Factors Counter Circuit
PC/CP220 Project Phase II

Fall 2020

Creating Equations
Truth Table

Table 1 is a truth table for the Factors Counter Circuit.

number	$a_3a_2a_1a_0$	number of factors	$b_2b_1b_0$
0	0000	0	000
1	0001	1	001
2	0010	2	010
3	0011	2	010
4	0100	3	011
5	0101	2	010
6	0110	4	100
7	0111	2	010
8	1000	4	100
9	1001	3	011
10	1010	4	100
11	1011	2	010
12	1100	6	110
13	1101	2	010
14	1110	4	100
15	1111	4	100

Table 1: Truth Table

In order to determine the logic equations for the Number of Segments Circuit, the only quantities to consider are the binary inputs and outputs. The fact that the bits are grouped together to represent numbers is irrelevant.

The number of binary outputs for the circuit is the number of equations for the circuit, since each output implements a specific logic function (i.e. an equation).

The truth table containing only binary quantities is shown in Table 2

a_3	a_2	a_1	a_0	b_2	b_1	b_0
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	0	1	0
1	1	0	0	1	1	0
1	1	0	1	0	1	0
1	1	1	0	1	0	0
1	1	1	1	1	0	0

Table 2: Truth Table showing only binary inputs and outputs

Each bit of output produces an equation. In this case, a Karnaugh map can be used to determine simplified sum-of-products logic equations for each bit of output. After that, each equation can be tested independently using a computer algebra system.

Output b_2

The section of the truth table for output b_2 is shown in Table 3.

a_3	a_2	a_1	a_0	b_2
0	0	0	0	0

0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

Table 3: Truth Table for b_2

This produces the Karnaugh map for bit b_2 shown in Table 4.

b_2		a_1a_0			
		00	01	11	10
a_3a_2	00	0	0	0	0
	01	0	0	0	1
	11	1	0	1	1
	10	1	0	0	1

Table 4: Karnaugh Map Table for b_2

This will produce the following equation for b_2.

$$a_3 \acute{a}_0 + a_1a_3a_2 + a_2a_1 \acute{a}_0$$

b_2		a_1a_0			
		00	01	11	10
a_3a_2	00	0	0	0	0
	01	0	0	0	1
	11	1	0	1	1
	10	1	0	0	1

Table 5: Karnaugh Map Table for b_2 highlighting two terms

b_2		a_1a_0 00	01	11	10
a_3a_2	00	0	0	0	0
	01	0	0	0	1
	11	1	0	1	1
	10	1	0	0	1

Table 6: Karnaugh Map Table for b_2 highlighting two more terms

Output b_1

The section of the truth table for output b_1 is shown in Table 7.

a_3	a_2	a_1	a_0	b_1
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

Table 7: Truth Table for b_1

This produces the Karnaugh map for bit b_1 shown in Table 8.

b1		a_1a_0 00	01	11	10
a_3a_2	00	0	0	1	1
	01	1	1	1	0
	11	0	1	1	0
	10	1	1	0	0

Table 8: Karnaugh Map Table for b_1

This will produce the following equation for b_1.

$$\acute{a}_1 \acute{a}_3 a_2 + \acute{a}_2 a_3 a_0 + a_1 \acute{a}_3 \acute{a}_2 + \acute{a}_3 a_1 a_0 + a_3 a_0 \acute{a}_1 + a_3 a_2 \acute{a}_1$$

b1		a_1a_0			
		00	01	11	10
a_3a_2	00	0	0	1	1
	01	1	1	1	0
	11	1	1	0	0
	10	0	1	1	0

Table 9: Karnaugh Map Table for b_1

b1		a_1a_0			
		00	01	11	10
a_3a_2	00	0	0	1	1
	01	1	1	1	0
	11	1	1	1	0
	10	0	1	0	0

Table 10: Karnaugh Map Table for b_1

Outputs b_0

The section of the truth table for output b_0 is shown in Table 11.

a_3	a_2	a_1	a_0	b_0
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Table 11: Truth Table for b_0

This produces the Karnaugh map for bit b_0 shown in Table 12.

b0		a_1a_0			
		00	01	11	10
a_3a_2	00	0	1	0	0
	01	1	0	0	0
	11	0	0	0	0

	10	0	1	0	0

Table 12: Karnaugh Map Table for b_0

This will produce the following equation for b_0.

