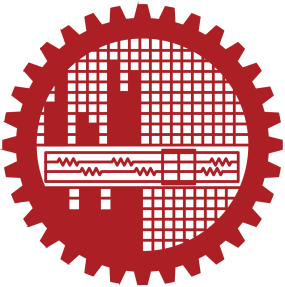
ALU Design

CSE 306



Group – 5 Section – B1

Rolls: 1505076 1505079

1505080 1505081

1505086 1505089

1305065

Department of Computer Science and Engineering

Bangladesh University of Engineering and Technology

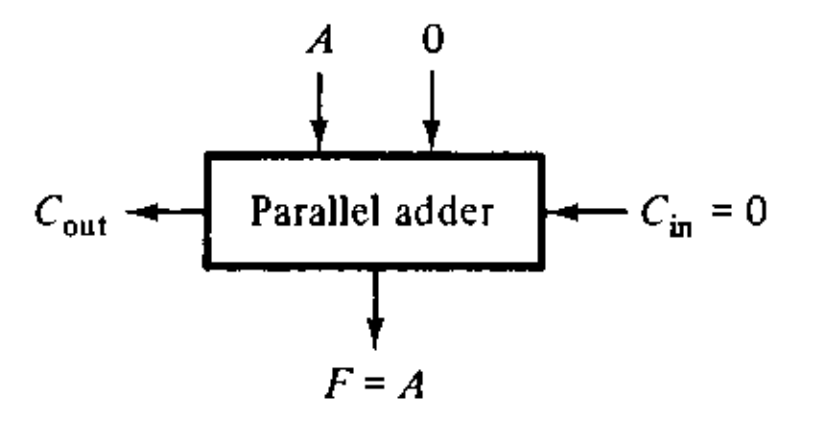
**Problem Specification:**

|  |  |  |  |
| --- | --- | --- | --- |
| cin | | | Functions |
| cs2 | cs1 | Cs0 |
| 0 | 0 | 0 | Transfer A |
| 0 | 1 | 0 | Increment A |
| 0 | x | 1 | OR |
| 1 | 0 | 0 | Subtract with borrow |
| 1 | 1 | 0 | Subtract |
| 1 | x | 1 | AND |

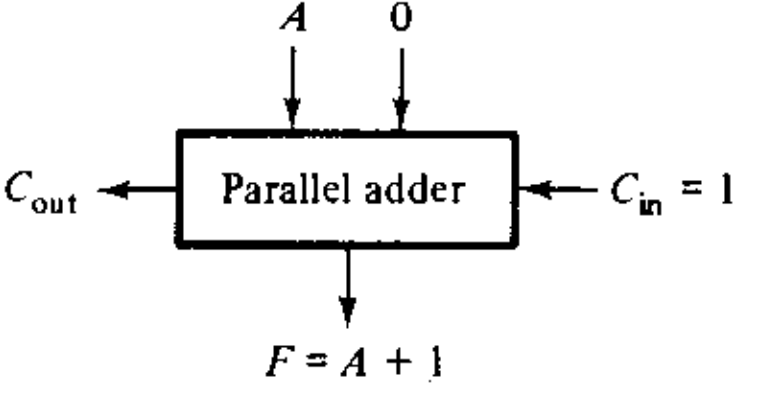
**Arithmetic Part:**

When cs0 = 0,

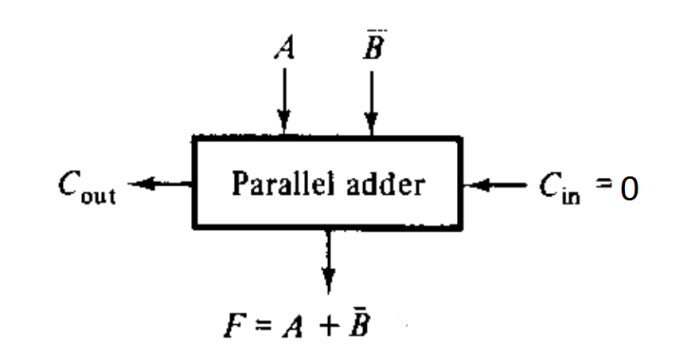
(a) Transfer A



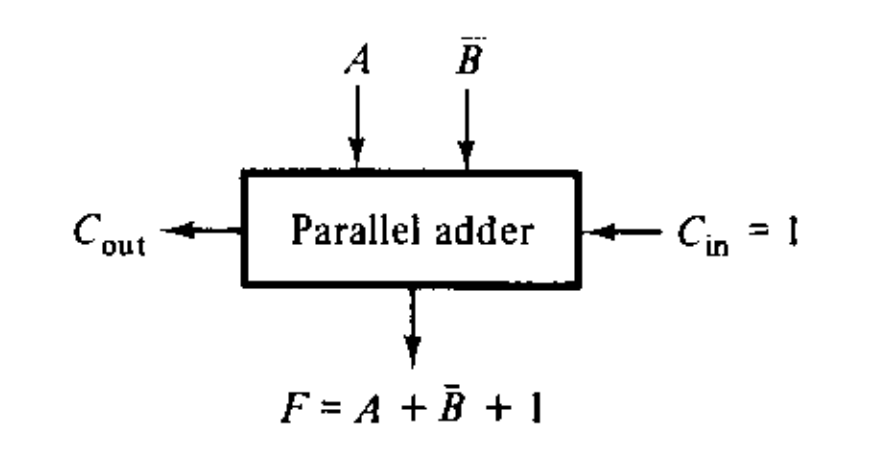
(b) Increment A



(c) Subtract with borrow



(d) Subtract



Table**:**

|  |  |  |
| --- | --- | --- |
| cs2 | cs1 | Y |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | B’ |
| 1 | 1 | B’ |

Yi = cs2.Bi’+ cs2.cs1.Bi’

= cs2.Bi’

Xi = Ai

**Logical Part:**

When cs0 = 1 logical part is activated and cs1 is don’t-care. We can force it to 0 by making Cin = cs0’.cs1. We also make every carry to next adder 0 in the same way.

Now,

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| cs2 | cs1 | cs0 | Xi | Yi | Cin | Operation | Required Operation |
| 0 | X | 1 | Ai | 0 | 0 | Fi = Ai | OR |
| 1 | X | 1 | Ai | Bi’ | 0 | Fi = Ai **⊕** Bi’ | AND |

So when cs0 = 1 and cs2 = 0, we can OR Bi with Ai and the result will be A+B.

∴ Xi = Ai + cs2’.cs0.Bi

Again when cs2 = 1 and cs0 = 1, we have to get AND operation where the output is equivalence.

Fi = Ai **⊕** Bi’ = Ai.Bi + Ai’.Bi’

Let us investigate the possibility of ORing each input Ai with some Boolean function Ki when cs2 = 1 and cs0 = 1.

Fi = Xi **⊕** Yi

= (Ai + Ki) **⊕** Bi’

= Ai.Bi + Ki.Bi+ Ai’.Ki’.Bi’

Taking Ki = Bi’, we get, Fi = Ai.Bi + Bi’.Bi+ Ai’.Bi.Bi’ = Ai.Bi, which is the required operation.

So, finally Xi = Ai + cs2’.cs0.Bi + cs2.cs0.Bi’

= Ai + cs0.(cs2’. Bi + cs2.Bi’)

= Ai + cs0.(cs2**⊕** Bi)

Yi = cs2.Bi’

Cin = cs0’.cs1

