THEN ERROR(1);
      PIN[11,9,7,5] : LOAD HIGH;
      IF (NOT(PIN[11])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[5]))<>D THEN ERROR(1);
      LOADMODEOFF;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74290,SN74290

#TEXT
Decade Counter

This device contains a
divide by 2 and a divide
```

```
by 5 counter.

#PIN 14
 1 : Input   R9(1)
 2 : N.C.
 3 : Input   R9(2)
 4 : Output  QC
 5 : Output  QB
 6 : N.C.
 7 : GND
 8 : Output  QD
 9 : Output  QA
10 : Clock A
11 : Clock B
12 : Input   R0(1)
13 : Input   R0(2)
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,3,10,11,12,13] : INPUT;
PIN[4,5,8,9] : OUTPUT;
PIN[14] : +5V;
PIN[7] : GND;
PIN[1,3,12,13]:=LOW;
PIN[10,11]:=HIGH;

PIN[1,3]:=HIGH;
PIN[1,3]:=LOW;
IF (PIN[8]&PIN[4]&PIN[5]&PIN[9])<>9 THEN ERROR(1);

PIN[12,13]:=HIGH;
PIN[12,13]:=LOW;
IF (PIN[8]&PIN[4]&PIN[5]&PIN[9])<>0 THEN ERROR(1);

PIN[10]:=LOW;PIN[10]:=HIGH;
IF PIN[9]<>HIGH THEN ERROR(1);

FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[8]&PIN[4]&PIN[5])<>I THEN ERROR(1);
   PIN[11]:=LOW;PIN[11]:=HIGH;
   END;

ERROR(0);
END.

#NAME 74293,SN74293

#TEXT
4-Bit Binary Counter

This device contains a
divide by 2 and a divide
by 8 counter.

#PIN 14
 1 : N.C.
 2 : N.C.
 3 : N.C.
 4 : Output  QC
 5 : Output  QB
 6 : N.C.
 7 : GND
```

```
 8 : Output  QD
 9 : Output  QA
10 : Clock A
11 : Clock B
12 : Input   R0(1)
13 : Input   R0(2)
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[10,11,12,13] : INPUT;
PIN[4,5,8,9] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[10,11]:=HIGH;

PIN[12,13]:=HIGH;PIN[12,13]:=LOW;
IF (PIN[8]&PIN[4]&PIN[5]&PIN[9])<>0 THEN ERROR(1);

PIN[10]:=LOW;PIN[10]:=HIGH;
IF PIN[9]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 DO
   BEGIN
   IF (PIN[8]&PIN[4]&PIN[5])<>I THEN ERROR(1);
   PIN[11]:=LOW;PIN[11]:=HIGH;
   END;

ERROR(0);
END.

#NAME 74295,SN74295

#TEXT
4-Bit Shift Register

This device contains
a 4-bit right-shift
register with parallel
or serial input and
output.

#PIN 14
 1 : Serial Input
 2 : Input   P0
 3 : Input   P1
 4 : Input   P2
 5 : Input   P3
 6 : Mode
 7 : GND
 8 : Out.-Enable OE
 9 : Clock
10 : Output  Q3
11 : Output  Q2
12 : Output  Q1
13 : Output  Q0
14 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,8,9] : INPUT;
```

```
PIN[10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[6,8]:=LOW;
PIN[9]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[1]:=X AND 1;
      PIN[9]:=LOW;PIN[9]:=HIGH;
      X:=X SHR 1;
      END;
   PIN[8]:=HIGH;
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>D THEN ERROR(1);
   PIN[8]:=LOW;
   D:=D EXOR %1111;
   END;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]:=HIGH;
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=D;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   PIN[6]:=LOW;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[8]:=HIGH;
      IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>X THEN ERROR(1);
      PIN[8]:=LOW;
      PIN[1]:=LOW;
      PIN[9]:=LOW;PIN[9]:=HIGH;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[13,12,11,10] : LOAD LOW;
IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>%0000 THEN ERROR(1);
PIN[13,12,11,10] : LOAD HIGH;
IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74298,SN74298

#TEXT
Quad 2-Input Multiplexer
with Storage

This device selects one
of two 4-bit data sources
and stores the selected
data.

#PIN 16
 1 : Input   B2
 2 : Input   A2
 3 : Input   A1
```

```
 4 : Input   B1
 5 : Input   C2
 6 : Input   D2
 7 : Input   D1
 8 : GND
 9 : Input   C1
10 : Word Select
11 : Clock
12 : Output  QD
13 : Output  QC
14 : Output  QB
15 : Output  QA
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10,11] : INPUT;
PIN[12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11]:=HIGH;

PIN[1,2,3,4,5,6,7,9]:=LOW;
PIN[10]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7]&PIN[9]&PIN[4]&PIN[3]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   PIN[7]&PIN[9]&PIN[4]&PIN[3]:=0;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,2,3,4,5,6,7,9]:=LOW;
PIN[10]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[1]&PIN[2]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   PIN[6]&PIN[5]&PIN[1]&PIN[2]:=0;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74299,SN74299

#TEXT
8-Bit TRI-STATE
Universal Shift
Register

This device contains a
universal right/left-
shift register with
parallel or serial input
and output and storage
capacity.

#PIN 20
```

```
 1 : S0
 2 : Out.-Enable -OE1
 3 : Out.-Enable -OE2
 4 : In-/Output   I/O6
 5 : In-/Output   I/O4
 6 : In-/Output   I/O2
 7 : In-/Output   I/O0
 8 : Output  Q0
 9 : -MR
10 : GND
11 : Input   DS0
12 : Clock CP
13 : In-/Output   I/O1
14 : In-/Output   I/O3
15 : In-/Output   I/O5
16 : In-/Output   I/O7
17 : Output  Q7
18 : Input   DS7
19 : S1
20 : +5V

#FAMILY ALS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,9,11,12,18,19] : INPUT;
PIN[8,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[12]:=LOW;
PIN[2,3,9]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;PIN[19]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[11]:=X AND 1;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      X:=X SHR 1;
      END;
   PIN[7,13,6,14,5,15,4,16] : OUTPUT;
   PIN[2,3]:=LOW;
   IF (PIN[7]&PIN[13]&PIN[6]&PIN[14]&
       PIN[5]&PIN[15]&PIN[4]&PIN[16])<>D THEN ERROR(1);
   PIN[2,3]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;PIN[19]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[18]:=X AND 1;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      X:=X SHR 1;
      END;
   PIN[16,4,15,5,14,6,13,7] : OUTPUT;
   PIN[2,3]:=LOW;
   IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
       PIN[14]&PIN[6]&PIN[13]&PIN[7])<>D THEN ERROR(1);
```

```
   PIN[2,3]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9]:=LOW;PIN[9]:=HIGH;
PIN[16,4,15,5,14,6,13,7] : OUTPUT;
PIN[2,3]:=LOW;
IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
    PIN[14]&PIN[6]&PIN[13]&PIN[7])<>0 THEN ERROR(1);
PIN[2,3]:=HIGH;

PIN[1,19]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7,13,6,14,5,15,4,16] : INPUT;
   PIN[1,19]:=HIGH;
   PIN[7]&PIN[13]&PIN[6]&PIN[14]&
   PIN[5]&PIN[15]&PIN[4]&PIN[16]:=D;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   PIN[1,19]:=LOW;
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[7,13,6,14,5,15,4,16] : OUTPUT;
      PIN[2,3]:=LOW;
      IF (PIN[7]&PIN[13]&PIN[6]&PIN[14]&
          PIN[5]&PIN[15]&PIN[4]&PIN[16])<>X THEN ERROR(1);
      IF PIN[17]<>(X AND 1) THEN ERROR(1);
      PIN[2,3]:=HIGH;
      PIN[1]:=HIGH;PIN[19]:=LOW;
      PIN[11]:=LOW;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      PIN[1,19]:=LOW;
      X:=X SHR 1;
      END;
   D:=D EXOR %11111111;
   END;

PIN[1,19]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[16,4,15,5,14,6,13,7] : INPUT;
   PIN[1,19]:=HIGH;
   PIN[16]&PIN[4]&PIN[15]&PIN[5]&
   PIN[14]&PIN[6]&PIN[13]&PIN[7]:=D;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   PIN[1,19]:=LOW;
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[16,4,15,5,14,6,13,7] : OUTPUT;
      PIN[2,3]:=LOW;
      IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
          PIN[14]&PIN[6]&PIN[13]&PIN[7])<>X THEN ERROR(1);
      IF PIN[8]<>(X AND 1) THEN ERROR(1);
      PIN[2,3]:=HIGH;
      PIN[1]:=LOW;PIN[19]:=HIGH;
      PIN[18]:=LOW;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      PIN[1,19]:=LOW;
      X:=X SHR 1;
      END;
   D:=D EXOR %11111111;
   END;

PIN[16,4,15,5,14,6,13,7] : OUTPUT;
LOADMODEON;
```

```
PIN[16,4,15,5,14,6,13,7] : LOAD LOW;
IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
    PIN[14]&PIN[6]&PIN[13]&PIN[7])<>%00000000 THEN ERROR(1);
PIN[16,4,15,5,14,6,13,7] : LOAD HIGH;
IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
    PIN[14]&PIN[6]&PIN[13]&PIN[7])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74301,SN74301

#TEXT
256-Bit RAM (O.C.)

This device contains a
read/write memory (RAM)
arranged as 256x1 bit
words.
The outputs are OPEN
COLLECTOR.

#PIN 16
 1 : Address  A0
 2 : Address  A1
 3 : -CS1
 4 : -CS2
 5 : -CS3
 6 : Output  -Q
 7 : Address  A3
 8 : GND
 9 : Address  A4
10 : Address  A5
11 : Address  A6
12 : Read/-Write
13 : Data
14 : Address  A7
15 : Address  A2
16 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,7,9,10,11,12,13,14,15] : INPUT;
PIN[6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=LOW;
PIN[12]:=HIGH;

D:=%0;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 255 DO
      BEGIN
      PIN[14]&PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[15]&PIN[2]&PIN[1]:=J;
      PIN[13]:=D;
      PIN[12]:=LOW;PIN[12]:=HIGH;
      D:=D EXOR %1;
      END;
   FOR J:=0 TO 255 DO
      BEGIN
      PIN[14]&PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[15]&PIN[2]&PIN[1]:=J;
      LOADMODEON;
```

```
      PIN[6] : LOAD LOW;
      IF PIN[6]<>LOW THEN ERROR(1);
      PIN[6] : LOAD HIGH;
      IF NOT(PIN[6])<>D THEN ERROR(1);
      LOADMODEOFF;
      D:=D EXOR %1;
      END;
   D:=D EXOR %1;
   END;

ERROR(0);
END.

##NAME 74319,SN74319

#TEXT
16x4-Bit RAM (O.C.)

This device contains
a read/write memory (RAM)
arranged as 16x4 bit
words.
The outputs are OPEN-
COLLECTOR.

#PIN 16
 1 : Address A0
 2 : -CS
 3 : -WE
 4 : Input   D1
 5 : Output  Q1
 6 : Input   D2
 7 : Output  Q2
 8 : GND
 9 : Output  Q3
10 : Input   D3
11 : Output  Q4
12 : Input   D4
13 : Address A3
14 : Address A2
15 : Address A1
16 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,6,10,12,13,14,15] : INPUT;
PIN[5,7,9,11] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[3]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      PIN[12]&PIN[10]&PIN[6]&PIN[4]:=D;
      PIN[3]:=LOW;PIN[3]:=HIGH;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 15 DO
      BEGIN
```

```
      PIN[13]&PIN[14]&PIN[15]&PIN[1]:=J;
      LOADMODEON;
      PIN[11,9,7,5] : LOAD LOW;
      IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>0 THEN ERROR(1);
      PIN[11,9,7,5] : LOAD HIGH;
      IF (PIN[11]&PIN[9]&PIN[7]&PIN[5])<>D THEN ERROR(1);
      LOADMODEOFF;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74322,SN74322

#TEXT
8-Bit TRI-STATE Universal
Shift Register with
Asynchronous Clear,
Cascadable

This device contains a
universal left/right-
shift register for
parallel or serial load
with multiplexed data
input and output ports.

#PIN 20
 1 : -RE
 2 : Serial/-Parallel
 3 : Input   D0
 4 : In-/Output   A/QA
 5 : In-/Output   C/QC
 6 : In-/Output   E/QE
 7 : In-/Output   G/QG
 8 : Out.-Enable -QE
 9 : -Clear
10 : GND
11 : Clock
12 : Output  QH
13 : In-/Output   H/QH
14 : In-/Output   F/QF
15 : In-/Output   D/QD
16 : In-/Output   B/QB
17 : Input   D1
18 : -Sign Extend -SE
19 : Data Select  S
20 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[1,2,3,8,9,11,17,19] : INPUT;
PIN[12,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,11]:=LOW;
PIN[2,8,9]:=HIGH;

D:=%01010101;

PIN[19]:=LOW;
```

```
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[3]:=X AND 1;
      PIN[11]:=HIGH;PIN[11]:=LOW;
      X:=X SHR 1;
      END;
   PIN[4,16,5,15,6,14,7,13] : OUTPUT;
   PIN[8]:=LOW;
   IF (PIN[4]&PIN[16]&PIN[5]&PIN[15]&
       PIN[6]&PIN[14]&PIN[7]&PIN[13])<>D THEN ERROR(1);
   PIN[8]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[19]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[17]:=X AND 1;
      PIN[11]:=HIGH;PIN[11]:=LOW;
      X:=X SHR 1;
      END;
   PIN[4,16,5,15,6,14,7,13] : OUTPUT;
   PIN[8]:=LOW;
   IF (PIN[4]&PIN[16]&PIN[5]&PIN[15]&
       PIN[6]&PIN[14]&PIN[7]&PIN[13])<>D THEN ERROR(1);
   PIN[8]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9]:=LOW;PIN[9]:=HIGH;
PIN[13,7,14,6,15,5,16,4] : OUTPUT;
PIN[8]:=LOW;
IF (PIN[13]&PIN[7]&PIN[14]&PIN[6]&
    PIN[15]&PIN[5]&PIN[16]&PIN[4])<>0 THEN ERROR(1);
PIN[8]:=HIGH;

PIN[1,19]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[4,16,5,15,6,14,7,13] : INPUT;
   PIN[2]:=LOW;
   PIN[4]&PIN[16]&PIN[5]&PIN[15]&
   PIN[6]&PIN[14]&PIN[7]&PIN[13]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[2]:=HIGH;
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[4,16,5,15,6,14,7,13] : OUTPUT;
      PIN[8]:=LOW;
      IF (PIN[4]&PIN[16]&PIN[5]&PIN[15]&
          PIN[6]&PIN[14]&PIN[7]&PIN[13])<>X THEN ERROR(1);
      IF PIN[12]<>(X AND 1) THEN ERROR(1);
      PIN[8]:=HIGH;
      PIN[3,17]:=LOW;
      PIN[11]:=HIGH;PIN[11]:=LOW;
      X:=X SHR 1;
      END;
   D:=D EXOR %11111111;
   END;
```

```
LOADMODEON;
PIN[13,7,14,6,15,5,16,4] : LOAD LOW;
IF (PIN[13]&PIN[7]&PIN[14]&PIN[6]&
    PIN[15]&PIN[5]&PIN[16]&PIN[4])<>%00000000 THEN ERROR(1);
PIN[13,7,14,6,15,5,16,4] : LOAD HIGH;
IF (PIN[13]&PIN[7]&PIN[14]&PIN[6]&
    PIN[15]&PIN[5]&PIN[16]&PIN[4])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74323,SN74323

#TEXT
8-Bit TRI-STATE
Universal Shift
Register with Clear,
Cascadable

This device contains
a universal bidirectional
shift register with
parallel or serial input
and output and storage
capacity.

#PIN 20
 1 : S0
 2 : Out.-Enable -OE1
 3 : Out.-Enable -OE2
 4 : In-/Output   I/O6
 5 : In-/Output   I/O4
 6 : In-/Output   I/O2
 7 : In-/Output   I/O0
 8 : Output  Q0
 9 : -SR
10 : GND
11 : Input   DS0
12 : Clock CP
13 : In-/Output   I/O1
14 : In-/Output   I/O3
15 : In-/Output   I/O5
16 : In-/Output   I/O7
17 : Output  Q7
18 : Input   DS7
19 : S1
20 : +5V

#FAMILY ALS,F,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,9,11,12,18,19] : INPUT;
PIN[8,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[12]:=LOW;
PIN[2,3,9]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;PIN[19]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
```

```
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[11]:=X AND 1;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      X:=X SHR 1;
      END;
   PIN[7,13,6,14,5,15,4,16] : OUTPUT;
   PIN[2,3]:=LOW;
   IF (PIN[7]&PIN[13]&PIN[6]&PIN[14]&
       PIN[5]&PIN[15]&PIN[4]&PIN[16])<>D THEN ERROR(1);
   PIN[2,3]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;PIN[19]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[18]:=X AND 1;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      X:=X SHR 1;
      END;
   PIN[16,4,15,5,14,6,13,7] : OUTPUT;
   PIN[2,3]:=LOW;
   IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
       PIN[14]&PIN[6]&PIN[13]&PIN[7])<>D THEN ERROR(1);
   PIN[2,3]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9]:=LOW;
PIN[12]:=HIGH;PIN[12]:=LOW;
PIN[9]:=HIGH;
PIN[16,4,15,5,14,6,13,7] : OUTPUT;
PIN[2,3]:=LOW;
IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
    PIN[14]&PIN[6]&PIN[13]&PIN[7])<>0 THEN ERROR(1);
PIN[2,3]:=HIGH;

PIN[1,19]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7,13,6,14,5,15,4,16] : INPUT;
   PIN[1,19]:=HIGH;
   PIN[7]&PIN[13]&PIN[6]&PIN[14]&
   PIN[5]&PIN[15]&PIN[4]&PIN[16]:=D;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   PIN[1,19]:=LOW;
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[7,13,6,14,5,15,4,16] : OUTPUT;
      PIN[2,3]:=LOW;
      IF (PIN[7]&PIN[13]&PIN[6]&PIN[14]&
          PIN[5]&PIN[15]&PIN[4]&PIN[16])<>X THEN ERROR(1);
      IF PIN[17]<>(X AND 1) THEN ERROR(1);
      PIN[2,3]:=HIGH;
      PIN[1]:=HIGH;PIN[19]:=LOW;
      PIN[11]:=LOW;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      PIN[1,19]:=LOW;
      X:=X SHR 1;
      END;
   D:=D EXOR %11111111;
   END;
```

```
PIN[1,19]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[16,4,15,5,14,6,13,7] : INPUT;
   PIN[1,19]:=HIGH;
   PIN[16]&PIN[4]&PIN[15]&PIN[5]&
   PIN[14]&PIN[6]&PIN[13]&PIN[7]:=D;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   PIN[1,19]:=LOW;
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[16,4,15,5,14,6,13,7] : OUTPUT;
      PIN[2,3]:=LOW;
      IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
          PIN[14]&PIN[6]&PIN[13]&PIN[7])<>X THEN ERROR(1);
      IF PIN[8]<>(X AND 1) THEN ERROR(1);
      PIN[2,3]:=HIGH;
      PIN[1]:=LOW;PIN[19]:=HIGH;
      PIN[18]:=LOW;
      PIN[12]:=HIGH;PIN[12]:=LOW;
      PIN[1,19]:=LOW;
      X:=X SHR 1;
      END;
   D:=D EXOR %11111111;
   END;

PIN[9]:=LOW;
PIN[12]:=HIGH;PIN[12]:=LOW;
PIN[9]:=HIGH;
PIN[16,4,15,5,14,6,13,7] : OUTPUT;
PIN[2,3]:=LOW;
IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
    PIN[14]&PIN[6]&PIN[13]&PIN[7])<>0 THEN ERROR(1);
PIN[2,3]:=HIGH;

LOADMODEON;
PIN[16,4,15,5,14,6,13,7] : LOAD LOW;
IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
    PIN[14]&PIN[6]&PIN[13]&PIN[7])<>%00000000 THEN ERROR(1);
PIN[16,4,15,5,14,6,13,7] : LOAD HIGH;
IF (PIN[16]&PIN[4]&PIN[15]&PIN[5]&
    PIN[14]&PIN[6]&PIN[13]&PIN[7])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74340,SN74340

#TEXT
TRI-STATE Inverting
Octal Buffer

This device contains
eight inverting
buffers with three-state
outputs.

#PIN 20
 1 : -G1
 2 : Input    E0
 3 : Output  -Q7
 4 : Input    E1
 5 : Output  -Q6
 6 : Input    E2
```

```
 7 : Output  -Q5
 8 : Input    E3
 9 : Output  -Q4
10 : GND
11 : Input    E4
12 : Output  -Q3
13 : Input    E5
14 : Output  -Q2
15 : Input    E6
16 : Output  -Q1
17 : Input    E7
18 : Output  -Q0
19 : -G2
20 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[18])&NOT(PIN[16])&NOT(PIN[14])&NOT(PIN[12])&
       NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[5])&NOT(PIN[3]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74341,SN74341

#TEXT
TRI-STATE Octal
Buffer

This device contains
eight non-inverting
buffers with three-
state outputs.

#PIN 20
 1 : -G1
 2 : Input   E0
 3 : Output  Q7
 4 : Input   E1
 5 : Output  Q6
```

```
 6 : Input   E2
 7 : Output  Q5
 8 : Input   E3
 9 : Output  Q4
10 : GND
11 : Input   E4
12 : Output  Q3
13 : Input   E5
14 : Output  Q2
15 : Input   E6
16 : Output  Q1
17 : Input   E7
18 : Output  Q0
19 : -G2
20 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=LOW;
PIN[1]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1]:=LOW;PIN[19]:=HIGH;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1]:=HIGH;PIN[19]:=LOW;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74344,SN74344

#TEXT
TRI-STATE Octal
Buffer

This device contains
eight non-inverting
buffers with three-
state outputs.

#PIN 20
 1 : -G1
 2 : Input   E0
 3 : Output  Q7
```

```
 4 : Input   E1
 5 : Output  Q6
 6 : Input   E2
 7 : Output  Q5
 8 : Input   E3
 9 : Output  Q4
10 : GND
11 : Input   E4
12 : Output  Q3
13 : Input   E5
14 : Output  Q2
15 : Input   E6
16 : Output  Q1
17 : Input   E7
18 : Output  Q0
19 : -G2
20 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74347,SN74347

#TEXT
BCD-to-7-Segment-Decoder/
Driver (O.C., 7V)

This device converts
BCD input data into
control data for 7-segment
displays. The outputs are
OPEN-COLLECTOR with a
maximum output voltage of
+7V.
```

```
#PIN 16
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : -LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input   D
 7 : BCD-Input   A
 8 : GND
 9 : 7-Seg.-Output  -e
10 : 7-Seg.-Output  -d
11 : 7-Seg.-Output  -c
12 : 7-Seg.-Output  -b
13 : 7-Seg.-Output  -a
14 : 7-Seg.-Output  -g
15 : 7-Seg.-Output  -f
16 : +5V...+7V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   LOADMODEON;
   PIN[9,10,11,12,13,14,15] : LOAD LOW;
   IF (PIN[14]&PIN[15]&PIN[9]&PIN[10]&
       PIN[11]&PIN[12]&PIN[13])<>0 THEN ERROR(1);
   PIN[9,10,11,12,13,14,15] : LOAD HIGH;
   D:=NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[9])&NOT(PIN[10])&
      NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13]);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74348,SN74348

#TEXT
Binary 8-to-3-Priority
Decoder (TRI-STATE)

This device arranges the
eight input signals in
order of importance.
The input with the
highest number is con-
verted into a BCD number.
```

Several devices can be
cascaded.

```
#PIN 16
 1 : Input   4
 2 : Input   5
 3 : Input   6
 4 : Input   7
 5 : -E1
 6 : Output  C
 7 : Output  B
 8 : GND
 9 : Output  A
10 : Input   0
11 : Input   1
12 : Input   2
13 : Input   3
14 : -GS
15 : -E0
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,10,11,12,13] : INPUT;
PIN[6,7,9,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[5]:=LOW;

D:=%11111111;
PIN[10]&PIN[11]&PIN[12]&PIN[13]&PIN[1]&
PIN[2]&PIN[3]&PIN[4]:=D;
LOADMODEON;
PIN[6,7,9] : LOAD LOW;
IF (PIN[6]&PIN[7]&PIN[9])<>%000 THEN ERROR(1);
PIN[6,7,9] : LOAD HIGH;
IF (PIN[6]&PIN[7]&PIN[9])<>%111 THEN ERROR(1);
LOADMODEOFF;
IF (PIN[14]<>HIGH) OR (PIN[15]<>LOW) THEN ERROR(1);
D:=D SHR 1;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[10]&PIN[11]&PIN[12]&PIN[13]&PIN[1]&
   PIN[2]&PIN[3]&PIN[4]:=D;
   IF (NOT(PIN[6])&NOT(PIN[7])&NOT(PIN[9]))<>I THEN ERROR(1);
   IF (PIN[14]<>LOW) OR (PIN[15]<>HIGH) THEN ERROR(1);
   D:=D SHR 1;
   END;

ERROR(0);
END.
```
•

```
#NAME 74351,SN74351

#TEXT
Dual TRI-STATE 8-Input
Multiplexer
(Inverted Output)

This device contains
two 8-input data selec-
tors with common
address and four common
data lines.

#PIN 20
 1 : Output  -Q MP 1
 2 : -G
 3 : Address A
 4 : Address B
 5 : Address C
 6 : Input   D0 MP 1
 7 : Input   D1 MP 1
 8 : Input   D2 MP 1
 9 : Input   D3 MP 1
10 : GND
11 : Input   D7
12 : Input   D6
13 : Input   D5
14 : Input   D4
15 : Input   D3 MP 2
16 : Input   D2 MP 2
17 : Input   D1 MP 2
18 : Input   D0 MP 2
19 : Output  -Q MP 2
20 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[1,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[2]:=HIGH;

D:=%10100101;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[9]&PIN[8]&PIN[7]&PIN[6]:=D;
   X:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[5]&PIN[4]&PIN[3]:=J;
      PIN[2]:=LOW;
      X:=(X SHL 1) OR NOT(PIN[1]);
      PIN[2]:=HIGH;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[1] : LOAD LOW;
IF PIN[1]<>LOW THEN ERROR(1);
PIN[1] : LOAD HIGH;
```

```
IF PIN[1]<>HIGH THEN ERROR(1);
LOADMODEOFF;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   X:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[5]&PIN[4]&PIN[3]:=J;
      PIN[2]:=LOW;
      X:=(X SHL 1) OR NOT(PIN[19]);
      PIN[2]:=HIGH;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19] : LOAD LOW;
IF PIN[19]<>LOW THEN ERROR(1);
PIN[19] : LOAD HIGH;
IF PIN[19]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74352,SN74352

#TEXT
Dual 4-Input Multi-
plexer (Inverted
Output)

This device contains
two 1-of-4 data selec-
tors with common
address inputs, sepa-
rate strobe inputs and
inverted outputs.

#PIN 16
 1 : Strobe     MP 1
 2 : Input    B
 3 : Input    3 MP 1
 4 : Input    2 MP 1
 5 : Input    1 MP 1
 6 : Input    0 MP 1
 7 : Output  -Q MP 1
 8 : GND
 9 : Output  -Q MP 2
10 : Input    0 MP 2
11 : Input    1 MP 2
12 : Input    2 MP 2
13 : Input    3 MP 2
14 : Input    A
15 : Strobe     MP 2
16 : +5V

#FAMILY ALS,AS,F,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,10,11,12,13,14,15] : INPUT;
PIN[7,9] : OUTPUT;
```

```
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[1]:=LOW;
      X:=(X SHL 1) OR NOT(PIN[7]);
      PIN[1]:=HIGH;
      IF PIN[7]<>HIGH THEN ERROR(1);
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[11]&PIN[10]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[15]:=LOW;
      X:=(X SHL 1) OR NOT(PIN[9]);
      PIN[15]:=HIGH;
      IF PIN[9]<>HIGH THEN ERROR(1);
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74353,SN74353

#TEXT
Dual 4-Input TRI-STATE
Multiplexer
(Inverted Output)

This device contains
two  1-of-4 data selec-
tors with common
address inputs, sepa-
rate output enable
inputs and inverting
three-state outputs.

#PIN 16
 1 : -OE        MP 1
 2 : Input     B
 3 : Input     3 MP 1
 4 : Input     2 MP 1
 5 : Input     1 MP 1
 6 : Input     0 MP 1
 7 : Output   -Q MP 1
 8 : GND
 9 : Output   -Q MP 2
10 : Input     0 MP 2
```

```
11 : Input    1 MP 2
12 : Input    2 MP 2
13 : Input    3 MP 2
14 : Input    A
15 : -OE        MP 2
16 : +5V

#FAMILY ALS,AS,F,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,10,11,12,13,14,15] : INPUT;
PIN[7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[1]:=LOW;
      X:=(X SHL 1) OR NOT(PIN[7]);
      PIN[1]:=HIGH;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[7] : LOAD LOW;
IF PIN[7]<>LOW THEN ERROR(1);
PIN[7] : LOAD HIGH;
IF PIN[7]<>HIGH THEN ERROR(1);
LOADMODEOFF;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[11]&PIN[10]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[15]:=LOW;
      X:=(X SHL 1) OR NOT(PIN[9]);
      PIN[15]:=HIGH;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[9] : LOAD LOW;
IF PIN[9]<>LOW THEN ERROR(1);
PIN[9] : LOAD HIGH;
IF PIN[9]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
```

```
#NAME 74354,SN74354

#TEXT
8-Input TRI-STATE
Multiplexer with Latch

This device contains a
data selector which
selects one of eight
input signals using a 3-
bit binary code.
The address and data
signals can be stored
by means of special
control inputs.

#PIN 20
 1 : Input   D7
 2 : Input   D6
 3 : Input   D5
 4 : Input   D4
 5 : Input   D3
 6 : Input   D2
 7 : Input   D1
 8 : Input   D0
 9 : -Data Control
10 : GND
11 : -Address Control
12 : Address A2
13 : Address A1
14 : Address A0
15 : Enable -E1
16 : Enable -E2
17 : Enable  E3
18 : Output  -Q
19 : Output   Q
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17] : INPUT;
PIN[18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[15,16,17]:=LOW;
PIN[9,11]:=HIGH;

D:=%10100101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=D;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=0;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[12]&PIN[13]&PIN[14]:=J;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      PIN[12]&PIN[13]&PIN[14]:=0;
      PIN[17]:=HIGH;
      X:=(X SHL 1) OR PIN[19];
      Y:=(Y SHL 1) OR NOT(PIN[18]);
      PIN[17]:=LOW;
      END;
```

```
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,18] : LOAD LOW;
IF (PIN[19]<>LOW) OR (PIN[18]<>LOW) THEN ERROR(1);
PIN[19,18] : LOAD HIGH;
IF (PIN[19]<>HIGH) OR (PIN[18]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74355,SN74355

#TEXT
8-Input Multiplexer
with Latch (O.C.)

This device contains a
data selector which
selects one of eight
input signals using a
3-bit binary code.
The address and data
signals are stored by
means of special control
inputs.

#PIN 20
 1 : Input   D7
 2 : Input   D6
 3 : Input   D5
 4 : Input   D4
 5 : Input   D3
 6 : Input   D2
 7 : Input   D1
 8 : Input   D0
 9 : -Data Control
10 : GND
11 : -Address Control
12 : Address A2
13 : Address A1
14 : Address A0
15 : Enable -E1
16 : Enable -E2
17 : Enable  E3
18 : Output -Q
19 : Output  Q
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17] : INPUT;
PIN[18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[15,16,17]:=LOW;
PIN[9,11]:=HIGH;

D:=%10100101;
FOR I:=0 TO 1 DO
   BEGIN
```

```
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=D;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=0;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[12]&PIN[13]&PIN[14]:=J;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      PIN[12]&PIN[13]&PIN[14]:=0;
      PIN[17]:=HIGH;
      LOADMODEON;
      PIN[19,18] : LOAD LOW;
      IF (PIN[19]<>LOW) OR (PIN[18]<>LOW) THEN ERROR(1);
      PIN[19,18] : LOAD HIGH;
      X:=(X SHL 1) OR PIN[19];
      Y:=(Y SHL 1) OR NOT(PIN[18]);
      LOADMODEOFF;
      PIN[17]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74356,SN74356

#TEXT
8-Input TRI-STATE
Multiplexer with Latches

This device contains a
data selector which
selects one of eight
input signals using a
3-bit binary code.
The address and data
signals are stored by
means of special control
inputs.

#PIN 20
 1 : Input   D7
 2 : Input   D6
 3 : Input   D5
 4 : Input   D4
 5 : Input   D3
 6 : Input   D2
 7 : Input   D1
 8 : Input   D0
 9 : Clock
10 : GND
11 : -Address Control
12 : Address A2
13 : Address A1
14 : Address A0
15 : Enable -E1
16 : Enable -E2
17 : Enable  E3
18 : Output -Q
19 : Output   Q
20 : +5V

#FAMILY LS

#PROGRAM
```

```
BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17] : INPUT;
PIN[18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[9,15,16,17]:=LOW;
PIN[11]:=HIGH;

D:=%10100101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=0;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[12]&PIN[13]&PIN[14]:=J;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      PIN[12]&PIN[13]&PIN[14]:=0;
      PIN[17]:=HIGH;
      X:=(X SHL 1) OR PIN[19];
      Y:=(Y SHL 1) OR NOT(PIN[18]);
      PIN[17]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,18] : LOAD LOW;
IF (PIN[19]<>LOW) OR (PIN[18]<>LOW) THEN ERROR(1);
PIN[19,18] : LOAD HIGH;
IF (PIN[19]<>HIGH) OR (PIN[18]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74357,SN74357

#TEXT
8-Input Multiplexer
with Latches (O.C.)

This device contains a
data selector which
selects one of eight
input signals using a
3-bit binary code.
The address and data
signals can be stored
by means of special
control inputs.

#PIN 20
 1 : Input   D7
 2 : Input   D6
 3 : Input   D5
 4 : Input   D4
 5 : Input   D3
 6 : Input   D2
 7 : Input   D1
 8 : Input   D0
 9 : Clock
10 : GND
```

```
11 : -Address Control
12 : Address A2
13 : Address A1
14 : Address A0
15 : Enable -E1
16 : Enable -E2
17 : Enable  E3
18 : Output -Q
19 : Output  Q
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17] : INPUT;
PIN[18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[9,15,16,17]:=LOW;
PIN[11]:=HIGH;

D:=%10100101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]&PIN[8]:=0;
   X:=0;Y:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[12]&PIN[13]&PIN[14]:=J;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      PIN[12]&PIN[13]&PIN[14]:=0;
      PIN[17]:=HIGH;
      LOADMODEON;
      PIN[19,18] : LOAD LOW;
      IF (PIN[19]<>LOW) OR (PIN[18]<>LOW) THEN ERROR(1);
      PIN[19,18] : LOAD HIGH;
      X:=(X SHL 1) OR PIN[19];
      Y:=(Y SHL 1) OR NOT(PIN[18]);
      LOADMODEOFF;
      PIN[17]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74365,SN74365

#TEXT
Hex TRI-STATE Buffer

This device contains
six non-inverting buffers
with common enable.

#PIN 16
 1 : Enable   -E1
 2 : Input     Gate 1
 3 : Output    Gate 1
 4 : Input     Gate 2
 5 : Output    Gate 2
```

```
 6 : Input     Gate   3
 7 : Output    Gate   3
 8 : GND
 9 : Output    Gate   4
10 : Input     Gate   4
11 : Output    Gate   5
12 : Input     Gate   5
13 : Output    Gate   6
14 : Input     Gate   6
15 : Enable   -E2
16 : +5V

#FAMILY Std,F,LS

#PROGRAM

BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&
       PIN[9]&PIN[11]&PIN[13])<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74366,SN74366

#TEXT
Inverting Hex TRI-
STATE Buffer

This device contains
six inverting buffers
with common enable.

#PIN 16
 1 : Enable   -E1
 2 : Input     Gate   1
 3 : Output    Gate   1
 4 : Input     Gate   2
 5 : Output    Gate   2
 6 : Input     Gate   3
 7 : Output    Gate   3
 8 : GND
 9 : Output    Gate   4
10 : Input     Gate   4
```

```
11 : Output    Gate   5
12 : Input     Gate   5
13 : Output    Gate   6
14 : Input     Gate   6
15 : Enable   -E2
16 : +5V

#FAMILY Std,F,LS

#PROGRAM

BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (NOT(PIN[3])&NOT(PIN[5])&NOT(PIN[7])&
       NOT(PIN[9])&NOT(PIN[11])&NOT(PIN[13]))<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74367,SN74367

#TEXT
Hex TRI-STATE Buffer

This device contains
six non-inverting buffers
with two enable inputs.

#PIN 16
 1 : Enable   -E1
 2 : Input     Gate   1
 3 : Output    Gate   1
 4 : Input     Gate   2
 5 : Output    Gate   2
 6 : Input     Gate   3
 7 : Output    Gate   3
 8 : GND
 9 : Output    Gate   4
10 : Input     Gate   4
11 : Output    Gate   5
12 : Input     Gate   5
13 : Output    Gate   6
14 : Input     Gate   6
15 : Enable   -E2
16 : +5V
```

```
#FAMILY Std,F,LS

#PROGRAM

BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&
       PIN[9]&PIN[11]&PIN[13])<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74368,SN74368

#TEXT
Inverting Hex TRI-
STATE Buffer

This device contains
six inverting buffers
with two enable inputs.

#PIN 16
 1 : Enable   -E1
 2 : Input     Gate   1
 3 : Output    Gate   1
 4 : Input     Gate   2
 5 : Output    Gate   2
 6 : Input     Gate   3
 7 : Output    Gate   3
 8 : GND
 9 : Output    Gate   4
10 : Input     Gate   4
11 : Output    Gate   5
12 : Input     Gate   5
13 : Output    Gate   6
14 : Input     Gate   6
15 : Enable   -E2
16 : +5V

#FAMILY Std,F,LS

#PROGRAM
```

```
BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (NOT(PIN[3])&NOT(PIN[5])&NOT(PIN[7])&
       NOT(PIN[9])&NOT(PIN[11])&NOT(PIN[13]))<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74373,SN74373

#TEXT
TRI-STATE 8-Bit-D-Type
Latch

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
 2 : Output  Q0
 3 : Input   D0
 4 : Input   D1
 5 : Output  Q1
 6 : Output  Q2
 7 : Input   D2
 8 : Input   D3
 9 : Output  Q3
10 : GND
11 : Latch Enable
12 : Output  Q4
13 : Input   D4
14 : Input   D5
15 : Output  Q5
16 : Output  Q6
17 : Input   D6
18 : Input   D7
19 : Output  Q7
20 : +5V

#FAMILY ALS,AS,F,LS,S

#PROGRAM
```

```
BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
   PIN[1]:=LOW;
   IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,16,15,12,9,6,5,2] : LOAD LOW;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[19,16,15,12,9,6,5,2] : LOAD HIGH;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74374,SN74374

#TEXT
TRI-STATE 8-Bit-D-Type
Register

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
 2 : Output  Q0
 3 : Input   D0
 4 : Input   D1
 5 : Output  Q1
 6 : Output  Q2
 7 : Input   D2
 8 : Input   D3
 9 : Output  Q3
10 : GND
11 : Clock
12 : Output  Q4
13 : Input   D4
14 : Input   D5
15 : Output  Q5
16 : Output  Q6
17 : Input   D6
18 : Input   D7
19 : Output  Q7
20 : +5V

#FAMILY ALS,AS,F,LS,S
```

```
#PROGRAM

BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
   PIN[1]:=LOW;
   IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,16,15,12,9,6,5,2] : LOAD LOW;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[19,16,15,12,9,6,5,2] : LOAD HIGH;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74375,SN74375

#TEXT
Dual 2-Bit-D-Type
Latch with Enable

This device contains
four bistable memory
elements.

#PIN 16
 1 : Input    D SE 1
 2 : Output  -Q SE 1
 3 : Output   Q SE 1
 4 : Enable     SE 1+2
 5 : Output   Q SE 2
 6 : Output  -Q SE 2
 7 : Input    D SE 2
 8 : GND
 9 : Input    D SE 3
10 : Output  -Q SE 3
11 : Output   Q SE 3
12 : Enable     SE 3+4
13 : Output   Q SE 4
14 : Output  -Q SE 4
15 : Input    D SE 4
16 : +5V

#FAMILY LS

#PROGRAM
```

```
BEGIN
PIN[1,4,7,9,12,15] : INPUT;
PIN[2,3,5,6,10,11,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[4,12]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[9]&PIN[7]&PIN[1]:=D;
   PIN[4,12]:=HIGH;PIN[4,12]:=LOW;
   PIN[15]&PIN[9]&PIN[7]&PIN[1]:=0;
   IF (PIN[13]&PIN[11]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   IF (NOT(PIN[14])&NOT(PIN[10])&NOT(PIN[6])&NOT(PIN[2]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74376,SN74376

#TEXT
4-Bit-J-K -Register
with Clear

This device contains
four J-K flip-flops
with common clock and
common clear.

#PIN 16
 1 : -Clear
 2 : Input    J  FF 1
 3 : Input   -K  FF 1
 4 : Output   Q  FF 1
 5 : Output   Q  FF 2
 6 : Input   -K  FF 2
 7 : Input    J  FF 2
 8 : GND
 9 : Clock
10 : Input    J  FF 3
11 : Input   -K  FF 3
12 : Output   Q  FF 3
13 : Output   Q  FF 4
14 : Input   -K  FF 4
15 : Input    J  FF 4
16 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,3,6,7,9,10,11,14,15] : INPUT;
PIN[4,5,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=LOW;
PIN[1]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[10]&PIN[7]&PIN[2]:=D;
```

```
   PIN[14]&PIN[11]&PIN[6]&PIN[3]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   IF (PIN[13]&PIN[12]&PIN[5]&PIN[4])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[13]&PIN[12]&PIN[5]&PIN[4])<>%0000 THEN ERROR(1);
PIN[15,10,7,2]:=HIGH;
PIN[14,11,6,3]:=LOW;
PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[13]&PIN[12]&PIN[5]&PIN[4])<>%1111 THEN ERROR(1);

ERROR(0);
END.

#NAME 74377,SN74377

#TEXT
8-Bit-D-Type Register
with Enable

This device contains
eight high speed edge-
triggered bistable
memory elements with
one enable input.

#PIN 20
 1 : -Enable
 2 : Output  Q0
 3 : Input   D0
 4 : Input   D1
 5 : Output  Q1
 6 : Output  Q2
 7 : Input   D2
 8 : Input   D3
 9 : Output  Q3
10 : GND
11 : Clock
12 : Output  Q4
13 : Input   D4
14 : Input   D5
15 : Output  Q5
16 : Output  Q6
17 : Input   D6
18 : Input   D7
19 : Output  Q7
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
```

```
   IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74378,SN74378

#TEXT
6-Bit-D-Type Register
with Enable

This device contains
six high-speed edge-
triggered bistable
memory elements with
one enable input.

#PIN 16
 1 : -Enable
 2 : Output  Q0
 3 : Input   D0
 4 : Input   D1
 5 : Output  Q1
 6 : Input   D2
 7 : Output  Q2
 8 : GND
 9 : Clock
10 : Output  Q4
11 : Input   D4
12 : Output  Q5
13 : Input   D5
14 : Input   D6
15 : Output  Q6
16 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[1,3,4,6,9,11,13,14] : INPUT;
PIN[2,5,7,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,9]:=LOW;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=0;
   IF (PIN[15]&PIN[12]&PIN[10]&PIN[7]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 74379,SN74379

#TEXT
4-Bit-D-Type Register
with Enable
```

```
This device contains
four high-speed edge-
triggered bistable
memory elements with
one enable input and
complementary outputs.