$$\acute{a}_1 a_0 a_3 \acute{a}_2 + \acute{a}_1 a_0 \acute{a}_3 \acute{a}_2 + \acute{a}_1 \acute{a}_0 \acute{a}_3 a_2$$

b0		a_1a_0			
		00	01	11	10
a_3a_2	00	0	1	0	0
	01	1	0	0	0
	11	0	0	0	0
	10	0	1	0	0

Table 13: Karnaugh Map Table for b_0 highlighting three terms

Equation Testing

The equations were tested with Maxima.

Output b_2

```
(%i1) b2: (a3 and (not a0)) or (a1 and a3 and a2) or (a2 and a1 and (not a0));
```
(%o1) $a_3 \wedge \neg a_0 \vee a_1 \wedge a_3 \wedge a_2 \vee a_2 \wedge a_1 \wedge \neg a_0$

```
(%i2) b2, a3 = false, a2 = false, a1 = false, a0 = false;
```
(%o2) **false**

```
(%i3) b2, a3 = false, a2 = false, a1 = false, a0 = true;
```
(%o3) **false**

```
(%i4) b2, a3 = false, a2 = false, a1 = true, a0 = false;
```
(%o4) **false**

```
(%i5) b2, a3 = false, a2 = false, a1 = true, a0 = true;
```
(%o5) **false**

```
(%i6) b2, a3 = false, a2 = true, a1 = false, a0 = false;
```
(%o6) **false**

```
(%i7) b2, a3 = false, a2 = true, a1 = false, a0 = true;
```
(%o7) **false**

```
(%i8) b2, a3 = false, a2 = true, a1 = true, a0 = false;
```
(%o8) **true**

```
(%i9) b2, a3 = false, a2 = true, a1 = true, a0 = true;
```
(%o9) **false**

```
(%i10) b2, a3 = true, a2 = false, a1 = false, a0 = false;
```
(%o10) **true**

```
(%i11) b2, a3 = true, a2 = false, a1 = false, a0 = true;
```
(%o11) **false**

```
(%i12) b2, a3 = true, a2 = false, a1 = true, a0 = false;
```
(%o12) **true**

```
(%i13) b2, a3 = true, a2 = false, a1 = true, a0 = true;
```
(%o13) **false**

```
(%i14) b2, a3 = true, a2 = true, a1 = false, a0 = false;
```
(%o14) **true**

```
(%i15) b2, a3 = true, a2 = true, a1 = false, a0 = true;
```
(%o15) **false**

```
(%i16) b2, a3 = true, a2 = true, a1 = true, a0 = false;
```
(%o16) **true**

```
(%i17) b2, a3 = true, a2 = true, a1 = true, a0 = true;
```
(%o17) **true**

Figure 1: Test of b_2

This matches the truth table, so the equation for b_2 is correct.

Output b_1

```
(%i1) b1: (a2 and (not a3) and (not a1)) or (a3 and (not a2) and a0) or (a1 and (not a3) and (not a2)) or (a1 and (not a3) and a0) or ((not a1) and a3 and a0) or (a3 and a2 and (not a1));
```

