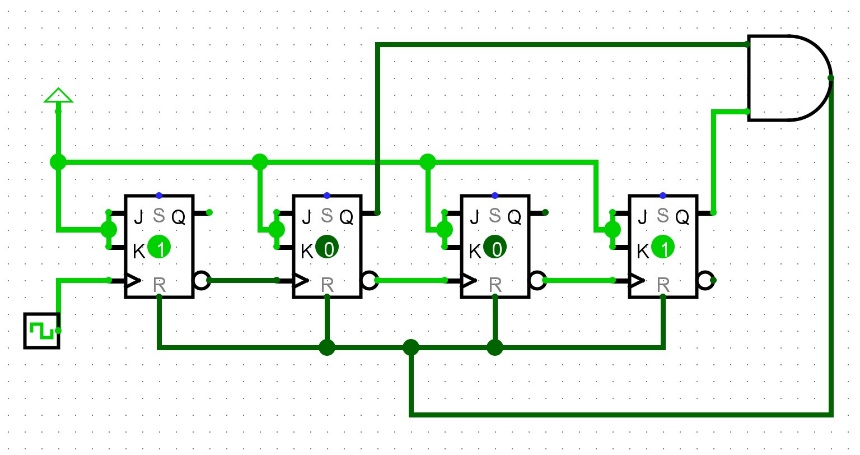
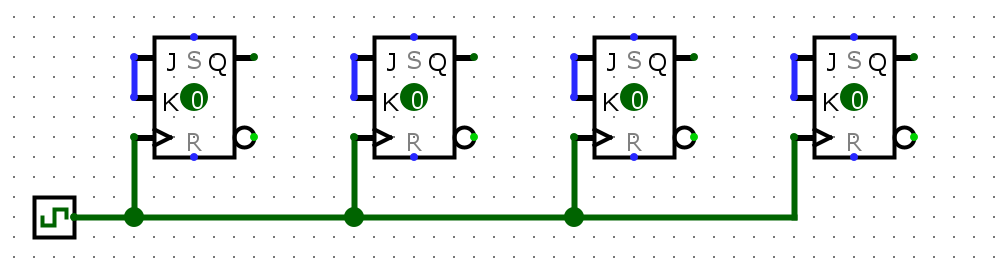
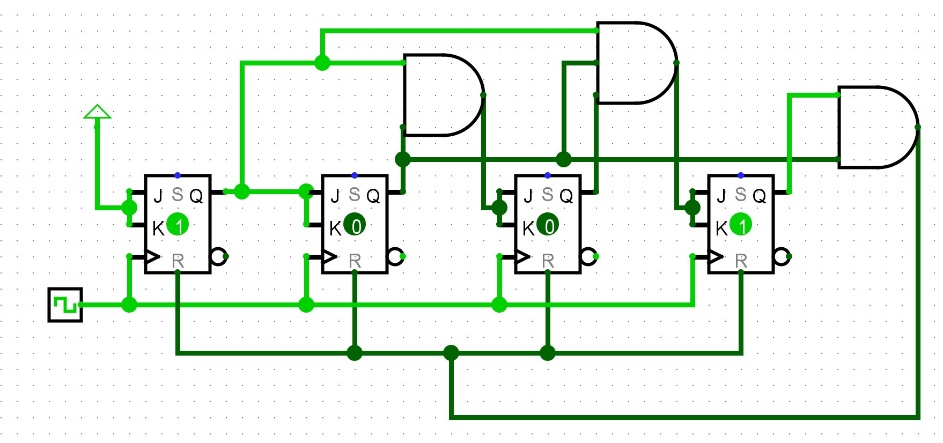
**Problem A: Mod-10 synchronous JK counter**



Construct a mod-10 (counts up to decimal 9 / binary 1001, then loops back to 0) counter) using JK flip flops. **~~It must be synchronous – no glitch states!~~** Use JK in “toggle” mode. The below configuration has been provided for you as a starting point.





**<- asynchronous**

**synchronous ->**

**Problem B: Custom synchronous 3-bit counter**

Given this state table, construct a **synchronous custom counter** using **D-type flip flops**. Hint: Construct Karnaugh maps by filling in where Dc, Db, and Da are 0, 1 or x respectively.

Note: None of the tn states of the counter have a tn+1 state of 010 or 110. However, if the counter were to ever enter those states (say, a random glitch happened), it should return to the sequence and not get stuck in those states.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| tn | | | | | | tn+1 | | |
| Qc | Qb | Qa | Dc | Db | Da | Qc | Qb | Qa |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | x | x | x | x | x | x |
| 1 | 1 | 0 | x | x | x | x | x | x |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Qc\QbQa | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 1 | x |
| 1 | 1 | 0 | 0 | x |

Dc

**!QcQa + Qc!Qa**

**Qc XOR Qa**

Db

!Qb!Qa + Qb!Qa

**= !Qa**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Qc\QbQa | 00 | 01 | 11 | 10 |
| 0 | 1 | 0 | 0 | x |
| 1 | 1 | 0 | 0 | x |

Da

**!Qa + !QcQb + Qc!Qb**

**!Qa + Qc XOR Qb**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Qc\QbQa | 00 | 01 | 11 | 10 |
| 0 | 1 | 0 | 1 | x |
| 1 | 1 | 1 | 0 | x |