#PIN 16
 1 : -Enable
 2 : Output   Q0
 3 : Output  -Q0
 4 : Input    D0
 5 : Input    D1
 6 : Output  -Q1
 7 : Output   Q1
 8 : GND
 9 : Clock
10 : Output   Q2
11 : Output  -Q2
12 : Input    D2
13 : Input    D3
14 : Output  -Q3
15 : Output   Q3
16 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[1,4,5,9,12,13] : Input;
PIN[2,3,6,7,10,11,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,9]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=0;
   IF (PIN[15]&PIN[10]&PIN[7]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[14])&NOT(PIN[11])&NOT(PIN[6])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74385,SN74385

#TEXT
Quad 1-Bit Full Adder

This device contains
four synchronous serial
adders/subtractors for
executing dual comple-
ment operations with
common clock and clear.

#PIN 20
 1 : Clock
 2 : Output  ä Add.1
 3 : S/-A      Add.1
```

```
 4 : Input   B Add.1
 5 : Input   A Add.1
 6 : Input   A Add.2
 7 : Input   B Add.2
 8 : S/-A      Add.2
 9 : Output  ä Add.2
10 : GND
11 : Clear
12 : Output  ä Add.3
13 : S/-A      Add.3
14 : Input   B Add.3
15 : Input   A Add.3
16 : Input   A Add.4
17 : Input   B Add.4
18 : S/-A      Add.4
19 : Output  ä Add.4
20 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[1,3,4,5,6,7,8,11,13,14,15,16,17,18] : INPUT;
PIN[2,9,12,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;
PIN[11]:=HIGH;

PIN[11]:=LOW;PIN[11]:=HIGH;

PIN[3,8,13,18]:=LOW;
FOR A:=0 TO 1 DO
   BEGIN
   PIN[5,6,15,16]:=A;
   FOR B:=0 TO 1 DO
      BEGIN
      PIN[4,7,14,17]:=B;
      PIN[1]:=HIGH;PIN[1]:=LOW;
      IF PIN[2]<>(A EXOR B) THEN ERROR(1);
      IF PIN[9]<>(A EXOR B) THEN ERROR(1);
      IF PIN[12]<>(A EXOR B) THEN ERROR(1);
      IF PIN[19]<>(A EXOR B) THEN ERROR(1);
      END;
   END;

PIN[3,8,13,18]:=HIGH;
FOR A:=1 DOWNTO 0 DO
   BEGIN
   PIN[5,6,15,16]:=A;
   FOR B:=1 DOWNTO 0 DO
      BEGIN
      PIN[4,7,14,17]:=B;
      PIN[1]:=HIGH;PIN[1]:=LOW;
      IF PIN[2]<>(A NAND B) THEN ERROR(1);
      IF PIN[9]<>(A NAND B) THEN ERROR(1);
      IF PIN[12]<>(A NAND B) THEN ERROR(1);
      IF PIN[19]<>(A NAND B) THEN ERROR(1);
      END;
   END;

PIN[11]:=LOW;PIN[11]:=HIGH;
IF PIN[2]<>LOW THEN ERROR(1);
IF PIN[9]<>LOW THEN ERROR(1);
IF PIN[12]<>LOW THEN ERROR(1);
IF PIN[19]<>LOW THEN ERROR(1);
```

```
ERROR(0);
END.

#NAME 74386,SN74386

#TEXT
Quad 2-Input EXOR-Gate

This device contains
four EXOR gates each
with 2 inputs.

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Output    Gate   1
 4 : Output    Gate   2
 5 : Input   1 Gate   2
 6 : Input   2 Gate   2
 7 : GND
 8 : Input   1 Gate   3
 9 : Input   2 Gate   3
10 : Output    Gate   3
11 : Output    Gate   4
12 : Input   1 Gate   4
13 : Input   2 Gate   4
14 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] EXOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] EXOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] EXOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] EXOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74390,SN74390

#TEXT
Dual Decade Counter

This device is composed
of a divide-by-two
counter and a divide-by-
five counter.

#PIN 16
 1 : Clock A    Counter 1
 2 : Reset      Counter 1
```

```
 3 : Output  QA Counter 1
 4 : Clock B    Counter 1
 5 : Output  QB Counter 1
 6 : Output  QC Counter 1
 7 : Output  QD Counter 1
 8 : GND
 9 : Output  QD Counter 2
10 : Output  QC Counter 2
11 : Output  QB Counter 2
12 : Clock B    Counter 2
13 : Output  QA Counter 2
14 : Reset      Counter 2
15 : Clock A    Counter 2
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,4,12,14,15] : INPUT;
PIN[3,5,6,7,9,10,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,14]:=LOW;
PIN[1,4,12,15]:=HIGH;

PIN[2]:=HIGH;PIN[2]:=LOW;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF PIN[3]<>HIGH THEN ERROR(1);
FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[5])<>I THEN ERROR(1);
   PIN[4]:=LOW;PIN[4]:=HIGH;
   END;

PIN[2]:=HIGH;PIN[2]:=LOW;
IF (PIN[7]&PIN[6]&PIN[5])<>0 THEN ERROR(1);

PIN[14]:=HIGH;PIN[14]:=LOW;

PIN[15]:=LOW;PIN[15]:=HIGH;
IF PIN[13]<>HIGH THEN ERROR(1);
FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[9]&PIN[10]&PIN[11])<>I THEN ERROR(1);
   PIN[12]:=LOW;PIN[12]:=HIGH;
   END;

PIN[14]:=HIGH;PIN[14]:=LOW;
IF (PIN[9]&PIN[10]&PIN[11]&PIN[13])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74393,SN74393

#TEXT
Dual 4-Bit Binary
Counter

This device contains
two completely indepen-
dent binary counters
with a preset input.
```

```
#PIN 14
 1 : Clock      Counter 1
 2 : Reset      Counter 1
 3 : Output  QA Counter 1
 4 : Output  QB Counter 1
 5 : Output  QC Counter 1
 6 : Output  QD Counter 1
 7 : GND
 8 : Output  QD Counter 2
 9 : Output  QC Counter 2
10 : Output  QB Counter 2
11 : Output  QA Counter 2
12 : Reset      Counter 2
13 : Clock      Counter 2
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,12,13] : INPUT;
PIN[3,4,5,6,8,9,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[2,12]:=LOW;
PIN[1,13]:=HIGH;

PIN[2,12]:=HIGH;
PIN[2,12]:=LOW;

FOR I:=0 TO 7 DO
   BEGIN
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>I THEN ERROR(1);
   IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>I THEN ERROR(1);
   PIN[1,13]:=LOW;PIN[1,13]:=HIGH;
   END;

PIN[2,12]:=HIGH;
PIN[2,12]:=LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>0 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74395,SN74395

#TEXT
4-Bit Shift Register with
Parallel or Serial Input
and Output, Clear,
Cascadable (TRI-STATE)

This device contains
a 4-bit right-shift
register with parallel
or serial input and out-
put and clear input.

#PIN 16
 1 : -Clear
 2 : Serial Input
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
```

```
 7 : Load/-Shift
 8 : GND
 9 : Out.-Enable -OE
10 : Clock
11 : Output  Q3'
12 : Output  Q3
13 : Output  Q2
14 : Output  Q1
15 : Output  Q0
16 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7]:=LOW;
PIN[1,9,10]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2]:=X AND 1;
      PIN[10]:=LOW;PIN[10]:=HIGH;
      X:=X SHR 1;
      END;
   PIN[9]:=LOW;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>D THEN ERROR(1);
   PIN[9]:=HIGH;
   IF PIN[11]<>(D AND 1) THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
PIN[9]:=LOW;
IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>0 THEN ERROR(1);
PIN[9]:=HIGH;
IF PIN[11]<>LOW THEN ERROR(0);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7]:=HIGH;
   PIN[3]&PIN[4]&PIN[5]&PIN[6]:=D;
   PIN[10]:=LOW;PIN[10]:=HIGH;
   PIN[7]:=LOW;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[9]:=LOW;
      IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>X THEN ERROR(1);
      PIN[9]:=HIGH;
      IF PIN[11]<>(X AND 1) THEN ERROR(1);
      PIN[2]:=LOW;
      PIN[10]:=LOW;PIN[10]:=HIGH;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;
```

```
LOADMODEON;
PIN[15,14,13,12] : LOAD LOW;
IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>%0000 THEN ERROR(1);
PIN[15,14,13,12] : LOAD HIGH;
IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74396,SN74396

#TEXT
8-Bit-D-Type Register
with 2x4-Bit Serial
Input and Parallel
Output

This device contains
two 4-bit memories
enabling 4-bit word
lengths to be converted
into 8-bit word lengths.

#PIN 16
 1 : Output  Q0 Word 2
 2 : Output  Q0 Word 1
 3 : Input   D0
 4 : Output  Q1 Word 2
 5 : Output  Q1 Word 1
 6 : Input   D1
 7 : Clock
 8 : GND
 9 : Input   D2
10 : Output  Q2 Word 1
11 : Output  Q2 Word 2
12 : Input   D3
13 : Output  Q3 Word 1
14 : Output  Q3 Word 2
15 : -Strobe
16 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[3,6,7,9,12,15] : INPUT;
PIN[1,2,4,5,10,11,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7]:=LOW;
PIN[15]:=HIGH;

D:=%01011010;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[12]&PIN[9]&PIN[6]&PIN[3]:=D SHR 4;
   PIN[7]:=HIGH;PIN[7]:=LOW;
   PIN[12]&PIN[9]&PIN[6]&PIN[3]:=D AND %1111;
   PIN[7]:=HIGH;PIN[7]:=LOW;
   PIN[15]:=LOW;
   IF (PIN[14]&PIN[11]&PIN[4]&PIN[1]&
       PIN[13]&PIN[10]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   IF (PIN[14]&PIN[11]&PIN[4]&PIN[1]&
       PIN[13]&PIN[10]&PIN[5]&PIN[2])<>%00000000 THEN ERROR(1);
```

```
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74398,SN74398

#TEXT
Quad 2-Input Multi-
plexer with Storage

With this device one of
two 4-bit data sources
can be selected and the
selected data can be
stored.

#PIN 20
 1 : Word Select
 2 : Output   QA
 3 : Output  -QA
 4 : Input    A1
 5 : Input    A2
 6 : Input    B2
 7 : Input    B1
 8 : Output  -QB
 9 : Output   QB
10 : GND
11 : Clock
12 : Output   QC
13 : Output  -QC
14 : Input    C1
15 : Input    C2
16 : Input    D2
17 : Input    D1
18 : Output  -QD
19 : Output   QD
20 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[1,4,5,6,7,11,14,15,16,17] : INPUT;
PIN[2,3,8,9,12,13,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[11]:=LOW;

PIN[4,5,6,7,14,15,16,17]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[17]&PIN[14]&PIN[7]&PIN[4]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[17]&PIN[14]&PIN[7]&PIN[4]:=0;
   IF (PIN[19]&PIN[12]&PIN[9]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[18])&NOT(PIN[13])&NOT(PIN[8])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[4,5,6,7,14,15,16,17]:=LOW;
PIN[1]:=HIGH;
D:=%0101;
```

```
FOR I:=0 TO 1 DO
   BEGIN
   PIN[16]&PIN[15]&PIN[6]&PIN[5]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[16]&PIN[15]&PIN[6]&PIN[5]:=0;
   IF (PIN[19]&PIN[12]&PIN[9]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[18])&NOT(PIN[13])&NOT(PIN[8])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74399,SN74399

#TEXT
Quad 2-Input Multiplexer
with Storage

This device enables one
of two 4-bit data sources
to be selected and stored.


#PIN 16
 1 : Word Select
 2 : Output   QA
 3 : Input    A1
 4 : Input    A2
 5 : Input    B2
 6 : Input    B1
 7 : Output   QB
 8 : GND
 9 : Clock
10 : Output   QC
11 : Input    C1
12 : Input    C2
13 : Input    D2
14 : Input    D1
15 : Output   QD
16 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[1,3,4,5,6,9,11,12,13,14] : INPUT;
PIN[2,7,10,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=LOW;

PIN[3,4,5,6,11,12,13,14]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[11]&PIN[6]&PIN[3]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[14]&PIN[11]&PIN[6]&PIN[3]:=0;
   IF (PIN[15]&PIN[10]&PIN[7]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[3,4,5,6,11,12,13,14]:=LOW;
PIN[1]:=HIGH;
```

```
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=0;
   IF (PIN[15]&PIN[10]&PIN[7]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74422,SN74422

#TEXT
Retriggerable Monostable
Multivibrator with Clear

This device contains a
retriggerable one-shot
with complementary out-
puts and clear input.

#PIN 14
 1 : Input   A1
 2 : Input   A2
 3 : Input   B1
 4 : Input   B2
 5 : -Clear
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : R int.
10 : N.C.
11 : C
12 : N.C.
13 : C/R
14 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[14,13,11] : RC;

PIN[1,2,3,4]:=LOW;
PIN[5]:=LOW;
PIN[5]:=HIGH;
IF PIN[8]<>LOW THEN ERROR(1);
C:=256;
PIN[3,4]:=HIGH;
IF PIN[8]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[8]<>LOW) DO
   C:=C-1;
IF PIN[8]<>LOW THEN ERROR(1);

PIN[1,2,3,4]:=HIGH;
PIN[5]:=LOW;
PIN[5]:=HIGH;
IF PIN[6]<>HIGH THEN ERROR(1);
```

```
C:=256;
PIN[1]:=LOW;
IF PIN[6]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[6]<>HIGH) DO
   C:=C-1;
IF PIN[6]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 74423,SN74423

#TEXT
Dual Retriggerable
Monostable Multivibrator
with Clear

This device contains
two retriggerable one-
shots with complementary
outputs and clear input.

#PIN 16
 1 : Input    A MF 1
 2 : Input    B MF 1
 3 : -Clear     MF 1
 4 : Output  -Q MF 1
 5 : Output   Q MF 2
 6 : C ext.     MF 2
 7 : C/R        MF 2
 8 : GND
 9 : Input    A MF 2
10 : Input    B MF 2
11 : -Clear     MF 2
12 : Output  -Q MF 2
13 : Output   Q MF 1
14 : C ext.     MF 1
15 : C/R        MF 1
16 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,9,10,11] : INPUT;
PIN[4,5,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[16,15,14] : RC;

PIN[1,2]:=LOW;
PIN[3]:=LOW;
PIN[3]:=HIGH;
IF PIN[13]<>LOW THEN ERROR(1);
C:=256;
PIN[2]:=HIGH;
IF PIN[13]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[13]<>LOW) DO
   C:=C-1;
IF PIN[13]<>LOW THEN ERROR(1);

PIN[1,2]:=HIGH;
PIN[3]:=LOW;
PIN[3]:=HIGH;
IF PIN[4]<>HIGH THEN ERROR(1);
```

```
C:=256;
PIN[1]:=LOW;
IF PIN[4]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[4]<>HIGH) DO
   C:=C-1;
IF PIN[4]<>HIGH THEN ERROR(1);

PIN[16,15,14] : RC OFF;
PIN[16,7,6] : RC;

PIN[9,10]:=LOW;
PIN[11]:=LOW;
PIN[11]:=HIGH;
IF PIN[5]<>LOW THEN ERROR(1);
C:=256;
PIN[10]:=HIGH;
IF PIN[5]<>HIGH THEN ERROR(1);
WHILE (C>0) AND (PIN[5]<>LOW) DO
   C:=C-1;
IF PIN[5]<>LOW THEN ERROR(1);

PIN[9,10]:=HIGH;
PIN[11]:=LOW;
PIN[11]:=HIGH;
IF PIN[12]<>HIGH THEN ERROR(1);
C:=256;
PIN[9]:=LOW;
IF PIN[12]<>LOW THEN ERROR(1);
WHILE (C>0) AND (PIN[12]<>HIGH) DO
   C:=C-1;
IF PIN[12]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 74425,SN74425

#TEXT
Quad TRI-STATE Buffer

This device contains
four independent non-
inverting buffers with
three-state outputs.

#PIN 14
 1 : -Control  Gate   1
 2 : Input     Gate   1
 3 : Output    Gate   1
 4 : -Control  Gate   2
 5 : Input     Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input     Gate   3
10 : -Control  Gate   3
11 : Output    Gate   4
12 : Input     Gate   4
13 : -Control  Gate   4
14 : +5V

#FAMILY Std

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
```

```
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,10,13]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[5]&PIN[9]&PIN[12]:=D;
   PIN[1,4,10,13]:=LOW;
   IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>D THEN ERROR(1);
   PIN[1,4,10,13]:=HIGH;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[3,6,8,11] : LOAD LOW;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,6,8,11] : LOAD HIGH;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74426,SN74426

#TEXT
Quad TRI-STATE Buffer

This device contains
four separate non-
inverting buffers with
three-state outputs.

#PIN 14
 1 : Control   Gate   1
 2 : Input     Gate   1
 3 : Output    Gate   1
 4 : Control   Gate   2
 5 : Input     Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input     Gate   3
10 : Control   Gate   3
11 : Output    Gate   4
12 : Input     Gate   4
13 : Control   Gate   4
14 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,10,13]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[5]&PIN[9]&PIN[12]:=D;
   PIN[1,4,10,13]:=HIGH;
```

```
   IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>D THEN ERROR(1);
   PIN[1,4,10,13]:=LOW;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[3,6,8,11] : LOAD LOW;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,6,8,11] : LOAD HIGH;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```

```
#NAME 74440,SN74440

#TEXT
Quad Tridirectional
Transceiver (O.C.)

With this device data
can be transferred
between three 4-bit
data buses. The data is
not inverted.

#PIN 20
 1 : -CS
 2 : In-/Output  B1
 3 : In-/Output  C1
 4 : In-/Output  C2
 5 : In-/Output  B2
 6 : In-/Output  B3
 7 : In-/Output  C3
 8 : In-/Output  C4
 9 : In-/Output  B4
10 : GND
11 : S0
12 : S1
13 : In-/Output  A4
14 : In-/Output  A3
15 : In-/Output  A2
16 : In-/Output  A1
17 : -GA
18 : -GB
19 : -GC
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,11,12,17,18,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[12]&PIN[11]:=0;
PIN[13,14,15,16] : INPUT;
PIN[2,3,4,5,6,7,8,9] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[14]&PIN[15]&PIN[16]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   LOADMODEON;
   PIN[9,6,5,2] : LOAD HIGH;
   IF (PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%011;
   LOADMODEON;
   PIN[8,7,4,3] : LOAD HIGH;
   IF (PIN[8]&PIN[7]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=1;
PIN[9,6,5,2] : INPUT;
PIN[3,4,7,8,13,14,15,16] : OUTPUT;
```

```
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[6]&PIN[5]&PIN[2]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   LOADMODEON;
   PIN[13,14,15,16] : LOAD HIGH;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%011;
   LOADMODEON;
   PIN[8,7,4,3] : LOAD HIGH;
   IF (PIN[8]&PIN[7]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=2;
PIN[3,4,7,8] : INPUT;
PIN[2,5,6,9,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   LOADMODEON;
   PIN[13,14,15,16] : LOAD HIGH;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   LOADMODEON;
   PIN[9,6,5,2] : LOAD HIGH;
   IF (PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74441,SN74441

#TEXT
Inverting Quad Tri-
directional Transceiver
(OPEN COLLECTOR)

With this device data
can be transferred
between three 4-bit
data buses. The data are
inverted.

#PIN 20
 1 : -CS
 2 : In-/Output  B1
 3 : In-/Output  C1
 4 : In-/Output  C2
 5 : In-/Output  B2
 6 : In-/Output  B3
 7 : In-/Output  C3
 8 : In-/Output  C4
 9 : In-/Output  B4
10 : GND
11 : S0
12 : S1
13 : In-/Output  A4
```

```
14 : In-/Output  A3
15 : In-/Output  A2
16 : In-/Output  A1
17 : -GA
18 : -GB
19 : -GC
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,11,12,17,18,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[12]&PIN[11]:=0;
PIN[13,14,15,16] : INPUT;
PIN[2,3,4,5,6,7,8,9] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[14]&PIN[15]&PIN[16]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   LOADMODEON;
   PIN[9,6,5,2] : LOAD HIGH;
   IF (NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%011;
   LOADMODEON;
   PIN[8,7,4,3] : LOAD HIGH;
   IF (NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=1;
PIN[9,6,5,2] : INPUT;
PIN[3,4,7,8,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[6]&PIN[5]&PIN[2]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   LOADMODEON;
   PIN[13,14,15,16] : LOAD HIGH;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%011;
   LOADMODEON;
   PIN[8,7,4,3] : LOAD HIGH;
   IF (NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=2;
PIN[3,4,7,8] : INPUT;
PIN[2,5,6,9,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   LOADMODEON;
```

```
   PIN[13,14,15,16] : LOAD HIGH;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   LOADMODEON;
   PIN[9,6,5,2] : LOAD HIGH;
   IF (NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74442,SN74442

#TEXT
TRI-STATE Quad Tri-
directional Transceiver

With this device data
can be transferred
between three 4-bit data
buses. The data are not
inverted.

#PIN 20
 1 : -CS
 2 : In-/Output  B1
 3 : In-/Output  C1
 4 : In-/Output  C2
 5 : In-/Output  B2
 6 : In-/Output  B3
 7 : In-/Output  C3
 8 : In-/Output  C4
 9 : In-/Output  B4
10 : GND
11 : S0
12 : S1
13 : In-/Output  A4
14 : In-/Output  A3
15 : In-/Output  A2
16 : In-/Output  A1
17 : -GA
18 : -GB
19 : -GC
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,11,12,17,18,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[12]&PIN[11]:=0;
PIN[13,14,15,16] : INPUT;
PIN[2,3,4,5,6,7,8,9] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[14]&PIN[15]&PIN[16]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   IF (PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
```

```
   PIN[19]&PIN[18]&PIN[17]:=%011;
   IF (PIN[8]&PIN[7]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=1;
PIN[9,6,5,2] : INPUT;
PIN[3,4,7,8,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[6]&PIN[5]&PIN[2]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%011;
   IF (PIN[8]&PIN[7]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=2;
PIN[3,4,7,8] : INPUT;
PIN[2,5,6,9,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%101;
   IF (PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=3;
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : OUTPUT;
LOADMODEON;
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : LOAD LOW;
IF (PIN[16]&PIN[15]&PIN[14]&PIN[13]&PIN[9]&PIN[8]&
    PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%000000000000 THEN ERROR(1);
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : LOAD HIGH;
IF (PIN[16]&PIN[15]&PIN[14]&PIN[13]&PIN[9]&PIN[8]&
    PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%111111111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74443,SN74443

#TEXT
Inverting Quad Tri-
directional Transceiver
(TRI-STATE)

Using this device data
can be transferred
between three 4-bit
data buses. The data
are inverted.

#PIN 20
 1 : -CS
 2 : In-/Output  B1
 3 : In-/Output  C1
 4 : In-/Output  C2
 5 : In-/Output  B2
 6 : In-/Output  B3
```

```
 7 : In-/Output  C3
 8 : In-/Output  C4
 9 : In-/Output  B4
10 : GND
11 : S0
12 : S1
13 : In-/Output  A4
14 : In-/Output  A3
15 : In-/Output  A2
16 : In-/Output  A1
17 : -GA
18 : -GB
19 : -GC
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,11,12,17,18,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[12]&PIN[11]:=0;
PIN[13,14,15,16] : INPUT;
PIN[2,3,4,5,6,7,8,9] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[14]&PIN[15]&PIN[16]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   IF (NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%011;
   IF (NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=1;
PIN[9,6,5,2] : INPUT;
PIN[3,4,7,8,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[6]&PIN[5]&PIN[2]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%011;
   IF (NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=2;
PIN[3,4,7,8] : INPUT;
PIN[2,5,6,9,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%101;
   IF (NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
```

```
PIN[12]&PIN[11]:=3;
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : OUTPUT;
LOADMODEON;
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : LOAD LOW;
IF (PIN[16]&PIN[15]&PIN[14]&PIN[13]&PIN[9]&PIN[8]&
    PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%000000000000 THEN ERROR(1);
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : LOAD HIGH;
IF (PIN[16]&PIN[15]&PIN[14]&PIN[13]&PIN[9]&PIN[8]&
    PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%111111111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74444,SN74444

#TEXT
TRI-STATE Quad Tri-
directional Transceiver

With this device data
can be transferred
between three 4-bit
data buses. The data
are partly inverted
and partly non-inverted.

#PIN 20
 1 : -CS
 2 : In-/Output  B1
 3 : In-/Output  C1
 4 : In-/Output  C2
 5 : In-/Output  B2
 6 : In-/Output  B3
 7 : In-/Output  C3
 8 : In-/Output  C4
 9 : In-/Output  B4
10 : GND
11 : S0
12 : S1
13 : In-/Output  A4
14 : In-/Output  A3
15 : In-/Output  A2
16 : In-/Output  A1
17 : -GA
18 : -GB
19 : -GC
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,11,12,17,18,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[12]&PIN[11]:=0;
PIN[13,14,15,16] : INPUT;
PIN[2,3,4,5,6,7,8,9] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[14]&PIN[15]&PIN[16]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%101;
```

```
   IF (NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%011;
   IF (NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=1;
PIN[9,6,5,2] : INPUT;
PIN[3,4,7,8,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[6]&PIN[5]&PIN[2]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%011;
   IF (PIN[8]&PIN[7]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=2;
PIN[3,4,7,8] : INPUT;
PIN[2,5,6,9,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
   PIN[19]&PIN[18]&PIN[17]:=%101;
   IF (PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=3;
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : OUTPUT;
LOADMODEON;
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : LOAD LOW;
IF (PIN[16]&PIN[15]&PIN[14]&PIN[13]&PIN[9]&PIN[8]&
    PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%000000000000 THEN ERROR(1);
PIN[2,3,4,5,6,7,8,9,13,14,15,16] : LOAD HIGH;
IF (PIN[16]&PIN[15]&PIN[14]&PIN[13]&PIN[9]&PIN[8]&
    PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%111111111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74445,SN74445

#TEXT
BCD-to-Decimal Decoder/
Driver (O.C., 7V)

This device converts a
standard BCD code with
4-bits into a decimal
number from 0 to 9.
The outputs are OPEN-
COLLECTOR with a maximum
output voltage of 7V.

#PIN 16
 1 : Output  0
 2 : Output  1
 3 : Output  2
 4 : Output  3
```

```
 5 : Output  4
 6 : Output  5
 7 : Output  6
 8 : GND
 9 : Output  7
10 : Output  8
11 : Output  9
12 : Input   D
13 : Input   C
14 : Input   B
15 : Input   A
16 : +5V...+7V

#FAMILY LS

#PROGRAM

BEGIN
PIN[12,13,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9,10,11] : Output ;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]:=I;
   LOADMODEON;
   PIN[1,2,3,4,5,6,7,9,10,11] : LOAD LOW;
   IF (PIN[11]&PIN[10]&PIN[9]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1])
       <>0 THEN ERROR(1);
   PIN[11,10,9,7,6,5,4,3,2,1] : LOAD HIGH;
   IF (NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2])&NOT(PIN[1]))
       <>(1 SHL I) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74446,SN74446

#TEXT
Quad TRI-STATE
Transceiver with
Individual Control
Inputs.

This device contains
four inverting
bidirectional buffers
with three-state out-
puts.

#PIN 16
 1 : -GBA
 2 : In-/Output  A1
 3 : DIR 2
 4 : In-/Output  A2
 5 : In-/Output  A3
 6 : DIR 3
 7 : In-/Output  A4
 8 : GND
 9 : In-/Output  B4
10 : DIR 4
11 : In-/Output  B3
```

```
12 : In-/Output  B2
13 : DIR 1
14 : In-/Output  B1
15 : -GAB
16 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,3,6,10,13,15] : INPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%0101;
PIN[3,6,10,13]:=HIGH;
PIN[2,4,5,7] : INPUT;
PIN[9,11,12,14] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7]&PIN[5]&PIN[4]&PIN[2]:=D;
   PIN[1,15]:=LOW;
   IF (NOT(PIN[9])&NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[14]))<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %1111;
   END;

D:=%0101;
PIN[3,6,10,13]:=LOW;
PIN[9,11,12,14] : INPUT;
PIN[2,4,5,7] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[11]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (NOT(PIN[7])&NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[2,4,5,7,9,11,12,14] : OUTPUT;
LOADMODEON;
PIN[2,4,5,7,9,11,12,14] : LOAD LOW;
IF (PIN[7]&PIN[5]&PIN[4]&PIN[2])<>%0000 THEN ERROR(1);
IF (PIN[9]&PIN[11]&PIN[12]&PIN[14])<>%0000 THEN ERROR(1);
PIN[2,4,5,7,9,11,12,14] : LOAD HIGH;
IF (PIN[7]&PIN[5]&PIN[4]&PIN[2])<>%1111 THEN ERROR(1);
IF (PIN[9]&PIN[11]&PIN[12]&PIN[14])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74447,SN74447

#TEXT
BCD-to-7-Segment Decoder/
Driver (O.C., 7V)

This device converts BCD
input data into control
data for 7-segment
displays. The outputs
are OPEN-COLLECTOR with
a maximum output voltage
```

of 7V.

```
#PIN 16
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : -LT (Lamp Test)
 4 : -BI/-RBO
 5 : -RBI
 6 : BCD-Input   D
 7 : BCD-Input   A
 8 : GND
 9 : 7-Seg.-Output  -e
10 : 7-Seg.-Output  -d
11 : 7-Seg.-Output  -c
12 : 7-Seg.-Output  -b
13 : 7-Seg.-Output  -a
14 : 7-Seg.-Output  -g
15 : 7-Seg.-Output  -f
16 : +5V...+7V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   LOADMODEON;
   PIN[9,10,11,12,13,14,15] : LOAD LOW;
   IF (PIN[14]&PIN[15]&PIN[9]&PIN[10]&
       PIN[11]&PIN[12]&PIN[13])<>0 THEN ERROR(1);
   PIN[9,10,11,12,13,14,15] : LOAD HIGH;
   D:=NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[9])&NOT(PIN[10])&
      NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13]);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 74448,SN74448

#TEXT
Quad Tridirectional
Transceiver, inverting/
non-inverting (O.C.)

With this device data
can be transferred
between three 4-bit data
```

buses. The data are
partly inverted and
partly non-inverted.

```
#PIN 20
 1 : -CS
 2 : In-/Output  B1
 3 : In-/Output  C1
 4 : In-/Output  C2
 5 : In-/Output  B2
 6 : In-/Output  B3
 7 : In-/Output  C3
 8 : In-/Output  C4
 9 : In-/Output  B4
10 : GND
11 : S0
12 : S1
13 : In-/Output  A4
14 : In-/Output  A3
15 : In-/Output  A2
16 : In-/Output  A1
17 : -GA
18 : -GB
19 : -GC
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,11,12,17,18,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[12]&PIN[11]:=0;
PIN[13,14,15,16] : INPUT;
PIN[2,3,4,5,6,7,8,9] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[14]&PIN[15]&PIN[16]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   LOADMODEON;
   PIN[9,6,5,2] : LOAD HIGH;
   IF (NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%011;
   LOADMODEON;
   PIN[8,7,4,3] : LOAD HIGH;
   IF (NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=1;
PIN[9,6,5,2] : INPUT;
PIN[3,4,7,8,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[6]&PIN[5]&PIN[2]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   LOADMODEON;
   PIN[13,14,15,16] : LOAD HIGH;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
```

```
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%011;
   LOADMODEON;
   PIN[8,7,4,3] : LOAD HIGH;
   IF (PIN[8]&PIN[7]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

PIN[12]&PIN[11]:=2;
PIN[3,4,7,8] : INPUT;
PIN[2,5,6,9,13,14,15,16] : OUTPUT;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[19]&PIN[18]&PIN[17]:=%110;
   LOADMODEON;
   PIN[13,14,15,16] : LOAD HIGH;
   IF (NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[16]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]&PIN[18]&PIN[17]:=%101;
   LOADMODEON;
   PIN[9,6,5,2] : LOAD HIGH;
   IF (PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74449,SN74449

#TEXT
Quad TRI-STATE
Transceiver with
Individual Control
Inputs

This device contains
four non-inverting
bidirectional buffers
with three-state
outputs.

#PIN 16
 1 : -GBA
 2 : In-/Output  A1
 3 : DIR 2
 4 : In-/Output  A2
 5 : In-/Output  A3
 6 : DIR 3
 7 : In-/Output  A4
 8 : GND
 9 : In-/Output  B4
10 : DIR 4
11 : In-/Output  B3
12 : In-/Output  B2
13 : DIR 1
14 : In-/Output  B1
15 : -GAB
16 : +5V

#FAMILY LS

#PROGRAM
```

```
BEGIN
PIN[1,3,6,10,13,15] : INPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%0101;
PIN[3,6,10,13]:=HIGH;
PIN[2,4,5,7] : INPUT;
PIN[9,11,12,14] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[7]&PIN[5]&PIN[4]&PIN[2]:=D;
   PIN[1,15]:=LOW;
   IF (PIN[9]&PIN[11]&PIN[12]&PIN[14])<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %1111;
   END;

D:=%0101;
PIN[3,6,10,13]:=LOW;
PIN[9,11,12,14] : INPUT;
PIN[2,4,5,7] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[11]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (PIN[7]&PIN[5]&PIN[4]&PIN[2])<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[2,4,5,7,9,11,12,14] : OUTPUT;
LOADMODEON;
PIN[2,4,5,7,9,11,12,14] : LOAD LOW;
IF (PIN[7]&PIN[5]&PIN[4]&PIN[2])<>%0000 THEN ERROR(1);
IF (PIN[9]&PIN[11]&PIN[12]&PIN[14])<>%0000 THEN ERROR(1);
PIN[2,4,5,7,9,11,12,14] : LOAD HIGH;
IF (PIN[7]&PIN[5]&PIN[4]&PIN[2])<>%1111 THEN ERROR(1);
IF (PIN[9]&PIN[11]&PIN[12]&PIN[14])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74465,SN74465

#TEXT
TRI-STATE Octal Non-
Inverting Buffer

This device contains
eight non-inverting
buffers with common
enable and three-state
outputs.

#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
 5 : Output    Buffer 2
 6 : Input     Buffer 3
 7 : Output    Buffer 3
 8 : Input     Buffer 4
```

```
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
       PIN[11]&PIN[13]&PIN[15]&PIN[17])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74466,SN74466

#TEXT
Inverting Octal TRI-
STATE Buffer with
Common Enable

This device contains
eight inverting buffers
with common enable and
three-state outputs.

#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
 5 : Output    Buffer 2
 6 : Input     Buffer 3
 7 : Output    Buffer 3
```

```
 8 : Input     Buffer 4
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[3])&NOT(PIN[5])&NOT(PIN[7])&NOT(PIN[9])&
       NOT(PIN[11])&NOT(PIN[13])&NOT(PIN[15])&NOT(PIN[17]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74467,SN74467

#TEXT
Non-Inverting Octal TRI-
STATE Buffer with two
Enable Inputs

This device contains
eight non-inverting
buffers with two enable
inputs and three-state
outputs.

#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
 5 : Output    Buffer 2
```

```
 6 : Input     Buffer 3
 7 : Output    Buffer 3
 8 : Input     Buffer 4
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
       PIN[11]&PIN[13]&PIN[15]&PIN[17])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74468,SN74468

#TEXT
Inverting Octal TRI-
STATE Buffer

This device contains
eight inverting buffers
with two enable inputs
and three-state out-
puts.

#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
```

```
 5 : Output    Buffer 2
 6 : Input     Buffer 3
 7 : Output    Buffer 3
 8 : Input     Buffer 4
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[3])&NOT(PIN[5])&NOT(PIN[7])&NOT(PIN[9])&
       NOT(PIN[11])&NOT(PIN[13])&NOT(PIN[15])&NOT(PIN[17]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

##NAME 74484,SN74484

#TEXT
BCD-to-Binary Decoder
(TRI-STATE)

This device contains
a decoder which converts
a 9-bit BCD code into a
binary code.
The outputs are three-
state.

#PIN 20
 1 : Input   E
 2 : Input   D
```

```
 3 : Input   C
 4 : Input   B
 5 : Input   A
 6 : Output  Q1
 7 : Output  Q2
 8 : Output  Q3
 9 : Output  Q4
10 : GND
11 : Output  Q5
12 : Output  Q6
13 : Output  Q7
14 : Output  Q8
15 : Enable -E1
16 : Enable -E2
17 : Input   H
18 : Input   G
19 : Input   F
20 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,15,16,17,18,19] : INPUT;
PIN[6,7,8,9,11,12,13,14] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[15,16]:=HIGH;

FOR I:=0 TO 199 BY 2 DO
   BEGIN
   X:=((I DIV 100) SHL 8)+((I DIV 10) SHL 4)+(I MOD 10);
   PIN[17]&PIN[18]&PIN[19]&PIN[1]&PIN[2]&
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=X SHR 1;
   PIN[15,16]:=LOW;
   IF I<>(PIN[14]&PIN[13]&PIN[12]&PIN[11]&PIN[9]&
          PIN[8]&PIN[7]&PIN[6]&(X AND 1)) THEN ERROR(1);
   PIN[15,16]:=HIGH;
   END;

LOADMODEON;
PIN[14,13,12,11,9,8,7,6] : LOAD LOW;
IF (PIN[14]&PIN[13]&PIN[12]&PIN[11]&
    PIN[9]&PIN[8]&PIN[7]&PIN[6])<>%00000000 THEN ERROR(1);
PIN[14,13,12,11,9,8,7,6] : LOAD HIGH;
IF (PIN[14]&PIN[13]&PIN[12]&PIN[11]&
    PIN[9]&PIN[8]&PIN[7]&PIN[6])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

##NAME 74485,SN74485

#TEXT
Binary-to-BCD Decoder
(TRI-STATE)

This device contains a
decoder which converts a
9-bit binary code into a
BCD code.

The outputs are three-
state.
```

```
#PIN 20
 1 : Input   E
 2 : Input   D
 3 : Input   C
 4 : Input   B
 5 : Input   A
 6 : Output  Q1
 7 : Output  Q2
 8 : Output  Q3
 9 : Output  Q4
10 : GND
11 : Output  Q5
12 : Output  Q6
13 : Output  Q7
14 : Output  Q8
15 : Enable -E1
16 : Enable -E2
17 : Input   H
18 : Input   G
19 : Input   F
20 : +5V

#FAMILY S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,15,16,17,18,19] : INPUT;
PIN[6,7,8,9,11,12,13,14] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[15,16]:=HIGH;

FOR I:=0 TO 319 BY 2 DO
   BEGIN
   X:=((I DIV 100) SHL 8)+((I DIV 10) SHL 4)+(I MOD 10);
   PIN[17]&PIN[18]&PIN[19]&PIN[1]&PIN[2]&
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I SHR 1;
   PIN[15,16]:=LOW;
   IF X<>(PIN[14]&PIN[13]&PIN[12]&PIN[11]&PIN[9]&
          PIN[8]&PIN[7]&PIN[6]&(I AND 1)) THEN ERROR(1);
   PIN[15,16]:=HIGH;
   END;

LOADMODEON;
PIN[14,13,12,11,9,8,7,6] : LOAD LOW;
IF (PIN[14]&PIN[13]&PIN[12]&PIN[11]&
    PIN[9]&PIN[8]&PIN[7]&PIN[6])<>%00000000 THEN ERROR(1);
PIN[14,13,12,11,9,8,7,6] : LOAD HIGH;
IF (PIN[14]&PIN[13]&PIN[12]&PIN[11]&
    PIN[9]&PIN[8]&PIN[7]&PIN[6])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74490,SN74490

#TEXT
Dual Decade Counter

This device contains a
divide-by-2 counter and
a divide-by-5 counter.

#PIN 16
 1 : Clock      Counter 1
```

```
 2 : Clear      Counter 1
 3 : Output  QA Counter 1
 4 : Set 9      Counter 1
 5 : Output  QB Counter 1
 6 : Output  QC Counter 1
 7 : Output  QD Counter 1
 8 : GND
 9 : Output  QD Counter 2
10 : Output  QC Counter 2
11 : Output  QB Counter 2
12 : Set 9      Counter 2
13 : Output  QA Counter 2
14 : Clear      Counter 2
15 : Clock      Counter 2
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,4,12,14,15] : INPUT;
PIN[3,5,6,7,9,10,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,4,12,14]:=LOW;
PIN[1,15]:=HIGH;

PIN[4,12]:=HIGH;
PIN[4,12]:=LOW;
IF (PIN[7]&PIN[6]&PIN[5]&PIN[3])<>9 THEN ERROR(1);
IF (PIN[9]&PIN[10]&PIN[11]&PIN[13])<>9 THEN ERROR(1);

PIN[2,14]:=HIGH;
PIN[2,14]:=LOW;
FOR I:=0 TO 9 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[5]&PIN[3])<>I THEN ERROR(1);
   IF (PIN[9]&PIN[10]&PIN[11]&PIN[13])<>I THEN ERROR(1);
   PIN[1,15]:=LOW;PIN[1,15]:=HIGH;
   END;

ERROR(0);
END.

#NAME 74518,SN74518

#TEXT
8-Bit Comparator,
(non-inverted output,
O.C.)

This device compares two
8-bit words, A and B and
indicates whether they
are equal. The output is
OPEN COLLECTOR.

#PIN 20
 1 : -EN
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
```

```
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : A=B
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
LOADMODEON;
PIN[19] : LOAD HIGH;
IF PIN[19]<>HIGH THEN ERROR(1);
LOADMODEOFF;

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74519,SN74519

#TEXT
8-Bit Comparator (non-
inverted output, O.C.)
```

```
This device compares two
8-bit words, A and B, and
indicates whether they
are equal. The output is
OPEN COLLECTOR.

#PIN 20
 1 : -EN
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : A=B
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
LOADMODEON;
PIN[19] : LOAD HIGH;
IF PIN[19]<>HIGH THEN ERROR(1);
LOADMODEOFF;

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
```

```
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74520,SN74520

#TEXT
8-Bit Comparator (Inverted
Output, O.C.)

This device compares two
8-bit words, A and B,
and indicates whether they
are equal. The output
is OPEN COLLECTOR.

#PIN 20
 1 : -EN
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : -(A=B)
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
LOADMODEON;
PIN[19] : LOAD HIGH;
IF PIN[19]<>LOW THEN ERROR(1);
LOADMODEOFF;

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
```

```
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74521,SN74521

#TEXT
8-Bit Comparator (Inverted
Output, O.C.)

This device compares
two 8-bit words, A and B,
and indicates whether or
not they are equal. The
output is OPEN COLLECTOR.

#PIN 20
 1 : -EN
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : -(A=B)
20 : +5V

#FAMILY ALS,F

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
```

```
PIN[1]:=LOW;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
LOADMODEON;
PIN[19] : LOAD HIGH;
IF PIN[19]<>LOW THEN ERROR(1);
LOADMODEOFF;

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74522,SN74522

#TEXT
8-Bit Comparator (Inverted
Output, O.C.)

This device compares two
8-bit words, A and B,
and indicates whether or
not they are equal. The
output is OPEN COLLECTOR.

#PIN 20
 1 : -EN
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
```

```
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : -(A=B)
20 : +5V

#FAMILY ALS,F

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
LOADMODEON;
PIN[19] : LOAD HIGH;
IF PIN[19]<>LOW THEN ERROR(1);
LOADMODEOFF;

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74533,SN74533

#TEXT
TRI-STATE 8-Bit-D-
Latch with Enable

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
```

```
 2 : Output  -Q0
 3 : Input    D0
 4 : Input    D1
 5 : Output  -Q1
 6 : Output  -Q2
 7 : Input    D2
 8 : Input    D3
 9 : Output  -Q3
10 : GND
11 : Latch Enable
12 : Output  -Q4
13 : Input    D4
14 : Input    D5
15 : Output  -Q5
16 : Output  -Q6
17 : Input    D6
18 : Input    D7
19 : Output  -Q7
20 : +5V

#FAMILY ALS,AS,F,LS

#PROGRAM

BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
   PIN[1]:=LOW;
   IF (NOT(PIN[19])&NOT(PIN[16])&NOT(PIN[15])&NOT(PIN[12])&
       NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,16,15,12,9,6,5,2] : LOAD LOW;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
      PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[19,16,15,12,9,6,5,2] : LOAD HIGH;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
      PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```

```
#NAME 74534,SN74534

#TEXT
8-Bit-D-Register
(TRI-STATE)

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
 2 : Output -Q0
 3 : Input   D0
 4 : Input   D1
 5 : Output -Q1
 6 : Output -Q2
 7 : Input   D2
 8 : Input   D3
 9 : Output -Q3
10 : GND
11 : Clock
12 : Output -Q4
13 : Input   D4
14 : Input   D5
15 : Output -Q5
16 : Output -Q6
17 : Input   D6
18 : Input   D7
19 : Output -Q7
20 : +5V

#FAMILY ALS,AS,F,LS

#PROGRAM

BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
   PIN[1]:=LOW;
   IF (NOT(PIN[19])&NOT(PIN[16])&NOT(PIN[15])&NOT(PIN[12])&
       NOT(PIN[9])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,16,15,12,9,6,5,2] : LOAD LOW;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[19,16,15,12,9,6,5,2] : LOAD HIGH;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;
```

```
ERROR(0);
END.