(%o1) $a_2 \wedge \neg a_3 \wedge \neg a_1 \vee a_3 \wedge \neg a_2 \wedge a_0 \vee a_1 \wedge \neg a_3 \wedge \neg a_2 \vee a_1 \wedge \neg a_3 \wedge a_0 \vee \neg a_1 \wedge a_3 \wedge a_0 \vee a_3 \wedge a_2 \wedge \neg a_1$

```
(%i2) b1, a3 = false, a2 = false, a1 = false, a0 = false;
```
(%o2) **false**

```
(%i3) b1, a3 = false, a2 = false, a1 = false, a0 = true;
```
(%o3) **false**

```
(%i4) b1, a3 = false, a2 = false, a1 = true, a0 = false;
```
(%o4) **true**

```
(%i5) b1, a3 = false, a2 = false, a1 = true, a0 = true;
```
(%o5) **true**

```
(%i6) b1, a3 = false, a2 = true, a1 = false, a0 = false;
```
(%o6) **true**

```
(%i7) b1, a3 = false, a2 = true, a1 = false, a0 = true;
```
(%o7) **true**

```
(%i8) b1, a3 = false, a2 = true, a1 = true, a0 = false;
```
(%o8) **false**

```
(%i9) b1, a3 = false, a2 = true, a1 = true, a0 = true;
```
(%o9) **true**

```
(%i10) b1, a3 = true, a2 = false, a1 = false, a0 = false;
```
(%o10) **false**

```
(%i11) b1, a3 = true, a2 = false, a1 = false, a0 = true;
```
(%o11) **true**

```
(%i12) b1, a3 = true, a2 = false, a1 = true, a0 = false;
```
(%o12) **false**

```
(%i13) b1, a3 = true, a2 = false, a1 = true, a0 = true;
```
(%o13) **true**

```
(%i14) b1, a3 = true, a2 = true, a1 = false, a0 = false;
```
(%o14) **true**

```
(%i15) b1, a3 = true, a2 = true, a1 = false, a0 = true;
```
(%o15) **true**

```
(%i16) b1, a3 = true, a2 = true, a1 = true, a0 = false;
```
(%o16) **false**

```
(%i17) b1, a3 = true, a2 = true, a1 = true, a0 = true;
```
(%o17) **false**

Figure 1: Test of b_1

This matches the truth table, so the equation for b_1 is correct.

Output b_0

```
(%i1) b0: ((not a1) and (not a2) and a0 and a3) or ((not a1) and (not a3) and (not a2) and a0) or ((not a1) and (not a0) and (not a3) and a2);
(%o1) ¬a1 ∧ ¬a2 ∧ a0 ∧ a3 ∨ ¬a1 ∧ ¬a3 ∧ ¬a2 ∧ a0 ∨ ¬a1 ∧ ¬a0 ∧ ¬a3 ∧ a2

(%i2) b0, a3 = false, a2 = false, a1 = false, a0 = false;
(%o2) false

(%i3) b0, a3 = false, a2 = false, a1 = false, a0 = true;
(%o3) true

(%i4) b0, a3 = false, a2 = false, a1 = true, a0 = false;
(%o4) false

(%i5) b0, a3 = false, a2 = false, a1 = true, a0 = true;
(%o5) false

(%i6) b0, a3 = false, a2 = true, a1 = false, a0 = false;
(%o6) true

(%i7) b0, a3 = false, a2 = true, a1 = false, a0 = true;
(%o7) false

(%i8) b0, a3 = false, a2 = true, a1 = true, a0 = false;
(%o8) false

(%i9) b0, a3 = false, a2 = true, a1 = true, a0 = true;
(%o9) false

(%i10) b0, a3 = true, a2 = false, a1 = false, a0 = false;
(%o10) false

(%i11) b0, a3 = true, a2 = false, a1 = false, a0 = true;
(%o11) true

(%i12) b0, a3 = true, a2 = false, a1 = true, a0 = false;
(%o12) false

(%i13) b0, a3 = true, a2 = false, a1 = true, a0 = true;
(%o13) false

(%i14) b0, a3 = true, a2 = true, a1 = false, a0 = false;
(%o14) false

(%i15) b0, a3 = true, a2 = true, a1 = false, a0 = true;
(%o15) false

(%i16) b0, a3 = true, a2 = true, a1 = true, a0 = false;
(%o16) false

(%i17) b0, a3 = true, a2 = true, a1 = true, a0 = true;
(%o17) false
```

Figure 1: Test of b_0

This matches the truth table, so the equation for b_0 is correct.

Summary

The equations for the outputs

b_2: $a_3 \acute{a}_0 + a_1a_3a_2 + a_2a_1\acute{a}_0$

b1: $\acute{a}_1\acute{a}_3a_2 + \acute{a}_2a_3a_0 + a_1\acute{a}_3\acute{a}_2 + \acute{a}_3a_1a_0 + a_3a_0\acute{a}_1 + a_3a_2\acute{a}_1$

b_0: $\acute{a}_1a_0a_3\acute{a}_2 + \acute{a}_1a_0\acute{a}_3\acute{a}_2 + \acute{a}_1\acute{a}_0\acute{a}_3a_2$

These equations have been tested and verified to be correct.