##NAME 74538,SN74538

#TEXT
1-of-8-TRI-STATE Decoder

This device contains a
1-of-8 decoder (3-to-8)
with four enable inputs
and three-state
outputs.

#PIN 20
 1 : Output    Q2
 2 : Output    Q1
 3 : Output    Q0
 4 : -OE1
 5 : -OE2
 6 : Address   A0
 7 : Address   A1
 8 : Output    Q5
 9 : Output    Q6
10 : GND
11 : Output    Q7
12 : Polarity  AL
13 : Enable    G1
14 : Enable    G2
15 : Enable   -G3
16 : Enable   -G4
17 : Address   A2
18 : Output    Q4
19 : Output    Q3
20 : +5V

#FAMILY ALS,F

#PROGRAM

BEGIN
PIN[4,5,6,7,12,13,14,15,16,17] : INPUT;
PIN[1,2,3,8,9,11,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[15,16]:=LOW;
PIN[4,5,13,14]:=HIGH;


FOR I:=0 TO 7 DO
   BEGIN
   PIN[17]&PIN[7]&PIN[6]:=I;
   PIN[4,5]:=LOW;
   PIN[12]:=LOW;
   IF (PIN[11]&PIN[9]&PIN[8]&PIN[18]&
       PIN[19]&PIN[1]&PIN[2]&PIN[3])<>(1 SHL I) THEN ERROR(1);
   PIN[12]:=HIGH;
   IF (NOT(PIN[11])&NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[18])&
       NOT(PIN[19])&NOT(PIN[1])&NOT(PIN[2])&NOT(PIN[3]))<>(1 SHL I) THEN
ERROR(1);
   PIN[4,5]:=HIGH;
   END;

LOADMODEON;
PIN[11,9,8,18,19,1,2,3] : LOAD LOW;
IF (PIN[11]&PIN[9]&PIN[8]&PIN[18]&
    PIN[19]&PIN[1]&PIN[2]&PIN[3])<>%00000000 THEN ERROR(1);
```

```
PIN[11,9,8,18,19,1,2,3] : LOAD HIGH;
IF (PIN[11]&PIN[9]&PIN[8]&PIN[18]&
    PIN[19]&PIN[1]&PIN[2]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

##NAME 74539,SN74539

#TEXT
Dual 1-of-4 TRI-STATE
Decoder

This device contains
two 1-of-4 decoders
(2-to-4) with enable
input and three-
state outputs.

#PIN 20
 1 : Output   Q2 DMP 1
 2 : Output   Q1 DMP 1
 3 : Output   Q0 DMP 1
 4 : Polarity    DMP 1
 5 : -OE         DMP 1
 6 : Address  A0 DMP 2
 7 : Address  A1 DMP 2
 8 : Output   Q3 DMP 2
 9 : Output   Q2 DMP 2
10 : GND
11 : Output   Q1 DMP 2
12 : Output   Q0 DMP 2
13 : Polarity    DMP 2
14 : -OE         DMP 2
15 : Enable   -G DMP 2
16 : Enable   -G DMP 1
17 : Address  A0 DMP 1
18 : Address  A1 DMP 1
19 : Output   Q3 DMP 1
20 : +5V

#FAMILY ALS,F

#PROGRAM

BEGIN
PIN[4,5,6,7,13,14,15,16,17,18] : INPUT;
PIN[1,2,3,8,9,11,12,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[15,16]:=LOW;
PIN[5,14]:=HIGH;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[18]&PIN[17]:=I;
   PIN[7]&PIN[6]:=I;
   PIN[5,14]:=LOW;
   PIN[4,13]:=LOW;
   IF (PIN[19]&PIN[1]&PIN[2]&PIN[3])<>(1 SHL I) THEN ERROR(1);
   IF (PIN[8]&PIN[9]&PIN[11]&PIN[12])<>(1 SHL I) THEN ERROR(1);
   PIN[4,13]:=HIGH;
   IF (NOT(PIN[19])&NOT(PIN[1])&NOT(PIN[2])&NOT(PIN[3]))<>(1 SHL I) THEN
ERROR(1);
   IF (NOT(PIN[8])&NOT(PIN[9])&NOT(PIN[11])&NOT(PIN[12]))<>(1 SHL I) THEN
ERROR(1);
```

```
   PIN[5,14]:=HIGH;
   END;

LOADMODEON;
PIN[19,1,2,3,8,9,11,12] : LOAD LOW;
IF (PIN[19]&PIN[1]&PIN[2]&PIN[3])<>%0000 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[11]&PIN[12])<>%0000 THEN ERROR(1);
PIN[19,1,2,3,8,9,11,12] : LOAD HIGH;
IF (PIN[19]&PIN[1]&PIN[2]&PIN[3])<>%1111 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[11]&PIN[12])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74540,SN74540

#TEXT
Inverting Octal TRI-
STATE Buffer

This device contains
eight inverting buffers
with three-state out-
puts.

#PIN 20
 1 : -G1
 2 : Input   E0
 3 : Input   E1
 4 : Input   E2
 5 : Input   E3
 6 : Input   E4
 7 : Input   E5
 8 : Input   E6
 9 : Input   E7
10 : GND
11 : Output -Q7
12 : Output -Q6
13 : Output -Q5
14 : Output -Q4
15 : Output -Q3
16 : Output -Q2
17 : Output -Q1
18 : Output -Q0
19 : -G2
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,19] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&
   PIN[6]&PIN[7]&PIN[8]&PIN[9]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[18])&NOT(PIN[17])&NOT(PIN[16])&NOT(PIN[15])&
       NOT(PIN[14])&NOT(PIN[13])&NOT(PIN[12])&NOT(PIN[11]))<>D THEN ERROR(1);
```

```
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,17,16,15,14,13,12,11] : LOAD LOW;
IF (PIN[18]&PIN[17]&PIN[16]&PIN[15]&
    PIN[14]&PIN[13]&PIN[12]&PIN[11])<>%00000000 THEN ERROR(1);
PIN[18,17,16,15,14,13,12,11] : LOAD HIGH;
IF (PIN[18]&PIN[17]&PIN[16]&PIN[15]&
    PIN[14]&PIN[13]&PIN[12]&PIN[11])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74541,SN74541

#TEXT
Non-Inverting Octal
TRI-STATE Buffer

This device contains
eight non-inverting
buffers with three-
state outputs.

#PIN 20
 1 : -G1
 2 : Input  E0
 3 : Input  E1
 4 : Input  E2
 5 : Input  E3
 6 : Input  E4
 7 : Input  E5
 8 : Input  E6
 9 : Input  E7
10 : GND
11 : Output Q7
12 : Output Q6
13 : Output Q5
14 : Output Q4
15 : Output Q3
16 : Output Q2
17 : Output Q1
18 : Output Q0
19 : -G2
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,19] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&
   PIN[6]&PIN[7]&PIN[8]&PIN[9]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[18]&PIN[17]&PIN[16]&PIN[15]&
```

```
       PIN[14]&PIN[13]&PIN[12]&PIN[11])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,17,16,15,14,13,12,11] : LOAD LOW;
IF (PIN[18]&PIN[17]&PIN[16]&PIN[15]&
    PIN[14]&PIN[13]&PIN[12]&PIN[11])<>%00000000 THEN ERROR(1);
PIN[18,17,16,15,14,13,12,11] : LOAD HIGH;
IF (PIN[18]&PIN[17]&PIN[16]&PIN[15]&
    PIN[14]&PIN[13]&PIN[12]&PIN[11])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

##NAME 74560,SN74560

#TEXT
4-Bit TRI-STATE
Decade Counter

This device contains
a decade counter which
counts up in BCD code and
can be synchronously or
asynchronously loaded
and cleared.

The outputs are three-
state.

#PIN 20
 1 : -ALOAD
 2 : Clock
 3 : Input   A
 4 : Input   B
 5 : Input   C
 6 : Input   D
 7 : ENP
 8 : -ACLR
 9 : -SCLR
10 : GND
11 : -SLOAD
12 : ENT
13 : Output  QD
14 : Output  QC
15 : Output  QB
16 : Output  QA
17 : Enable  -G
18 : CCO
19 : RCO
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,17] : INPUT;
PIN[13,14,15,16,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[2,17]:=LOW;
PIN[1,7,8,9,11,12]:=HIGH;
```

```
PIN[8]:=LOW;PIN[8]:=HIGH;
FOR I:=0 TO 9 DO
   BEGIN
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[1]:=HIGH;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
PIN[8]:=LOW;PIN[8]:=HIGH;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[9]:=HIGH;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
PIN[9]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[9]:=HIGH;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);

ERROR(0);
END.

##NAME 74561,SN74561

#TEXT
4-Bit Binary Counter
with Synchronous/
Asynchronous Load and
Clear (TRI-STATE)

This binary counter
counts up and can be
synchronously or asyn-
chronously loaded or
cleared.

The outputs are three-
state.

#PIN 20
 1 : -ALOAD
 2 : Clock
 3 : Input   A
 4 : Input   B
 5 : Input   C
 6 : Input   D
 7 : ENP
 8 : -ACLR
 9 : -SCLR
10 : GND
11 : -SLOAD
```

```
12 : ENT
13 : Output  QD
14 : Output  QC
15 : Output  QB
16 : Output  QA
17 : Enable -G
18 : CCO
19 : RCO
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,17] : INPUT;
PIN[13,14,15,16,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[2,17]:=LOW;
PIN[1,7,8,9,11,12]:=HIGH;

PIN[8]:=LOW;PIN[8]:=HIGH;
FOR I:=0 TO 15 DO
   BEGIN
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[1]:=HIGH;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
PIN[8]:=LOW;PIN[8]:=HIGH;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[9]:=HIGH;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
PIN[9]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[9]:=HIGH;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 74563,SN74563

#TEXT
TRI-STATE 8-Bit-D-
Type Latch with Enable
(Inverted Outputs)
```

```
This device contains
eight inverting bistable
memory elements with
three-state outputs.

#PIN 20
 1 : -OE
 2 : Input   D0
 3 : Input   D1
 4 : Input   D2
 5 : Input   D3
 6 : Input   D4
 7 : Input   D5
 8 : Input   D6
 9 : Input   D7
10 : GND
11 : Latch Enable
12 : Output -Q7
13 : Output -Q6
14 : Output -Q5
15 : Output -Q4
16 : Output -Q3
17 : Output -Q2
18 : Output -Q1
19 : Output -Q0
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11] : INPUT;
PIN[12,13,14,15,16,17,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=0;
   PIN[1]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&
       NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18])&NOT(PIN[19]))<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[12,13,14,15,16,17,18,19] : LOAD LOW;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%00000000 THEN ERROR(1);
PIN[12,13,14,15,16,17,18,19] : LOAD HIGH;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74564,SN74564

#TEXT
```

```
TRI-STATE 8-Bit-D-
Type Register with
Inverted Outputs

This device contains
eight inverting bistable
memory elements with
three-state outputs.

#PIN 20
 1 : -OE
 2 : Input  D0
 3 : Input  D1
 4 : Input  D2
 5 : Input  D3
 6 : Input  D4
 7 : Input  D5
 8 : Input  D6
 9 : Input  D7
10 : GND
11 : Clock
12 : Output Q7
13 : Output Q6
14 : Output Q5
15 : Output Q4
16 : Output Q3
17 : Output Q2
18 : Output Q1
19 : Output Q0
20 : +5V

#FAMILY ALS,AS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11] : INPUT;
PIN[12,13,14,15,16,17,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=0;
   PIN[1]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&
       NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18])&NOT(PIN[19]))<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[12,13,14,15,16,17,18,19] : LOAD LOW;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%00000000 THEN ERROR(1);
PIN[12,13,14,15,16,17,18,19] : LOAD HIGH;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
```

```
#NAME 74568,SN74568

#TEXT
4-Bit Up/Down Decade
Counter (TRI-STATE)

This device contains a
sychronous, programmable
up/down decimal counter.

#PIN 20
 1 : Up/-Down
 2 : Clock
 3 : Input  P0
 4 : Input  P1
 5 : Input  P2
 6 : Input  P3
 7 : -CEP
 8 : -MR
 9 : -SR
10 : GND
11 : -PE
12 : -CET
13 : Output Q3
14 : Output Q2
15 : Output Q1
16 : Output Q0
17 : OE
18 : GC
19 : -TC
20 : +5V

#FAMILY ALS,F,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,17] : INPUT;
PIN[13,14,15,16,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[2,7,12]:=LOW;
PIN[8,9,11,17]:=HIGH;

PIN[8]:=LOW;PIN[8]:=HIGH;

PIN[1]:=HIGH;
FOR I:=0 TO 8 DO
   BEGIN
   PIN[17]:=LOW;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
   PIN[17]:=HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
PIN[17]:=LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
PIN[17]:=HIGH;
IF PIN[19]<>LOW THEN ERROR(1);

PIN[1]:=LOW;
FOR I:=9 DOWNTO 1 DO
   BEGIN
   PIN[17]:=LOW;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
   PIN[17]:=HIGH;
```

```
   IF PIN[19]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
PIN[17]:=LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
PIN[17]:=HIGH;
IF PIN[19]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[11]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[11]:=HIGH;
   PIN[17]:=LOW;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   PIN[17]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
PIN[17]:=LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);
PIN[17]:=HIGH;

LOADMODEON;
PIN[13,14,15,16] : LOAD LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>%0000 THEN ERROR(1);
PIN[13,14,15,16] : LOAD HIGH;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74569,SN74569

#TEXT
4-Bit Up/Down Binary
Counter (TRI-STATE)

This device contains a
synchronous, programmable
up/down binary counter.

#PIN 20
 1 : Up/-Down
 2 : Clock
 3 : Input  P0
 4 : Input  P1
 5 : Input  P2
 6 : Input  P3
 7 : -CEP
 8 : -MR
 9 : -SR
10 : GND
11 : -PE
12 : -CET
13 : Output Q3
14 : Output Q2
15 : Output Q1
16 : Output Q0
17 : OE
18 : GC
19 : -TC
20 : +5V

#FAMILY ALS,F,LS
```

```
#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,17] : INPUT;
PIN[13,14,15,16,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[2,7,12]:=LOW;
PIN[8,9,11,17]:=HIGH;

PIN[8]:=LOW;PIN[8]:=HIGH;

PIN[1]:=HIGH;
FOR I:=0 TO 14 DO
   BEGIN
   PIN[17]:=LOW;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
   PIN[17]:=HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
PIN[17]:=LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
PIN[17]:=HIGH;
IF PIN[19]<>LOW THEN ERROR(1);

PIN[1]:=LOW;
FOR I:=15 DOWNTO 1 DO
   BEGIN
   PIN[17]:=LOW;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
   PIN[17]:=HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
PIN[17]:=LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>I THEN ERROR(1);
PIN[17]:=HIGH;
IF PIN[19]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[11]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[11]:=HIGH;
   PIN[17]:=LOW;
   IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>D THEN ERROR(1);
   PIN[17]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
PIN[17]:=LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>0 THEN ERROR(1);
PIN[17]:=HIGH;

LOADMODEON;
PIN[13,14,15,16] : LOAD LOW;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>%0000 THEN ERROR(1);
PIN[13,14,15,16] : LOAD HIGH;
IF (PIN[13]&PIN[14]&PIN[15]&PIN[16])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
```

```
#NAME 74573,SN74573

#TEXT
TRI-STATE 8-Bit-D-
Type Latch with Enable

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
 2 : Input  D0
 3 : Input  D1
 4 : Input  D2
 5 : Input  D3
 6 : Input  D4
 7 : Input  D5
 8 : Input  D6
 9 : Input  D7
10 : GND
11 : Latch Enable
12 : Output Q7
13 : Output Q6
14 : Output Q5
15 : Output Q4
16 : Output Q3
17 : Output Q2
18 : Output Q1
19 : Output Q0
20 : +5V

#FAMILY ALS,AS,LS,S

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11] : INPUT;
PIN[12,13,14,15,16,17,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=0;
   PIN[1]:=LOW;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
       PIN[16]&PIN[17]&PIN[18]&PIN[19])<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[12,13,14,15,16,17,18,19] : LOAD LOW;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%00000000 THEN ERROR(1);
PIN[12,13,14,15,16,17,18,19] : LOAD HIGH;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%11111111 THEN ERROR(1);
LOADMODEOFF;
```

```
ERROR(0);
END.

#NAME 74574,SN74574

#TEXT
TRI-STATE 8-Bit-D-
Type Register (non-
inverting)

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
 2 : Input  D0
 3 : Input  D1
 4 : Input  D2
 5 : Input  D3
 6 : Input  D4
 7 : Input  D5
 8 : Input  D6
 9 : Input  D7
10 : GND
11 : Clock
12 : Output Q7
13 : Output Q6
14 : Output Q5
15 : Output Q4
16 : Output Q3
17 : Output Q2
18 : Output Q1
19 : Output Q0
20 : +5V

#FAMILY ALS,AS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11] : INPUT;
PIN[12,13,14,15,16,17,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=0;
   PIN[1]:=LOW;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
       PIN[16]&PIN[17]&PIN[18]&PIN[19])<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[12,13,14,15,16,17,18,19] : LOAD LOW;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%00000000 THEN ERROR(1);
PIN[12,13,14,15,16,17,18,19] : LOAD HIGH;
```

```
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74576,SN74576

#TEXT
TRI-STATE 8-Bit-D-
Type Register with
Inverted Outputs

This device contains
eight bistable memory
elements with inverting
three-state outputs.

#PIN 20
 1 : -OE
 2 : Input   D0
 3 : Input   D1
 4 : Input   D2
 5 : Input   D3
 6 : Input   D4
 7 : Input   D5
 8 : Input   D6
 9 : Input   D7
10 : GND
11 : Clock
12 : Output -Q7
13 : Output -Q6
14 : Output -Q5
15 : Output -Q4
16 : Output -Q3
17 : Output -Q2
18 : Output -Q1
19 : Output -Q0
20 : +5V

#FAMILY ALS,AS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11] : INPUT;
PIN[12,13,14,15,16,17,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=0;
   PIN[1]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&
       NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18])&NOT(PIN[19]))<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
```

```
PIN[12,13,14,15,16,17,18,19] : LOAD LOW;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%00000000 THEN ERROR(1);
PIN[12,13,14,15,16,17,18,19] : LOAD HIGH;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74580,SN74580

#TEXT
TRI-STATE 8-Bit-D-
Type Latch with
Inverted Outputs

This device contains
eight bistable memory
elements with inverting
three-state outputs.

#PIN 20
 1 : -OE
 2 : Input   D0
 3 : Input   D1
 4 : Input   D2
 5 : Input   D3
 6 : Input   D4
 7 : Input   D5
 8 : Input   D6
 9 : Input   D7
10 : GND
11 : Latch Enable
12 : Output -Q7
13 : Output -Q6
14 : Output -Q5
15 : Output -Q4
16 : Output -Q3
17 : Output -Q2
18 : Output -Q1
19 : Output -Q0
20 : +5V

#FAMILY ALS,AS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11] : INPUT;
PIN[12,13,14,15,16,17,18,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=0;
   PIN[1]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&NOT(PIN[15])&
       NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18])&NOT(PIN[19]))<>D THEN ERROR(1);
   PIN[1]:=HIGH;
```

```
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[12,13,14,15,16,17,18,19] : LOAD LOW;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%00000000 THEN ERROR(1);
PIN[12,13,14,15,16,17,18,19] : LOAD HIGH;
IF (PIN[12]&PIN[13]&PIN[14]&PIN[15]&
    PIN[16]&PIN[17]&PIN[18]&PIN[19])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

##NAME 74590,SN74590

#TEXT
TRI-STATE 8-Bit-Binary
Counter with Output
Latches and Clear

This device contains an
8-bit binary counter and
an 8-bit register with
three-state outputs.

#PIN 16
 1 : Output Q1
 2 : Output Q2
 3 : Output Q3
 4 : Output Q4
 5 : Output Q5
 6 : Output Q6
 7 : Output Q7
 8 : GND
 9 : -RCO
10 : -CCLR
11 : CCK
12 : -CCKEN
13 : RCK
14 : -OE
15 : Output Q0
16 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[10,11,12,13,14] : INPUT;
PIN[1,2,3,4,5,6,7,9,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11,12,13]:=LOW;
PIN[10,14]:=HIGH;

PIN[10]:=LOW;PIN[10]:=HIGH;
FOR I:=0 TO 255 DO
   BEGIN
   PIN[13]:=HIGH;PIN[13]:=LOW;
   PIN[14]:=LOW;
   IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
       PIN[3]&PIN[2]&PIN[1]&PIN[15])<>I THEN ERROR(1);
   PIN[14]:=HIGH;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   END;
```

```
PIN[14]:=LOW;
IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
    PIN[3]&PIN[2]&PIN[1]&PIN[15])<>0 THEN ERROR(1);
PIN[14]:=HIGH;

LOADMODEON;
PIN[7,6,5,4,3,2,1,15] : LOAD LOW;
IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
    PIN[3]&PIN[2]&PIN[1]&PIN[15])<>%00000000 THEN ERROR(1);
PIN[7,6,5,4,3,2,1,15] : LOAD HIGH;
IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
    PIN[3]&PIN[2]&PIN[1]&PIN[15])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

##NAME 74591,SN74591

#TEXT
8-Bit Binary Counter
with Output Latches and
Clear (O.C.)

This device contains
an 8-bit binary counter
and an 8-bit register
with OPEN COLLECTOR
outputs.

#PIN 16
 1 : Output Q1
 2 : Output Q2
 3 : Output Q3
 4 : Output Q4
 5 : Output Q5
 6 : Output Q6
 7 : Output Q7
 8 : GND
 9 : -RCO
10 : -CCLR
11 : CCK
12 : -CCKEN
13 : RCK
14 : -OE
15 : Output Q0
16 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[10,11,12,13,14] : INPUT;
PIN[1,2,3,4,5,6,7,9,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11,12,13,14]:=LOW;
PIN[10]:=HIGH;

PIN[10]:=LOW;PIN[10]:=HIGH;
FOR I:=0 TO 255 DO
   BEGIN
   PIN[13]:=HIGH;PIN[13]:=LOW;
   LOADMODEON;
   PIN[7,6,5,4,3,2,1,15] : LOAD LOW;
   IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
```

```
       PIN[3]&PIN[2]&PIN[1]&PIN[15])<>0 THEN ERROR(1);
   PIN[7,6,5,4,3,2,1,15] : LOAD HIGH;
   IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
       PIN[3]&PIN[2]&PIN[1]&PIN[15])<>I THEN ERROR(1);
   LOADMODEOFF;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   END;

LOADMODEON;
PIN[7,6,5,4,3,2,1,15] : LOAD LOW;
IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
    PIN[3]&PIN[2]&PIN[1]&PIN[15])<>0 THEN ERROR(1);
PIN[7,6,5,4,3,2,1,15] : LOAD HIGH;
IF (PIN[7]&PIN[6]&PIN[5]&PIN[4]&
    PIN[3]&PIN[2]&PIN[1]&PIN[15])<>0 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74620,SN74620

#TEXT
Inverting 8-Bit TRI-
STATE Transceiver
with Data Storage

This device contains
eight bidirectional
buffers with three-
state outputs.

#PIN 20
 1 : GAB
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -GBA
20 : +5V

#FAMILY ALS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

PIN[1,19]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
```

```
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[1,19]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   IF (NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;PIN[19]:=HIGH;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74621,SN74621

#TEXT
Non-Inverting 8-Bit
Transceiver with
Storage (O.C.)

This device contains
eight bidirectional
buffers with non-
inverting outputs
(OPEN COLLECTOR).

#PIN 20
 1 : GAB
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
```

```
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -GBA
20 : +5V

#FAMILY ALS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

PIN[1,19]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %11111111;
   END;

PIN[1,19]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   LOADMODEON;
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74622,SN74622

#TEXT
Inverting 8-Bit
Transceiver with
Storage (O.C.)

This device contains
eight bidirectional
buffers with inverting
outputs (OPEN COLLECTOR)

#PIN 20
 1 : GAB
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
```

```
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -GBA
20 : +5V

#FAMILY ALS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

PIN[1,19]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %11111111;
   END;

PIN[1,19]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   LOADMODEON;
   PIN[9,8,7,6,5,4,3,2] : LOAD LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74623,SN74623

#TEXT
Non-Inverting Octal TRI-
STATE Transceiver with
Storage
```

```
This device contains
eight bidirectional
buffers with three-
state outputs.

#PIN 20
 1 : GAB
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -GBA
20 : +5V

#FAMILY ALS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

PIN[1,19]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[1,19]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;PIN[19]:=HIGH;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
```

```
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74638,SN74638

#TEXT
Inverting Octal TRI-
STATE Transceiver

This device contains
eight inverting bi-
directional buffers
with three-state
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -Enable
20 : +5V

#FAMILY ALS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
```

```
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74639,SN74639

#TEXT
Octal TRI-STATE
Transceiver

This device contains
eight non-inverting
bidirectional buffers
with three-state
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -Enable
20 : +5V
```

```
#FAMILY ALS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74640,SN74640

#TEXT
Inverting Octal TRI-
STATE Transceiver

This device contains
eight inverting bi-
directional buffers
with three-state
outputs.
```

```
#PIN 20
 1 : DIR
 2 : In-/Output A1
 3 : In-/Output A2
 4 : In-/Output A3
 5 : In-/Output A4
 6 : In-/Output A5
 7 : In-/Output A6
 8 : In-/Output A7
 9 : In-/Output A8
10 : GND
11 : In-/Output B8
12 : In-/Output B7
13 : In-/Output B6
14 : In-/Output B5
15 : In-/Output B4
16 : In-/Output B3
17 : In-/Output B2
18 : In-/Output B1
19 : -Enable
20 : +5V

#FAMILY ALS,AS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
```

```
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```

```
#NAME 74641,SN74641

#TEXT
Non-Inverting Octal
Transceiver

This device contains
eight non-inverting bi-
directional buffers with
OPEN COLLECTOR outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS,AS,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>0 THEN ERROR(1);
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
```

```
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[9,8,7,6,5,4,3,2] : LOAD LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74642,SN74642

#TEXT
Inverting Octal
Transceiver (O.C.)

This device contains
eight inverting bi-
directional buffers
with OPEN COLLECTOR
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS,AS,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
```

```
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>0 THEN ERROR(1);
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[9,8,7,6,5,4,3,2] : LOAD LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74643,SN74643

#TEXT
True-Inverting Octal
TRI-STATE Transceiver

This device contains
eight inverting/non-
inverting bidirectional
buffers with three-state
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
```

```
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS,AS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74644,SN74644

#TEXT
True-Inverting Octal
Transceiver (O.C.)
```

```
This device contains
eight inverting/non-
inverting bidirectional
buffers with OPEN
COLLECTOR outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS,AS,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : Output ;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>0 THEN ERROR(1);
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
```

```
   PIN[9,8,7,6,5,4,3,2] : LOAD LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 74645,SN74645

#TEXT
Non-Inverting Octal
TRI-STATE Transceiver

This device contains
eight non-inverting bi-
directional buffers
with three-state
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS,AS,F,LS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
```

```
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74668,SN74668

#TEXT
Synchronous Up/Down
Decade Counter

This device contains a
synchronous, programmable
up/down decade counter.

#PIN 16
 1 : Up/-Down
 2 : Clock
 3 : Input   D0
 4 : Input   D1
 5 : Input   D2
 6 : Input   D3
 7 : -CEP
 8 : GND
 9 : -PE
10 : -CET
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
15 : -TC
16 : +5V

#FAMILY LS

#PROGRAM
```

```
BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,7,10]:=LOW;
PIN[9]:=HIGH;

PIN[6]&PIN[5]&PIN[4]&PIN[3]:=0;
PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;

PIN[1]:=HIGH;
FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

PIN[1]:=LOW;
FOR I:=9 DOWNTO 1 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74669,SN74669

#TEXT
Synchronous Up/Down
Binary Counter

This device contains a
synchronous, programmable
up/down binary counter.

#PIN 16
 1 : Up/-Down
 2 : Clock
 3 : Input   D0
 4 : Input   D1
 5 : Input   D2
 6 : Input   D3
 7 : -CEP
 8 : GND
 9 : -PE
10 : -CET
11 : Output  Q3
12 : Output  Q2
```

```
13 : Output  Q1
14 : Output  Q0
15 : -TC
16 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,7,10]:=LOW;
PIN[9]:=HIGH;

PIN[9]:=LOW;
PIN[6]&PIN[5]&PIN[4]&PIN[3]:=0;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[9]:=HIGH;

PIN[1]:=HIGH;
FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

PIN[1]:=LOW;
FOR I:=15 DOWNTO 1 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 74670,SN74670

#TEXT
16-Bit RAM (4x4, TRI-
STATE)

This device contains a
read/write memory (RAM)
arranged in 4x4 words
allowing simultaneous
reading and writing.
```

```
The outputs are three-
state.

#PIN 16
 1 : Input   D2
 2 : Input   D3
 3 : Input   D4
 4 : Address RB
 5 : Address RA
 6 : Output  Q4
 7 : Output  Q3
 8 : GND
 9 : Output  Q2
10 : Output  Q1
11 : -RE
12 : -WE
13 : Address WB
14 : Address WA
15 : Input   D1
16 : +5V

#FAMILY Std,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,11,12,13,14,15] : INPUT;
PIN[6,7,9,10] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11,12]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[13]&PIN[14]:=J;
      PIN[12]:=LOW;
      PIN[3]&PIN[2]&PIN[1]&PIN[15]:=D;
      PIN[12]:=HIGH;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[4]&PIN[5]:=J;
      PIN[11]:=LOW;
      IF (PIN[6]&PIN[7]&PIN[9]&PIN[10])<>D THEN ERROR(1);
      PIN[11]:=HIGH;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[6,7,9,10] : LOAD LOW;
IF (PIN[6]&PIN[7]&PIN[9]&PIN[10])<>%0000 THEN ERROR(1);
PIN[6,7,9,10] : LOAD HIGH;
IF (PIN[6]&PIN[7]&PIN[9]&PIN[10])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74682,SN74682
```

```
#TEXT
8-Bit Comparator with
Internal 20kê-Pull-Up

This device compares two
8-bit words, A and B,
and indicates whether or
not they are equal, or
if A is greater than B.

#PIN 20
 1 : - (A>B)
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : - (A=B)
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[1,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   IF (PIN[1]<>LOW) OR (PIN[19]<>HIGH) THEN ERROR(1);
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>HIGH) THEN ERROR(1);
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74683,SN74683

#TEXT
```

8-Bit Comparator with
Internal 20kê-Pull-Up
(O.C.)

This device compares two
8-bit words, A and B,
and indicates whether
or not they are equal or
if A is greater than B.
The outputs are OPEN
COLLECTOR.

```
#PIN 20
 1 : - (A>B)
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : - (A=B)
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[1,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>LOW) OR (PIN[19]<>HIGH) THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
```

```
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>HIGH) THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74684,SN74684

#TEXT
8-Bit Comparator

This device compares two
8-bit words, A and B,
and indicates whether or
not they are equal or if
A is greater than B.

#PIN 20
 1 : - (A>B)
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : - (A=B)
20 : +5V

#FAMILY F,LS

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[1,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);

C:=0;
```

```
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   IF (PIN[1]<>LOW) OR (PIN[19]<>HIGH) THEN ERROR(1);
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>HIGH) THEN ERROR(1);
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74685,SN74685

#TEXT
8-Bit Comparator (O.C.)

This device compares two
8-bit words, A and B,
and indicates whether or
not they are equal or
if A is greater than B.
The outputs are OPEN
COLLECTOR.

#PIN 20
 1 : - (A>B)
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
19 : - (A=B)
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[1,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);

C:=0;
```

```
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>LOW) OR (PIN[19]<>HIGH) THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>HIGH) THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[1,19] : LOAD LOW;
   IF (PIN[1]<>LOW) OR (PIN[19]<>LOW) THEN ERROR(1);
   PIN[1,19] : LOAD HIGH;
   IF (PIN[1]<>HIGH) OR (PIN[19]<>LOW) THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74688,SN74688

#TEXT
8-Bit Comparator with
Output Enable

This device compares two
8-bit words, A and B and
indicates whether or not
they are equal.

#PIN 20
 1 : -G
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
```

```
19 : - (A=B)
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
IF PIN[19]<>LOW THEN ERROR(1);

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   IF PIN[19]<>HIGH THEN ERROR(1);
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   IF PIN[19]<>LOW THEN ERROR(1);
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   IF PIN[19]<>HIGH THEN ERROR(1);
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   IF PIN[19]<>LOW THEN ERROR(1);
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74689,SN74689

#TEXT
8-Bit Comparator with
Output Enable (O.C.)

This device compares two
8-bit words, A and B, and
indicates whether or not
they are equal. The out-
puts are OPEN COLLECTOR.

#PIN 20
 1 : -G
 2 : Input   A0
 3 : Input   B0
 4 : Input   A1
 5 : Input   B1
 6 : Input   A2
 7 : Input   B2
 8 : Input   A3
 9 : Input   B3
10 : GND
11 : Input   A4
12 : Input   B4
13 : Input   A5
14 : Input   B5
15 : Input   A6
16 : Input   B6
17 : Input   A7
18 : Input   B7
```

```
19 : - (A=B)
20 : +5V

#FAMILY ALS,LS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,11,12,13,14,15,16,17,18] : INPUT;
PIN[19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=LOW;

PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=0;
PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=0;
IF PIN[19]<>LOW THEN ERROR(1);

C:=0;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD LOW;
   IF PIN[19]<>LOW THEN ERROR(1);
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD LOW;
   IF PIN[19]<>LOW THEN ERROR(1);
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   PIN[18]&PIN[16]&PIN[14]&PIN[12]&PIN[9]&PIN[7]&PIN[5]&PIN[3]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD LOW;
   IF PIN[19]<>LOW THEN ERROR(1);
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>HIGH THEN ERROR(1);
   LOADMODEOFF;
   PIN[17]&PIN[15]&PIN[13]&PIN[11]&PIN[8]&PIN[6]&PIN[4]&PIN[2]:=1 SHL C;
   LOADMODEON;
   PIN[19] : LOAD LOW;
   IF PIN[19]<>LOW THEN ERROR(1);
   PIN[19] : LOAD HIGH;
   IF PIN[19]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 74795,SN74795

#TEXT
Non-Inverting Octal
TRI-STATE Buffer

This device contains
eight non-inverting
buffers with common
enable and three-state
outputs.
```

```
#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
 5 : Output    Buffer 2
 6 : Input     Buffer 3
 7 : Output    Buffer 3
 8 : Input     Buffer 4
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
       PIN[11]&PIN[13]&PIN[15]&PIN[17])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74796,SN74796

#TEXT
Inverting Octal TRI-STATE
Buffer

This device contains
eight inverting buffers
with common enable and
three-state outputs.
```

```
#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
 5 : Output    Buffer 2
 6 : Input     Buffer 3
 7 : Output    Buffer 3
 8 : Input     Buffer 4
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[3])&NOT(PIN[5])&NOT(PIN[7])&NOT(PIN[9])&
       NOT(PIN[11])&NOT(PIN[13])&NOT(PIN[15])&NOT(PIN[17]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74797,SN74797

#TEXT
Octal TRI-STATE Buffer

This device contains
eight non-inverting
buffers with two enable
inputs and three-state
outputs.
```

```
#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
 5 : Output    Buffer 2
 6 : Input     Buffer 3
 7 : Output    Buffer 3
 8 : Input     Buffer 4
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
       PIN[11]&PIN[13]&PIN[15]&PIN[17])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74798,SN74798

#TEXT
Inverting Octal TRI-STATE
Buffer

This device contains
eight inverting buffers
with two enable inputs
and three-state outputs.
```

```
#PIN 20
 1 : Enable   -E1
 2 : Input     Buffer 1
 3 : Output    Buffer 1
 4 : Input     Buffer 2
 5 : Output    Buffer 2
 6 : Input     Buffer 3
 7 : Output    Buffer 3
 8 : Input     Buffer 4
 9 : Output    Buffer 4
10 : GND
11 : Output    Buffer 5
12 : Input     Buffer 5
13 : Output    Buffer 6
14 : Input     Buffer 6
15 : Output    Buffer 7
16 : Input     Buffer 7
17 : Output    Buffer 8
18 : Input     Buffer 8
19 : Enable   -E2
20 : +5V

#FAMILY LS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,12,14,16,18,19] : INPUT;
PIN[3,5,7,9,11,13,15,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[12]&PIN[14]&PIN[16]&PIN[18]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[3])&NOT(PIN[5])&NOT(PIN[7])&NOT(PIN[9])&
       NOT(PIN[11])&NOT(PIN[13])&NOT(PIN[15])&NOT(PIN[17]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13,15,17] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%00000000 THEN ERROR(1);
PIN[3,5,7,9,11,13,15,17] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&PIN[9]&
    PIN[11]&PIN[13]&PIN[15]&PIN[17])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 74800,SN74800

#TEXT
Triple 4-Input AND/NAND
Driver

This device contains
three AND/NAND drivers
each with four inputs.
```

```
#PIN 20
 1 : Input   1 Driver  1
 2 : Input   1 Driver  2
 3 : Input   2 Driver  2
 4 : Input   3 Driver  2
 5 : Input   4 Driver  2
 6 : Input   1 Driver  3
 7 : Input   2 Driver  3
 8 : Input   3 Driver  3
 9 : Input   4 Driver  3
10 : GND
11 : NAND-Out. Driver  3
12 : AND-Out.  Driver  3
13 : NAND-Out. Driver  2
14 : AND-Out.  Driver  2
15 : NAND-Out. Driver  1
16 : AND-Out.  Driver  1
17 : Input   2 Driver  1
18 : Input   3 Driver  1
19 : Input   4 Driver  1
20 : +5V

#FAMILY AS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,17,18,19] : INPUT;
PIN[11,12,13,14,15,16] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[17]&PIN[18]&PIN[19]:=I;
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[6]&PIN[7]&PIN[8]&PIN[9]:=I;
   IF PIN[16]<>(PIN[1] AND PIN[17] AND PIN[18] AND PIN[19]) THEN ERROR(1);
   IF PIN[15]<>NOT(PIN[1] AND PIN[17] AND PIN[18] AND PIN[19]) THEN ERROR(1);
   IF PIN[14]<>(PIN[2] AND PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[13]<>NOT(PIN[2] AND PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[12]<>(PIN[6] AND PIN[7] AND PIN[8] AND PIN[9]) THEN ERROR(1);
   IF PIN[11]<>NOT(PIN[6] AND PIN[7] AND PIN[8] AND PIN[9]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74802,SN74802

#TEXT
Triple 4-Input OR/NOR
Driver

This device contains
three OR/NOR drivers
each with four inputs.

#PIN 20
 1 : Input   1 Driver  1
 2 : Input   1 Driver  2
 3 : Input   2 Driver  2
 4 : Input   3 Driver  2
 5 : Input   4 Driver  2
 6 : Input   1 Driver  3
 7 : Input   2 Driver  3
 8 : Input   3 Driver  3
```

```
 9 : Input   4 Driver  3
10 : GND
11 : NOR-Out.  Driver  3
12 : OR-Out.   Driver  3
13 : NOR-Out.  Driver  2
14 : OR-Out.   Driver  2
15 : NOR-Out.  Driver  1
16 : OR-Out.   Driver  1
17 : Input   2 Driver  1
18 : Input   3 Driver  1
19 : Input   4 Driver  1
20 : +5V

#FAMILY AS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8,9,17,18,19] : INPUT;
PIN[11,12,13,14,15,16] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[17]&PIN[18]&PIN[19]:=I;
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[6]&PIN[7]&PIN[8]&PIN[9]:=I;
   IF PIN[16]<>(PIN[1] OR PIN[17] OR PIN[18] OR PIN[19]) THEN ERROR(1);
   IF PIN[15]<>NOT(PIN[1] OR PIN[17] OR PIN[18] OR PIN[19]) THEN ERROR(1);
   IF PIN[14]<>(PIN[2] OR PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[13]<>NOT(PIN[2] OR PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[12]<>(PIN[6] OR PIN[7] OR PIN[8] OR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>NOT(PIN[6] OR PIN[7] OR PIN[8] OR PIN[9]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74804,SN74804

#TEXT
Hex 2-Input NAND Driver

This device contains
six NAND drivers each
with two inputs.

#PIN 20
 1 : Input   1 Driver  1
 2 : Input   2 Driver  1
 3 : Output    Driver  1
 4 : Input   1 Driver  2
 5 : Input   2 Driver  2
 6 : Output    Driver  2
 7 : Input   1 Driver  2
 8 : Input   2 Driver  2
 9 : Output    Driver  2
10 : GND
11 : Output    Driver  3
12 : Input   1 Driver  3
13 : Input   2 Driver  3
14 : Output    Driver  4
15 : Input   1 Driver  4
16 : Input   2 Driver  4
17 : Output    Driver  4
18 : Input   1 Driver  4
```

```
19 : Input   2 Driver  4
20 : +5V

#FAMILY ALS,AS

#PROGRAM

BEGIN
PIN[1,2,4,5,7,8,12,13,15,16,18,19] : INPUT;
PIN[3,6,9,11,14,17] : Output ;
PIN[10] : GND;
PIN[20] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[7]&PIN[8]:=I;
   PIN[12]&PIN[13]:=I;
   PIN[15]&PIN[16]:=I;
   PIN[18]&PIN[19]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[9]<>(PIN[7] NAND PIN[8]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   IF PIN[14]<>(PIN[15] NAND PIN[16]) THEN ERROR(1);
   IF PIN[17]<>(PIN[18] NAND PIN[19]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74805,SN74805

#TEXT
Hex 2-Input NOR Driver

This device contains
six NOR drivers each
with two inputs.

#PIN 20
 1 : Input   1 Driver  1
 2 : Input   2 Driver  1
 3 : Output    Driver  1
 4 : Input   1 Driver  2
 5 : Input   2 Driver  2
 6 : Output    Driver  2
 7 : Input   1 Driver  2
 8 : Input   2 Driver  2
 9 : Output    Driver  2
10 : GND
11 : Output    Driver  3
12 : Input   1 Driver  3
13 : Input   2 Driver  3
14 : Output    Driver  4
15 : Input   1 Driver  4
16 : Input   2 Driver  4
17 : Output    Driver  4
18 : Input   1 Driver  4
19 : Input   2 Driver  4
20 : +5V

#FAMILY ALS,AS

#PROGRAM
```

```
BEGIN
PIN[1,2,4,5,7,8,12,13,15,16,18,19] : INPUT;
PIN[3,6,9,11,14,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[7]&PIN[8]:=I;
   PIN[12]&PIN[13]:=I;
   PIN[15]&PIN[16]:=I;
   PIN[18]&PIN[19]:=I;
   IF PIN[3]<>(PIN[1] NOR PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NOR PIN[5]) THEN ERROR(1);
   IF PIN[9]<>(PIN[7] NOR PIN[8]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NOR PIN[13]) THEN ERROR(1);
   IF PIN[14]<>(PIN[15] NOR PIN[16]) THEN ERROR(1);
   IF PIN[17]<>(PIN[18] NOR PIN[19]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74808,SN74808

#TEXT
Hex 2-Input AND Driver

This device contains
six AND drivers each
with two inputs.

#PIN 20
 1 : Input   1 Driver  1
 2 : Input   2 Driver  1
 3 : Output    Driver  1
 4 : Input   1 Driver  2
 5 : Input   2 Driver  2
 6 : Output    Driver  2
 7 : Input   1 Driver  2
 8 : Input   2 Driver  2
 9 : Output    Driver  2
10 : GND
11 : Output    Driver  3
12 : Input   1 Driver  3
13 : Input   2 Driver  3
14 : Output    Driver  4
15 : Input   1 Driver  4
16 : Input   2 Driver  4
17 : Output    Driver  4
18 : Input   1 Driver  4
19 : Input   2 Driver  4
20 : +5V

#FAMILY ALS,AS

#PROGRAM

BEGIN
PIN[1,2,4,5,7,8,12,13,15,16,18,19] : INPUT;
PIN[3,6,9,11,14,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

FOR I:=0 TO 3 DO
```

```
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[7]&PIN[8]:=I;
   PIN[12]&PIN[13]:=I;
   PIN[15]&PIN[16]:=I;
   PIN[18]&PIN[19]:=I;
   IF PIN[3]<>(PIN[1] AND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[9]<>(PIN[7] AND PIN[8]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] AND PIN[13]) THEN ERROR(1);
   IF PIN[14]<>(PIN[15] AND PIN[16]) THEN ERROR(1);
   IF PIN[17]<>(PIN[18] AND PIN[19]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 74832,SN74832

#TEXT
Hex 2-Input OR Driver

This device contains
six OR drivers each
with two inputs.

#PIN 20
 1 : Input   1 Driver  1
 2 : Input   2 Driver  1
 3 : Output    Driver  1
 4 : Input   1 Driver  2
 5 : Input   2 Driver  2
 6 : Output    Driver  2
 7 : Input   1 Driver  2
 8 : Input   2 Driver  2
 9 : Output    Driver  2
10 : GND
11 : Output    Driver  3
12 : Input   1 Driver  3
13 : Input   2 Driver  3
14 : Output    Driver  4
15 : Input   1 Driver  4
16 : Input   2 Driver  4
17 : Output    Driver  4
18 : Input   1 Driver  4
19 : Input   2 Driver  4
20 : +5V

#FAMILY ALS,AS

#PROGRAM

BEGIN
PIN[1,2,4,5,7,8,12,13,15,16,18,19] : INPUT;
PIN[3,6,9,11,14,17] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[7]&PIN[8]:=I;
   PIN[12]&PIN[13]:=I;
   PIN[15]&PIN[16]:=I;
   PIN[18]&PIN[19]:=I;
```

```
   IF PIN[3]<>(PIN[1] OR PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[9]<>(PIN[7] OR PIN[8]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] OR PIN[13]) THEN ERROR(1);
   IF PIN[14]<>(PIN[15] OR PIN[16]) THEN ERROR(1);
   IF PIN[17]<>(PIN[18] OR PIN[19]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 741000,SN741000

#TEXT
Buffered Quad 2-Input
NAND Gate

This device contains
four NAND gates each
with two inputs and
buffered outputs.

#PIN 14
 1 : Input   1 Gate   1
 2 : Input   2 Gate   1
 3 : Output    Gate   1
 4 : Input   1 Gate   2
 5 : Input   2 Gate   2
 6 : Output    Gate   2
 7 : GND
 8 : Output    Gate   3
 9 : Input   1 Gate   3
10 : Input   2 Gate   3
11 : Output    Gate   4
12 : Input   1 Gate   4
13 : Input   2 Gate   4
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.
•
```

```
#NAME 741002,SN741002

#TEXT
Buffered Quad 2-Input
NOR Gate

This device contains
four NOR gates each
with 2 inputs and
buffered outputs.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Input   1 Gate 4
12 : Input   2 Gate 4
13 : Output    Gate 4
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[2,3,5,6,8,9,11,12] : INPUT;
PIN[1,4,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[2]&PIN[3]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[11]&PIN[12]:=I;
   IF PIN[1]<>(PIN[2] NOR PIN[3]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NOR PIN[9]) THEN ERROR(1);
   IF PIN[13]<>(PIN[11] NOR PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 741003,SN741003

#TEXT
Buffered Quad 2-Input
NAND Gate (O.C.)

This device contains
four NAND gates each
with two inputs; the
outputs are buffered
and OPEN COLLECTOR.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
```

```
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   LOADMODEON;
   PIN[3,6,8,11] : LOAD HIGH;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] NAND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] NAND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   PIN[3,6,8,11] : LOAD LOW;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[6]<>LOW THEN ERROR(1);
   IF PIN[8]<>LOW THEN ERROR(1);
   IF PIN[11]<>LOW THEN ERROR(1);
   LOADMODEOFF;
   END;

ERROR(0);
END.

#NAME 741004,SN741004

#TEXT
Buffered Hex Inverter

This device contains
six independent inverters
with buffered outputs.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
```

```
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 741005,SN741005

#TEXT
Buffered Hex Inverter
(O.C.)

This device contains
six independent inverters;
the outputs are buffered
and OPEN COLLECTOR.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
```

```
   LOADMODEON;
   PIN[2,4,6,8,10,12] : LOAD HIGH;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   PIN[2,4,6,8,10,12] : LOAD LOW;
   IF (PIN[2]&PIN[4]&PIN[6]&PIN[8]&PIN[10]&PIN[12])<>0 THEN ERROR(1);
   LOADMODEOFF;
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 741008,SN741008

#TEXT
Buffered Quad 2-Input
AND Gate

This device contains
four AND gates each
with two inputs and
buffered outputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] AND PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] AND PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 741010,SN741010
```

```
#TEXT
Buffered Triple 3-Input
NAND Gate

This device contains
three NAND gates each
with three inputs and
buffered outputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Input   1 Gate 2
 4 : Input   2 Gate 2
 5 : Input   3 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Input   3 Gate 3
12 : Output    Gate 1
13 : Input   3 Gate 1
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,13] : INPUT;
PIN[6,8,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[13]:=I;
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[12]<>NOT(PIN[1] AND PIN[2] AND PIN[13]) THEN ERROR(1);
   IF PIN[6]<>NOT(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 741011,SN741011

#TEXT
Buffered Triple 3-Input
AND Gate

This device contains
three AND gates each
with three inputs and
buffered outputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Input   1 Gate 2
 4 : Input   2 Gate 2
 5 : Input   3 Gate 2
 6 : Output    Gate 2
 7 : GND
```

```
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Input   3 Gate 3
12 : Output    Gate 1
13 : Input   3 Gate 1
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,3,4,5,9,10,11,13] : INPUT;
PIN[6,8,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[13]:=I;
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]:=I;
   IF PIN[12]<>(PIN[1] AND PIN[2] AND PIN[13]) THEN ERROR(1);
   IF PIN[6]<>(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] AND PIN[10] AND PIN[11]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 741020,SN741020

#TEXT
Buffered Dual 4-Input
NAND Gate

This device contains
two NAND gates each
with four inputs and
buffered outputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : N.C.
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Output    Gate 2
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : N.C.
12 : Input   3 Gate 2
13 : Input   4 Gate 2
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
```

```
FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[1] AND PIN[2] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[8]<>NOT(PIN[9] AND PIN[10] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 741032,SN741032

#TEXT
Buffered Quad 2-Input
OR Gate

This device contains
four OR gates each
with two inputs and
buffered outputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Input   1 Gate 2
 5 : Input   2 Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input   1 Gate 3
10 : Input   2 Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] OR PIN[2]) THEN ERROR(1);
   IF PIN[6]<>(PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[8]<>(PIN[9] OR PIN[10]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] OR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 741034,SN741034

#TEXT
```

```
Non-Inverting Hex Buffer

This device contains
six independent drivers
with buffered outputs.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (PIN[2]&PIN[4]&PIN[6]&
       PIN[8]&PIN[10]&PIN[12])<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 741035,SN741035

#TEXT
Inverting Hex Buffer

This device contains
six separate inverting
drivers with buffered
outputs.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
```

```
13 : Input     Gate 6
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 741240,SN741240

#TEXT
Inverting Octal TRI-
STATE Buffer

This device contains
eight inverting buffers
with three-state
outputs.

#PIN 20
 1 : -G1
 2 : Input    E0
 3 : Output  -Q7
 4 : Input    E1
 5 : Output  -Q6
 6 : Input    E2
 7 : Output  -Q5
 8 : Input    E3
 9 : Output  -Q4
10 : GND
11 : Input    E4
12 : Output  -Q3
13 : Input    E5
14 : Output  -Q2
15 : Input    E6
16 : Output  -Q1
17 : Input    E7
18 : Output  -Q0
19 : -G2
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
```

```
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[18])&NOT(PIN[16])&NOT(PIN[14])&NOT(PIN[12])&
       NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[5])&NOT(PIN[3]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741241,SN741241

#TEXT
Octal TRI-STATE Buffer

This device contains
eight non-inverting
buffers with three-state
outputs.

#PIN 20
 1 : -G1
 2 : Input   E0
 3 : Output  Q7
 4 : Input   E1
 5 : Output  Q6
 6 : Input   E2
 7 : Output  Q5
 8 : Input   E3
 9 : Output  Q4
10 : GND
11 : Input   E4
12 : Output  Q3
13 : Input   E5
14 : Output  Q2
15 : Input   E6
16 : Output  Q1
17 : Input   E7
18 : Output  Q0
19 : -G2
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
```

```
PIN[19]:=LOW;
PIN[1]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1]:=LOW;PIN[19]:=HIGH;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1]:=HIGH;PIN[19]:=LOW;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741242,SN741242

#TEXT
Inverting Quad TRI-
STATE Transceiver

This device contains
four inverting bi-
directional buffers
with three-state
outputs.

#PIN 14
 1 : -GA
 2 : N.C.
 3 : In-/Output  A0
 4 : In-/Output  A1
 5 : In-/Output  A2
 6 : In-/Output  A3
 7 : GND
 8 : In-/Output  B3
 9 : In-/Output  B2
10 : In-/Output  B1
11 : In-/Output  B0
12 : N.C.
13 : GB
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,13] : INPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%0101;
PIN[1,13]:=LOW;
PIN[3,4,5,6] : INPUT;
```

```
PIN[8,9,10,11] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   IF (NOT(PIN[8])&NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,13]:=HIGH;
PIN[8,9,10,11] : INPUT;
PIN[3,4,5,6] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[9]&PIN[10]&PIN[11]:=D;
   IF (NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=HIGH;PIN[13]:=LOW;
PIN[3,4,5,6,8,9,10,11] : OUTPUT;
LOADMODEON;
PIN[3,4,5,6,8,9,10,11] : LOAD LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,4,5,6,8,9,10,11] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741243,SN741243

#TEXT
Quad TRI-STATE
Transceiver

This device contains
four non-inverting bi-
directional buffers
with three-state
outputs.

#PIN 14
 1 : -GA
 2 : N.C.
 3 : In-/Output  A0
 4 : In-/Output  A1
 5 : In-/Output  A2
 6 : In-/Output  A3
 7 : GND
 8 : In-/Output  B3
 9 : In-/Output  B2
10 : In-/Output  B1
11 : In-/Output  B0
12 : N.C.
13 : GB
14 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,13] : INPUT;
PIN[7] : GND;
```

```
PIN[14] : +5V;

D:=%0101;
PIN[1,13]:=LOW;
PIN[3,4,5,6] : INPUT;
PIN[8,9,10,11] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,13]:=HIGH;
PIN[8,9,10,11] : INPUT;
PIN[3,4,5,6] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[9]&PIN[10]&PIN[11]:=D;
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=HIGH;PIN[13]:=LOW;
PIN[3,4,5,6,8,9,10,11] : OUTPUT;
LOADMODEON;
PIN[3,4,5,6,8,9,10,11] : LOAD LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,4,5,6,8,9,10,11] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741244,SN741244

#TEXT
Octal TRI-STATE Buffer

This device contains
eight non-inverting
buffers with three-state
outputs.

#PIN 20
 1 : -G1
 2 : Input   E0
 3 : Output  Q7
 4 : Input   E1
 5 : Output  Q6
 6 : Input   E2
 7 : Output  Q5
 8 : Input   E3
 9 : Output  Q4
10 : GND
11 : Input   E4
12 : Output  Q3
13 : Input   E5
14 : Output  Q2
15 : Input   E6
16 : Output  Q1
17 : Input   E7
18 : Output  Q0
19 : -G2
```

```
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741245,SN741245

#TEXT
Octal TRI-STATE
Transceiver

This device contains
eight non-inverting bi-
directional buffers
with three-state
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
```

```
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741640,SN741640

#TEXT
Inverting Octal TRI-STATE
Transceiver

This device contains
eight inverting bi-
```

directional buffers
with three-state
outputs.

```
#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
```

```
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741641,SN741641

#TEXT
Non-Inverting Octal
Transceiver  (O.C.)

This device contains
eight non-inverting bi-
directional buffers
with OPEN COLLECTOR
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
```

```
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>0 THEN ERROR(1);
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[9,8,7,6,5,4,3,2] : LOAD LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 741642,SN741642

#TEXT
Inverting Octal
Transceiver (O.C.)

This device contains
eight inverting bi-
directional buffers
with OPEN COLLECTOR
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V
```

```
#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>0 THEN ERROR(1);
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[9,8,7,6,5,4,3,2] : LOAD LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (NOT(PIN[9])&NOT(PIN[8])&NOT(PIN[7])&NOT(PIN[6])&
       NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3])&NOT(PIN[2]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 741643,SN741643

#TEXT
True-Inverting Octal
TRI-STATE Transceiver

This device contains
eight inverting/non-
inverting bidirectional
buffers with three-state
outputs.

#PIN 20
```

```
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
```

```
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 741644,SN741644

#TEXT
True-Inverting Octal
Transceiver (O.C.)

This device contains
eight inverting/non-
inverting bidirectional
buffers with OPEN
COLLECTOR outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[11,12,13,14,15,16,17,18] : LOAD LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>0 THEN ERROR(1);
   PIN[11,12,13,14,15,16,17,18] : LOAD HIGH;
```

```
   IF (NOT(PIN[11])&NOT(PIN[12])&NOT(PIN[13])&NOT(PIN[14])&
       NOT(PIN[15])&NOT(PIN[16])&NOT(PIN[17])&NOT(PIN[18]))<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   LOADMODEON;
   PIN[9,8,7,6,5,4,3,2] : LOAD LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>0 THEN ERROR(1);
   PIN[9,8,7,6,5,4,3,2] : LOAD HIGH;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   LOADMODEOFF;
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 741645,SN741645

#TEXT
Non-Inverting Octal TRI-
State Transceiver

This device contains
eight non-inverting
bidirectional buffers
with three-state
outputs.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : +5V

#FAMILY ALS

#PROGRAM
```

```
BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```

```
#NAME 4000,CD4000

#TEXT
Dual 3-Input NOR Gate
Plus Inverter

This device contains
two NOR gates each with
3 inputs and an inverter.

#PIN 14
 1 : N.C.
 2 : N.C.
 3 : Input   1 Gate   1
 4 : Input   2 Gate   1
 5 : Input   3 Gate   1
 6 : Output    Gate   1
 7 : GND
 8 : Input     Inverter
 9 : Output    Inverter
10 : Output    Gate   2
11 : Input   1 Gate   2
12 : Input   2 Gate   2
13 : Input   3 Gate   2
14 : Udd

#PROGRAM

BEGIN
PIN[3,4,5,8,11,12,13] : INPUT;
PIN[6,9,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[11]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[10]<>NOT(PIN[11] OR PIN[12] OR PIN[13]) THEN ERROR(1);
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]:=I;
   IF PIN[9]<>NOT(PIN[8]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4001,CD4001

#TEXT
Quad 2-Input NOR Gate

This device contains
four NOR gates each
with two inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
```

```
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4002,CD4002

#TEXT
Dual 4-Input NOR Gate

This device contains
two NOR gates each
with four inputs.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : N.C.
 7 : GND
 8 : N.C.
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : Input   3 Gate 2
12 : Input   4 Gate 2
13 : Output    Gate 2
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12] : INPUT;
PIN[1,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
```

```
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]&PIN[12]:=I;
   IF PIN[1]<>NOT(PIN[2] OR PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[13]<>NOT(PIN[9] OR PIN[10] OR PIN[11] OR PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4006,CD4006

#TEXT
18-Stage Shift Register

This device contains four
separate shift register
sections. Two of these are
four stages long and two
can be employed either as
a four stage or five stage
register.

#PIN 14
 1 : Input   A
 2 : N.C.
 3 : Clock
 c
 4 : Input   C
 5 : Input   B
 6 : Input   D
 7 : GND
 8 : Output  3D
 9 : Output  4D
10 : Output  3B
11 : Output  3C
12 : Output  4C
13 : Output  3A
14 : Udd
#PROGRAM

BEGIN
PIN[1,3,4,5,6] : INPUT;
PIN[8,9,10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[1,5]:=X AND 1;
      X:=X SHR 1;
      PIN[3]:=HIGH;PIN[3]:=LOW;
      END;
   X:=0;Y:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[13] SHL 3);
      Y:=(Y SHR 1) OR (PIN[10] SHL 3);
      PIN[3]:=HIGH;PIN[3]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   D:=D EXOR %1111;
   END;
```

```
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[4,6]:=X AND 1;
      X:=X SHR 1;
      PIN[3]:=HIGH;PIN[3]:=LOW;
      END;
   X:=0;Y:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      X:=(X SHR 1) OR (PIN[11] SHL 3);
      Y:=(Y SHR 1) OR (PIN[8] SHL 3);
      PIN[3]:=HIGH;PIN[3]:=LOW;
      END;
   IF (X<>D) OR (Y<>D) THEN ERROR(1);
   IF (PIN[12] SHL 3)<>(D AND 8) THEN ERROR(1);
   IF (PIN[9] SHL 3)<>(D AND 8) THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4008,CD4008

#TEXT
4-Bit Full Adder

This device consists of
four full-adder stages
with fast look-ahead
carry provision from
stage to stage.

#PIN 16
 1 : Input   A4
 2 : Input   B4
 3 : Input   A3
 4 : Input   B2
 5 : Input   A2
 6 : Input   B1
 7 : Input   A1
 8 : GND
 9 : Input   C0
10 : Output  ä1
11 : Output  ä2
12 : Output  ä1
13 : Output  ä4
14 : Output  C4
15 : Input   B4
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,15] : INPUT;
PIN[10,11,12,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR C:=0 TO 1 DO
   BEGIN
   PIN[9]:=C;
```

```
   FOR A:=0 TO 15 DO
      BEGIN
      PIN[1]&PIN[3]&PIN[5]&PIN[7]:=A;
      FOR B:=A MOD 3 TO 15 BY 3 DO
         BEGIN
         PIN[15]&PIN[2]&PIN[4]&PIN[6]:=B;
         IF (PIN[14]&PIN[13]&PIN[12]&PIN[11]&PIN[10])<>(C+A+B) THEN ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 4009,CD4009

#TEXT
Hex Inverting Buffer

This device contains
six separate inverting
buffers.

#PIN 16
 1 : +3V...+15V
 2 : Output    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 3
 7 : Input     Gate 3
 8 : GND
 9 : Input     Gate 4
10 : Output    Gate 4
11 : Input     Gate 5
12 : Output    Gate 5
13 : N.C.
14 : Input     Gate 6
15 : Output    Gate 6
16 : +3V...+15V

#PROGRAM

BEGIN
PIN[1,3,5,7,9,11,14] : INPUT;
PIN[2,4,6,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[10])&NOT(PIN[12])&NOT(PIN[15]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4010,CD4010

#TEXT
Hex Buffer
(Non-Inverting)
```

```
This device contains
six separate non-
inverting buffers.

#PIN 16
 1 : +3V...+15V
 2 : Output    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 3
 7 : Input     Gate 3
 8 : GND
 9 : Input     Gate 4
10 : Output    Gate 4
11 : Input     Gate 5
12 : Output    Gate 5
13 : N.C.
14 : Input     Gate 6
15 : Output    Gate 6
16 : +3V...+15V

#PROGRAM

BEGIN
PIN[1,3,5,7,9,11,14] : INPUT;
PIN[2,4,6,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14]:=D;
   IF (PIN[2]&PIN[4]&PIN[6]&
       PIN[10]&PIN[12]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4011,CD4011

#TEXT
Quad 2-Input NAND Gate

This device contains
four NAND gates each
with two inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
```

```
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NAND PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NAND PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4012,CD4012

#TEXT
Dual 4-Input NAND Gate

This device contains
two NAND gates each
with four inputs.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : N.C.
 7 : GND
 8 : N.C.
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : Input   3 Gate 2
12 : Input   4 Gate 2
13 : Output    Gate 2
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12] : INPUT;
PIN[1,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]&PIN[12]:=I;
   IF PIN[1]<>NOT(PIN[2] AND PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[13]<>NOT(PIN[9] AND PIN[10] AND PIN[11] AND PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
```

```
END.

#NAME 4013,CD4013

#TEXT
Dual D-Flip Flop with
Preset and Clear

This device consists of
two separate D-flip-
flops with triggering on
the positive edge of the
clock and independent
set and reset inputs.

#PIN 14
 1 : Output   Q  FF 1
 2 : Output  -Q  FF 1
 3 : Clock       FF 1
 4 : Reset       FF 1
 5 : Input    D  FF 1
 6 : Set         FF 1
 7 : GND
 8 : Set         FF 2
 9 : Input    D  FF 2
10 : Reset       FF 2
11 : Clock       FF 2
12 : Output  -Q  FF 2
13 : Output   Q  FF 2
14 : Udd

#PROGRAM

BEGIN
PIN[3,4,5,6,8,9,10,11] : INPUT;
PIN[1,2,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,6,8,10,11]:=LOW;

PIN[4,10]:=HIGH;PIN[4,10]:=LOW;
IF (PIN[1]<>LOW) OR (PIN[2]<>HIGH) THEN ERROR(1);
IF (PIN[13]<>LOW) OR (PIN[12]<>HIGH) THEN ERROR(1);

PIN[6,8]:=HIGH;PIN[6,8]:=LOW;
IF (PIN[1]<>HIGH) OR (PIN[2]<>LOW) THEN ERROR(1);
IF (PIN[13]<>HIGH) OR (PIN[12]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5,9]:=I;
   PIN[3,11]:=HIGH;PIN[3,11]:=LOW;
   IF (PIN[1]<>I) OR NOT(PIN[2]<>I) THEN ERROR(1);
   IF (PIN[13]<>I) OR NOT(PIN[12]<>I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4014,CD4014

#TEXT
8-Stage Shift Register
with Serial and Parallel
Input and Serial Output

This device can be
```

```
employed as a 6-, 7- or
8-stage shift register.
Data can be parallel or
serial input and serial
output.

#PIN 16
 1 : Input   P8
 2 : Output  Q6
 3 : Output  Q8
 4 : Input   P4
 5 : Input   P3
 6 : Input   P2
 7 : Input   P1
 8 : GND
 9 : P/-S
10 : Clock
11 : Serial In
12 : Output  Q7
13 : Input   P5
14 : Input   P6
15 : Input   P7
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,6,7,9,10,11,13,14,15] : INPUT;
PIN[2,3,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10]:=LOW;

D:=%01010101;

PIN[9]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[11]:=X AND 1;
      PIN[10]:=HIGH;PIN[10]:=LOW;
      X:=X SHR 1;
      END;
   IF (PIN[3]&PIN[12]&PIN[2])<>(D AND 7) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=HIGH;
   PIN[1]&PIN[15]&PIN[14]&PIN[13]&
   PIN[4]&PIN[5]&PIN[6]&PIN[7]:=D;
   PIN[10]:=HIGH;PIN[10]:=LOW;
   PIN[9]:=LOW;
   X:=0;
   FOR J:=0 TO 7 DO
      BEGIN
      X:=(X SHL 1) OR PIN[3];
      PIN[10]:=HIGH;PIN[10]:=LOW;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
```

```
END.

#NAME 4015,CD4015

#TEXT
Dual 4-Bit Shift
Register (serial input/
parallel output)

This device contains
two identical 4-stage
serial input/parallel
output shift registers
with independent data,
clock and reset inputs.

#PIN 16
 1 : Clock     SR B
 2 : Output  4 SR B
 3 : Output  3 SR A
 4 : Output  2 SR A
 5 : Output  1 SR A
 6 : Reset     SR A
 7 : Serial In SR A
 8 : GND
 9 : Clock     SR A
10 : Output  4 SR A
11 : Output  3 SR B
12 : Output  2 SR B
13 : Output  1 SR B
14 : Reset     SR B
15 : Serial In SR B
16 : Udd

#PROGRAM

BEGIN
PIN[1,6,7,9,14,15] : INPUT;
PIN[2,3,4,5,10,11,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,6,9,14]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[7,15]:=X AND 1;
      PIN[9,1]:=HIGH;PIN[9,1]:=LOW;
      X:=X SHR 1;
      END;
   IF (PIN[5]&PIN[4]&PIN[3]&PIN[10])<>D THEN ERROR(1);
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[6,14]:=HIGH;PIN[6,14]:=LOW;
IF (PIN[5]&PIN[4]&PIN[3]&PIN[10])<>0 THEN ERROR(1);
IF (PIN[13]&PIN[12]&PIN[11]&PIN[2])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4016,CD4016
```

```
#TEXT
Quad Bilateral Switch

This device contains
four separate bilateral
switches intended for the
transmission or multi-
plexing of analog or
digital signals.

#PIN 14
 1 : Input   Switch 1
 2 : Output  Switch 1
 3 : Input   Switch 2
 4 : Output  Switch 2
 5 : Control Switch 2
 6 : Control Switch 3
 7 : GND
 8 : Input   Switch 3
 9 : Output  Switch 3
10 : Input   Switch 4
11 : Output  Switch 4
12 : Control Switch 4
13 : Control Switch 1
14 : +3V...+15V

#PROGRAM

BEGIN
PIN[1,3,5,6,8,10,12,13] : Input  ;
PIN[2,4,9,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 1 DO
    BEGIN
    PIN[13,5,6,12]:=I;
    PIN[1,3,8,10]:=I;
    IF PIN[2]<>I THEN ERROR(1);
    IF PIN[4]<>I THEN ERROR(1);
    IF PIN[9]<>I THEN ERROR(1);
    IF PIN[11]<>I THEN ERROR(1);
    END;

ERROR(0);
END.

#NAME 4017,CD4017

#TEXT
Decade Counter/Divider
with 10 Decoded Outputs

This device is a synchro-
nous decade divide by 10
counter.

#PIN 16
 1 : Output  5
 2 : Output  1
 3 : Output  0
 4 : Output  2
 5 : Output  6
 6 : Output  7
 7 : Output  3
 8 : GND
 9 : Output  8
```

```
10 : Output  4
11 : Output  9
12 : Carry Out
13 : -Enable
14 : Clock
15 : Reset
16 : Udd

#PROGRAM

BEGIN
PIN[13,14,15] : INPUT;
PIN[1,2,3,4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[13,14,15]:=LOW;

PIN[15]:=HIGH;PIN[15]:=LOW;

FOR I:=0 TO 9 DO
   BEGIN
   IF (PIN[11]&PIN[9]&PIN[6]&PIN[5]&PIN[1]&
       PIN[10]&PIN[7]&PIN[4]&PIN[2]&PIN[3])<>(1 SHL I) THEN ERROR(1);
   IF (I<5) AND (PIN[12]<>HIGH) THEN ERROR(1);
   IF (I>=5) AND (PIN[12]<>LOW) THEN ERROR(1);
   PIN[14]:=HIGH;PIN[14]:=LOW;
   END;

PIN[15]:=HIGH;PIN[15]:=LOW;
IF (PIN[11]&PIN[9]&PIN[6]&PIN[5]&PIN[1]&
    PIN[10]&PIN[7]&PIN[4]&PIN[2]&PIN[3])<>1 THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 4018,CD4018

#TEXT
Presettable Divide-by-N
Counter

This device can divide the
input signal at the clock
by 2,3,4,.....10, depen-
ding on the configuration.
The output signal is
continually square wave.

#PIN 16
 1 : In
 2 : Input    P1
 3 : Input    P2
 4 : Output  -Q2
 5 : Output  -Q1
 6 : Output  -Q3
 7 : Input    P3
 8 : GND
 9 : Input    P4
10 : Load
11 : Output  -Q4
12 : Input    P5
13 : Output  -Q5
14 : Clock
15 : Reset
16 : Udd
```

```
#PROGRAM

BEGIN
PIN[1,2,3,7,9,10,12,14,15] : INPUT;
PIN[4,5,6,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10,14,15]:=LOW;

FOR I:=1 TO 5 DO
   BEGIN
   PIN[15]:=HIGH;PIN[15]:=LOW;
   FOR J:=1 DOWNTO 0 DO
      FOR K:=1 TO I DO
         BEGIN
         B:=((PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[5]) SHR (I-1)) AND 1;
         PIN[1]:=B;
         PIN[14]:=HIGH;PIN[14]:=LOW;
         IF B<>J THEN ERROR(1);
         END;
   END;

D:=%01010;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[12]&PIN[9]&PIN[7]&PIN[3]&PIN[2]:=D;
   PIN[10]:=HIGH;PIN[10]:=LOW;
   IF (NOT(PIN[13])&NOT(PIN[11])&NOT(PIN[6])&
       NOT(PIN[4])&NOT(PIN[5]))<>D THEN ERROR(1);
   D:=D EXOR %11111;
   END;

ERROR(0);
END.

#NAME 4019,CD4019

#TEXT
Quad AND/OR Select Gate

This device contains
four AND/OR select gates
with common select logic.
It is normally used as a
4-pole selector switch or
data selector.

#PIN 16
 1 : Input   B MP 4
 2 : Input   A MP 3
 3 : Output  B MP 3
 4 : Input   A MP 2
 5 : Input   B MP 2
 6 : Input   A MP 1
 7 : Input   B MP 1
 8 : GND
 9 : Select  A
10 : Output  Q MP 1
11 : Output  Q MP 2
12 : Output  Q MP 3
13 : Output  Q MP 4
14 : Select  B
15 : Input   A MP 4
16 : Udd

#PROGRAM
```

```
BEGIN
PIN[1,2,3,4,5,6,7,9,14,15] : INPUT;
PIN[10,11,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[6,7,4,5,2,3,15,1]:=LOW;
PIN[9]:=HIGH;PIN[14]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[4]&PIN[6]:=D;
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[6,7,4,5,2,3,15,1]:=LOW;
PIN[9]:=LOW;PIN[14]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[7]:=D;
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4020,CD4020

#TEXT
14-Stage Binary
Counter (:16384)

This binary counter
counts up asynchronously.

#PIN 16
 1 : Output  12
 2 : Output  13
 3 : Output  14
 4 : Output   6
 5 : Output   5
 6 : Output   7
 7 : Output   4
 8 : GND
 9 : Output   1
10 : -Clock
11 : Reset
12 : Output   9
13 : Output   8
14 : Output  10
15 : Output  11
16 : Udd

#PROGRAM

BEGIN
PIN[10,11] : INPUT;
PIN[1,2,3,4,5,6,7,9,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11]:=LOW;
PIN[10]:=HIGH;

PIN[11]:=HIGH;PIN[11]:=LOW;
```

```
WINDOW (31,11,48,14);
D:=0;

FOR I:=0 TO 13 DO
   BEGIN
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      PIN[10]:=LOW;PIN[10]:=HIGH;
      END;
   GOTOXY (I+2,1);WRITE ('Û');
   GOTOXY (I+2,2);WRITE ('Û');
   IF (I<>1) AND (I<>2) THEN
      BEGIN
      D:=(D SHL 1) OR 1;
      IF (PIN[3]&PIN[2]&PIN[1]&PIN[15]&PIN[14]&PIN[12]&
          PIN[13]&PIN[6]&PIN[4]&PIN[5]&PIN[7]&PIN[9])<>D THEN ERROR(1);
      END;
   END;

PIN[11]:=HIGH;PIN[11]:=LOW;
IF (PIN[3]&PIN[2]&PIN[1]&PIN[15]&PIN[14]&PIN[12]&
    PIN[13]&PIN[6]&PIN[4]&PIN[5]&PIN[7]&PIN[9])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4021,CD4021

#TEXT
8-Stage Shift Register
(Parallel-In/Serial-Out)

This device can be used
as a 6-, 7-, or 8-stage
shift register. Data is
serial or parallel input
and serial output.

#PIN 16
 1 : Input   P8
 2 : Output  Q6
 3 : Output  Q8
 4 : Input   P4
 5 : Input   P3
 6 : Input   P2
 7 : Input   P1
 8 : GND
 9 : Load P/-S
10 : Clock
11 : Serial In
12 : Output  Q7
13 : Input   P5
14 : Input   P6
15 : Input   P7
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,6,7,9,10,11,13,14,15] : INPUT;
PIN[2,3,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10]:=LOW;

D:=%01010101;
```

```
PIN[9]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[11]:=X AND 1;
      PIN[10]:=HIGH;PIN[10]:=LOW;
      X:=X SHR 1;
      END;
   IF (PIN[3]&PIN[12]&PIN[2])<>(D AND 7) THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=HIGH;
   PIN[1]&PIN[15]&PIN[14]&PIN[13]&
   PIN[4]&PIN[5]&PIN[6]&PIN[7]:=D;
   PIN[10]:=HIGH;PIN[10]:=LOW;
   PIN[9]:=LOW;
   X:=0;
   FOR J:=0 TO 7 DO
      BEGIN
      X:=(X SHL 1) OR PIN[3];
      PIN[10]:=HIGH;PIN[10]:=LOW;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 4022,CD4022

#TEXT
Divide-By-8 Counter/
Divider with 8 Decoded
Outputs

This is a 4-stage divide-
by-8 counter with 8
decoded inputs and a
carry-out bit.

#PIN 16
 1 : Output  1
 2 : Output  0
 3 : Output  2
 4 : Output  5
 5 : Output  6
 6 : N.C.
 7 : Output  3
 8 : GND
 9 : N.C.
10 : Output  7
11 : Output  4
12 : Carry Out
13 : -Enable
14 : Clock
15 : Reset
16 : Udd

#PROGRAM

BEGIN
```

```
PIN[13,14,15] : INPUT;
PIN[1,2,3,4,5,7,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[13,14,15]:=LOW;

PIN[15]:=HIGH;PIN[15]:=LOW;

FOR I:=0 TO 7 DO
   BEGIN
   IF (PIN[10]&PIN[5]&PIN[4]&PIN[11]&
       PIN[7]&PIN[3]&PIN[1]&PIN[2])<>(1 SHL I) THEN ERROR(1);
   IF (I<4) AND (PIN[12]<>HIGH) THEN ERROR(1);
   IF (I>=4) AND (PIN[12]<>LOW) THEN ERROR(1);
   PIN[14]:=HIGH;PIN[14]:=LOW;
   END;

PIN[15]:=HIGH;PIN[15]:=LOW;
IF (PIN[10]&PIN[5]&PIN[4]&PIN[11]&
    PIN[7]&PIN[3]&PIN[1]&PIN[2])<>1 THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 4023,CD4023

#TEXT
Triple 3-Input NAND Gate

This device contains
three NAND gates each
with three inputs.

#PIN 14
 1 : Input   1 Gate 2
 2 : Input   2 Gate 2
 3 : Input   1 Gate 1
 4 : Input   2 Gate 1
 5 : Input   3 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Input   3 Gate 2
 9 : Output    Gate 2
10 : Output    Gate 3
11 : Input   1 Gate 3
12 : Input   2 Gate 3
13 : Input   3 Gate 3
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,8,11,12,13] : INPUT;
PIN[6,9,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[1]&PIN[2]&PIN[8]:=I;
   PIN[11]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[9]<>NOT(PIN[1] AND PIN[2] AND PIN[8]) THEN ERROR(1);
   IF PIN[10]<>NOT(PIN[11] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;
```

```
ERROR(0);
END.

#NAME 4024,CD4024

#TEXT
7-Stage Binary Counter

This device contains
a binary counter which
counts up asynchronously.

#PIN 14
 1 : Clock
 2 : Reset
 3 : Output  7
 4 : Output  6
 5 : Output  5
 6 : Output  4
 7 : GND
 8 : N.C.
 9 : Output  3
10 : N.C.
11 : Output  2
12 : Output  1
13 : N.C.
14 : Udd

#PROGRAM

BEGIN
PIN[1,2] : INPUT;
PIN[3,4,5,6,9,11,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[2]:=LOW;
PIN[1]:=HIGH;

PIN[2]:=HIGH;PIN[2]:=LOW;

D:=0;
FOR I:=0 TO 6 DO
   BEGIN
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      PIN[1]:=LOW;PIN[1]:=HIGH;
      END;
      D:=(D SHL 1) OR 1;
      IF (PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[9]&
          PIN[11]&PIN[12])<>D THEN ERROR(1);
      END;
   END;

PIN[2]:=HIGH;PIN[2]:=LOW;
IF (PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[9]&
    PIN[11]&PIN[12])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4025,CD4025

#TEXT
Triple 3-Input NOR Gate

This device contains
```

```
three NOR gates each
with three inputs.

#PIN 14
 1 : Input   1 Gate 2
 2 : Input   2 Gate 2
 3 : Input   1 Gate 1
 4 : Input   2 Gate 1
 5 : Input   3 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Input   3 Gate 2
 9 : Output    Gate 2
10 : Output    Gate 3
11 : Input   1 Gate 3
12 : Input   2 Gate 3
13 : Input   3 Gate 3
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,8,11,12,13] : INPUT;
PIN[6,9,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[1]&PIN[2]&PIN[8]:=I;
   PIN[11]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>NOT(PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[9]<>NOT(PIN[1] OR PIN[2] OR PIN[8]) THEN ERROR(1);
   IF PIN[10]<>NOT(PIN[11] OR PIN[12] OR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4026,CD4026

#TEXT
Decade Counter/Divider
with 7-Segment Decoded
Outputs

This synchronous decade
divide-by-10 decade
counter provides inter-
nal decoding for con-
trolling a 7-segment
display.

#PIN 16
 1 : Clock
 2 : Clock Enable
 3 : Display Enable
 4 : Enable Out
 5 : Out ö10
 6 : 7-Seg.-Output  f
 7 : 7-Seg.-Output  g
 8 : GND
 9 : 7-Seg.-Output  d
10 : 7-Seg.-Output  a
11 : 7-Seg.-Output  e
12 : 7-Seg.-Output  b
```

```
13 : 7-Seg.-Output  c
14 : Out 2
15 : Reset
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,15] : INPUT;
PIN[4,5,6,7,9,10,11,12,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,2,15]:=LOW;
PIN[3]:=HIGH;

IF PIN[4]<>HIGH THEN ERROR(1);

PIN[15]:=HIGH;PIN[15]:=LOW;
FOR I:=0 TO 9 DO
   BEGIN
   D:=PIN[7]&PIN[6]&PIN[11]&PIN[9]&PIN[13]&PIN[12]&PIN[10];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I<5) AND (PIN[5]<>HIGH) THEN ERROR(1);
   IF (I>=5) AND (PIN[5]<>LOW) THEN ERROR(1);
   IF (I=2) AND (PIN[14]<>LOW) THEN ERROR(1);
   IF (I<>2) AND (PIN[14]<>HIGH) THEN ERROR(1);
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;

ERROR(0);
END.

#NAME 4027,CD4027

#TEXT
Dual J-K Flip-Flop with
Set and Clear

This device consists of
two separate J-K flip-
flops with preset and
clear.

#PIN 16
 1 : Output   Q  FF 2
 2 : Output  -Q  FF 2
 3 : Clock       FF 2
 4 : Reset       FF 2
 5 : Input    K  FF 2
 6 : Input    J  FF 2
 7 : Set         FF 2
 8 : GND
 9 : Set         FF 1
10 : Input    J  FF 1
11 : Input    K  FF 1
12 : Reset      FF 1
13 : Clock      FF 1
14 : Output  -Q  FF 1
```

```
15 : Output   Q  FF 1
16 : Udd

#PROGRAM

BEGIN
PIN[3,4,5,6,7,9,10,11,12,13] : INPUT;
PIN[1,2,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5,6,7,9,10,11,12,13]:=LOW;

PIN[4,12]:=HIGH;PIN[4,12]:=LOW;
IF (PIN[1]<>LOW) OR (PIN[2]<>HIGH) THEN ERROR(1);
IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);

PIN[7,9]:=HIGH;PIN[7,9]:=LOW;
IF (PIN[1]<>HIGH) OR (PIN[2]<>LOW) THEN ERROR(1);
IF (PIN[15]<>HIGH) OR (PIN[14]<>LOW) THEN ERROR(1);

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[5,11]:=I;
   PIN[3,13]:=HIGH;PIN[3,13]:=LOW;
   IF (PIN[1]<>LOW) OR (PIN[2]<>HIGH) THEN ERROR(1);
   IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);
   END;

FOR I:=1 DOWNTO 0 DO
   BEGIN
   PIN[6,10]:=I;
   PIN[3,13]:=HIGH;PIN[3,13]:=LOW;
   IF (PIN[1]<>HIGH) OR (PIN[2]<>LOW) THEN ERROR(1);
   IF (PIN[15]<>HIGH) OR (PIN[14]<>LOW) THEN ERROR(1);
   END;

PIN[6,5,10,11]:=HIGH;
PIN[3,13]:=HIGH;PIN[3,13]:=LOW;
IF (PIN[1]<>LOW) OR (PIN[2]<>HIGH) THEN ERROR(1);
IF (PIN[15]<>LOW) OR (PIN[14]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 4028,CD4028

#TEXT
BCD-to-Decimal Decoder
(1-of-10)

This device converts
a standard BCD code into
a decimal number between
0 and 9.

#PIN 16
 1 : Output  4
 2 : Output  2
 3 : Output  0
 4 : Output  7
 5 : Output  9
 6 : Output  5
 7 : Output  6
 8 : GND
 9 : Output  9
10 : Input   A
11 : Input   D
```

```
12 : Input   C
13 : Input   B
14 : Output  1
15 : Output  3
16 : Udd

#PROGRAM

BEGIN
PIN[10,11,12,13] : INPUT;
PIN[1,2,3,4,5,6,7,9,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[10]:=I;
   IF (PIN[5]&PIN[9]&PIN[4]&PIN[7]&PIN[6]&
       PIN[1]&PIN[15]&PIN[2]&PIN[14]&PIN[3])<>(1 SHL I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4029,CD4029

#TEXT
Presettable Binary/Decade/
Up/Down Counter

This device contains a
synchronous, presettable
binary or decade up/down
counter.

#PIN 16
 1 : Load
 2 : Output  QD
 3 : Input   PD
 4 : Input   PA
 5 : -CE/CI
 6 : Output  QA
 7 : Carry Out
 8 : GND
 9 : Binary/-BCD
10 : Up/-Down
11 : Output  QB
12 : Input   PB
13 : Input   PC
14 : Output  QC
15 : Clock
16 : Udd

#PROGRAM

BEGIN
PIN[1,3,4,5,9,10,12,13,15] : INPUT;
PIN[2,6,7,11,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,5,15]:=LOW;

PIN[1]:=HIGH;
PIN[3]&PIN[13]&PIN[12]&PIN[4]:=0;
PIN[15]:=HIGH;PIN[15]:=LOW;
PIN[1]:=LOW;
```

```
PIN[9]:=HIGH;

PIN[10]:=HIGH;
FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

PIN[10]:=LOW;
FOR I:=15 DOWNTO 1 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

PIN[9]:=LOW;

PIN[10]:=HIGH;
FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

PIN[10]:=LOW;
FOR I:=9 DOWNTO 1 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]:=HIGH;
   PIN[3]&PIN[13]&PIN[12]&PIN[4]:=D;
   PIN[15]:=HIGH;PIN[15]:=LOW;
   PIN[1]:=HIGH;
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4030,CD4030

#TEXT
Quad 2-Input EXOR Gate

This device contains
four EXOR gates each
with two inputs.
```

```
#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] EXOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] EXOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] EXOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] EXOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4031,CD4031

#TEXT
64-Stage Static Shift
Register with Serial
Input and Output and
Recirculation

This device contains a
fully static shift
register with recirculate
input.

#PIN 16
 1 : REC IN
 2 : Clock In
 3 : N.C.
 4 : N.C.
 5 : N.C.
 6 : Output   Q
 7 : Output  -Q
 8 : GND
 9 : Clock Out
10 : Mode Control
11 : N.C.
12 : N.C.
13 : N.C.
14 : N.C.
```

```
15 : Data In
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,10,15] : INPUT;
PIN[6,7] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

D:=%01010101;

PIN[10]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 7 DO
      BEGIN
      X:=D;
      FOR K:=0 TO 7 DO
         BEGIN
         PIN[15]:=X AND 1;
         PIN[2]:=HIGH;PIN[2]:=LOW;
         X:=X SHR 1;
         END;
      END;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      X:=0;Y:=0;
      FOR K:=7 DOWNTO 0 DO
         BEGIN
         X:=(X SHR 1) OR (PIN[6] SHL 7);
         Y:=(Y SHR 1) OR (NOT(PIN[7]) SHL 7);
         PIN[2]:=HIGH;PIN[2]:=LOW;
         END;
      IF X<>D THEN ERROR(1);
      IF Y<>D THEN ERROR(1);
      END;
   D:=D EXOR %11111111;
   END;

PIN[10]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 7 DO
      BEGIN
      X:=D;
      FOR K:=0 TO 7 DO
         BEGIN
         PIN[1]:=X AND 1;
         PIN[2]:=HIGH;PIN[2]:=LOW;
         X:=X SHR 1;
         END;
      END;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      X:=0;Y:=0;
      FOR K:=7 DOWNTO 0 DO
         BEGIN
         X:=(X SHR 1) OR (PIN[6] SHL 7);
         Y:=(Y SHR 1) OR (NOT(PIN[7]) SHL 7);
         PIN[2]:=HIGH;PIN[2]:=LOW;
         END;
      IF X<>D THEN ERROR(1);
      IF Y<>D THEN ERROR(1);
      END;
   D:=D EXOR %11111111;
```

```
   END;

PIN[2]:=HIGH;IF PIN[9]<>HIGH THEN ERROR(1);
PIN[2]:=LOW;IF PIN[9]<>LOW THEN ERROR(1);

ERROR(0);
END.

#NAME 4032,CD4032

#TEXT
Triple Serial Adder
(positive logic)

This device consists of
three separate adders
which carry out binary
addition of sequential
words of every length.

#PIN 16
 1 : Output  ä Add.3
 2 : Input   I Add.3
 3 : Clock
 4 : Output  ä Add.2
 5 : Input   I Add.2
 6 : Carry Reset
 7 : Input   I Add.1
 8 : GND
 9 : Output  ä Add.1
10 : Input   A Add.1
11 : Input   B Add.1
12 : Input   B Add.2
13 : Input   A Add.2
14 : Input   B Add.3
15 : Input   A Add.3
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,5,6,7,10,11,12,13,14,15] : INPUT;
PIN[1,4,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,6]:=LOW;

PIN[6]:=HIGH;PIN[3]:=HIGH;PIN[3]:=LOW;PIN[6]:=LOW;

PIN[2,5,7]:=LOW;
FOR A:=1 DOWNTO 0 DO
   BEGIN
   PIN[15,13,10]:=A;
   FOR B:=1 DOWNTO 0 DO
      BEGIN
      PIN[14,12,11]:=B;
      PIN[3]:=HIGH;PIN[3]:=LOW;
      PIN[2,5,7]:=LOW;
      IF PIN[1]<>(A EXOR B) THEN ERROR(1);
      IF PIN[4]<>(A EXOR B) THEN ERROR(1);
      IF PIN[9]<>(A EXOR B) THEN ERROR(1);
      PIN[2,5,7]:=HIGH;
      IF PIN[1]<>(A NEXOR B) THEN ERROR(1);
      IF PIN[4]<>(A NEXOR B) THEN ERROR(1);
      IF PIN[9]<>(A NEXOR B) THEN ERROR(1);
      END;
   END;
```

```
PIN[3]:=HIGH;PIN[3]:=LOW;
PIN[2,5,7]:=LOW;
IF PIN[1]<>LOW THEN ERROR(1);
IF PIN[4]<>LOW THEN ERROR(1);
IF PIN[9]<>LOW THEN ERROR(1);

ERROR(0);
END.

#NAME 4033,CD4033

#TEXT
Decade Counter/Divider
with 7-Segment Decoded
Outputs (synchronous)

This synchronous decade
divide-by-10 counter
provides internal deco-
ding for controlling a
7-segment display.

#PIN 16
 1 : Clock
 2 : Clock Enable
 3 : RB In
 4 : RB Out
 5 : Out ö10
 6 : 7-Seg.-Output  f
 7 : 7-Seg.-Output  g
 8 : GND
 9 : 7-Seg.-Output  d
10 : 7-Seg.-Output  a
11 : 7-Seg.-Output  e
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  c
14 : Test
15 : Reset
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,14,15] : INPUT;
PIN[6,7,9,10,11,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,2,14,15]:=LOW;
PIN[3]:=HIGH;

PIN[15]:=HIGH;PIN[15]:=LOW;
FOR I:=0 TO 9 DO
   BEGIN
   D:=PIN[7]&PIN[6]&PIN[11]&PIN[9]&PIN[13]&PIN[12]&PIN[10];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I<5) AND (PIN[5]<>HIGH) THEN ERROR(1);
   IF (I>=5) AND (PIN[5]<>LOW) THEN ERROR(1);
```

```
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;

ERROR(0);
END.

#NAME 4035,CD4035

#TEXT
4-Bit Parallel-In/Parallel
Out Shift Register

This device contains a
4-bit right shift register
with parallel input and
output.

#PIN 16
 1 : Output   Q1
 2 : Compl.
 3 : Input   -K
 4 : Input    J
 5 : Reset
 6 : Clock
 7 : Load P/S
 8 : GND
 9 : Input    P1
10 : Input    P2
11 : Input    P3
12 : Input    P4
13 : Output   Q4
14 : Output   Q3
15 : Output   Q2
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7,9,10,11,12] : INPUT;
PIN[1,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[5,6,7]:=LOW;

D:=%0101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[3,4]:=NOT(X AND 1);
      PIN[6]:=HIGH;PIN[6]:=LOW;
      X:=X SHR 1;
      END;
   PIN[2]:=LOW;
   IF (PIN[1]&PIN[15]&PIN[14]&PIN[13])<>D THEN ERROR(1);
   PIN[2]:=HIGH;
   IF (NOT(PIN[1])&NOT(PIN[15])&NOT(PIN[14])&NOT(PIN[13]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[7]:=HIGH;
   PIN[12]&PIN[11]&PIN[10]&PIN[9]:=D;
   PIN[6]:=HIGH;PIN[6]:=LOW;
```

```
   PIN[7]:=LOW;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2]:=LOW;
      IF (PIN[1]&PIN[15]&PIN[14]&PIN[13])<>X THEN ERROR(1);
      PIN[2]:=HIGH;
      IF (NOT(PIN[1])&NOT(PIN[15])&NOT(PIN[14])&NOT(PIN[13]))<>X THEN
      ERROR(1);
      PIN[3,4]:=HIGH;
      PIN[6]:=HIGH;PIN[6]:=LOW;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

PIN[5]:=HIGH;PIN[5]:=LOW;
PIN[2]:=LOW;
IF (NOT(PIN[1])&NOT(PIN[15])&NOT(PIN[14])&NOT(PIN[13]))<>0 THEN ERROR(1);
PIN[2]:=HIGH;
IF (PIN[1]&PIN[15]&PIN[14]&PIN[13])<>0 THEN ERROR(1);

ERROR(0)
END.

#NAME 4037,CD4037

#TEXT
Triple AND/OR Pair

This device consists of
three AND/OR pairs with
common control inputs
and separate data inputs.

#PIN 14
 1 : Ucc
 2 : Input   B
 3 : Data      Gate 1
 4 : Input   A
 5 : Data      Gate 2
 6 : Data      Gate 3
 7 : GND
 8 : Output  D Gate 3
 9 : Output  E Gate 3
10 : Output  D Gate 2
11 : Output  E Gate 2
12 : Output  E Gate 1
13 : Output  D Gate 1
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6] : INPUT;
PIN[8,9,10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1]:=HIGH;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[2,4]:=I;
   IF (NOT(PIN[8])<>I) OR (NOT(PIN[9])<>I) THEN ERROR(1);
   IF (NOT(PIN[10])<>I) OR (NOT(PIN[11])<>I) THEN ERROR(1);
   IF (NOT(PIN[13])<>I) OR (NOT(PIN[12])<>I) THEN ERROR(1);
   END;
```

```
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3,5,6]:=I;
   PIN[4]:=HIGH;PIN[2]:=LOW;
   IF (PIN[8]<>I) OR (NOT(PIN[9])<>I) THEN ERROR(1);
   IF (PIN[10]<>I) OR (NOT(PIN[11])<>I) THEN ERROR(1);
   IF (PIN[13]<>I) OR (NOT(PIN[12])<>I) THEN ERROR(1);
   PIN[4]:=LOW;PIN[2]:=HIGH;
   IF (NOT(PIN[8])<>I) OR (PIN[9]<>I) THEN ERROR(1);
   IF (NOT(PIN[10])<>I) OR (PIN[11]<>I) THEN ERROR(1);
   IF (NOT(PIN[13])<>I) OR (PIN[12]<>I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4038,CD4038

#TEXT
Triple Serial Adder
(negative logic)

This device contains
three separate adders
which carry out binary
addition of sequential
words of every length.

#PIN 16
 1 : Output  ä Add.3
 2 : Input   I Add.3
 3 : Clock
 4 : Output  ä Add.2
 5 : Input   I Add.2
 6 : Carry Reset
 7 : Input   I Add.1
 8 : GND
 9 : Output  ä Add.1
10 : Input   A Add.1
11 : Input   B Add.1
12 : Input   B Add.2
13 : Input   A Add.2
14 : Input   B Add.3
15 : Input   A Add.3
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,5,6,7,10,11,12,13,14,15] : INPUT;
PIN[1,4,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,6]:=LOW;

PIN[6]:=HIGH;PIN[3]:=HIGH;PIN[3]:=LOW;PIN[6]:=LOW;

PIN[2,5,7]:=LOW;
FOR A:=1 DOWNTO 0 DO
   BEGIN
   PIN[15,13,10]:=A;
   FOR B:=1 DOWNTO 0 DO
      BEGIN
      PIN[14,12,11]:=B;
      PIN[3]:=HIGH;PIN[3]:=LOW;
      PIN[2,5,7]:=LOW;
```

```
      IF PIN[1]<>(A EXOR B) THEN ERROR(1);
      IF PIN[4]<>(A EXOR B) THEN ERROR(1);
      IF PIN[9]<>(A EXOR B) THEN ERROR(1);
      PIN[2,5,7]:=HIGH;
      IF PIN[1]<>(A NEXOR B) THEN ERROR(1);
      IF PIN[4]<>(A NEXOR B) THEN ERROR(1);
      IF PIN[9]<>(A NEXOR B) THEN ERROR(1);
      END;
   END;

PIN[3]:=HIGH;PIN[3]:=LOW;
PIN[2,5,7]:=LOW;
IF PIN[1]<>LOW THEN ERROR(1);
IF PIN[4]<>LOW THEN ERROR(1);
IF PIN[9]<>LOW THEN ERROR(1);

ERROR(0);
END.
•
```

```
#NAME 4040,CD4040

#TEXT
12-Stage Binary Counter
(:4096)

This device contains a
binary counter which
counts up asynchronously.

#PIN 16
 1 : Output  12
 2 : Output   6
 3 : Output   5
 4 : Output   7
 5 : Output   4
 6 : Output   3
 7 : Output   2
 8 : GND
 9 : Output   1
10 : -Clock
11 : Reset
12 : Output   9
13 : Output   8
14 : Output  10
15 : Output  11
16 : Udd

#PROGRAM

BEGIN
PIN[10,11] : INPUT;
PIN[1,2,3,4,5,6,7,9,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11]:=LOW;
PIN[10]:=HIGH;

PIN[11]:=HIGH;PIN[11]:=LOW;
WINDOW (32,11,47,14);
D:=0;

FOR I:=0 TO 11 DO
   BEGIN
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      PIN[10]:=LOW;PIN[10]:=HIGH;
      END;
   GOTOXY (I+2,1);WRITE ('Û');
   GOTOXY (I+2,2);WRITE ('Û');
   D:=(D SHL 1) OR 1;
   IF (PIN[1]&PIN[15]&PIN[14]&PIN[12]&PIN[13]&PIN[4]&
       PIN[2]&PIN[3]&PIN[5]&PIN[6]&PIN[7]&PIN[9])<>D THEN
ERROR(1);
   END;

PIN[11]:=HIGH;PIN[11]:=LOW;
IF (PIN[1]&PIN[15]&PIN[14]&PIN[12]&PIN[13]&PIN[4]&
    PIN[2]&PIN[3]&PIN[5]&PIN[6]&PIN[7]&PIN[9])<>0 THEN
ERROR(1);

ERROR(0);
END.

#NAME 4041,CD4041

#TEXT
```

```
Quad TTL-Buffer

This device contains
four separate buffers
having inverting and
non-inverting outputs.

#PIN 14
 1 : Output   Q Buffer 1
 2 : Output  -Q Buffer 1
 3 : Input      Buffer 1
 4 : Output   Q Buffer 2
 5 : Output  -Q Buffer 2
 6 : Input      Buffer 2
 7 : GND
 8 : Output   Q Buffer 3
 9 : Output  -Q Buffer 3
10 : Input      Buffer 3
11 : Output   Q Buffer 4
12 : Output  -Q Buffer 4
13 : Input      Buffer 4
14 : +3V...+15V

#PROGRAM

BEGIN
PIN[3,6,10,13] : INPUT;
PIN[1,2,4,5,8,9,11,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[6]&PIN[10]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[5])&NOT(PIN[9])&NOT(PIN[12]))<>D
THEN ERROR(1);
   IF (PIN[1]&PIN[4]&PIN[8]&PIN[11])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4042,CD4042

#TEXT
Quad D Latch

This device contains
four latches with a
common level-controlled
command input.

#PIN 16
 1 : Output   Q4
 2 : Output   Q1
 3 : Output  -Q1
 4 : Input    D1
 5 : Store
 6 : Pol
 7 : Input    D2
 8 : GND
 9 : Output  -Q2
10 : Output   Q2
11 : Output   Q3
12 : Output  -Q3
```

```
13 : Input    D3
14 : Input    D4
15 : Output  -Q4
16 : Udd

#PROGRAM

BEGIN
PIN[4,5,6,7,13,14] : INPUT;
PIN[1,2,3,9,10,11,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

D:=%0101;

PIN[5]:=LOW;PIN[6]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[13]&PIN[7]&PIN[4]:=D;
   PIN[5]:=HIGH;PIN[5]:=LOW;
   PIN[14]&PIN[13]&PIN[7]&PIN[4]:=0;
   IF (PIN[1]&PIN[11]&PIN[10]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[15])&NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[5]:=HIGH;PIN[6]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[13]&PIN[7]&PIN[4]:=D;
   PIN[5]:=LOW;PIN[5]:=HIGH;
   PIN[14]&PIN[13]&PIN[7]&PIN[4]:=0;
   IF (PIN[1]&PIN[11]&PIN[10]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[15])&NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4043,CD4043

#TEXT
Quad TRI-STATE NOR R/S
Flip-Flops

This device contains four
independent set/reset
flip-flops having a common
three-state enable input.

#PIN 16
 1 : Output   Q FF 4
 2 : Output   Q FF 1
 3 : Input    R FF 1
 4 : Input    S FF 1
 5 : Enable
 6 : Input    S FF 2
 7 : Input    R FF 2
 8 : GND
 9 : Output   Q FF 2
10 : Output   Q FF 3
11 : Input    R FF 3
12 : Input    S FF 3
13 : N.C.
14 : Input    S FF 4
15 : Input    R FF 4
```

```
16 : Udd

#PROGRAM

BEGIN
PIN[3,4,5,6,7,11,12,14,15] : INPUT;
PIN[1,2,9,10] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[5]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[12]&PIN[6]&PIN[4]:=D;
   PIN[15]&PIN[11]&PIN[7]&PIN[3]:=D EXOR %1111;
   PIN[5]:=HIGH;
   IF (PIN[1]&PIN[10]&PIN[9]&PIN[2])<>D THEN ERROR(1);
   PIN[5]:=LOW;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[1,10,9,2] : LOAD LOW;
IF (PIN[1]&PIN[10]&PIN[9]&PIN[2])<>%0000 THEN ERROR(1);
PIN[1,10,9,2] : LOAD HIGH;
IF (PIN[1]&PIN[10]&PIN[9]&PIN[2])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4044,CD4044

#TEXT
Quad TRI-STATE NAND R/S
Flip-Flops

This device contains four
independent set/reset
flip-flops having a common
three-state enable input.

#PIN 16
 1 : Output   Q FF 4
 2 : N.C.
 3 : Input    S FF 1
 4 : Input    R FF 1
 5 : Enable
 6 : Input    R FF 2
 7 : Input    S FF 2
 8 : GND
 9 : Output   Q FF 2
10 : Output   Q FF 3
11 : Input    S FF 3
12 : Input    R FF 3
13 : Output   Q FF 1
14 : Input    R FF 4
15 : Input    S FF 4
16 : Udd

#PROGRAM

BEGIN
PIN[3,4,5,6,7,11,12,14,15] : INPUT;
PIN[1,9,10,13] : OUTPUT;
PIN[8] : GND;
```

```
PIN[16] : +5V;
PIN[5]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[11]&PIN[7]&PIN[3]:=D EXOR %1111;
   PIN[14]&PIN[12]&PIN[6]&PIN[4]:=D;
   PIN[5]:=HIGH;
   IF (PIN[1]&PIN[10]&PIN[9]&PIN[13])<>D THEN ERROR(1);
   PIN[5]:=LOW;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[1,10,9,13] : LOAD LOW;
IF (PIN[1]&PIN[10]&PIN[9]&PIN[13])<>%0000 THEN ERROR(1);
PIN[1,10,9,13] : LOAD HIGH;
IF (PIN[1]&PIN[10]&PIN[9]&PIN[13])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4048,CD4048

#TEXT
8-Function 8-Input Gate

This device consists of
a programmable gate with
eight inputs. Three
control lines determine
the eight different logic
functions of the gate.

#PIN 16
 1 : Output  J
 2 : Tristate-Stg.  Kd
 3 : Input   H
 4 : Input   G
 5 : Input   F
 6 : Input   E
 7 : Functions-Stg. Kb
 8 : GND
 9 : Functions-Stg. Kc
10 : Functions-Stg. Ka
11 : Input   D
12 : Input   C
13 : Input   B
14 : Input   A
15 : Expand Input
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7,9,10,11,12,13,14,15] : INPUT;
PIN[1] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;

PIN[10]&PIN[7]&PIN[9]:=1;
FOR I:=0 TO 63 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[11]&PIN[12]&PIN[13]&PIN[14]:=I;
```

```
   PIN[2]:=HIGH;
   IF PIN[1]<>(PIN[3] OR PIN[4] OR PIN[5] OR PIN[6] OR
               PIN[11] OR PIN[12] OR PIN[13] OR PIN[14]) THEN ERROR(1);
   PIN[2]:=LOW;
   END;

PIN[10]&PIN[7]&PIN[9]:=2;
FOR I:=0 TO 63 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[11]&PIN[12]&PIN[13]&PIN[14]:=I;
   PIN[2]:=HIGH;
   IF PIN[1]<>((PIN[3] OR PIN[4] OR PIN[5] OR PIN[6]) AND
               (PIN[11] OR PIN[12] OR PIN[13] OR PIN[14])) THEN ERROR(1);
   PIN[2]:=LOW;
   END;

PIN[10]&PIN[7]&PIN[9]:=4;
FOR I:=0 TO 63 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[11]&PIN[12]&PIN[13]&PIN[14]:=I;
   PIN[2]:=HIGH;
   IF PIN[1]<>(PIN[3] AND PIN[4] AND PIN[5] AND PIN[6] AND
               PIN[11] AND PIN[12] AND PIN[13] AND PIN[14]) THEN ERROR(1);
   PIN[2]:=LOW;
   END;

LOADMODEON;
PIN[1] : LOAD LOW;
IF PIN[1]<>LOW THEN ERROR(1);
PIN[1] : LOAD HIGH;
IF PIN[1]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4049,CD4049

#TEXT
Hex Inverting Buffer
and TTL Driver

This device consists of
six separate, inverting
buffers which can be used
as a driver for inter-
facing with TTL or other
logic.

#PIN 16
 1 : +3V...+15V
 2 : Output    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 3
 7 : Input     Gate 3
 8 : GND
 9 : Input     Gate 4
10 : Output    Gate 4
11 : Input     Gate 5
12 : Output    Gate 5
13 : N.C.
14 : Input     Gate 6
15 : Output    Gate 6
16 : +3V...+15V
```

```
#PROGRAM

BEGIN
PIN[1,3,5,7,9,11,14] : INPUT;
PIN[2,4,6,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[10])&NOT(PIN[12])&NOT(PIN[15]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4050,CD4050

#TEXT
Hex Non-Inverting Buffer
and TTL Driver

This device contains six
separate, non-inverting
buffers which can be used
as a driver for inter-
facing with TTL or other
logic.

#PIN 16
 1 : +3V...+15V
 2 : Output    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 3
 7 : Input     Gate 3
 8 : GND
 9 : Input     Gate 4
10 : Output    Gate 4
11 : Input     Gate 5
12 : Output    Gate 5
13 : N.C.
14 : Input     Gate 6
15 : Output    Gate 6
16 : +3V...+15V

#PROGRAM

BEGIN
PIN[1,3,5,7,9,11,14] : INPUT;
PIN[2,4,6,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14]:=D;
   IF (PIN[2]&PIN[4]&PIN[6]&
       PIN[10]&PIN[12]&PIN[15])<>D THEN ERROR(1);
```

```
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4051,CD4051

#TEXT
8-Channel Analog Multi-
plexer/Demultiplexer

This device can be
employed as a 1-of-8
multiplexer or demulti-
plexer or as a 1-of-8
digital selector or
distributor.

#PIN 16
 1 : In-/Output  4
 2 : In-/Output  6
 3 : In/Out
 4 : In-/Output  7
 5 : In-/Output  5
 6 : Inhibit
 7 : Digital/Analog
 8 : GND
 9 : Address C
10 : Address B
11 : Address A
12 : In-/Output  3
13 : In-/Output  0
14 : In-/Output  1
15 : In-/Output  2
16 : Udd

#PROGRAM

BEGIN
PIN[6,7,9,10,11] : INPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[6,7]:=LOW;

PIN[3] : INPUT;
PIN[1,2,4,5,12,13,14,15] : OUTPUT;
FOR I:=0 TO 7 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[11]:=I;
   PIN[3]:=LOW;
   IF (((PIN[4]&PIN[2]&PIN[5]&PIN[1]&
         PIN[12]&PIN[15]&PIN[14]&PIN[13]) SHR I) AND 1)<>LOW THEN ERROR(1);
   PIN[3]:=HIGH;
   IF (((PIN[4]&PIN[2]&PIN[5]&PIN[1]&
         PIN[12]&PIN[15]&PIN[14]&PIN[13]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   END;

PIN[1,2,4,5,12,13,14,15] : INPUT;
PIN[3] : OUTPUT;
FOR I:=0 TO 7 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[11]:=I;
   PIN[4]&PIN[2]&PIN[5]&PIN[1]&PIN[12]&PIN[15]&PIN[14]&PIN[13]:=0;
   IF PIN[3]<>LOW THEN ERROR(1);
   PIN[4]&PIN[2]&PIN[5]&PIN[1]&PIN[12]&PIN[15]&PIN[14]&PIN[13]:=1 SHL I;
   IF PIN[3]<>HIGH THEN ERROR(1);
```

```
   END;

ERROR(0);
END.

#NAME 4052,CD4052

#TEXT
Dual 4-Channel Analog
Multiplexer/Demultiplexer

This device can be
employed as a 1-of-4
analog data multiplexer
or demultiplexer with
common address, or as
a dual 1-of-4 digital
selector or distributor.

#PIN 16
 1 : In-/Output  Y0
 2 : In-/Output  Y2
 3 : In/Out       Y
 4 : In-/Output  Y3
 5 : In-/Output  Y1
 6 : Inhibit
 7 : Digital/Analog
 8 : GND
 9 : Address B
10 : Address A
11 : In-/Output  X3
12 : In-/Output  X0
13 : In/Out       X
14 : In-/Output  X1
15 : In-/Output  X2
16 : Udd

#PROGRAM

BEGIN
PIN[6,7,9,10] : INPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[6,7]:=LOW;

PIN[3,13] : INPUT;
PIN[1,2,4,5,11,12,14,15] : OUTPUT;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   PIN[3,13]:=LOW;
   IF (((PIN[4]&PIN[2]&PIN[5]&PIN[1]) SHR I) AND 1)<>LOW THEN ERROR(1);
   IF (((PIN[3]&PIN[15]&PIN[14]&PIN[12]) SHR I) AND 1)<>LOW THEN ERROR(1);
   PIN[3,13]:=HIGH;
   IF (((PIN[4]&PIN[2]&PIN[5]&PIN[1]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   IF (((PIN[11]&PIN[15]&PIN[14]&PIN[12]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   END;

PIN[1,2,4,5,11,12,14,15] : INPUT;
PIN[3,13] : OUTPUT;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[9]&PIN[10]:=I;
   PIN[4]&PIN[2]&PIN[5]&PIN[1]:=0;
   PIN[11]&PIN[15]&PIN[14]&PIN[12]:=0;
   IF PIN[3]<>LOW THEN ERROR(1);
   IF PIN[13]<>LOW THEN ERROR(1);
```

```
   PIN[4]&PIN[2]&PIN[5]&PIN[1]:=1 SHL I;
   PIN[11]&PIN[15]&PIN[14]&PIN[12]:=1 SHL I;
   IF PIN[3]<>HIGH THEN ERROR(1);
   IF PIN[13]<>HIGH THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4053,CD4053

#TEXT
Triple 2-Channel Analog
Multiplexer/Demultiplexer

This device can be used
as a triple 1-of-2 analog
data multiplexer or demul-
tiplexer with common
address, or as triple 1-
of-2 digital selector or
distributor.

#PIN 16
 1 : In-/Output  Y1
 2 : In-/Output  Y0
 3 : In-/Output  Z1
 4 : In/Out       Z
 5 : In-/Output  Z0
 6 : Inhibit
 7 : Digital/Analog
 8 : GND
 9 : Select Z
10 : Select Y
11 : Select X
12 : In-/Output  X0
13 : In-/Output  X1
14 : In/Out       X
15 : In/Out       Y
16 : Udd

#PROGRAM

BEGIN
PIN[6,7,9,10,11] : INPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[6,7]:=LOW;

PIN[4,14,15] : INPUT;
PIN[1,2,3,5,12,13] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11,10,9]:=I;
   PIN[14,15,4]:=LOW;
   IF (((PIN[13]&PIN[12]) SHR I) AND 1)<>LOW THEN ERROR(1);
   IF (((PIN[1]&PIN[2]) SHR I) AND 1)<>LOW THEN ERROR(1);
   IF (((PIN[3]&PIN[5]) SHR I) AND 1)<>LOW THEN ERROR(1);
   PIN[14,15,4]:=HIGH;
   IF (((PIN[13]&PIN[12]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   IF (((PIN[1]&PIN[2]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   IF (((PIN[3]&PIN[5]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   END;

PIN[1,2,3,5,12,13] : INPUT;
PIN[4,14,15] : OUTPUT;
FOR I:=0 TO 1 DO
```

```
   BEGIN
   PIN[11,10,9]:=I;
   PIN[13]&PIN[12]:=0;
   PIN[1]&PIN[2]:=0;
   PIN[3]&PIN[5]:=0;
   IF PIN[14]<>LOW THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   IF PIN[4]<>LOW THEN ERROR(1);
   PIN[13]&PIN[12]:=1 SHL I;
   PIN[1]&PIN[2]:=1 SHL I;
   PIN[3]&PIN[5]:=1 SHL I;
   IF PIN[14]<>HIGH THEN ERROR(1);
   IF PIN[15]<>HIGH THEN ERROR(1);
   IF PIN[4]<>HIGH THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4054,CD4054

#TEXT
4-Segment Driver

This device drives 4-
segment liquid crystal
displays.

#PIN 16
 1 : Strobe  Driver 4
 2 : DF in
 3 : Output  Driver 4
 4 : Output  Driver 3
 5 : Output  Driver 2
 6 : Output  Driver 1
 7 : Uee
 8 : Uss
 9 : Input   Driver 1
10 : Strobe  Driver 1
11 : Input   Driver 2
12 : Strobe  Driver 2
13 : Input   Driver 3
14 : Strobe  Driver 3
15 : Input   Driver 4
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,7,9,10,11,12,13,14,15] : INPUT;
PIN[3,4,5,6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7,9,11,13,15]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[13]&PIN[11]&PIN[9]:=D;
   PIN[1,14,12,10]:=HIGH;
   PIN[1,14,12,10]:=LOW;
   PIN[15]&PIN[13]&PIN[11]&PIN[9]:=0;
   PIN[2]:=LOW;
   IF (PIN[3]&PIN[4]&PIN[5]&PIN[6])<>D THEN ERROR(1);
   PIN[2]:=HIGH;
   IF (NOT(PIN[3])&NOT(PIN[4])&NOT(PIN[5])&NOT(PIN[6]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
```

```
   END;

ERROR(0);
END.

#NAME 4055,CD4055

#TEXT
BCD-to-7-Segment-Decoder/
Driver for LC-Displays

This device converts a
BCD input code into a
7-segment code for dri-
ving a liquid crystal
display.

#PIN 16
 1 : DF Out
 2 : BCD-Input   2
 3 : BCD-Input   1
 4 : BCD-Input   3
 5 : BCD-Input   0
 6 : DF In
 7 : Uee
 8 : Uss
 9 : 7-Seg.-Output  a
10 : 7-Seg.-Output  b
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  d
13 : 7-Seg.-Output  e
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7] : INPUT;
PIN[1,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7]:=LOW;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[2]&PIN[3]&PIN[5]:=I;
   PIN[6]:=LOW;
   D:=PIN[14]&PIN[15]&PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9];
   IF PIN[1]<>LOW THEN ERROR(1);
   PIN[6]:=HIGH;
   IF D<>(NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[13])&NOT(PIN[12])&
          NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])) THEN ERROR(1);
   IF PIN[1]<>HIGH THEN ERROR(1);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
```

```
END.

#NAME 4056,CD4056

#TEXT
BCD-to-7-Segment-Decoder/
Driver for LC-Displays

This device converts a
BCD input code into a
7-segment code for
driving liquid crystal
displays.

#PIN 16
 1 : Strobe
 2 : BCD-Input   2
 3 : BCD-Input   1
 4 : BCD-Input   3
 5 : BCD-Input   0
 6 : DF In
 7 : Uee
 8 : Uss
 9 : 7-Seg.-Output  a
10 : 7-Seg.-Output  b
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  d
13 : 7-Seg.-Output  e
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,7]:=LOW;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[2]&PIN[3]&PIN[5]:=I;
   PIN[1]:=HIGH;PIN[1]:=LOW;
   PIN[4]&PIN[2]&PIN[3]&PIN[5]:=0;
   PIN[6]:=LOW;
   D:=PIN[14]&PIN[15]&PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9];
   PIN[6]:=HIGH;
   IF D<>(NOT(PIN[14])&NOT(PIN[15])&NOT(PIN[13])&NOT(PIN[12])&
          NOT(PIN[11])&NOT(PIN[10])&NOT(PIN[9])) THEN ERROR(1);
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4060,CD4060
```

```
#TEXT
14-Stage Binary Counter
(:16384) with Internal
Oscillator

This device contains a
binary counter which
counts up asynchro-
nously. Feed back avai-
lable at the clock input
allows the input signal
to be conditioned or an
internal crystal or RC
oscillator can be
employed.

#PIN 16
 1 : Output  12
 2 : Output  13
 3 : Output  14
 4 : Output   6
 5 : Output   5
 6 : Output   7
 7 : Output   4
 8 : GND
 9 :  í0
10 : -í0
11 :  í1
12 : Reset
13 : Output   9
14 : Output   8
15 : Output  10
16 : Udd

#PROGRAM

BEGIN
PIN[9,10,11,12] : INPUT;
PIN[1,2,3,4,5,6,7,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9,12]:=LOW;
PIN[10,11]:=HIGH;

PIN[12]:=HIGH;PIN[12]:=LOW;
WINDOW (31,11,48,14);
D:=0;

FOR I:=0 TO 13 DO
   BEGIN
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      END;
   GOTOXY (I+2,1);WRITE ('Û');
   GOTOXY (I+2,2);WRITE ('Û');
   IF (I>2) AND (I<>10) THEN
      BEGIN
      D:=(D SHL 1) OR 1;
      IF (PIN[3]&PIN[2]&PIN[1]&PIN[15]&PIN[13]&
          PIN[14]&PIN[6]&PIN[4]&PIN[5]&PIN[7])<>D THEN ERROR(1);
      END;
   END;

PIN[12]:=HIGH;PIN[12]:=LOW;
IF (PIN[3]&PIN[2]&PIN[1]&PIN[15]&PIN[13]&
```

```
    PIN[14]&PIN[6]&PIN[4]&PIN[5]&PIN[7])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4063,CD4063

#TEXT
4-Bit Comparator

This device consists of
a 2 x 4-bit comparator.
The device indicates
which 4-bit word is the
largest or whether they
both equal.

#PIN 16
 1 : Input   B3
 2 : carryinput. A<B
 3 : carryinput. A=B
 4 : carryinput. A>B
 5 : Output  A>B
 6 : Output  A=B
 7 : Output  A<B
 8 : GND
 9 : Input   B0
10 : Input   A0
11 : Input   B1
12 : Input   A1
13 : Input   A2
14 : Input   B2
15 : Input   A3
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,9,10,11,12,13,14,15] : INPUT;
PIN[5,6,7] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[2]:=HIGH;PIN[3]:=LOW;PIN[4]:=LOW;
PIN[15]&PIN[13]&PIN[12]&PIN[10]:=0;
PIN[1]&PIN[14]&PIN[11]&PIN[9]:=0;
IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[2]:=LOW;PIN[3]:=HIGH;PIN[4]:=LOW;
PIN[15]&PIN[13]&PIN[12]&PIN[10]:=0;
PIN[1]&PIN[14]&PIN[11]&PIN[9]:=0;
IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
```

```
C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>HIGH) OR (PIN[5]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[2]:=LOW;PIN[3]:=LOW;PIN[4]:=HIGH;
PIN[15]&PIN[13]&PIN[12]&PIN[10]:=0;
PIN[1]&PIN[14]&PIN[11]&PIN[9]:=0;
IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   C:=C+1;
   PIN[1]&PIN[14]&PIN[11]&PIN[9]:=1 SHL C;
   IF (PIN[7]<>HIGH) OR (PIN[6]<>LOW) OR (PIN[5]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[13]&PIN[12]&PIN[10]:=1 SHL C;
   IF (PIN[7]<>LOW) OR (PIN[6]<>LOW) OR (PIN[5]<>HIGH) THEN ERROR(1);
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 4066,CD4066

#TEXT
Quad Bilateral Switch

This device contains
four separate bilateral
switches intended for
the transmission or
multiplexing of analog
or digital signals.

#PIN 14
 1 : Input   Switch 1
 2 : Output  Switch 1
 3 : Input   Switch 2
 4 : Output  Switch 2
 5 : Control Switch 2
 6 : Control Switch 3
 7 : GND
 8 : Input   Switch 3
 9 : Output  Switch 3
10 : Input   Switch 4
11 : Output  Switch 4
12 : Control Switch 4
13 : Control Switch 1
14 : +3V...+15V
```

```
#PROGRAM

BEGIN
PIN[1,3,5,6,8,10,12,13] : INPUT;
PIN[2,4,9,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 1 DO
    BEGIN
    PIN[13,5,6,12]:=I;
    PIN[1,3,8,10]:=I;
    IF PIN[2]<>I THEN ERROR(1);
    IF PIN[4]<>I THEN ERROR(1);
    IF PIN[9]<>I THEN ERROR(1);
    IF PIN[11]<>I THEN ERROR(1);
    END;

ERROR(0);
END.

#NAME 4068,CD4068

#TEXT
8-Input NAND Gate

This device contains
a NAND gate with eight
inputs.

#PIN 14
 1 : N.C.
 2 : Input   1
 3 : Input   2
 4 : Input   3
 5 : Input   4
 6 : N.C.
 7 : GND
 8 : N.C.
 9 : Input   5
10 : Input   6
11 : Input   7
12 : Input   8
13 : Output
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12] : INPUT;
PIN[13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[2,3,4,5,9,10,11,12]:=LOW;
IF PIN[13]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[9]&PIN[10]&PIN[11]&PIN[12]:=1 SHL I;
   IF PIN[13]<>HIGH THEN ERROR(1);
   END;

PIN[2,3,4,5,9,10,11,12]:=HIGH;
IF PIN[13]<>LOW THEN ERROR(1);

ERROR(0);
```

```
END.

#NAME 4069,CD4069

#TEXT
Hex Inverter

This device contains
six separate inverters.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : Udd

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4070,CD4070

#TEXT
Quad 2-Input EXOR Gate

This device contains
four EXOR gates each with
two inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
```

```
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] EXOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] EXOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] EXOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] EXOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4071,CD4071

#TEXT
Quad 2-Input OR Gate

This device contains
four OR gates each with
two inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] OR PIN[2]) THEN ERROR(1);
```

```
   IF PIN[4]<>(PIN[5] OR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] OR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] OR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4072,CD4072

#TEXT
Dual 4-Input Or Gate

This device contains
two OR gates each with
four inputs.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : N.C.
 7 : GND
 8 : N.C.
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : Input   3 Gate 2
12 : Input   4 Gate 2
13 : Output    Gate 2
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12] : INPUT;
PIN[1,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]&PIN[12]:=I;
   IF PIN[1]<>(PIN[2] OR PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[13]<>(PIN[9] OR PIN[10] OR PIN[11] OR PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4073,CD4073

#TEXT
Triple 3-Input AND Gate

This device contains
three AND gates each with
three inputs.

#PIN 14
 1 : Input   1 Gate 2
 2 : Input   2 Gate 2
 3 : Input   1 Gate 1
 4 : Input   2 Gate 1
 5 : Input   3 Gate 1
```

```
 6 : Output    Gate 1
 7 : GND
 8 : Input   3 Gate 2
 9 : Output    Gate 2
10 : Output    Gate 3
11 : Input   1 Gate 3
12 : Input   2 Gate 3
13 : Input   3 Gate 3
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,8,11,12,13] : INPUT;
PIN[6,9,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[1]&PIN[2]&PIN[8]:=I;
   PIN[11]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>(PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[9]<>(PIN[1] AND PIN[2] AND PIN[8]) THEN ERROR(1);
   IF PIN[10]<>(PIN[11] AND PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4075,CD4075

#TEXT
Triple 3-Input OR Gate

This device contains
three OR gates each
with three inputs.

#PIN 14
 1 : Input   1 Gate 2
 2 : Input   2 Gate 2
 3 : Input   1 Gate 1
 4 : Input   2 Gate 1
 5 : Input   3 Gate 1
 6 : Output    Gate 1
 7 : GND
 8 : Input   3 Gate 2
 9 : Output    Gate 2
10 : Output    Gate 3
11 : Input   1 Gate 3
12 : Input   2 Gate 3
13 : Input   3 Gate 3
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,8,11,12,13] : INPUT;
PIN[6,9,10] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
```

```
   PIN[1]&PIN[2]&PIN[8]:=I;
   PIN[11]&PIN[12]&PIN[13]:=I;
   IF PIN[6]<>(PIN[3] OR PIN[4] OR PIN[5]) THEN ERROR(1);
   IF PIN[9]<>(PIN[1] OR PIN[2] OR PIN[8]) THEN ERROR(1);
   IF PIN[10]<>(PIN[11] OR PIN[12] OR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4076,CD4076

#TEXT
Quad TRI-STATE D-Register

This device contains
four bistable memory
elements with three-
state outputs.

#PIN 16
 1 : -Enable Output  1
 2 : -Enable Output  2
 3 : Output  Q1
 4 : Output  Q2
 5 : Output  Q3
 6 : Output  Q4
 7 : Clock
 8 : GND
 9 : -Enable Input   1
10 : -Enable Input   2
11 : Input   D4
12 : Input   D3
13 : Input   D2
14 : Input   D1
15 : Reset
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,7,9,10,11,12,13,14,15] : INPUT;
PIN[3,4,5,6] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[7,15]:=LOW;
PIN[1,2,9,10]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9,10]:=LOW;
   PIN[11]&PIN[12]&PIN[13]&PIN[14]:=D;
   PIN[7]:=HIGH;PIN[7]:=LOW;
   PIN[11]&PIN[12]&PIN[13]&PIN[14]:=0;
   PIN[9,10]:=HIGH;
   PIN[1,2]:=LOW;
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   PIN[1,2]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[15]:=HIGH;PIN[15]:=LOW;
PIN[1,2]:=LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>0 THEN ERROR(1);
PIN[1,2]:=HIGH;
```

```
LOADMODEON;
PIN[6,5,4,3] : LOAD LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
PIN[6,5,4,3] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4077,CD4077

#TEXT
Quad 2-Input Exclusive
NOR Gate

This device contains
four EXNOR gates each
two inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NEXOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NEXOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NEXOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NEXOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4078,CD4078

#TEXT
8-Input NOR Gate

This device contains
a NOR gate with eight
inputs.
```

```
#PIN 14
 1 : N.C.
 2 : Input   1
 3 : Input   2
 4 : Input   3
 5 : Input   4
 6 : N.C.
 7 : GND
 8 : N.C.
 9 : Input   5
10 : Input   6
11 : Input   7
12 : Input   8
13 : Output
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12] : INPUT;
PIN[13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[2,3,4,5,9,10,11,12]:=LOW;
IF PIN[13]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[9]&PIN[10]&PIN[11]&PIN[12]:=1 SHL I;
   IF PIN[13]<>LOW THEN ERROR(1);
   END;

PIN[2,3,4,5,9,10,11,12]:=HIGH;
IF PIN[13]<>LOW THEN ERROR(1);

ERROR(0);
END.

#NAME 4081,CD4081

#TEXT
Quad 2-Input AND Gate

This device contains
four AND gates each
with two inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM
```

```
BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] AND PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] AND PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] AND PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] AND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4082,CD4082

#TEXT
Dual 4-Input AND Gate

This device contains
two AND gates each
with four inputs.

#PIN 14
 1 : Output    Gate 1
 2 : Input   1 Gate 1
 3 : Input   2 Gate 1
 4 : Input   3 Gate 1
 5 : Input   4 Gate 1
 6 : N.C.
 7 : GND
 8 : N.C.
 9 : Input   1 Gate 2
10 : Input   2 Gate 2
11 : Input   3 Gate 2
12 : Input   4 Gate 2
13 : Output    Gate 2
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12] : INPUT;
PIN[1,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]&PIN[12]:=I;
   IF PIN[1]<>(PIN[2] AND PIN[3] AND PIN[4] AND PIN[5]) THEN ERROR(1);
   IF PIN[13]<>(PIN[9] AND PIN[10] AND PIN[11] AND PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4085,CD4085
```

```
#TEXT
Dual AOI Gate

This device contains
two separate AND-OR-
INVERT gates.

#PIN 14
 1 : Input   I1 Gate 1
 2 : Input   I2 Gate 1
 3 : Output  Q  Gate 1
 4 : Output  Q  Gate 2
 5 : Input   I1 Gate 2
 6 : Input   I2 Gate 2
 7 : GND
 8 : Input   I3 Gate 2
 9 : Input   I4 Gate 2
10 : Inhibit    Gate 1
11 : Inhibit    Gate 2
12 : Input   I3 Gate 1
13 : Input   I4 Gate 1
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,10,11,12,13] : INPUT;
PIN[3,4] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 31 DO
   BEGIN
   PIN[10]&PIN[1]&PIN[2]&PIN[12]&PIN[13]:=I;
   PIN[11]&PIN[5]&PIN[6]&PIN[8]&PIN[9]:=I;
   IF PIN[3]<>NOT(PIN[10] OR (PIN[1] AND PIN[2]) OR
                  (PIN[12] AND PIN[13])) THEN ERROR(1);
   IF PIN[4]<>NOT(PIN[11] OR (PIN[5] AND PIN[6]) OR
                  (PIN[8] AND PIN[9])) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4086,CD4086

#TEXT
2-2-2-2-Input AOI Gate

This device contains an
AND-OR-INVERT gate with
4 x 2 inputs.

#PIN 14
 1 : Input   I1
 2 : Input   I2
 3 : Output  Q
 4 : N.C.
 5 : Input   I3
 6 : Input   I4
 7 : GND
 8 : Input   I5
 9 : Input   I6
10 : Inhibit
11 : Enable
12 : Input   I7
13 : Input   I8
```

```
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,10,11,12,13] : INPUT;
PIN[3] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

PIN[1,2,5,6,8,9,10,11,12,13]:=LOW;
IF PIN[3]<>LOW THEN ERROR(1);
PIN[11]:=HIGH;
IF PIN[3]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 BY 2 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[5]&PIN[6]&PIN[8]&PIN[9]&PIN[12]&PIN[13]:=
   (1 SHL I) OR (1 SHL (I+1));
   IF PIN[3]<>LOW THEN ERROR(1);
   PIN[1]&PIN[2]&PIN[5]&PIN[6]&PIN[8]&PIN[9]&PIN[12]&PIN[13]:=0;
   IF PIN[3]<>HIGH THEN ERROR(1);
   END;

PIN[10]:=HIGH;
IF PIN[3]<>LOW THEN ERROR(1);

ERROR(0);
END.

#NAME 4089,CD4089

#TEXT
Binary Rate-Multiplier

This device contains a
programmable 4-bit
frequency divider, other-
wise known as a bit rate-
multiplier (Modulo 16).


#PIN 16
 1 : "15" Out
 2 : Input   C
 3 : Input   D
 4 : "15" Set
 5 : -Out
 6 : Out
 7 : Inhibit Out
 8 : GND
 9 : Clock
10 : Strobe
11 : Inhibit
12 : Cascade
13 : Clear
14 : Input   A
15 : Input   B
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,9,10,11,12,13,14,15] : INPUT;
PIN[1,5,6,7] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
```

```
PIN[9,10,11,12,13]:=LOW;

PIN[12]:=LOW;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]:=1 SHL I;
   Q:=0;C:=0;E:=0;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      E:=E+PIN[7];
      END;
   IF Q<>(16-(1 SHL I)) THEN ERROR(1);
   IF C<>1 SHL I THEN ERROR(1);
   IF E<>15 THEN ERROR(1);
   END;

PIN[12]:=HIGH;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]:=1 SHL I;
   Q:=0;C:=0;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      END;
   IF Q<>(16-(1 SHL I)) THEN ERROR(1);
   IF C<>16 THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4093,CD4093

#TEXT
Quad 2-Input NAND
Schmitt Trigger

This device contains
four NAND Schmitt-
Triggers each with
two inputs.

#PIN 14
 1 : Input   1 Gate 1
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM
```

```
BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] NAND PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] NAND PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] NAND PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] NAND PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4094,CD4094

#TEXT
8-Bit Shift Register/Latch

This device contains an
eight-stage shift register
an 8-bit latch and three-
state outputs.

#PIN 16
 1 : Strobe
 2 : Input   D
 3 : Clock
 4 : Output  Q1
 5 : Output  Q2
 6 : Output  Q3
 7 : Output  Q4
 8 : GND
 9 : Output  QS
10 : Output  QS'
11 : Output  Q8
12 : Output  Q7
13 : Output  Q6
14 : Output  Q5
15 : Output Enable
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,15] : INPUT;
PIN[4,5,6,7,11,12,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,3,15]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[2]:=X AND 1;
      PIN[3]:=HIGH;PIN[3]:=LOW;
```

```
      X:=X SHR 1;
      END;
   PIN[1]:=HIGH;PIN[1]:=LOW;
   X:=0;
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[2]:=X AND 1;
      PIN[3]:=HIGH;PIN[3]:=LOW;
      X:=X SHR 1;
      END;
   PIN[15]:=HIGH;
   IF (PIN[4]&PIN[5]&PIN[6]&PIN[7]&
       PIN[14]&PIN[13]&PIN[12]&PIN[11])<>D THEN ERROR(1);
   PIN[15]:=LOW;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[4,5,6,7,14,13,12,11] : LOAD LOW;
IF (PIN[4]&PIN[5]&PIN[6]&PIN[7]&
      PIN[14]&PIN[13]&PIN[12]&PIN[11])<>%00000000 THEN ERROR(1);
PIN[4,5,6,7,14,13,12,11] : LOAD HIGH;
IF (PIN[4]&PIN[5]&PIN[6]&PIN[7]&
      PIN[14]&PIN[13]&PIN[12]&PIN[11])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```

```
#NAME 4095,CD4095

#TEXT
3-Input J-K Master/Slave
Flip-Flop with Preset
and Clear

This device contains
a pulse-triggered J-K
master/slave flip-flop
with 3 inputs, preset
and clear.

#PIN 14
 1 : N.C.
 2 : Reset
 3 : Input    J1
 4 : Input    J2
 5 : Input    J3
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : Input    K3
10 : Input    K2
11 : Input    K1
12 : Clock
13 : Preset
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[2,3,4,5,9,10,11,12,13]:=LOW;

PIN[2]:=HIGH;PIN[2]:=LOW;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

PIN[13]:=HIGH;PIN[13]:=LOW;
IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[11]&PIN[10]&PIN[9]:=I;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

PIN[3,4,5,11,10,9]:=HIGH;
PIN[12]:=HIGH;PIN[12]:=LOW;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 4096,CD4096
```

```
#TEXT
3-Input J-K Master/Slave
Flip-Flop with Preset and
Clear

This device contains a
pulse-triggered J-K master
slave flip-flop with three
AND inputs, preset and
clear.

#PIN 14
 1 : N.C.
 2 : Reset
 3 : Input    J1
 4 : Input    J2
 5 : Input   -J3
 6 : Output  -Q
 7 : GND
 8 : Output   Q
 9 : Input   -K3
10 : Input    K2
11 : Input    K1
12 : Clock
13 : Preset
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,9,10,11,12,13] : INPUT;
PIN[6,8] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[2,3,4,10,11,12,13]:=LOW;
PIN[5,9]:=HIGH;

PIN[2]:=HIGH;PIN[2]:=LOW;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

PIN[13]:=HIGH;PIN[13]:=LOW;
IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[11]&PIN[10]&PIN[9]:=I EXOR %001;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);
   END;

FOR I:=7 DOWNTO 0 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]:=I EXOR %001;
   PIN[12]:=HIGH;PIN[12]:=LOW;
   IF (PIN[8]<>HIGH) OR (PIN[6]<>LOW) THEN ERROR(1);
   END;

PIN[3,4,11,10]:=HIGH;
PIN[5,9]:=LOW;
PIN[12]:=HIGH;PIN[12]:=LOW;
IF (PIN[8]<>LOW) OR (PIN[6]<>HIGH) THEN ERROR(1);

ERROR(0);
END.

#NAME 4099,CD4099
```

```
#TEXT
8-Bit Addressable
Latches

This device contains an
addressable 8-bit latch
with enable and clear.

#PIN 16
 1 : Output  Q7
 2 : Reset
 3 : Data
 4 : Write Disable
 5 : Address A0
 6 : Address A1
 7 : Address A2
 8 : GND
 9 : Output  Q0
10 : Output  Q1
11 : Output  Q2
12 : Output  Q3
13 : Output  Q4
14 : Output  Q5
15 : Output  Q6
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7] : INPUT;
PIN[1,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[4]:=HIGH;

D:=%01011010;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[3]:=(D SHR J) AND 1;
      PIN[7]&PIN[6]&PIN[5]:=J;
      PIN[4]:=LOW;PIN[4]:=HIGH;
      END;
   IF (PIN[1]&PIN[15]&PIN[14]&PIN[13]&
       PIN[12]&PIN[11]&PIN[10]&PIN[9])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[2]:=HIGH;PIN[2]:=LOW;
IF (PIN[1]&PIN[15]&PIN[14]&PIN[13]&
    PIN[12]&PIN[11]&PIN[10]&PIN[9])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4104,CD4104

#TEXT
Quad TTL/CMOS-Level
Converter with Comple-
mentary TRI-STATE Outputs

This device contains
four separate level
converters which convert
```

```
low level signals to a
higher level.

#PIN 16
 1 : Udd0
 2 : Output  -Q Buffer 1
 3 : Output   Q Buffer 1
 4 : Input      Buffer 1
 5 : Input      Buffer 2
 6 : Output   Q Buffer 2
 7 : Output  -Q Buffer 2
 8 : GND
 9 : Output  -Q Buffer 3
10 : Output   Q Buffer 3
11 : Input      Buffer 3
12 : Input      Buffer 4
13 : Output   Q Buffer 4
14 : Output  -Q Buffer 4
15 : Output Enable
16 : Udd1

#PROGRAM

BEGIN
PIN[1,4,5,11,12,15] : INPUT;
PIN[2,3,6,7,9,10,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=LOW;
PIN[1]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[4]&PIN[5]&PIN[11]&PIN[12]:=D;
   PIN[15]:=HIGH;
   IF (PIN[3]&PIN[6]&PIN[10]&PIN[13])<>D THEN ERROR(1);
   IF (NOT(PIN[2])&NOT(PIN[7])&NOT(PIN[9])&NOT(PIN[14]))<>D THEN ERROR(1);
   PIN[15]:=LOW;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[2,3,6,7,9,10,13,14] : LOAD LOW;
IF (PIN[3]&PIN[6]&PIN[10]&PIN[13])<>%0000 THEN ERROR(1);
IF (PIN[2]&PIN[7]&PIN[9]&PIN[14])<>%0000 THEN ERROR(1);
PIN[2,3,6,7,9,10,13,14] : LOAD HIGH;
IF (PIN[3]&PIN[6]&PIN[10]&PIN[13])<>%1111 THEN ERROR(1);
IF (PIN[2]&PIN[7]&PIN[9]&PIN[14])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4160,CD4160

#TEXT
Decade Counter with
Asynchronous Clear

This device consists of
a programmable, synchro-
nous decade counter which
counts up in BCD code and
is cleared asynchronously.

#PIN 16
```

```
 1 : -R asyn
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
15 : Carry Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4161,CD4161

#TEXT
Binary Counter with
Asynchronous Clear

This device consists of
a programmable, synchro-
nous 4-bit binary counter
which counts up in binary
code and is cleared
asychronously.

#PIN 16
 1 : -R asyn
```

```
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
15 : Carry Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4162,CD4162

#TEXT
Decade Counter with
Synchronous Clear

This device contains a
programmable, synchronous
decade counter which
counts up in BCD code and
is cleared synchronously.

#PIN 16
 1 : -R syn
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
```

```
 6 : Input   P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
15 : Carry Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[1]:=HIGH;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4163,CD4163

#TEXT
Binary Counter with
Synchronous Clear

This device contains a
programmable, synchronous
4-bit binary counter which
counts up in binary code
and is synchronously
reset.

#PIN 16
 1 : -R syn
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
```

```
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
15 : Carry Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[1]:=HIGH;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4174,CD4174

#TEXT
Hex D-Register with
Clear

This device contains
six bistable memory
elements.

#PIN 16
 1 : -Clear
 2 : Output  Q SE 1
 3 : Input   D SE 1
 4 : Input   D SE 2
 5 : Output  Q SE 2
 6 : Input   D SE 3
 7 : Output  Q SE 3
 8 : GND
 9 : Clock
10 : Output  Q SE 4
```

```
11 : Input   D SE 4
12 : Output  Q SE 5
13 : Input   D SE 5
14 : Input   D SE 6
15 : Output  Q SE 6
16 : Udd

#PROGRAM

BEGIN
PIN[1,3,4,6,9,11,13,14] : INPUT;
PIN[2,5,7,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=LOW;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=0;
   IF (PIN[15]&PIN[12]&PIN[10]&PIN[7]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[15]&PIN[12]&PIN[10]&PIN[7]&PIN[5]&PIN[2])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4175,CD4175

#TEXT
Quad D-Register with
Clear

This device contains
four bistable memory
elements.

#PIN 16
 1 : -Clear
 2 : Output   Q SE 1
 3 : Output  -Q SE 1
 4 : Input    D SE 1
 5 : Input    D SE 2
 6 : Output  -Q SE 2
 7 : Output   Q SE 2
 8 : GND
 9 : Clock
10 : Output   Q SE 3
11 : Output  -Q SE 3
12 : Input    D SE 3
13 : Input    D SE 4
14 : Output  -Q SE 4
15 : Output   Q SE 4
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,9,12,13] : INPUT;
PIN[2,3,6,7,10,11,14,15] : OUTPUT;
PIN[8] : GND;
```

```
PIN[16] : +5V;
PIN[9]:=LOW;
PIN[1]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=0;
   IF (PIN[15]&PIN[10]&PIN[7]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[14])&NOT(PIN[11])&NOT(PIN[6])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4192,CD4192

#TEXT
Synchronous Up/Down Decade
Counter with Clear

This device contains a
programmable, synchronous
up/down decade BCD counter
having separate clock
inputs and a clear input.

#PIN 16
 1 : Input   P1
 2 : Output  Q1
 3 : Output  Q0
 4 : Down
 5 : Up
 6 : Output  Q2
 7 : Output  Q3
 8 : GND
 9 : Input   P3
10 : Input   P2
11 : -Load
12 : -Carry Up
13 : -Carry Down
14 : Reset
15 : Input   P0
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,5,14]:=LOW;
PIN[4,5,11]:=HIGH;

PIN[14]:=HIGH;PIN[14]:=LOW;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>HIGH THEN ERROR(1);
   PIN[5]:=LOW;PIN[5]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
```

```
FOR I:=9 DOWNTO 1 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[13]<>HIGH THEN ERROR(1);
   PIN[4]:=LOW;PIN[4]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4193,CD4193

#TEXT
Synchronous 4-Bit Up/
Down Binary Counter
with Clear

This device contains a
programmable, synchro-
nous 4-bit up/down binary
counter with separate
clock inputs and clear
input.

#PIN 16
 1 : Input   P1
 2 : Output  Q1
 3 : Output  Q0
 4 : Down
 5 : Up
 6 : Output  Q2
 7 : Output  Q3
 8 : GND
 9 : Input   P3
10 : Input   P2
11 : -Load
12 : -Carry Up
13 : -Carry Down
14 : Reset
15 : Input   P0
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,5,14]:=LOW;
PIN[4,5,11]:=HIGH;

PIN[14]:=HIGH;PIN[14]:=LOW;

FOR I:=0 TO 14 DO
   BEGIN
```

```
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>HIGH THEN ERROR(1);
   PIN[5]:=LOW;PIN[5]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

FOR I:=15 DOWNTO 1 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[13]<>HIGH THEN ERROR(1);
   PIN[4]:=LOW;PIN[4]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4194,CD4194

#TEXT
4-Bit-Right/Left Shift
Register with Synchronous
Parallel Input/Output
and Clear

This device contains a
4-bit shift register
with parallel input and
output. Data can be
shifted to the right or
left. The device also
has a clear input.

#PIN 16
 1 : -Reset
 2 : Data    SR
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : Data    SL
 8 : GND
 9 : Mode    S0
10 : Mode    S1
11 : Clock
12 : Output  Q3
13 : Output  Q2
14 : Output  Q1
15 : Output  Q0
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10,11] : INPUT;
PIN[12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
```

```
PIN[11]:=LOW;
PIN[1]:=HIGH;

D:=%0101;

PIN[9]:=LOW;
PIN[10]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[7]:=X AND 1;
      X:=X SHR 1;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      END;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[9]:=HIGH;
PIN[10]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2]:=X AND 1;
      X:=X SHR 1;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      END;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[9]:=HIGH;
PIN[10]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4195,CD4195

#TEXT
4-Bit Serial-In/Parallel-
Out Shift Register with
Clear

This device contains a
4-bit right shift regis-
ter with parallel or
serial input and parallel
output and a clear input.

#PIN 16
 1 : -Reset
 2 : Input     J
 3 : Input    -K
 4 : Input    P0
 5 : Input    P1
```

```
 6 : Input    P2
 7 : Input    P3
 8 : GND
 9 : -Load
10 : Clock
11 : Output  -Q3
12 : Output   Q3
13 : Output   Q2
14 : Output   Q1
15 : Output   Q0
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10]:=LOW;
PIN[1,9]:=HIGH;

D:=%0101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2,3]:=X AND 1;
      X:=X SHR 1;
      PIN[10]:=LOW;PIN[10]:=HIGH;
      END;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>D THEN ERROR(1);
   IF NOT(PIN[11])<>(D AND 1) THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;PIN[2,3]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[4]&PIN[5]&PIN[6]&PIN[7]:=D;
   PIN[10]:=LOW;PIN[10]:=HIGH;
   PIN[9]:=HIGH;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>X THEN ERROR(1);
      IF NOT(PIN[11])<>(X AND 1) THEN ERROR(1);
      PIN[2,3]:=LOW;
      PIN[10]:=LOW;PIN[10]:=HIGH;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4311,CD4311

#TEXT
Binary-to- 7- Segment
```

```
Decoder/Driver with
Storage

This device converts
binary input data into
control data for 7-seg-
ment displays.

#PIN 16
 1 : Binary-Input   B
 2 : Binary-Input   C
 3 : -LT (Lamp Test)
 4 : -BL
 5 : Strobe
 6 : Binary-Input   D
 7 : Binary-Input   A
 8 : GND
 9 : 7-Seg.-Output  e
10 : 7-Seg.-Output  d
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  a
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   PIN[5]:=LOW;PIN[5]:=HIGH;
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=0;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&
      PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I=10) AND (D<>119) THEN ERROR(1);
   IF (I=11) AND (D<>124) THEN ERROR(1);
   IF (I=12) AND (D<>57) AND (D<>88) THEN ERROR(1);
   IF (I=13) AND (D<>94) THEN ERROR(1);
   IF (I=14) AND (D<>121) THEN ERROR(1);
   IF (I=15) AND (D<>113) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4368,CD4368

#TEXT
Binary-to-7-Segment
```

```
Decoder/Driver with
Storage

This device converts
binary input data into
control data for 7 seg-
ment displays.

#PIN 16
 1 : Binary-Input   B
 2 : Binary-Input   C
 3 : -Enable Latch
 4 : -RBO
 5 : -RBI
 6 : Binary-Input   D
 7 : Binary-Input   A
 8 : GND
 9 : 7-Seg.-Output  e
10 : 7-Seg.-Output  d
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  a
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,5,6,7] : INPUT;
PIN[4,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,5]:=HIGH;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   PIN[3]:=LOW;PIN[3]:=HIGH;
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=0;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&
      PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I=10) AND (D<>119) THEN ERROR(1);
   IF (I=11) AND (D<>124) THEN ERROR(1);
   IF (I=12) AND (D<>57) AND (D<>88) THEN ERROR(1);
   IF (I=13) AND (D<>94) THEN ERROR(1);
   IF (I=14) AND (D<>121) THEN ERROR(1);
   IF (I=15) AND (D<>113) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4404,CD4404

#TEXT
8-Stage Binary Counter
```

```
This device contains
a synchronous eight-
stage binary counter.

#PIN 14
 1 : -Clock
 2 : Reset
 3 : Output  7
 4 : Output  6
 5 : Output  5
 6 : Output  4
 7 : GND
 8 : N.C.
 9 : Output  3
10 : N.C.
11 : Output  2
12 : Output  1
13 : Output  8
14 : Udd

#PROGRAM

BEGIN
PIN[1,2] : INPUT;
PIN[3,4,5,6,9,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[2]:=LOW;
PIN[1]:=HIGH;

PIN[2]:=HIGH;PIN[2]:=LOW;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[1]:=LOW;PIN[1]:=HIGH;
   IF (PIN[13]&PIN[3]&PIN[4]&PIN[5]&
       PIN[6]&PIN[9]&PIN[11]&PIN[12])<>(1 SHL I) THEN ERROR(1);
   END;

PIN[2]:=HIGH;PIN[2]:=LOW;
IF (PIN[13]&PIN[3]&PIN[4]&PIN[5]&
    PIN[6]&PIN[9]&PIN[11]&PIN[12])<>0 THEN ERROR(1);

ERROR(0);
END.

##NAME 4416,CD4416

#TEXT
Quad Digital/Analog
Bilateral Switch

This device contains
four separate bilateral
switches intended for
the transmission or
multiplexing of digital
or analog signals.

#PIN 14
 1 : Input   Switch 1
 2 : Output  Switch 1
 3 : Input   Switch 2
 4 : Output  Switch 2
 5 : Control Switch 2
 6 : Control Switch 3
```

```
 7 : GND
 8 : Input   Switch 3
 9 : Output  Switch 3
10 : Input   Switch 4
11 : Output  Switch 4
12 : Control Switch 4
13 : Control Switch 1
14 : +3V...+15V

#PROGRAM

BEGIN
PIN[1,3,5,6,8,10,12,13] : INPUT;
PIN[2,4,9,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 1 DO
    BEGIN
    PIN[13,5,6,12]:=I;
    PIN[1,3,8,10]:=I;
    IF PIN[2]<>I THEN ERROR(1);
    IF PIN[4]<>I THEN ERROR(1);
    IF PIN[9]<>I THEN ERROR(1);
    IF PIN[11]<>I THEN ERROR(1);
    END;

ERROR(0);
END.

##NAME 4426,CD4426

#TEXT
Decade Counter with
Decoded 7-Segment-
Driver Outputs

This divide-by-10 decade
counter provides internal
decoding for controlling
a seven segment display.

#PIN 16
 1 : Clock
 2 : Clock Enable
 3 : Display Enable
 4 : Enable Out
 5 : Carry Out
 6 : 7-Seg.-Output  f
 7 : 7-Seg.-Output  g
 8 : GND
 9 : 7-Seg.-Output  d
10 : 7-Seg.-Output  a
11 : 7-Seg.-Output  e
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  c
14 : Ungated C-Segment
15 : Reset
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,15] : INPUT;
PIN[4,5,6,7,9,10,11,12,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
```

```
PIN[1,2,15]:=LOW;
PIN[3]:=HIGH;

IF PIN[4]<>HIGH THEN ERROR(1);

PIN[15]:=HIGH;PIN[15]:=LOW;
FOR I:=0 TO 9 DO
   BEGIN
   D:=PIN[7]&PIN[6]&PIN[11]&PIN[9]&PIN[13]&PIN[12]&PIN[10];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I<5) AND (PIN[5]<>HIGH) THEN ERROR(1);
   IF (I>=5) AND (PIN[5]<>LOW) THEN ERROR(1);
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;

ERROR(0);
END.

#NAME 4428,CD4428

#TEXT
Binary-to-Octal Decoder

This device converts a
binary code into a
decimal number from
0 to 7.

#PIN 14
 1 : Output  4
 2 : Output  2
 3 : Output  0
 4 : Output  7
 5 : Output  5
 6 : Output  6
 7 : GND
 8 : Input   A
 9 : Strobe
10 : Input   C
11 : Input   B
12 : Output  1
13 : Output  3
14 : Udd

#PROGRAM

BEGIN
PIN[8,9,10,11] : INPUT;
PIN[1,2,3,4,5,6,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[9]:=HIGH;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[10]&PIN[11]&PIN[8]:=I;
   IF (PIN[4]&PIN[6]&PIN[5]&PIN[1]&
       PIN[13]&PIN[2]&PIN[12]&PIN[3])<>(1 SHL I) THEN ERROR(1);
```

```
   END;

ERROR(0);
END.

##NAME 4433,CD4433

#TEXT
Decade Counter with
Decoded 7-Segment-
Driver Outputs

This divide-by-10 decade
counter provides internal
decoding for controlling
a seven segment display.

#PIN 16
 1 : Clock
 2 : Clock Enable
 3 : RBI
 4 : RBO
 5 : Carry Out
 6 : 7-Seg.-Output  f
 7 : 7-Seg.-Output  g
 8 : GND
 9 : 7-Seg.-Output  d
10 : 7-Seg.-Output  a
11 : 7-Seg.-Output  e
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  c
14 : Lamp Test
15 : Reset
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,2,14,15]:=LOW;
PIN[3]:=HIGH;

PIN[15]:=HIGH;PIN[15]:=LOW;
FOR I:=0 TO 9 DO
   BEGIN
   D:=PIN[7]&PIN[6]&PIN[11]&PIN[9]&PIN[13]&PIN[12]&PIN[10];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I<5) AND (PIN[5]<>HIGH) THEN ERROR(1);
   IF (I>=5) AND (PIN[5]<>LOW) THEN ERROR(1);
   PIN[1]:=HIGH;PIN[1]:=LOW;
   END;

ERROR(0);
END.
```

```
#NAME 4441,CD4441

#TEXT
Quad Inverting Buffer/
Driver

This device contains
four separate buffers
for high currents.

#PIN 14
 1 : N.C.
 2 : Output  Inverter 1
 3 : Input   Inverter 1
 4 : N.C.
 5 : Output  Inverter 2
 6 : Input   Inverter 2
 7 : GND
 8 : N.C.
 9 : Output  Inverter 3
10 : Input   Inverter 3
11 : N.C.
12 : Output  Inverter 4
13 : Input   Inverter 4
14 : Udd

#PROGRAM

BEGIN
PIN[3,6,10,13] : INPUT;
PIN[2,5,9,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[6]&PIN[10]&PIN[13]:=I;
   IF (NOT(PIN[2])&NOT(PIN[5])&NOT(PIN[9])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4449,CD4449

#TEXT
Hex Inverter

This device contains
six separate inverting
buffers.

#PIN 16
 1 : +3V...+15V
 2 : Output    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 3
 7 : Input     Gate 3
 8 : GND
 9 : Input     Gate 4
10 : Output    Gate 4
11 : Input     Gate 5
12 : Output    Gate 5
```

```
13 : N.C.
14 : Input     Gate 6
15 : Output    Gate 6
16 : +3V...+15V

#PROGRAM

BEGIN
PIN[1,3,5,7,9,11,14] : INPUT;
PIN[2,4,6,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[10])&NOT(PIN[12])&NOT(PIN[15]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4501,CD4501

#TEXT
Triple Gate

This device contains a
triple gate consisting
of two NAND gates each
with four inputs and a
NOR/OR gate with two
inputs.

#PIN 16
 1 : Input   1 NAND 1
 2 : Input   2 NAND 1
 3 : Input   3 NAND 1
 4 : Input   4 NAND 1
 5 : Input   1 NAND 2
 6 : Input   2 NAND 2
 7 : Input   3 NAND 2
 8 : GND
 9 : Input   4 NAND 2
10 : Output    NAND 2
11 : Input   1 NOR/OR
12 : Input   2 NOR/OR
13 : Output    NAND 1
14 : Output    NOR
15 : Output    OR
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,11,12] : INPUT;
PIN[10,13,14,16] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]:=I;
```

```
   PIN[5]&PIN[6]&PIN[7]&PIN[9]:=I;
   IF PIN[13]<>NOT(PIN[1] AND PIN[2] AND PIN[3] AND PIN[4]) THEN ERROR(1);
   IF PIN[10]<>NOT(PIN[5] AND PIN[6] AND PIN[7] AND PIN[9]) THEN ERROR(1);
   END;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[11]&PIN[12]:=I;
   IF PIN[14]<>(PIN[11] NOR PIN[12]) THEN ERROR(1);
   IF PIN[15]<>(PIN[11] OR PIN[12]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4502,CD4502

#TEXT
Hex TRI-STATE Inverter/
Driver

This device contains
six independent inverters
with inhibit and output
enable.

#PIN 16
 1 : Input   Driver 1
 2 : Output  Driver 1
 3 : Input   Driver 2
 4 : -Enable
 5 : Output  Driver 2
 6 : Input   Driver 3
 7 : Output  Driver 3
 8 : GND
 9 : Output  Driver 4
10 : Input   Driver 4
11 : Output  Driver 5
12 : Input   Driver 5
13 : Inhibit
14 : Output  Driver 6
15 : Input   Driver 6
16 : Udd

#PROGRAM

BEGIN
PIN[1,3,4,6,10,12,13,15] : INPUT;
PIN[2,5,7,9,11,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[4,12]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[6]&PIN[10]&PIN[13]&PIN[15]:=D;
   PIN[4]:=LOW;
   PIN[12]:=LOW;
   IF (NOT(PIN[2])&NOT(PIN[5])&NOT(PIN[7])&
       NOT(PIN[9])&NOT(PIN[11])&NOT(PIN[14]))<>D THEN ERROR(1);
   PIN[12]:=HIGH;
   IF (PIN[2]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14])<>0 THEN ERROR(1);
   PIN[4]:=HIGH;
   END;

LOADMODEON;
```

```
PIN[2,5,7,9,11,14] : LOAD LOW;
IF (PIN[2]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14])<>%000000 THEN ERROR(1);
PIN[2,5,7,9,11,14] : LOAD HIGH;
IF (PIN[2]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4503,CD4503

#TEXT
Hex Non-Inverting TRI-
STATE Buffer with Enable
Input

This device contains
six non-inverting buffers
with two enable inputs.

#PIN 16
 1 : Enable   -E1
 2 : Input     Gate 1
 3 : Output    Gate 1
 4 : Input     Gate 2
 5 : Output    Gate 2
 6 : Input     Gate 3
 7 : Output    Gate 3
 8 : GND
 9 : Output    Gate 4
10 : Input     Gate 4
11 : Output    Gate 5
12 : Input     Gate 5
13 : Output    Gate 6
14 : Input     Gate 6
15 : Enable   -E2
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&
       PIN[9]&PIN[11]&PIN[13])<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;
```

```
ERROR(0);
END.

#NAME 4504,CD4504

#TEXT
Hex Level Converter
(TTL-CMOS)

This device contains
six separate non-
inverting level
converters.

#PIN 16
 1 : Udd
 2 : Output    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 3
 7 : Input     Gate 3
 8 : GND
 9 : Input     Gate 4
10 : Output    Gate 4
11 : Input     Gate 5
12 : Output    Gate 5
13 : Mode
14 : Input     Gate 6
15 : Output    Gate 6
16 : Udd

#PROGRAM

BEGIN
PIN[1,3,5,7,9,11,13,14] : INPUT;
PIN[2,4,6,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,13]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14]:=D;
   IF (PIN[2]&PIN[4]&PIN[6]&
       PIN[10]&PIN[12]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4506,CD4506

#TEXT
Dual 2-Input AND/OR
Inverter Gate

This device contains
two expandable AND/OR
inverter gates each with
two inputs and inhibit
and three-state input.

#PIN 16
 1 : Input   A Gate 1
```

```
 2 : Input   B Gate 1
 3 : Input   C Gate 1
 4 : Input   D Gate 1
 5 : Input   E Gate 1
 6 : Inhibit
 7 : Output  Z Gate 2
 8 : GND
 9 : Input   A Gate 2
10 : Input   B Gate 2
11 : Input   C Gate 2
12 : Input   D Gate 2
13 : Input   E Gate 2
14 : Disable
15 : Output  Z Gate 1
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,9,10,11,12,13,14] : INPUT;
PIN[7,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[14]:=HIGH;

PIN[6]:=LOW;
FOR I:=31 DOWNTO 0 DO
   BEGIN
   PIN[1]&PIN[2]&PIN[3]&PIN[4]&PIN[5]:=I;
   PIN[9]&PIN[10]&PIN[11]&PIN[12]&PIN[13]:=I;
   PIN[14]:=LOW;
   IF PIN[15]<>(((PIN[1] AND PIN[2]) NOR (PIN[3] AND PIN[4])) AND PIN[5])
               THEN ERROR(1);
   IF PIN[7]<>(((PIN[9] AND PIN[10]) NOR (PIN[11] AND PIN[12])) AND PIN[13])
              THEN ERROR(1);
   PIN[14]:=HIGH;
   END;

PIN[6]:=HIGH;
PIN[14]:=LOW;
IF PIN[15]<>LOW THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);
PIN[14]:=HIGH;

LOADMODEON;
PIN[15,7] : LOAD LOW;
IF PIN[15]<>LOW THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);
PIN[15,7] : LOAD HIGH;
IF PIN[15]<>HIGH THEN ERROR(1);
IF PIN[7]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4507,CD4507

#TEXT
Quad 2-Input EXOR Gate

This device contains
four EXOR gates each
with two inputs.

#PIN 14
 1 : Input   1 Gate 1
```

```
 2 : Input   2 Gate 1
 3 : Output    Gate 1
 4 : Output    Gate 2
 5 : Input   1 Gate 2
 6 : Input   2 Gate 2
 7 : GND
 8 : Input   1 Gate 3
 9 : Input   2 Gate 3
10 : Output    Gate 3
11 : Output    Gate 4
12 : Input   1 Gate 4
13 : Input   2 Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,5,6,8,9,12,13] : INPUT;
PIN[3,4,10,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[1]&PIN[2]:=I;
   PIN[5]&PIN[6]:=I;
   PIN[8]&PIN[9]:=I;
   PIN[12]&PIN[13]:=I;
   IF PIN[3]<>(PIN[1] EXOR PIN[2]) THEN ERROR(1);
   IF PIN[4]<>(PIN[5] EXOR PIN[6]) THEN ERROR(1);
   IF PIN[10]<>(PIN[8] EXOR PIN[9]) THEN ERROR(1);
   IF PIN[11]<>(PIN[12] EXOR PIN[13]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4510,CD4510

#TEXT
Synchronous Up/Down Decade
Counter

This device contains a
synchronous programmable
up/down decade counter.

#PIN 16
 1 : Load
 2 : Output  Q4
 3 : Input   P4
 4 : Input   P1
 5 : -Carry
 6 : Output  Q1
 7 : -Carry Out
 8 : GND
 9 : Reset
10 : Up/-Down
11 : Output  Q2
12 : Input   P2
13 : Input   P3
14 : Output  Q3
15 : Clock
16 : Udd

#PROGRAM
```

```
BEGIN
PIN[1,3,4,5,9,10,12,13,15] : INPUT;
PIN[2,6,7,11,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,5,9]:=LOW;

PIN[9]:=HIGH;PIN[9]:=LOW;

PIN[10]:=HIGH;
FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

PIN[10]:=LOW;
FOR I:=9 DOWNTO 1 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[13]&PIN[12]&PIN[4]:=D;
   PIN[1]:=HIGH;PIN[1]:=LOW;
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>0 THEN ERROR(1);
PIN[5]:=HIGH;PIN[15]:=HIGH;PIN[15]:=LOW;PIN[5]:=LOW;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4511,CD4511

#TEXT
BCD-to-7-Segment-Latch
Decoder/Driver

This device decodes BCD
input data into control
data for 7-segment
displays.

#PIN 16
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : -LT (Lamp Test)
 4 : -BL
 5 : Strobe
 6 : BCD-Input   D
 7 : BCD-Input   A
 8 : GND
 9 : 7-Seg.-Output  e
```

```
10 : 7-Seg.-Output  d
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  a
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   PIN[5]:=LOW;PIN[5]:=HIGH;
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=0;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&
      PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4512,CD4512

#TEXT
8-Channel TRI-STATE
Data Selector

This device is normally
used as a digital signal
multiplexer selecting one
of eight inputs and rou-
ting the signal to a
three-state output. The
device has an inhibit
and an enable input.

#PIN 16
 1 : Input   0
 2 : Input   1
 3 : Input   2
 4 : Input   3
 5 : Input   4
 6 : Input   5
 7 : Input   6
 8 : GND
 9 : Input   7
10 : Inhibit
11 : Address S1
12 : Address S2
```

```
13 : Address S3
14 : Output
15 : Enable
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10,11,12,13,15] : INPUT;
PIN[14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10,15]:=HIGH;

D:=%01011010;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1]:=D;
   X:=0;
   FOR J:=7 DOWNTO 0 DO
      BEGIN
      PIN[13]&PIN[12]&PIN[11]:=J;
      PIN[15]:=LOW;
      PIN[10]:=LOW;
      X:=(X SHL 1) OR PIN[14];
      PIN[10]:=HIGH;
      IF PIN[14]<>LOW THEN ERROR(1);
      PIN[15]:=HIGH;
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[14] : LOAD LOW;
IF PIN[14]<>LOW THEN ERROR(1);
PIN[14] : LOAD HIGH;
IF PIN[14]<>HIGH THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4513,CD4513

#TEXT
BCD-to-7-Segment-Decoder/
Driver with Storage

This device decodes BCD
input data into control
data for 7-segment
displays.

#PIN 18
 1 : BCD-Input   B
 2 : BCD-Input   C
 3 : -LT (Lamp Test)
 4 : -BI
 5 : Latch Enable
 6 : BCD-Input   D
 7 : BCD-Input   A
 8 : RBI
 9 : GND
10 : RBO
11 : 7-Seg.-Output  e
12 : 7-Seg.-Output  d
```

```
13 : 7-Seg.-Output  c
14 : 7-Seg.-Output  b
15 : 7-Seg.-Output  a
16 : 7-Seg.-Output  g
17 : 7-Seg.-Output  f
18 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,8] : INPUT;
PIN[10,11,12,13,14,15,16,17] : OUTPUT;
PIN[9] : GND;
PIN[18] : +5V;
PIN[8]:=LOW;
PIN[3,4,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   PIN[5]:=LOW;PIN[5]:=HIGH;
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=0;
   D:=PIN[16]&PIN[17]&PIN[11]&PIN[12]&
      PIN[13]&PIN[14]&PIN[15];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.
•
```

```
#NAME 4516,CD4516

#TEXT
Synchronous Up/Down
Binary Counter

This device contains a
synchronous programmable
up/down binary counter.

#PIN 16
 1 : Load
 2 : Output  Q4
 3 : Input   P4
 4 : Input   P1
 5 : -Carry
 6 : Output  Q1
 7 : -Carry Out
 8 : GND
 9 : Reset
10 : Up/-Down
11 : Output  Q2
12 : Input   P2
13 : Input   P3
14 : Output  Q3
15 : Clock
16 : Udd

#PROGRAM

BEGIN
PIN[1,3,4,5,9,10,12,13,15] : INPUT;
PIN[2,6,7,11,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,5,9]:=LOW;

PIN[9]:=HIGH;PIN[9]:=LOW;

PIN[10]:=HIGH;
FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

PIN[10]:=LOW;
FOR I:=15 DOWNTO 1 DO
   BEGIN
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
   IF PIN[7]<>HIGH THEN ERROR(1);
   PIN[15]:=HIGH;PIN[15]:=LOW;
   END;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>I THEN ERROR(1);
IF PIN[7]<>LOW THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[13]&PIN[12]&PIN[4]:=D;
   PIN[1]:=HIGH;PIN[1]:=LOW;
   IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
```

```
PIN[9]:=HIGH;PIN[9]:=LOW;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>0 THEN ERROR(1);
PIN[5]:=HIGH;PIN[15]:=HIGH;PIN[15]:=LOW;PIN[5]:=LOW;
IF (PIN[2]&PIN[14]&PIN[11]&PIN[6])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4517,CD4517

#TEXT
Dual Static 64-Stage
Shift Register

This device contains
two static 64-stage
shift registers.

#PIN 16
 1 : Output  Q16  SR A
 2 : Output  Q48  SR A
 3 : Write Enable SR A
 4 : Clock        SR A
 5 : Output  Q64  SR A
 6 : Output  Q32  SR A
 7 : Data         SR A
 8 : GND
 9 : Data         SR B
10 : Output  Q32  SR B
11 : Output  Q64  SR B
12 : Clock        SR B
13 : Write Enable SR B
14 : Output  Q48  SR B
15 : Output  Q16  SR B
16 : Udd

#PROGRAM

BEGIN
PIN[3,4,7,9,12,13] : INPUT;
PIN[5,11] : OUTPUT;
PIN[8] : GND;
PIN[16] :+5V;
PIN[4,12]:=LOW;

D:=%0101010101010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1,2,6,10,14,15] : INPUT;
   PIN[3,13]:=HIGH;
   X:=D;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[7,1,6,2,9,15,10,14]:=X AND 1;
      PIN[4,12]:=HIGH;PIN[4,12]:=LOW;
      X:=X SHR 1;
      END;
   PIN[1,2,6,10,14,15] : OUTPUT;
   PIN[3,13]:=LOW;
   X:=0;Y:=0;
   FOR J:=15 DOWNTO 0 DO
      BEGIN
      IF (PIN[1]<>PIN[6]) OR (PIN[1]<>PIN[2]) OR
         (PIN[1]<>PIN[5]) THEN ERROR(1);
      X:=(X SHR 1) OR (PIN[1] SHL 15);
      IF (PIN[15]<>PIN[10]) OR (PIN[15]<>PIN[14]) OR
```

```
         (PIN[15]<>PIN[11]) THEN ERROR(1);
      Y:=(Y SHR 1) OR (PIN[15] SHL 15);
      PIN[4,12]:=HIGH;PIN[4,12]:=LOW;
      END;
   IF X<>D THEN ERROR(1);
   IF Y<>D THEN ERROR(1);
   D:=D EXOR %1111111111111111;
   END;

ERROR(0);
END.

#NAME 4518,CD4518

#TEXT
Dual Synchronous Decade
Counter

This device contains
two separate synchronous
up counters with enable
inputs and reset.

#PIN 16
 1 : Clock      Counter 1
 2 : Enable     Counter 1
 3 : Output  Q1 Counter 1
 4 : Output  Q2 Counter 1
 5 : Output  Q3 Counter 1
 6 : Output  Q4 Counter 1
 7 : Reset      Counter 1
 8 : GND
 9 : Clock      Counter 2
10 : Enable     Counter 2
11 : Output  Q1 Counter 2
12 : Output  Q2 Counter 2
13 : Output  Q3 Counter 2
14 : Output  Q4 Counter 2
15 : Reset      Counter 2
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,7,9,10,15] : INPUT;
PIN[3,4,5,6,11,12,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,2,9,10]:=LOW;
PIN[2,10]:=HIGH;

PIN[7,15]:=HIGH;PIN[7,15]:=LOW;

FOR I:=0 TO 9 DO
   BEGIN
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>I THEN ERROR(1);
   IF (PIN[14]&PIN[13]&PIN[12]&PIN[11])<>I THEN ERROR(1);
   PIN[1,9]:=HIGH;PIN[1,9]:=LOW;
   END;

PIN[7,15]:=HIGH;PIN[7,15]:=LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>0 THEN ERROR(1);
IF (PIN[14]&PIN[13]&PIN[12]&PIN[11])<>0 THEN ERROR(1);

ERROR(0);
END.
```

```
#NAME 4519,CD4519

#TEXT
Quad Common Addressable
2-to-1 Data Selectors/
Multiplexers

This device selects one
two 4-bit data sources.

#PIN 16
 1 : Input   B MP 4
 2 : Input   A MP 3
 3 : Input   B MP 3
 4 : Input   A MP 2
 5 : Input   B MP 2
 6 : Input   A MP 1
 7 : Input   B MP 1
 8 : GND
 9 : SA
10 : Output  Q MP 1
11 : Output  Q MP 2
12 : Output  Q MP 3
13 : Output  Q MP 4
14 : SB
15 : Input   A MP 4
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,14,15] : INPUT;
PIN[10,11,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9,14]:=LOW;

PIN[1,2,3,4,5,6,7,15]:=LOW;
PIN[9]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[4]&PIN[6]:=D;
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
PIN[9]:=LOW;

PIN[1,2,3,4,5,6,7,15]:=LOW;
PIN[14]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[7]:=D;
   IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;
PIN[14]:=LOW;

ERROR(0);
END.

#NAME 4520,CD4520

#TEXT
Dual Synchronous Binary
Counter
```

```
This device contains
two separate synchronous
up counters with enable
inputs and reset.

#PIN 16
 1 : Clock      Counter 1
 2 : Enable     Counter 1
 3 : Output  Q1 Counter 1
 4 : Output  Q2 Counter 1
 5 : Output  Q3 Counter 1
 6 : Output  Q4 Counter 1
 7 : Reset      Counter 1
 8 : GND
 9 : Clock      Counter 2
10 : Enable     Counter 2
11 : Output  Q1 Counter 2
12 : Output  Q2 Counter 2
13 : Output  Q3 Counter 2
14 : Output  Q4 Counter 2
15 : Reset      Counter 2
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,7,9,10,15] : INPUT;
PIN[3,4,5,6,11,12,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,2,9,10]:=LOW;
PIN[2,10]:=HIGH;

PIN[7,15]:=HIGH;PIN[7,15]:=LOW;

FOR I:=0 TO 15 DO
   BEGIN
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>I THEN ERROR(1);
   IF (PIN[14]&PIN[13]&PIN[12]&PIN[11])<>I THEN ERROR(1);
   PIN[1,9]:=HIGH;PIN[1,9]:=LOW;
   END;

PIN[7,15]:=HIGH;PIN[7,15]:=LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>0 THEN ERROR(1);
IF (PIN[14]&PIN[13]&PIN[12]&PIN[11])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 4522,CD4522

#TEXT
Programmable Decade
Counter

This decade counter
counts from a set
value back to 0 and
then restarts at 9.

#PIN 16
 1 : Output  Q3
 2 : Input   D3
 3 : Load
 4 : Inhibit
 5 : Input   D0
```

```
 6 : Clock
 7 : Output  Q0
 8 : GND
 9 : Output  Q1
10 : Reset
11 : Input   D1
12 : Zero Out
13 : Cascade
14 : Input   D2
15 : Output  Q2
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,10,11,13,14] : INPUT;
PIN[1,7,9,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,6,10]:=LOW;
PIN[13]:=HIGH;

FOR I:=9 DOWNTO 0 DO
   BEGIN
   PIN[10]:=HIGH;PIN[10]:=LOW;
   PIN[2]&PIN[14]&PIN[11]&PIN[5]:=I;
   PIN[3]:=HIGH;PIN[3]:=LOW;
   FOR J:=I DOWNTO 0 DO
      BEGIN
      IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>J THEN ERROR(1);
      PIN[6]:=HIGH;PIN[6]:=LOW;
      END;
   END;

IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>9 THEN ERROR(1);
IF PIN[12]<>LOW THEN ERROR(1);
PIN[10]:=HIGH;PIN[10]:=LOW;
IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>0 THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);
PIN[4]:=HIGH;PIN[6]:=HIGH;PIN[6]:=LOW;PIN[4]:=LOW;
IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>LOW THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 4526,CD4526

#TEXT
Programmable Binary
Counter

This binary counter counts
backwards from a set value
to 0 and then restarts at
9.

#PIN 16
 1 : Output  Q3
 2 : Input   D3
 3 : Load
 4 : Inhibit
 5 : Input   D0
 6 : Clock
 7 : Output  Q0
 8 : GND
 9 : Output  Q1
```

```
10 : Reset
11 : Input   D1
12 : Zero Out
13 : Cascade
14 : Input   D2
15 : Output  Q2
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,10,11,13,14] : INPUT;
PIN[1,7,9,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,6,10]:=LOW;
PIN[13]:=HIGH;

FOR I:=15 DOWNTO 0 DO
   BEGIN
   PIN[10]:=HIGH;PIN[10]:=LOW;
   PIN[2]&PIN[14]&PIN[11]&PIN[5]:=I;
   PIN[3]:=HIGH;PIN[3]:=LOW;
   FOR J:=I DOWNTO 0 DO
      BEGIN
      IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>J THEN ERROR(1);
      PIN[6]:=HIGH;PIN[6]:=LOW;
      END;
   END;

IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>15 THEN ERROR(1);
IF PIN[12]<>LOW THEN ERROR(1);
PIN[10]:=HIGH;PIN[10]:=LOW;
IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>0 THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);
PIN[4]:=HIGH;PIN[6]:=HIGH;PIN[6]:=LOW;PIN[4]:=LOW;
IF (PIN[1]&PIN[15]&PIN[9]&PIN[7])<>LOW THEN ERROR(1);
IF PIN[12]<>HIGH THEN ERROR(1);

ERROR(0);
END.

#NAME 4527,CD4527

#TEXT
Decimal Rate Multiplier

This device contains a
programmable decimal
rate multiplier
(Modulo 10).

#PIN 16
 1 : "9" Out
 2 : Input   C
 3 : Input   D
 4 : "9" Set
 5 : -Out
 6 : Out
 7 : Inhibit Out
 8 : GND
 9 : Clock
10 : Strobe
11 : Inhibit
12 : Cascade
13 : Clear
14 : Input   A
```

```
15 : Input   B
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,9,10,11,12,13,14,15] : INPUT;
PIN[1,5,6,7] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9,10,11,12,13]:=LOW;

PIN[12]:=LOW;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]:=1 SHL I;
   Q:=0;C:=0;E:=0;
   FOR J:=0 TO 9 DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      E:=E+PIN[7];
      END;
   IF Q<>(10-(1 SHL I)) THEN ERROR(1);
   IF C<>1 SHL I THEN ERROR(1);
   IF E<>9 THEN ERROR(1);
   END;

PIN[12]:=HIGH;
PIN[13]:=HIGH;PIN[13]:=LOW;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]&PIN[15]&PIN[14]:=1 SHL I;
   Q:=0;C:=0;
   FOR J:=0 TO 9 DO
      BEGIN
      PIN[9]:=HIGH;PIN[9]:=LOW;
      Q:=Q+PIN[5];
      C:=C+PIN[6];
      END;
   IF Q<>(10-(1 SHL I)) THEN ERROR(1);
   IF C<>10 THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4529,CD4529

#TEXT
Dual 4-Channel Analog
Data Selector

This device can be used
as a double 1-of-4 analog
data multiplexer or de-
multiplexer with common
address or as double 1-of-
4 digital selector or
distributor.

#PIN 16
 1 : STX
 2 : In-/Output  X0
 3 : In-/Output  X1
```

```
 4 : In-/Output  X2
 5 : In-/Output  X3
 6 : Address      A
 7 : Address      B
 8 : GND
 9 : In-/Output  Z
10 : In-/Output  W
11 : In-/Output  Y3
12 : In-/Output  Y2
13 : In-/Output  Y1
14 : In-/Output  Y0
15 : STY
16 : +5V...+15V

#PROGRAM

BEGIN
PIN[1,6,7,15] : INPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

PIN[9,10] : INPUT;
PIN[2,3,4,5,11,12,13,14] : OUTPUT;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[7]&PIN[6]:=I;
   PIN[9,10]:=LOW;
   IF (((PIN[5]&PIN[4]&PIN[3]&PIN[2]) SHR I) AND 1)<>LOW THEN ERROR(1);
   IF (((PIN[11]&PIN[12]&PIN[13]&PIN[14]) SHR I) AND 1)<>LOW THEN ERROR(1);
   PIN[9,10]:=HIGH;
   IF (((PIN[5]&PIN[4]&PIN[3]&PIN[2]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   IF (((PIN[11]&PIN[12]&PIN[13]&PIN[14]) SHR I) AND 1)<>HIGH THEN ERROR(1);
   END;

PIN[2,3,4,5,11,12,13,14] : INPUT;
PIN[9,10] : OUTPUT;
FOR I:=0 TO 3 DO
   BEGIN
   PIN[7]&PIN[6]:=I;
   PIN[5]&PIN[4]&PIN[3]&PIN[2]:=0;
   PIN[11]&PIN[12]&PIN[13]&PIN[14]:=0;
   IF PIN[9]<>LOW THEN ERROR(1);
   IF PIN[10]<>LOW THEN ERROR(1);
   PIN[5]&PIN[4]&PIN[3]&PIN[2]:=1 SHL I;
   PIN[11]&PIN[12]&PIN[13]&PIN[14]:=1 SHL I;
   IF PIN[9]<>HIGH THEN ERROR(1);
   IF PIN[10]<>HIGH THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4530,CD4530

#TEXT
Dual 5-Input Majority
Logic Gate

This device gives an
output of logic 1 when
more than half of the 5
input signals, A to E,
are logic 1.

#PIN 16
 1 : Input   A MjG 1
```

```
 2 : Input   B MjG 1
 3 : Input   C MjG 1
 4 : Input   D MjG 1
 5 : Input   E MjG 1
 6 : Input   W MjG 1
 7 : Output  Z MjG 1
 8 : GND
 9 : Input   A MjG 2
10 : Input   B MjG 2
11 : Input   C MjG 2
12 : Input   D MjG 2
13 : Input   E MjG 2
14 : Input   W MjG 2
15 : Output  Z MjG 2
16 : Udd


#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,9,10,11,12,13,14] : INPUT;
PIN[7,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   FOR J:=0 TO 4 DO
      BEGIN
      PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1]:=(1 SHL I) OR (1 SHL (I+1));
      PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9]:=(1 SHL I) OR (1 SHL (I+1));
      PIN[6,14]:=LOW;
      IF PIN[7]<>HIGH THEN ERROR(1);
      IF PIN[15]<>HIGH THEN ERROR(1);
      PIN[6,14]:=HIGH;
      IF PIN[7]<>LOW THEN ERROR(1);
      IF PIN[15]<>LOW THEN ERROR(1);
      PIN[5]&PIN[4]&PIN[3]&PIN[2]&PIN[1]:=(1 SHL I) OR (1 SHL (I+1)) OR
                                          (1 SHL J);
      PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[9]:=(1 SHL I) OR (1 SHL (I+1)) OR
                                          (1 SHL J);
      IF (J=I) OR (J=I+1) THEN
         BEGIN
         IF PIN[7]<>LOW THEN ERROR(1);
         IF PIN[15]<>LOW THEN ERROR(1);
         END
      ELSE
         BEGIN
         IF PIN[7]<>HIGH THEN ERROR(1);
         IF PIN[15]<>HIGH THEN ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 4531,CD4531

#TEXT
13-Bit Parity Generator/
Checker

This device contains a
parity generator/checker
for 13 bits (12 data bits
plus 1 parity bit).
```

```
#PIN 16
 1 : Input   D7
 2 : Input   D6
 3 : Input   D5
 4 : Input   D4
 5 : Input   D3
 6 : Input   D2
 7 : Input   D1
 8 : GND
 9 : Output
10 : Odd/-Even
11 : Input   D12
12 : Input   D11
13 : Input   D10
14 : Input   D9
15 : Input   D8
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,10,11,12,13,14,15] : INPUT;
PIN[9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[10]:=LOW;
D:=0;
FOR I:=0 TO 11 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[1]&
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]:=D;
   IF PIN[9]<>(I MOD 2) THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;

PIN[10]:=HIGH;
D:=0;
FOR I:=0 TO 11 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[1]&
   PIN[2]&PIN[3]&PIN[4]&PIN[5]&PIN[6]&PIN[7]:=D;
   IF PIN[9]<>NOT(I MOD 2) THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;

ERROR(0);
END.

#NAME 4532,CD4532

#TEXT
Binary 8-to-3 Priority
Encoder

This device arranges the
eight input signals in
order of importance.
The highest numbered
input is converted into
a binary code.
The devices can be
cascaded.

#PIN 16
 1 : Input   4
```

```
 2 : Input   5
 3 : Input   6
 4 : Input   7
 5 : Enable  In
 6 : Output  C
 7 : Output  B
 8 : GND
 9 : Output  A
10 : Input   0
11 : Input   1
12 : Input   2
13 : Input   3
14 : GS
15 : Enable  Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,10,11,12,13] : INPUT;
PIN[6,7,9,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[5]:=HIGH;

D:=%00000000;
PIN[4]&PIN[3]&PIN[2]&PIN[1]&PIN[13]&
PIN[12]&PIN[11]&PIN[10]:=D;
IF (PIN[14]<>LOW) OR (PIN[15]<>HIGH) THEN ERROR(1);
D:=(D SHL 1) OR 1;

FOR I:=0 TO 7 DO
   BEGIN
   PIN[4]&PIN[3]&PIN[2]&PIN[1]&PIN[13]&
   PIN[12]&PIN[11]&PIN[10]:=D;
   IF (PIN[6]&PIN[7]&PIN[9])<>I THEN ERROR(1);
   IF (PIN[14]<>HIGH) OR (PIN[15]<>LOW) THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;

ERROR(0);
END.

#NAME 4539,CD4539

#TEXT
Dual 4-Channel Data
Selector /Multiplexer

This device contains
two 1-of-4 data selec-
tors with common ad-
dress inputs and sepa-
rate strobe inputs.

#PIN 16
 1 : -Strobe    MP 1
 2 : Input    B
 3 : Input    3 MP 1
 4 : Input    2 MP 1
 5 : Input    1 MP 1
 6 : Input    0 MP 1
 7 : Output  -Q MP 1
 8 : GND
 9 : Output  -Q MP 2
10 : Input    0 MP 2
11 : Input    1 MP 2
```

```
12 : Input    2 MP 2
13 : Input    3 MP 2
14 : Input    A
15 : -Strobe    MP 2
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,10,11,12,13,14,15] : INPUT;
PIN[7,9] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[4]&PIN[5]&PIN[6]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[1]:=LOW;
      X:=(X SHL 1) OR PIN[7];
      PIN[1]:=HIGH;
      IF PIN[7]<>LOW THEN ERROR(1);
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

D:=%1001;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[11]&PIN[10]:=D;
   X:=0;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      PIN[2]&PIN[14]:=J;
      PIN[15]:=LOW;
      X:=(X SHL 1) OR PIN[9];
      PIN[15]:=HIGH;
      IF PIN[9]<>LOW THEN ERROR(1);
      END;
   IF X<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4543,CD4543

#TEXT
BCD-to-7-Segment-Latch
Decoder/Driver

This device converts
BCD input data into
control data for 7-seg-
ment liquid crystal or
LED displays.

#PIN 16
 1 : Latch Enable
 2 : Binary-Input   C
```

```
 3 : Binary-Input   B
 4 : Binary-Input   D
 5 : Binary-Input   A
 6 : Phase Input
 7 : Blanking Input
 8 : GND
 9 : 7-Seg.-Output  a
10 : 7-Seg.-Output  b
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  d
13 : 7-Seg.-Output  e
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,6,7]:=LOW;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[2]&PIN[3]&PIN[5]:=I;
   PIN[1]:=HIGH;PIN[1]:=LOW;
   PIN[4]&PIN[2]&PIN[3]&PIN[5]:=0;
   D:=PIN[14]&PIN[15]&PIN[13]&PIN[12]&
      PIN[11]&PIN[10]&PIN[9];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4544,CD4544

#TEXT
BCD-to-7-Segment-Latch
Decoder/Driver

This device converts
BCD input data into
control data for 7-seg-
ment liquid crystal or
LED displays.

#PIN 18
 1 : Latch Enable
 2 : Binary-Input   C
 3 : Binary-Input   B
 4 : Binary-Input   D
 5 : Binary-Input   A
 6 : Phase Input
 7 : Blanking Input
 8 : RBO
```

```
 9 : GND
10 : RBI
11 : 7-Seg.-Output  a
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  c
14 : 7-Seg.-Output  d
15 : 7-Seg.-Output  e
16 : 7-Seg.-Output  g
17 : 7-Seg.-Output  f
18 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,10] : INPUT;
PIN[8,11,12,13,14,15,16,17] : OUTPUT;
PIN[9] : GND;
PIN[18] : +5V;
PIN[1,5,6,7]:=LOW;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[2]&PIN[3]&PIN[5]:=I;
   PIN[1]:=HIGH;PIN[1]:=LOW;
   PIN[4]&PIN[2]&PIN[3]&PIN[5]:=0;
   D:=PIN[16]&PIN[17]&PIN[15]&PIN[14]&
      PIN[13]&PIN[12]&PIN[11];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4547,CD4547

#TEXT
Binary-to-7-Segment-
Decoder/Driver with
Storage

This device converts
binary input data into
control data for 7-seg-
ment displays.

#PIN 16
 1 : Binary-Input   B
 2 : Binary-Input   C
 3 : N.C.
 4 : -BI
 5 : N.C.
 6 : Binary-Input   D
 7 : Binary-Input   A
 8 : GND
 9 : 7-Seg.-Output  e
10 : 7-Seg.-Output  d
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  b
```

```
13 : 7-Seg.-Output  a
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,7] : INPUT;
PIN[9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[4]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&
      PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4551,CD4551

#TEXT
Dual Common Addressable
2-Channel Data Selector/
Multiplexer

With this device one of
two 4-bit data sources
can be selected with
digital or analog signal
processing.

#PIN 16
 1 : Input   W1
 2 : Input   X0
 3 : Input   X1
 4 : Output  X
 5 : Output  Y
 6 : Input   Y0
 7 : GND
 8 : -3V...-7.5V
 9 : Control
10 : Input   Y1
11 : Input   Z0
12 : Input   Z1
13 : Output  Z
14 : Output  W
15 : Input   W0
16 : +3V...+7.5V

#PROGRAM
```

```
BEGIN
PIN[1,2,3,6,7,9,10,11,12,15] : INPUT;
PIN[4,5,13,14] : OUTPUT;
PIN[7] : GND;
PIN[16] : +5V;
PIN[8]:=LOW;

PIN[1,2,3,6,10,11,12,15]:=LOW;
PIN[9]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[6]&PIN[2]&PIN[15]:=D;
   IF (PIN[13]&PIN[5]&PIN[4]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,2,3,6,10,11,12,15]:=LOW;
PIN[9]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[12]&PIN[10]&PIN[3]&PIN[1]:=D;
   IF (PIN[13]&PIN[5]&PIN[4]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 4553,CD4553

#TEXT
Three Digit BCD Counter

This device consists of
a counter for three-seg-
ment LED or LCD displays
(count up to 1000).

#PIN 16
 1 : -Disable 2
 2 : -Disable 1
 3 : C1b
 4 : C1a
 5 : Output  Q3
 6 : Output  Q2
 7 : Output  Q1
 8 : GND
 9 : Output  Q0
10 : Latch Enable
11 : Clock Disable
12 : Clock
13 : Master Reset
14 : Overflow
15 : -Disable 3
16 : Udd

#PROGRAM
PIN[4,10,11,12,13] : INPUT;
PIN[1,2,3,6,7,9,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10,11,12,13]:=LOW;

PIN[13]:=HIGH;PIN[13]:=LOW;
WINDOW (33,11,46,14);
```

```
FOR I:=0 TO 9 DO
   BEGIN
   IF (PIN[15]&PIN[1]&PIN[2])<>%110 THEN ERROR(1);
   FOR J:=0 TO 9 DO
      BEGIN
      PIN[4]:=HIGH;PIN[4]:=LOW;
      IF (PIN[15]&PIN[1]&PIN[2])<>%101 THEN ERROR(1);
      IF (PIN[5]&PIN[6]&PIN[7]&PIN[9])<>J THEN ERROR(1);
      PIN[4]:=HIGH;PIN[4]:=LOW;
      IF (PIN[15]&PIN[1]&PIN[2])<>%011 THEN ERROR(1);
      IF (PIN[5]&PIN[6]&PIN[7]&PIN[9])<>I THEN ERROR(1);
      PIN[4]:=HIGH;PIN[4]:=LOW;
      FOR K:=0 TO 9 DO
         BEGIN
         IF (PIN[5]&PIN[6]&PIN[7]&PIN[9])<>K THEN ERROR(1);
         PIN[12]:=HIGH;PIN[12]:=LOW;
         END;
      END;
   GOTOXY (I+2,1);WRITE ('Û');
   GOTOXY (I+2,2);WRITE ('Û');
   END;

ERROR(0)
END.

#NAME 4555,CD4555

#TEXT
Dual 1-of-4-Decoder/
Demultiplexer (2-to-4)

This device contains
two separate non-
inverting 1-of-4 (or
2-to-4) decoders which
can be used either as a
decoder or distributor.
With an external inverter
they can also be employed
as a 1-to-8-decoder or
distributor.

#PIN 16
 1 : Disable    DMP A
 2 : Input    A DMP A
 3 : Input    B DMP A
 4 : Output  Q0 DMP A
 5 : Output  Q1 DMP A
 6 : Output  Q2 DMP A
 7 : Output  Q3 DMP A
 8 : GND
 9 : Output  Q3 DMP B
10 : Output  Q2 DMP B
11 : Output  Q1 DMP B
12 : Output  Q0 DMP B
13 : Input    B DMP B
14 : Input    A DMP B
15 : Disable    DMP B
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
```

```
PIN[16] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]:=I;
   PIN[13]&PIN[14]:=I;
   PIN[1,15]:=LOW;
   IF (PIN[7]&PIN[6]&PIN[5]&PIN[4])
      <>(1 SHL I) THEN ERROR(1);
   IF (PIN[9]&PIN[10]&PIN[11]&PIN[12])
      <>(1 SHL I) THEN ERROR(1);
   PIN[1,15]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[5]&PIN[4])<>0 THEN ERROR(1);
   IF (PIN[9]&PIN[10]&PIN[11]&PIN[12])<>0 THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4556,CD4556

#TEXT
Dual 1-of-4-Decoder/
Demultiplexer (2-to-4)

This device contains
two separate inverting
1-of-4 (or 2-to-4) deco-
ders which can be used
either as a decoder or
distributor. With an
external inverter they
can also be employed as
a 1-to-8 decoder or
distributor.

#PIN 16
 1 : Disable     DMP A
 2 : Input    A  DMP A
 3 : Input    B  DMP A
 4 : Output  -Q0 DMP A
 5 : Output  -Q1 DMP A
 6 : Output  -Q2 DMP A
 7 : Output  -Q3 DMP A
 8 : GND
 9 : Output  -Q3 DMP B
10 : Output  -Q2 DMP B
11 : Output  -Q1 DMP B
12 : Output  -Q0 DMP B
13 : Input    B  DMP B
14 : Input    A  DMP B
15 : Disable     DMP B
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[3]&PIN[2]:=I;
   PIN[13]&PIN[14]:=I;
   PIN[1,15]:=LOW;
```

```
   IF (NOT(PIN[7])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[4]))
      <>(1 SHL I) THEN ERROR(1);
   IF (NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11])&NOT(PIN[12]))
      <>(1 SHL I) THEN ERROR(1);
   PIN[1,15]:=HIGH;
   IF (NOT(PIN[7])&NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[4]))<>0 THEN ERROR(1);
   IF (NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11])&NOT(PIN[12]))<>0 THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4557,CD4557

#TEXT
Variable Shift Register,
1 to 64 Bit

This device contains a
static, serial shift
register which can be
used from 1 to 64 stages.

#PIN 16
 1 : Input   L2
 2 : Input   L1
 3 : Reset
 4 : Clock
 5 : -Clock Enable
 6 : Input    B
 7 : Input    A
 8 : GND
 9 : Select   A/-B
10 : Output   Q
11 : Output  -Q
12 : Input   L32
13 : Input   L16
14 : Input   L8
15 : Input   L4
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,12,13,14,15] : INPUT;
PIN[10,11] : Output ;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,4,5]:=LOW;

FOR I:=0 TO 5 DO
   BEGIN
   PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[1]&PIN[2]:=(1 SHL I)-1;
   PIN[9]:=HIGH;
   PIN[3]:=HIGH;PIN[3]:=LOW;
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      PIN[7]:=J AND 1;
      PIN[4]:=HIGH;PIN[4]:=LOW;
      END;
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      IF PIN[10]<>(J AND 1) THEN ERROR(1);
      PIN[4]:=HIGH;PIN[4]:=LOW;
      END;
   PIN[9]:=LOW;
   PIN[3]:=HIGH;PIN[3]:=LOW;
```

```
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      PIN[6]:=J AND 1;
      PIN[4]:=HIGH;PIN[4]:=LOW;
      END;
   FOR J:=1 TO 1 SHL I DO
      BEGIN
      IF NOT(PIN[11])<>(J AND 1) THEN ERROR(1);
      PIN[4]:=HIGH;PIN[4]:=LOW;
      END;
   END;

ERROR(0);
END.

#NAME 4558,CD4558

#TEXT
Binary-to-7-Segment
Decoder/Driver with
Storage

This device converts
binary input data into
control data for 7-seg-
ment displays.

#PIN 16
 1 : Binary-Input   B
 2 : Binary-Input   C
 3 : Enable Latch
 4 : -RBO
 5 : -RBI
 6 : Binary-Input   D
 7 : Binary-Input   A
 8 : GND
 9 : 7-Seg.-Output  e
10 : 7-Seg.-Output  d
11 : 7-Seg.-Output  c
12 : 7-Seg.-Output  b
13 : 7-Seg.-Output  a
14 : 7-Seg.-Output  g
15 : 7-Seg.-Output  f
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,5,6,7] : INPUT;
PIN[4,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3]:=LOW;
PIN[3,5]:=HIGH;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[6]&PIN[2]&PIN[1]&PIN[7]:=I;
   D:=PIN[14]&PIN[15]&PIN[9]&PIN[10]&
      PIN[11]&PIN[12]&PIN[13];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
```

```
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4560,CD4560

#TEXT
4-Bit-BCD-Adder

This device contains
a BCD adder which gives
the sum of two 4-bit-
BCD numbers and has
look-ahead carry provi-
sion.

#PIN 16
 1 : Input   A2
 2 : Input   B2
 3 : Input   A3
 4 : Input   B3
 5 : Input   A4
 6 : Input   B4
 7 : Input   C0
 8 : GND
 9 : Output  C4
10 : Output  ä4
11 : Output  ä3
12 : Output  ä2
13 : Output  ä1
14 : Input   B1
15 : Input   A1
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,14,15] : INPUT;
PIN[9,10,11,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

FOR C:=0 TO 1 DO
   BEGIN
   PIN[7]:=C;
   FOR A:=0 TO 9 DO
      BEGIN
      PIN[5]&PIN[3]&PIN[1]&PIN[15]:=A;
      FOR B:=A MOD 3 TO 9 BY 3 DO
         BEGIN
         PIN[6]&PIN[4]&PIN[2]&PIN[14]:=B;
         IF (PIN[9]*10+(PIN[10]&PIN[11]&PIN[12]&PIN[13]))<>(C+A+B) THEN
         ERROR(1);
         END;
      END;
   END;

ERROR(0);
END.

#NAME 4561,CD4561

#TEXT
```

```
9-Complementer

This devicce provides
the nines complement
of a 4-bit BCD number.

#PIN 14
 1 : Input   A1
 2 : Input   A2
 3 : Input   A3
 4 : Input   A4
 5 : Complement disable
 6 : -Complement disable
 7 : GND
 8 : N.C.
 9 : -Enable
10 : Output  F4
11 : Output  F3
12 : Output  F2
13 : Output  F1
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,9] : INPUT;
PIN[10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[9]:=LOW;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[4]&PIN[3]&PIN[2]&PIN[1]:=I;
   PIN[5]:=LOW;PIN[6]:=HIGH;
   IF (PIN[10]&PIN[11]&PIN[12]&PIN[13])<>I THEN ERROR(1);
   PIN[5]:=HIGH;PIN[6]:=LOW;
   IF (PIN[10]&PIN[11]&PIN[12]&PIN[13])<>(9-I) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4562,CD4562

#TEXT
128-Stage Static Shift
Register

This device contains a
static, 128-stage shift
register. The output data
is available at 16-bit
stages.

#PIN 14
 1 : Output  Q64
 2 : Output  Q96
 3 : Output  Q128
 4 : N.C.
 5 : Clock
 6 : Output  Q112
 7 : GND
 8 : Output  Q80
 9 : Output  Q48
10 : Output  Q16
11 : N.C.
```

```
12 : Data In
13 : Output  Q32
14 : Udd

#PROGRAM

BEGIN
PIN[5,12] : INPUT;
PIN[1,2,3,6,8,9,10,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[5]:=LOW;

D:=%0101010101010101;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[12]:=D AND (1 SHL (15-J));
      PIN[5]:=HIGH;PIN[5]:=LOW;
      END;
   FOR J:=0 TO 7 DO
      BEGIN
      X:=0;
      FOR K:=0 TO 15 DO
         BEGIN
         B:=((PIN[3]&PIN[6]&PIN[2]&PIN[8]&PIN[1]&PIN[9]&PIN[13]&PIN[10])
              SHR J) AND 1;
         X:=(X SHL 1) OR B;
         PIN[5]:=HIGH;PIN[5]:=LOW;
         END;
      IF X<>D THEN ERROR(1);
      END;
   D:=D EXOR %1111111111111111;
   END;

ERROR(0);
END.

#NAME 4566,CD4566

#TEXT
Time Base Generator

This device is a univer-
sal time base generator
having divide-by-10,
divide-by-5 or 6 and a
monostable multivibrator.

Warning !
Unfortunately the
monostable multivibrator
cannot be tested due to
technical reasons concer-
ning the hardware.

#PIN 16
 1 : -Clock     Divider A
 2 : Reset
 3 : Output  Q0 Divider A
 4 : Output  Q1 Divider A
 5 : Output  Q2 Divider A
 6 : Output  Q3 Divider A
 7 : Input  -I1 (MV)
 8 : GND
 9 : Input   I0 (MV)
```

```
10 : Output  QM (MV)
11 : Control 56 Divider B
12 : Output  Q0 Divider B
13 : Output  Q1 Divider B
14 : Output  Q2 Divider B
15 : -Clock     Divider B
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,11,15] : INPUT;
PIN[3,4,5,6,12,13,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,15]:=HIGH;

PIN[2]:=HIGH;PIN[2]:=LOW;
FOR I:=0 TO 9 DO
   BEGIN
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>I THEN ERROR(1);
   PIN[1]:=LOW;PIN[1]:=HIGH;
   END;

PIN[11]:=HIGH;
PIN[2]:=HIGH;PIN[2]:=LOW;
FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[14]&PIN[13]&PIN[12])<>I THEN ERROR(1);
   PIN[15]:=LOW;PIN[15]:=HIGH;
   END;

PIN[11]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
FOR I:=0 TO 5 DO
   BEGIN
   IF (PIN[14]&PIN[13]&PIN[12])<>I THEN ERROR(1);
   PIN[15]:=LOW;PIN[15]:=HIGH;
   END;

ERROR(0);
END.

#NAME 4572,CD4572

#TEXT
Hex Gate

This device contains
four inverters, a NOR
and a NAND gate each
with two inputs.

#PIN 16
 1 : Output    Inv. 1
 2 : Input     Inv. 1
 3 : Output    Inv. 2
 4 : Input     Inv. 2
 5 : Output    NOR
 6 : Input   1 NOR
 7 : Input   2 NOR
 8 : GND
 9 : Output    Inv. 3
10 : Input     Inv. 3
11 : Output    Inv. 4
12 : Input     Inv. 4
```

```
13 : Output    NAND
14 : Input   1 NAND
15 : Input   2 NAND
16 : Udd

#PROGRAM

BEGIN
PIN[2,4,6,7,10,12,14,15] : INPUT;
PIN[1,3,5,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[10]&PIN[12]:=D;
   IF (NOT(PIN[1])&NOT(PIN[3])&NOT(PIN[9])&NOT(PIN[11]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[6]&PIN[7]:=I;
   PIN[14]&PIN[15]:=I;
   IF PIN[5]<>(PIN[6] NOR PIN[7]) THEN ERROR(1);
   IF PIN[13]<>(PIN[14] NAND PIN[15]) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 4583,CD4583

#TEXT
Dual Schmitt-Trigger

This device consists of
a schmitt trigger with
adjustable threshold
value, three-state out-
puts and an exclusive
NOR gate.

#PIN 16
 1 : Output   Bcom
 2 : Input    Bpos
 3 : Input    Bneg
 4 : Output   Aout
 5 : Input    Aneg
 6 : Input    Apos
 7 : Output   Acom
 8 : GND
 9 : Input    Ain
10 : Output   Bout
11 : Output  -Aout
12 : Output  -Bout
13 : Input    EO
14 : Output   A NEXOR B
15 : Input    Bin
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,5,6,9,13,15] : INPUT;
PIN[1,4,7,10,11,12,14] : OUTPUT;
```

```
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,5,13]:=LOW;
PIN[2,6]:=HIGH;

FOR I:=0 TO 3 DO
   BEGIN
   PIN[9]&PIN[15]:=I;
   IF PIN[4]<>PIN[9] THEN ERROR(1);
   IF PIN[10]<>PIN[15] THEN ERROR(1);
   IF NOT(PIN[7])<>PIN[9] THEN ERROR(1);
   IF NOT(PIN[1])<>PIN[15] THEN ERROR(1);
   PIN[13]:=HIGH;
   IF NOT(PIN[11])<>PIN[9] THEN ERROR(1);
   IF NOT(PIN[12])<>PIN[15] THEN ERROR(1);
   PIN[13]:=LOW;
   IF PIN[14]<>(PIN[9] EXOR PIN[15]) THEN ERROR(1);
   END;

LOADMODEON;
PIN[11,12] : LOAD LOW;
IF (PIN[11]<>LOW) OR (PIN[12]<>LOW) THEN ERROR(1);
PIN[11,12] : LOAD HIGH;
IF (PIN[11]<>HIGH) OR (PIN[12]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 4584,CD4584

#TEXT
Hex Inverting
Schmitt Trigger

This device contains
six separate inverting
schmitt triggers.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : Udd

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
```

```
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 4585,CD4585

#TEXT
4-Bit Comparator

This device contains a
2 x 4-bit comparator.
It indicates which of
two 4-bit words is larger
or whether they are equal.

#PIN 16
 1 : Input   B2
 2 : Input   A2
 3 : Output  A=B
 4 : Carry Input. A>B
 5 : Carry Input. A<B
 6 : Carry Input. A=B
 7 : Input   A1
 8 : GND
 9 : Input   B1
10 : Input   A0
11 : Input   B0
12 : Output  A<B
13 : Output  A>B
14 : Input   B3
15 : Input   A3
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,5,6,7,9,10,11,14,15] : INPUT;
PIN[3,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[5]:=HIGH;PIN[6]:=LOW;PIN[4]:=LOW;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW ) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[5]:=LOW;PIN[6]:=HIGH;PIN[4]:=LOW;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
```

```
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW ) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[5]:=LOW;PIN[6]:=LOW;PIN[4]:=HIGH;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   C:=C+1;
   END;

ERROR(0);
END.

#NAME 4724,CD4724

#TEXT
8-Bit Addressable
Latch

This device contains an
addressable 8-bit latch
with enable and clear.

#PIN 16
 1 : Address A0
 2 : Address A1
 3 : Address A2
 4 : Output  Q0
 5 : Output  Q1
 6 : Output  Q2
 7 : Output  Q3
 8 : GND
 9 : Output  Q4
10 : Output  Q5
11 : Output  Q6
12 : Output  Q7
13 : Input
14 : -Enable
15 : Clear
```

```
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,13,14,15] : INPUT;
PIN[4,5,6,7,9,10,11,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=LOW;
PIN[14]:=HIGH;

D:=%01011010;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 7 DO
      BEGIN
      PIN[13]:=(D SHR J) AND 1;
      PIN[3]&PIN[2]&PIN[1]:=J;
      PIN[14]:=LOW;PIN[14]:=HIGH;
      END;
   IF (PIN[12]&PIN[11]&PIN[10]&PIN[9]&
       PIN[7]&PIN[6]&PIN[5]&PIN[4])<>D THEN ERROR(1);
   D:=D EXOR %11111111;
   END;

PIN[15]:=HIGH;PIN[15]:=LOW;
IF (PIN[12]&PIN[11]&PIN[10]&PIN[9]&
    PIN[7]&PIN[6]&PIN[5]&PIN[4])<>0 THEN ERROR(1);

ERROR(0);
END.
•
```

```
#NAME 5002,CD5002

#TEXT
BCD-to-7-Segment-Decoder/
Driver

This device decodes
BCD input data into
control data for 7-seg-
ment displays.

#PIN 16
 1 : RBO
 2 : RBI
 3 : BI
 4 : BCD-Input   A
 5 : BCD-Input   B
 6 : BCD-Input   C
 7 : BCD-Input   D
 8 : GND
 9 : 7-Seg.-Output  g
10 : 7-Seg.-Output  f
11 : 7-Seg.-Output  e
12 : 7-Seg.-Output  d
13 : 7-Seg.-Output  c
14 : 7-Seg.-Output  b
15 : 7-Seg.-Output  a
16 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,7] : INPUT;
PIN[1,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,3]:=LOW;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[7]&PIN[6]&PIN[5]&PIN[4]:=I;
   D:=PIN[9]&PIN[10]&PIN[11]&PIN[12]&
      PIN[13]&PIN[14]&PIN[15];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I=10) AND (D<>63) THEN ERROR(1);
   IF (I=11) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=12) AND (D<>91) THEN ERROR(1);
   IF (I=13) AND (D<>79) THEN ERROR(1);
   IF (I=14) AND (D<>102) THEN ERROR(1);
   IF (I=15) AND (D<>109) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 5012,CD5012

#TEXT
```

```
Hex TRI-STATE Buffer
with 2 Enable Inputs

This device contains
six non-inverting buffers
with two enable inputs.

#PIN 16
 1 : Enable    DIS1
 2 : Input     Gate 1
 3 : Output    Gate 1
 4 : Input     Gate 2
 5 : Output    Gate 2
 6 : Input     Gate 3
 7 : Output    Gate 3
 8 : GND
 9 : Output    Gate 4
10 : Input     Gate 4
11 : Output    Gate 5
12 : Input     Gate 5
13 : Output    Gate 6
14 : Input     Gate 6
15 : Enable    DIS2
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&
       PIN[9]&PIN[11]&PIN[13])<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 5020,CD5020

#TEXT
Hex Inverting Level
Converter

This device contains
six separate inver-
ting level converters.
```

```
#PIN 16
 1 : Ucc
 2 : Output    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 3
 7 : Input     Gate 3
 8 : GND
 9 : Input     Gate 4
10 : Output    Gate 4
11 : Input     Gate 5
12 : Output    Gate 5
13 : N.C.
14 : Input     Gate 6
15 : Output    Gate 6
16 : Udd

#PROGRAM

BEGIN
PIN[1,3,5,7,9,11,14] : INPUT;
PIN[2,4,6,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[5]&PIN[7]&PIN[9]&PIN[11]&PIN[14]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[10])&NOT(PIN[12])&NOT(PIN[15]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 5022,CD5022

#TEXT
BCD-to-7-Segment-Decoder/
Driver

This device converts
BCD input data into
control data for 7-seg-
ment displays.

#PIN 16
 1 : RBO
 2 : RBI
 3 : BI
 4 : BCD-Input   A
 5 : BCD-Input   B
 6 : BCD-Input   C
 7 : BCD-Input   D
 8 : GND
 9 : 7-Seg.-Output  g
10 : 7-Seg.-Output  f
11 : 7-Seg.-Output  e
12 : 7-Seg.-Output  d
13 : 7-Seg.-Output  c
14 : 7-Seg.-Output  b
15 : 7-Seg.-Output  a
16 : Udd
```

```
#PROGRAM

BEGIN
PIN[2,3,4,5,6,7] : INPUT;
PIN[1,9,10,11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,3]:=LOW;

FOR I:=0 TO 15 DO
   BEGIN
   PIN[7]&PIN[6]&PIN[5]&PIN[4]:=I;
   D:=PIN[9]&PIN[10]&PIN[11]&PIN[12]&
      PIN[13]&PIN[14]&PIN[15];
   IF (I=0) AND (D<>63) THEN ERROR(1);
   IF (I=1) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=2) AND (D<>91) THEN ERROR(1);
   IF (I=3) AND (D<>79) THEN ERROR(1);
   IF (I=4) AND (D<>102) THEN ERROR(1);
   IF (I=5) AND (D<>109) THEN ERROR(1);
   IF (I=6) AND (D<>124) AND (D<>125) THEN ERROR(1);
   IF (I=7) AND (D<>7) AND (D<>39) THEN ERROR(1);
   IF (I=8) AND (D<>127) THEN ERROR(1);
   IF (I=9) AND (D<>103) AND (D<>111) THEN ERROR(1);
   IF (I=10) AND (D<>63) THEN ERROR(1);
   IF (I=11) AND (D<>6) AND (D<>48) THEN ERROR(1);
   IF (I=12) AND (D<>91) THEN ERROR(1);
   IF (I=13) AND (D<>79) THEN ERROR(1);
   IF (I=14) AND (D<>102) THEN ERROR(1);
   IF (I=15) AND (D<>109) THEN ERROR(1);
   END;

ERROR(0);
END.

#NAME 5024,CD5024

#TEXT
Quad TRI-STATE Buffer

This device contains
four separate non-
inverting buffers
with three-state
outputs.

#PIN 14
 1 : Disable   Gate 1
 2 : Input     Gate 1
 3 : Output    Gate 1
 4 : Disable   Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input     Gate 3
10 : Disable   Gate 3
11 : Output    Gate 4
12 : Input     Gate 4
13 : Disable   Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
```

```
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,10,13]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[5]&PIN[9]&PIN[12]:=D;
   PIN[1,4,10,13]:=LOW;
   IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>D THEN ERROR(1);
   PIN[1,4,10,13]:=HIGH;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[3,6,8,11] : LOAD LOW;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,6,8,11] : LOAD HIGH;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 5025,CD5025

#TEXT
Quad TRI-STATE Buffer

This device contains
four separate non-
inverting buffers
with three-state
outputs.

#PIN 14
 1 : -Disable  Gate 1
 2 : Input     Gate 1
 3 : Output    Gate 1
 4 : -Disable  Gate 2
 5 : Input     Gate 2
 6 : Output    Gate 2
 7 : GND
 8 : Output    Gate 3
 9 : Input     Gate 3
10 : -Disable  Gate 3
11 : Output    Gate 4
12 : Input     Gate 4
13 : -Disable  Gate 4
14 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,5,9,10,12,13] : INPUT;
PIN[3,6,8,11] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[1,4,10,13]:=LOW;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[5]&PIN[9]&PIN[12]:=D;
   PIN[1,4,10,13]:=HIGH;
   IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>D THEN ERROR(1);
```

```
   PIN[1,4,10,13]:=LOW;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[3,6,8,11] : LOAD LOW;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,6,8,11] : LOAD HIGH;
IF (PIN[3]&PIN[6]&PIN[8]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 5026,CD5026

#TEXT
Decade Counter

This device contains a
decade up counter having
two reset functions. The
device can be used as
binary, quinary or decade
counter.

#PIN 14
 1 : N.C.
 2 : Clock A
 3 : Reset 9
 4 : Reset 9
 5 : Reset 0
 6 : Reset 0
 7 : GND
 8 : N.C.
 9 : Clock B
10 : Output  QA
11 : Output  QB
12 : Output  QC
13 : Output  QD
14 : Udd

#PROGRAM

BEGIN
PIN[2,3,4,5,6,9] : INPUT;
PIN[10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[3,4,5,6]:=LOW;
PIN[2,9]:=HIGH;

PIN[3,4]:=HIGH;
PIN[3,4]:=LOW;
IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>9 THEN ERROR(1);

PIN[5,6]:=HIGH;
PIN[5,6]:=LOW;
IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>0 THEN ERROR(1);

PIN[2]:=LOW;PIN[2]:=HIGH;
IF PIN[10]<>HIGH THEN ERROR(1);

FOR I:=0 TO 4 DO
   BEGIN
   IF (PIN[13]&PIN[12]&PIN[11])<>I THEN ERROR(1);
   PIN[9]:=LOW;PIN[9]:=HIGH;
```

```
   END;

ERROR(0);
END.

#NAME 5027,CD5027

#TEXT
Binary Counter

This device contains a
binary up counter with
reset function. The
device can be used as a
binary, octal or hexa-
decimal counter.

#PIN 14
 1 : N.C.
 2 : Clock A
 3 : N.C.
 4 : N.C.
 5 : Reset 0
 6 : Reset 0
 7 : GND
 8 : N.C.
 9 : Clock B
10 : Output  QA
11 : Output  QB
12 : Output  QC
13 : Output  QD
14 : Udd

#PROGRAM

BEGIN
PIN[2,5,6,9] : INPUT;
PIN[10,11,12,13] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[5,6]:=LOW;
PIN[2,9]:=HIGH;

PIN[5,6]:=HIGH;
PIN[5,6]:=LOW;
IF (PIN[13]&PIN[12]&PIN[11]&PIN[10])<>0 THEN ERROR(1);

PIN[2]:=LOW;PIN[2]:=HIGH;
IF PIN[10]<>HIGH THEN ERROR(1);

FOR I:=0 TO 7 DO
   BEGIN
   IF (PIN[13]&PIN[12]&PIN[11])<>I THEN ERROR(1);
   PIN[9]:=LOW;PIN[9]:=HIGH;
   END;

ERROR(0);
END.

#NAME 40014,CD40014

#TEXT
Hex Inverting
Schmitt-Trigger

This device contains
six separate inverting
```

schmitt triggers.

```
#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : Udd

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 40085,CD40085

#TEXT
4-Bit Comparator

This device contains a
2 x 4-bit comparator. The
device indicates which of
two 4-bit words is the
largest or whether both
are equal.

#PIN 16
 1 : Input   B2
 2 : Input   A2
 3 : Output  A=B
 4 : Carry Input. A>B
 5 : Carry Input. A<B
 6 : Carry Input. A=B
 7 : Input   A1
 8 : GND
 9 : Input   B1
10 : Input   A0
11 : Input   B0
12 : Output  A<B
13 : Output  A>B
14 : Input   B3
15 : Input   A3
16 : Udd
```

```
#PROGRAM

BEGIN
PIN[1,2,4,5,6,7,9,10,11,14,15] : INPUT;
PIN[3,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

PIN[5]:=HIGH;PIN[6]:=LOW;PIN[4]:=LOW;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW ) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[5]:=LOW;PIN[6]:=HIGH;PIN[4]:=LOW;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW ) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>HIGH) OR (PIN[13]<>LOW) THEN ERROR(1);
   C:=C+1;
   END;

PIN[5]:=LOW;PIN[6]:=LOW;PIN[4]:=HIGH;
PIN[15]&PIN[2]&PIN[7]&PIN[10]:=0;
PIN[14]&PIN[1]&PIN[9]&PIN[11]:=0;
IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);

C:=0;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   C:=C+1;
   PIN[14]&PIN[1]&PIN[9]&PIN[11]:=1 SHL C;
   IF (PIN[12]<>HIGH) OR (PIN[3]<>LOW) OR (PIN[13]<>LOW) THEN ERROR(1);
   PIN[15]&PIN[2]&PIN[7]&PIN[10]:=1 SHL C;
   IF (PIN[12]<>LOW) OR (PIN[3]<>LOW) OR (PIN[13]<>HIGH) THEN ERROR(1);
   C:=C+1;
```

```
   END;

ERROR(0);
END.

#NAME 40097,CD40097

#TEXT
Hex TRI-STATE Buffer
with Two Enable Inputs

This device contains
six non-inverting buffers
with two enable inputs.

#PIN 16
 1 : Enable   -E1
 2 : Input     Gate 1
 3 : Output    Gate 1
 4 : Input     Gate 2
 5 : Output    Gate 2
 6 : Input     Gate 3
 7 : Output    Gate 3
 8 : GND
 9 : Output    Gate 4
10 : Input     Gate 4
11 : Output    Gate 5
12 : Input     Gate 5
13 : Output    Gate 6
14 : Input     Gate 6
15 : Enable   -E2
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (PIN[3]&PIN[5]&PIN[7]&
       PIN[9]&PIN[11]&PIN[13])<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40098,CD40098
```

```
#TEXT
Hex TRI-STATE Buffer
with Two Enable Inputs

This device contains
six inverting buffers
with two enable inputs.

#PIN 16
 1 : Enable   -E1
 2 : Input     Gate 1
 3 : Output    Gate 1
 4 : Input     Gate 2
 5 : Output    Gate 2
 6 : Input     Gate 3
 7 : Output    Gate 3
 8 : GND
 9 : Output    Gate 4
10 : Input     Gate 4
11 : Output    Gate 5
12 : Input     Gate 5
13 : Output    Gate 6
14 : Input     Gate 6
15 : Enable   -E2
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,10,12,14,15] : INPUT;
PIN[3,5,7,9,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[1,15]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[10]&PIN[12]&PIN[14]:=D;
   PIN[1,15]:=LOW;
   IF (NOT(PIN[3])&NOT(PIN[5])&NOT(PIN[7])&
       NOT(PIN[9])&NOT(PIN[11])&NOT(PIN[13]))<>D THEN ERROR(1);
   PIN[1,15]:=HIGH;
   D:=D EXOR %111111;
   END;

LOADMODEON;
PIN[3,5,7,9,11,13] : LOAD LOW;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%000000 THEN ERROR(1);
PIN[3,5,7,9,11,13] : LOAD HIGH;
IF (PIN[3]&PIN[5]&PIN[7]&
    PIN[9]&PIN[11]&PIN[13])<>%111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40100,CD40100

#TEXT
32-Stage Static Bi-
directional Shift
Register with Serial
Input and Output and
Recirculation
```

```
This device contains a
fully static shift re-
gister with recirculation.

#PIN 16
 1 : N.C.
 2 : Clock Inhibit
 3 : Clock In
 4 : Shift Left Out
 5 : N.C.
 6 : Shift Left In
 7 : N.C.
 8 : GND
 9 : REC Control
10 : N.C.
11 : Shift Right In
12 : Shift Right Out
13 : Left/-Right
14 : N.C.
15 : N.C.
16 : +5V

#PROGRAM

BEGIN
PIN[2,3,6,9,11,13] : INPUT;
PIN[4,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[9]:=HIGH;

D:=%01010101;

PIN[13]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 3 DO
      BEGIN
      X:=D;
      FOR K:=0 TO 7 DO
         BEGIN
         PIN[11]:=X AND 1;
         PIN[3]:=HIGH;PIN[3]:=LOW;
         X:=X SHR 1;
         END;
      END;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      X:=0;
      FOR K:=7 DOWNTO 0 DO
         BEGIN
         X:=(X SHR 1) OR (PIN[12] SHL 7);
         PIN[3]:=HIGH;PIN[3]:=LOW;
         END;
      IF X<>D THEN ERROR(1);
      END;
   D:=D EXOR %11111111;
   END;

PIN[13]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   FOR J:=0 TO 3 DO
      BEGIN
      X:=D;
      FOR K:=0 TO 7 DO
```

```
         BEGIN
         PIN[6]:=X AND 1;
         PIN[3]:=HIGH;PIN[3]:=LOW;
         X:=X SHR 1;
         END;
      END;
   FOR J:=3 DOWNTO 0 DO
      BEGIN
      X:=0;
      FOR K:=7 DOWNTO 0 DO
         BEGIN
         X:=(X SHR 1) OR (PIN[4] SHL 7);
         PIN[3]:=HIGH;PIN[3]:=LOW;
         END;
      IF X<>D THEN ERROR(1);
      END;
   D:=D EXOR %11111111;
   END;

ERROR(0);
END.

#NAME 40101,CD40101

#TEXT
9-Bit Parity Generator/
Checker

This device contains a
parity generator/checker
for 9 bits (8 data bits
plus 1 parity bit).

#PIN 14
 1 : Input   D1
 2 : Input   D2
 3 : Input   D3
 4 : Input   D4
 5 : Input   D9
 6 : Output  odd
 7 : GND
 8 : Inhibit
 9 : Output  even
10 : Input   D5
11 : Input   D6
12 : Input   D7
13 : Input   D8
14 : +5V

#PROGRAM

BEGIN
PIN[1,2,3,4,5,8,10,11,12,13] : INPUT;
PIN[6,9] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;
PIN[8]:=LOW;

D:=0;
FOR I:=0 TO 9 DO
   BEGIN
   PIN[5]&PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[4]&PIN[3]&PIN[2]&PIN[1]:=D;
   IF (NOT(PIN[9])<>(I MOD 2)) OR (PIN[6]<>(I MOD 2)) THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;

ERROR(0);
```

```
END.

#NAME 40102,CD40102

#TEXT
8-Digit Presettable
Synchronous Up Counter
(BCD)

This device contains an
eight-digit synchronous
BCD up counter. The device
is presettable and gives
a signal when the count
reaches zero.

#PIN 16
 1 : Clock
 2 : -Clear
 3 : -CI/CE
 4 : Input   J0
 5 : Input   J1
 6 : Input   J2
 7 : Input   J3
 8 : GND
 9 : -APE
10 : Input   J4
11 : Input   J5
12 : Input   J6
13 : Input   J7
14 : -CO/ZD
15 : -SPE
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,10,11,12,13,9,15] : INPUT;
PIN[14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3]:=LOW;
PIN[1,2,9,15]:=HIGH;

PIN[2]:=LOW;PIN[2]:=HIGH;
FOR I:=1 TO 99 DO
   BEGIN
   IF PIN[14]<>HIGH THEN ERROR(1);
   PIN[1]:=LOW;PIN[1]:=HIGH;
   END;
IF PIN[14]<>LOW THEN ERROR(1);


FOR I:=0 TO 7 DO
   BEGIN
   C:=1 SHL I;
   PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[7]&PIN[6]&PIN[5]&PIN[4]:=C;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   FOR J:=1 TO (C SHR 4)*10+(C AND 15) DO
      BEGIN
      IF PIN[14]<>HIGH THEN ERROR(1);
      PIN[1]:=LOW;PIN[1]:=HIGH;
      END;
   IF PIN[14]<>LOW THEN ERROR(1);
   END;

ERROR(0);
```

```
END.

#NAME 40103,CD40103

#TEXT
8-Digit Presettable
Synchronous Up Counter
(Binary)

This device contains an
eight-digit synchronous
binary up counter. The
device can be preset and
gives a signal when the
count reaches zero.

#PIN 16
 1 : Clock
 2 : -Clear
 3 : -CI/CE
 4 : Input   J0
 5 : Input   J1
 6 : Input   J2
 7 : Input   J3
 8 : GND
 9 : -APE
10 : Input   J4
11 : Input   J5
12 : Input   J6
13 : Input   J7
14 : -CO/ZD
15 : -SPE
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,10,11,12,13,9,15] : INPUT;
PIN[14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3]:=LOW;
PIN[1,2,9,15]:=HIGH;

PIN[2]:=LOW;PIN[2]:=HIGH;
FOR I:=1 TO 255 DO
   BEGIN
   IF PIN[14]<>HIGH THEN ERROR(1);
   PIN[1]:=LOW;PIN[1]:=HIGH;
   END;
IF PIN[14]<>LOW THEN ERROR(1);


FOR I:=0 TO 7 DO
   BEGIN
   C:=1 SHL I;
   PIN[13]&PIN[12]&PIN[11]&PIN[10]&PIN[7]&PIN[6]&PIN[5]&PIN[4]:=C;
   PIN[9]:=LOW;PIN[9]:=HIGH;
   FOR J:=1 TO C DO
      BEGIN
      IF PIN[14]<>HIGH THEN ERROR(1);
      PIN[1]:=LOW;PIN[1]:=HIGH;
      END;
   IF PIN[14]<>LOW THEN ERROR(1);
   END;

ERROR(0);
```

```
END.

#NAME 40105,CD40105

#TEXT
CMOS FIFO-Register
(4 Bits x 16 Words,
TRI-STATE)

The CD40105 is a flexible
first-in-first-out regis-
ter and can store up to
16 words of 4-bits each.
The outputs are three-
state.

#PIN 16
 1 : Output Enable
 2 : Data In Ready
 3 : Shift In
 4 : Input   D0
 5 : Input   D1
 6 : Input   D2
 7 : Input   D3
 8 : GND
 9 : Master Reset
10 : Output  Q3
11 : Output  Q2
12 : Output  Q1
13 : Output  Q0
14 : Data Out Ready
15 : Shift Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,14,15] : INPUT;
PIN[10,11,12,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[3,9,15]:=LOW;
PIN[1,2,14]:=HIGH;

D:=%0101;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=HIGH;PIN[9]:=LOW;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[7]&PIN[6]&PIN[5]&PIN[4]:=D;
      PIN[3]:=HIGH;PIN[3]:=LOW;
      D:=D EXOR %1111;
      END;
   FOR J:=0 TO 15 DO
      BEGIN
      PIN[1]:=LOW;
      IF (PIN[10]&PIN[11]&PIN[12]&PIN[13])<>D THEN ERROR(1);
      PIN[1]:=HIGH;
      PIN[15]:=HIGH;PIN[15]:=LOW;
      D:=D EXOR %1111;
      END;
   D:=D EXOR %1111;
   END;

LOADMODEON;
```

```
PIN[10,11,12,13] : LOAD LOW;
IF (PIN[10]&PIN[11]&PIN[12]&PIN[13])<>%0000 THEN ERROR(1);
PIN[10,11,12,13] : LOAD HIGH;
IF (PIN[10]&PIN[11]&PIN[12]&PIN[13])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40106,CD40106

#TEXT
Hex Inverting
Schmitt-Trigger

This device contains
six separate inverting
schmitt triggers.

#PIN 14
 1 : Input     Gate 1
 2 : Output    Gate 1
 3 : Input     Gate 2
 4 : Output    Gate 2
 5 : Input     Gate 3
 6 : Output    Gate 3
 7 : GND
 8 : Output    Gate 4
 9 : Input     Gate 4
10 : Output    Gate 5
11 : Input     Gate 5
12 : Output    Gate 6
13 : Input     Gate 6
14 : Udd

#PROGRAM

BEGIN
PIN[1,3,5,9,11,13] : INPUT;
PIN[2,4,6,8,10,12] : OUTPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[1]&PIN[3]&PIN[5]&PIN[9]&PIN[11]&PIN[13]:=D;
   IF (NOT(PIN[2])&NOT(PIN[4])&NOT(PIN[6])&
       NOT(PIN[8])&NOT(PIN[10])&NOT(PIN[12]))<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

ERROR(0);
END.

#NAME 40109,CD40109

#TEXT
Quad TRI-STATE Level
Converter

This device contains
four separate non-
inverting level con-
verters with three-
state outputs.
```

```
#PIN 16
 1 : Ucc
 2 : Enable    Gate 1
 3 : Input     Gate 1
 4 : Output    Gate 1
 5 : Output    Gate 2
 6 : Input     Gate 2
 7 : Enable    Gate 2
 8 : GND
 9 : Enable    Gate 3
10 : Input     Gate 3
11 : Output    Gate 3
12 : N.C.
13 : Output    Gate 4
14 : Input     Gate 4
15 : Enable    Gate 4
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,6,7,9,10,14,15] : INPUT;
PIN[4,5,11,13] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2,7,9,15]:=LOW;
PIN[1]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[3]&PIN[6]&PIN[10]&PIN[14]:=D;
   PIN[2,7,9,15]:=HIGH;
   IF (PIN[4]&PIN[5]&PIN[11]&PIN[13])<>D THEN ERROR(1);
   PIN[2,7,9,15]:=LOW;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[4,5,11,13] : LOAD LOW;
IF (PIN[4]&PIN[5]&PIN[11]&PIN[13])<>%0000 THEN ERROR(1);
PIN[4,5,11,13] : LOAD HIGH;
IF (PIN[4]&PIN[5]&PIN[11]&PIN[13])<>%1111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40147,CD40147

#TEXT
BCD-10-to-4-Priority
Encoder

This device arranges the
10 input signals in order
of priority. The highest-
numbered input is con-
verted into a BCD number.
The devices can be
cascaded.

#PIN 16
 1 : Input   4
 2 : Input   5
 3 : Input   6
 4 : Input   7
```

```
 5 : Input   8
 6 : Output  C
 7 : Output  B
 8 : GND
 9 : Output  A
10 : Input   9
11 : Input   1
12 : Input   2
13 : Input   3
14 : Output  D
15 : Input   0
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,10,11,12,13,15] : INPUT;
PIN[6,7,9,14] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;

D:=%00000000;
PIN[10]&PIN[5]&PIN[4]&PIN[3]&PIN[2]&
PIN[1]&PIN[13]&PIN[12]&PIN[11]&PIN[15]:=D;
IF (PIN[14]&PIN[6]&PIN[7]&PIN[9])<>%1111 THEN ERROR(1);
D:=(D SHL 1) OR 1;

FOR I:=0 TO 9 DO
   BEGIN
   PIN[10]&PIN[5]&PIN[4]&PIN[3]&PIN[2]&
   PIN[1]&PIN[13]&PIN[12]&PIN[11]&PIN[15]:=D;
   IF (PIN[14]&PIN[6]&PIN[7]&PIN[9])<>%I THEN ERROR(1);
   D:=(D SHL 1) OR 1;
   END;

ERROR(0);
END.

#NAME 40160,CD40160

#TEXT
Synchronous Programmable
Decade Counter with
Asynchronous Clear

This device contains
a programmable, synchro-
nous decade counter which
counts up in BCD code
and is reset asynchro-
nously.

#PIN 16
 1 : -R asyn
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
```

```
15 : Carry Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[2]:=HIGH;PIN[2]:=LOW;
   PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40161,CD40161

#TEXT
Synchronous Programmable
4-Bit Binary Counter with
Asynchronous Clear

This device contains a
programmable, synchro-
nous 4-bit binary counter
which counts up in binary
code and is reset asyn-
chronously.

#PIN 16
 1 : -R asyn
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
```

```
15 : Carry Out
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;PIN[1]:=HIGH;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40162,CD40162

#TEXT
Synchronous Programmable
Decade Counter with
Synchronous Clear

This device contains a
programmable, synchro-
nous decade counter which
counts up in BCD code
and is synchronously
reset.

#PIN 16
 1 : -R syn
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
15 : Carry Out
16 : Udd
```

```
#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[1]:=HIGH;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40163,CD40163

#TEXT
Synchronous Programmable
4-Bit Binary Counter with
Synchronous Clear

This device contains a
programmable, synchro-
nous 4-bit binary counter
which counts up in binary
code and is reset synchro-
nously.

#PIN 16
 1 : -R syn
 2 : Clock
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : PE
 8 : GND
 9 : -Load
10 : TE
11 : Output  Q3
12 : Output  Q2
13 : Output  Q1
14 : Output  Q0
15 : Carry Out
16 : Udd
```

```
#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[2]:=LOW;
PIN[1,7,9,10]:=HIGH;

PIN[1]:=LOW;
PIN[2]:=HIGH;PIN[2]:=LOW;
PIN[1]:=HIGH;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
   IF PIN[15]<>LOW THEN ERROR(1);
   PIN[2]:=HIGH;PIN[2]:=LOW;
   END;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>I THEN ERROR(1);
IF PIN[15]<>HIGH THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[9]:=LOW;PIN[2]:=HIGH;PIN[2]:=LOW;PIN[9]:=HIGH;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40174,CD40174

#TEXT
6-Bit-D-Register with
Clear

This device contains
six bistable memory
elements.

#PIN 16
 1 : -Clear
 2 : Output  Q SE 1
 3 : Input   D SE 1
 4 : Input   D SE 2
 5 : Output  Q SE 2
 6 : Input   D SE 3
 7 : Output  Q SE 3
 8 : GND
 9 : Clock
10 : Output  Q SE 4
11 : Input   D SE 4
12 : Output  Q SE 5
13 : Input   D SE 5
14 : Input   D SE 6
15 : Output  Q SE 6
16 : Udd

#PROGRAM

BEGIN
```

```
PIN[1,3,4,6,9,11,13,14] : INPUT;
PIN[2,5,7,10,12,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=LOW;
PIN[1]:=HIGH;

D:=%010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[14]&PIN[13]&PIN[11]&PIN[6]&PIN[4]&PIN[3]:=0;
   IF (PIN[15]&PIN[12]&PIN[10]&PIN[7]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   D:=D EXOR %111111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[15]&PIN[12]&PIN[10]&PIN[7]&PIN[5]&PIN[2])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 40175,CD40175

#TEXT
4-Bit-D-Register with
Clear

This device contains
four bistable memory
elements.

#PIN 16
 1 : -Clear
 2 : Output   Q SE 1
 3 : Output  -Q SE 1
 4 : Input    D SE 1
 5 : Input    D SE 2
 6 : Output  -Q SE 2
 7 : Output   Q SE 2
 8 : GND
 9 : Clock
10 : Output   Q SE 3
11 : Output  -Q SE 3
12 : Input    D SE 3
13 : Input    D SE 4
14 : Output  -Q SE 4
15 : Output   Q SE 4
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,9,12,13] : INPUT;
PIN[2,3,6,7,10,11,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[9]:=LOW;
PIN[1]:=HIGH;

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=D;
   PIN[9]:=HIGH;PIN[9]:=LOW;
   PIN[13]&PIN[12]&PIN[5]&PIN[4]:=0;
```

```
   IF (PIN[15]&PIN[10]&PIN[7]&PIN[2])<>D THEN ERROR(1);
   IF (NOT(PIN[14])&NOT(PIN[11])&NOT(PIN[6])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40192,CD40192

#TEXT
Synchronous Programmable
Up/Down Decade Counter
with Clear

This device consists of
a programmable, synchro-
nous up/down decade BCD
counter. The device has
separate clock inputs
and a clear input.

#PIN 16
 1 : Input   P1
 2 : Output  Q1
 3 : Output  Q0
 4 : Down
 5 : Up
 6 : Output  Q2
 7 : Output  Q3
 8 : GND
 9 : Input   P3
10 : Input   P2
11 : -Load
12 : -Carry Up
13 : -Carry Down
14 : Reset
15 : Input   P0
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,5,14]:=LOW;
PIN[4,5,11]:=HIGH;

PIN[14]:=HIGH;PIN[14]:=LOW;

FOR I:=0 TO 8 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>HIGH THEN ERROR(1);
   PIN[5]:=LOW;PIN[5]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

FOR I:=9 DOWNTO 1 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[13]<>HIGH THEN ERROR(1);
   PIN[4]:=LOW;PIN[4]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
```

```
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40193,CD40193

#TEXT
Synchronous Programmable
Up/Down 4-Bit Binary
Counter with Clear

This device contains a
programmable, synchro-
nous 4-bit binary up/
down counter with
separate clock inputs
and a clear input.

#PIN 16
 1 : Input   P1
 2 : Output  Q1
 3 : Output  Q0
 4 : Down
 5 : Up
 6 : Output  Q2
 7 : Output  Q3
 8 : GND
 9 : Input   P3
10 : Input   P2
11 : -Load
12 : -Carry Up
13 : -Carry Down
14 : Reset
15 : Input   P0
16 : Udd

#PROGRAM

BEGIN
PIN[1,4,5,9,10,11,14,15] : INPUT;
PIN[2,3,6,7,12,13] : OUTPUT;
PIN[8] :  GND;
PIN[16] : +5V;
PIN[4,5,14]:=LOW;
PIN[4,5,11]:=HIGH;

PIN[14]:=HIGH;PIN[14]:=LOW;

FOR I:=0 TO 14 DO
   BEGIN
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[12]<>HIGH THEN ERROR(1);
   PIN[5]:=LOW;PIN[5]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

FOR I:=15 DOWNTO 1 DO
   BEGIN
```

```
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);
   IF PIN[13]<>HIGH THEN ERROR(1);
   PIN[4]:=LOW;PIN[4]:=HIGH;
   END;
IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>I THEN ERROR(1);

D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[10]&PIN[1]&PIN[15]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[7]&PIN[6]&PIN[2]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40194,CD40194

#TEXT
4-Bit Bidirectional
Shift Register with
Synchronous Parallel
Input and Output and
Clear

This device contains
a 4-bit shift register
with parallel input and
output. Data can be
shifted to the right or
left and the device has
a clear input.

#PIN 16
 1 : -Reset
 2 : Data    SR
 3 : Input   P0
 4 : Input   P1
 5 : Input   P2
 6 : Input   P3
 7 : Data    SL
 8 : GND
 9 : Mode    S0
10 : Mode    S1
11 : Clock
12 : Output  Q3
13 : Output  Q2
14 : Output  Q1
15 : Output  Q0
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10,11] : INPUT;
PIN[12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[11]:=LOW;
PIN[1]:=HIGH;

D:=%0101;

PIN[9]:=LOW;
PIN[10]:=HIGH;
```

```
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[7]:=X AND 1;
      X:=X SHR 1;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      END;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[9]:=HIGH;
PIN[10]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2]:=X AND 1;
      X:=X SHR 1;
      PIN[11]:=LOW;PIN[11]:=HIGH;
      END;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[9]:=HIGH;
PIN[10]:=HIGH;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   PIN[11]:=LOW;PIN[11]:=HIGH;
   IF (PIN[12]&PIN[13]&PIN[14]&PIN[15])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

ERROR(0);
END.

#NAME 40195,CD40195

#TEXT
4-Bit Serial-In/Parallel-
Out Shift Register with
Clear

This device contains a
4-bit right shift regis-
ter with parallel or
serial input and parallel
output and clear input.

#PIN 16
 1 : -Reset
 2 : Input     J
 3 : Input    -K
 4 : Input    P0
 5 : Input    P1
 6 : Input    P2
 7 : Input    P3
 8 : GND
 9 : -Load
10 : Clock
11 : Output  -Q3
12 : Output   Q3
```

```
13 : Output   Q2
14 : Output   Q1
15 : Output   Q0
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,4,5,6,7,9,10] : INPUT;
PIN[11,12,13,14,15] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[10]:=LOW;
PIN[1,9]:=HIGH;

D:=%0101;

FOR I:=0 TO 1 DO
   BEGIN
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      PIN[2,3]:=X AND 1;
      X:=X SHR 1;
      PIN[10]:=LOW;PIN[10]:=HIGH;
      END;
   IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>D THEN ERROR(1);
   IF NOT(PIN[11])<>(D AND 1) THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;PIN[2,3]:=LOW;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]:=LOW;
   PIN[4]&PIN[5]&PIN[6]&PIN[7]:=D;
   PIN[10]:=LOW;PIN[10]:=HIGH;
   PIN[9]:=HIGH;
   X:=D;
   FOR J:=0 TO 3 DO
      BEGIN
      IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>X THEN ERROR(1);
      IF NOT(PIN[11])<>(X AND 1) THEN ERROR(1);
      PIN[2,3]:=LOW;
      PIN[10]:=LOW;PIN[10]:=HIGH;
      X:=X SHR 1;
      END;
   D:=D EXOR %1111;
   END;

PIN[1]:=LOW;PIN[1]:=HIGH;
IF (PIN[15]&PIN[14]&PIN[13]&PIN[12])<>0 THEN ERROR(1);

ERROR(0);
END.

#NAME 40240,CD40240

#TEXT
Octal Inverting TRI-
STATE Buffer

This device contains
eight inverting buffers
with three-state outputs.

#PIN 20
```

```
 1 : -G1
 2 : Input    E0
 3 : Output  -Q7
 4 : Input    E1
 5 : Output  -Q6
 6 : Input    E2
 7 : Output  -Q5
 8 : Input    E3
 9 : Output  -Q4
10 : GND
11 : Input    E4
12 : Output  -Q3
13 : Input    E5
14 : Output  -Q2
15 : Input    E6
16 : Output  -Q1
17 : Input    E7
18 : Output  -Q0
19 : -G2
20 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1,19]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (NOT(PIN[18])&NOT(PIN[16])&NOT(PIN[14])&NOT(PIN[12])&
       NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[5])&NOT(PIN[3]))<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40241,CD40241

#TEXT
Octal TRI-STATE Buffer

This device contains
eight non-inverting
buffers with three-state
outputs.

#PIN 20
 1 : -G1
 2 : Input   E0
```

```
 3 : Output  Q7
 4 : Input   E1
 5 : Output  Q6
 6 : Input   E2
 7 : Output  Q5
 8 : Input   E3
 9 : Output  Q4
10 : GND
11 : Input   E4
12 : Output  Q3
13 : Input   E5
14 : Output  Q2
15 : Input   E6
16 : Output  Q1
17 : Input   E7
18 : Output  Q0
19 : -G2
20 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=LOW;
PIN[1]:=HIGH;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1]:=LOW;PIN[19]:=HIGH;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1]:=HIGH;PIN[19]:=LOW;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```

```
#NAME 40242,CD40242

#TEXT
Quad Inverting TRI-STATE
Transceiver

This device contains
four inverting bi-
directional buffers
with three-state
outputs.

#PIN 14
 1 : -GA
 2 : N.C.
 3 : In-/Output  A0
 4 : In-/Output  A1
 5 : In-/Output  A2
 6 : In-/Output  A3
 7 : GND
 8 : In-/Output  B3
 9 : In-/Output  B2
10 : In-/Output  B1
11 : In-/Output  B0
12 : N.C.
13 : GB
14 : Udd

#PROGRAM

BEGIN
PIN[1,13] : INPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%0101;
PIN[1,13]:=LOW;
PIN[3,4,5,6] : INPUT;
PIN[8,9,10,11] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   IF (NOT(PIN[8])&NOT(PIN[9])&NOT(PIN[10])&NOT(PIN[11]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,13]:=HIGH;
PIN[8,9,10,11] : INPUT;
PIN[3,4,5,6] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[9]&PIN[10]&PIN[11]:=D;
   IF (NOT(PIN[6])&NOT(PIN[5])&NOT(PIN[4])&NOT(PIN[3]))<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=HIGH;PIN[13]:=LOW;
PIN[3,4,5,6,8,9,10,11] : OUTPUT;
LOADMODEON;
PIN[3,4,5,6,8,9,10,11] : LOAD LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,4,5,6,8,9,10,11] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%1111 THEN ERROR(1);
LOADMODEOFF;
```

```
ERROR(0);
END.

#NAME 40243,CD40243

#TEXT
Quad TRI-STATE
Transceiver

This device contains
four non-inverting bi-
directional buffers with
three-state outputs.

#PIN 14
 1 : -GA
 2 : N.C.
 3 : In-/Output  A0
 4 : In-/Output  A1
 5 : In-/Output  A2
 6 : In-/Output  A3
 7 : GND
 8 : In-/Output  B3
 9 : In-/Output  B2
10 : In-/Output  B1
11 : In-/Output  B0
12 : N.C.
13 : GB
14 : Udd

#PROGRAM

BEGIN
PIN[1,13] : INPUT;
PIN[7] : GND;
PIN[14] : +5V;

D:=%0101;
PIN[1,13]:=LOW;
PIN[3,4,5,6] : INPUT;
PIN[8,9,10,11] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[6]&PIN[5]&PIN[4]&PIN[3]:=D;
   IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1,13]:=HIGH;
PIN[8,9,10,11] : INPUT;
PIN[3,4,5,6] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[8]&PIN[9]&PIN[10]&PIN[11]:=D;
   IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>D THEN ERROR(1);
   D:=D EXOR %1111;
   END;

PIN[1]:=HIGH;PIN[13]:=LOW;
PIN[3,4,5,6,8,9,10,11] : OUTPUT;
LOADMODEON;
PIN[3,4,5,6,8,9,10,11] : LOAD LOW;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%0000 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%0000 THEN ERROR(1);
PIN[3,4,5,6,8,9,10,11] : LOAD HIGH;
IF (PIN[6]&PIN[5]&PIN[4]&PIN[3])<>%1111 THEN ERROR(1);
IF (PIN[8]&PIN[9]&PIN[10]&PIN[11])<>%1111 THEN ERROR(1);
```

```
LOADMODEOFF;

ERROR(0);
END.

#NAME 40244,CD40244

#TEXT
Octal TRI-STATE Buffer

This device contains
eight non-inverting
buffers with three-state
outputs.

#PIN 20
 1 : -G1
 2 : Input   E0
 3 : Output  Q7
 4 : Input   E1
 5 : Output  Q6
 6 : Input   E2
 7 : Output  Q5
 8 : Input   E3
 9 : Output  Q4
10 : GND
11 : Input   E4
12 : Output  Q3
13 : Input   E5
14 : Output  Q2
15 : Input   E6
16 : Output  Q1
17 : Input   E7
18 : Output  Q0
19 : -G2
20 : Udd

#PROGRAM

BEGIN
PIN[1,2,4,6,8,11,13,15,17,19] : INPUT;
PIN[3,5,7,9,12,14,16,18] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;

D:=%01010101;

FOR I:=0 TO 1 DO
   BEGIN
   PIN[2]&PIN[4]&PIN[6]&PIN[8]&
   PIN[11]&PIN[13]&PIN[15]&PIN[17]:=D;
   PIN[1,19]:=LOW;
   IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
       PIN[9]&PIN[7]&PIN[5]&PIN[3])<>D THEN ERROR(1);
   PIN[1,19]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[18,16,14,12,9,7,5,3] : LOAD LOW;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%00000000 THEN ERROR(1);
PIN[18,16,14,12,9,7,5,3] : LOAD HIGH;
IF (PIN[18]&PIN[16]&PIN[14]&PIN[12]&
    PIN[9]&PIN[7]&PIN[5]&PIN[3])<>%11111111 THEN ERROR(1);
LOADMODEOFF;
```

```
ERROR(0);
END.

#NAME 40245,CD40245

#TEXT
Octal TRI-STATE
Transceiver

This device contains
eight non-inverting
bidirectional buffers
with three-state out-
puts.

#PIN 20
 1 : DIR
 2 : In-/Output  A1
 3 : In-/Output  A2
 4 : In-/Output  A3
 5 : In-/Output  A4
 6 : In-/Output  A5
 7 : In-/Output  A6
 8 : In-/Output  A7
 9 : In-/Output  A8
10 : GND
11 : In-/Output  B8
12 : In-/Output  B7
13 : In-/Output  B6
14 : In-/Output  B5
15 : In-/Output  B4
16 : In-/Output  B3
17 : In-/Output  B2
18 : In-/Output  B1
19 : -Enable
20 : Udd

#PROGRAM

BEGIN
PIN[1,19] : INPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[19]:=HIGH;

D:=%01010101;

PIN[1]:=HIGH;
PIN[9,8,7,6,5,4,3,2] : INPUT;
PIN[11,12,13,14,15,16,17,18] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[9]&PIN[8]&PIN[7]&PIN[6]&PIN[5]&PIN[4]&PIN[3]&PIN[2]:=D;
   PIN[19]:=LOW;
   IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
       PIN[15]&PIN[16]&PIN[17]&PIN[18])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[1]:=LOW;
PIN[11,12,13,14,15,16,17,18] : INPUT;
PIN[9,8,7,6,5,4,3,2] : OUTPUT;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[11]&PIN[12]&PIN[13]&PIN[14]&PIN[15]&PIN[16]&PIN[17]&PIN[18]:=D;
   PIN[19]:=LOW;
```

```
   IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
       PIN[5]&PIN[4]&PIN[3]&PIN[2])<>D THEN ERROR(1);
   PIN[19]:=HIGH;
   D:=D EXOR %11111111;
   END;

PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : OUTPUT;
LOADMODEON;
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD LOW;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%00000000 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[9,8,7,6,5,4,3,2,11,12,13,14,15,16,17,18] : LOAD HIGH;
IF (PIN[11]&PIN[12]&PIN[13]&PIN[14]&
    PIN[15]&PIN[16]&PIN[17]&PIN[18])<>%11111111 THEN ERROR(1);
IF (PIN[9]&PIN[8]&PIN[7]&PIN[6]&
    PIN[5]&PIN[4]&PIN[3]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40257,CD40257

#TEXT
Quad TRI-STATE 2-Channel
Data Selector/Multiplexer

This device contains
four 1-of-2 data selec-
tors. The outputs are
three-state.

#PIN 16
 1 : Select
 2 : Input   A MP 1
 3 : Input   B MP 1
 4 : Output  Q MP 1
 5 : Input   A MP 2
 6 : Input   B MP 2
 7 : Output  Q MP 2
 8 : GND
 9 : Output  Q MP 3
10 : Input   B MP 3
11 : Input   A MP 3
12 : Output  Q MP 4
13 : Input   B MP 4
14 : Input   A MP 4
15 : -Enable
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,5,6,10,11,13,14,15] : INPUT;
PIN[4,7,9,12] : OUTPUT;
PIN[8] : GND;
PIN[16] : +5V;
PIN[15]:=HIGH;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[11]&PIN[5]&PIN[2]:=D;
```

```
   PIN[15]:=LOW;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[10]&PIN[6]&PIN[3]:=D;
   PIN[15]:=LOW;
   IF (PIN[12]&PIN[9]&PIN[7]&PIN[4])<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[4,7,9,12] : LOAD LOW;
IF (PIN[4]<>LOW) OR (PIN[7]<>LOW) OR
   (PIN[9]<>LOW) OR (PIN[12]<>LOW) THEN ERROR(1);
PIN[4,7,9,12] : LOAD HIGH;
IF (PIN[4]<>HIGH) OR (PIN[7]<>HIGH) OR
   (PIN[9]<>HIGH) OR (PIN[12]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40258,CD40258

#TEXT
Quad TRI-STATE 2-Channel
Data Selector/Multiplexer
(Inverted Outputs)

This device contains
four 1-of-2 data selec-
tors/multiplexers with
inverting three-state
outputs.

#PIN 16
 1 : Select
 2 : Input   A MP 1
 3 : Input   B MP 1
 4 : Output  Q MP 1
 5 : Input   A MP 2
 6 : Input   B MP 2
 7 : Output  Q MP 2
 8 : GND
 9 : Output  Q MP 3
10 : Input   B MP 3
11 : Input   A MP 3
12 : Output  Q MP 4
13 : Input   B MP 4
14 : Input   A MP 4
15 : -Enable
16 : Udd

#PROGRAM

BEGIN
PIN[1,2,3,5,6,10,11,13,14,15] : INPUT;
PIN[4,7,9,12] : OUTPUT;
PIN[8] : GND;
```

```
PIN[16] : +5V;
PIN[15]:=HIGH;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=LOW;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[14]&PIN[11]&PIN[5]&PIN[2]:=D;
   PIN[15]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

PIN[2,3,5,6,10,11,13,14]:=LOW;
PIN[1]:=HIGH;
D:=%0101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[13]&PIN[10]&PIN[6]&PIN[3]:=D;
   PIN[15]:=LOW;
   IF (NOT(PIN[12])&NOT(PIN[9])&NOT(PIN[7])&NOT(PIN[4]))<>D THEN ERROR(1);
   PIN[15]:=HIGH;
   D:=D EXOR %1111;
   END;

LOADMODEON;
PIN[4,7,9,12] : LOAD LOW;
IF (PIN[4]<>LOW) OR (PIN[7]<>LOW) OR
   (PIN[9]<>LOW) OR (PIN[12]<>LOW) THEN ERROR(1);
PIN[4,7,9,12] : LOAD HIGH;
IF (PIN[4]<>HIGH) OR (PIN[7]<>HIGH) OR
   (PIN[9]<>HIGH) OR (PIN[12]<>HIGH) THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40373,CD40373

#TEXT
8-Bit-D-Latch with Enable
(TRI-STATE)

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
 2 : Output  Q0
 3 : Input   D0
 4 : Input   D1
 5 : Output  Q1
 6 : Output  Q2
 7 : Input   D2
 8 : Input   D3
 9 : Output  Q3
10 : GND
11 : Latch Enable
12 : Output  Q4
13 : Input   D4
14 : Input   D5
15 : Output  Q5
16 : Output  Q6
```

```
17 : Input   D6
18 : Input   D7
19 : Output  Q7
20 : Udd

#PROGRAM

BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
   PIN[1]:=LOW;
   IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,16,15,12,9,6,5,2] : LOAD LOW;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[19,16,15,12,9,6,5,2] : LOAD HIGH;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.

#NAME 40374,CD40374

#TEXT
8-Bit-D-Register
(TRI-STATE)

This device contains
eight bistable memory
elements with three-
state outputs.

#PIN 20
 1 : -OE
 2 : Output  Q0
 3 : Input   D0
 4 : Input   D1
 5 : Output  Q1
 6 : Output  Q2
 7 : Input   D2
 8 : Input   D3
 9 : Output  Q3
10 : GND
11 : Clock
12 : Output  Q4
13 : Input   D4
14 : Input   D5
15 : Output  Q5
```

```
16 : Output  Q6
17 : Input   D6
18 : Input   D7
19 : Output  Q7
20 : Udd

#PROGRAM

BEGIN
PIN[1,3,4,7,8,11,13,14,17,18] : INPUT;
PIN[2,5,6,9,12,15,16,19] : OUTPUT;
PIN[10] : GND;
PIN[20] : +5V;
PIN[1]:=HIGH;
PIN[11]:=LOW;

D:=%01010101;
FOR I:=0 TO 1 DO
   BEGIN
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=D;
   PIN[11]:=HIGH;PIN[11]:=LOW;
   PIN[18]&PIN[17]&PIN[14]&PIN[13]&PIN[8]&PIN[7]&PIN[4]&PIN[3]:=0;
   PIN[1]:=LOW;
   IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
       PIN[9]&PIN[6]&PIN[5]&PIN[2])<>D THEN ERROR(1);
   PIN[1]:=HIGH;
   D:=D EXOR %11111111;
   END;

LOADMODEON;
PIN[19,16,15,12,9,6,5,2] : LOAD LOW;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%00000000 THEN ERROR(1);
PIN[19,16,15,12,9,6,5,2] : LOAD HIGH;
IF (PIN[19]&PIN[16]&PIN[15]&PIN[12]&
    PIN[9]&PIN[6]&PIN[5]&PIN[2])<>%11111111 THEN ERROR(1);
LOADMODEOFF;

ERROR(0);
END.
•
```