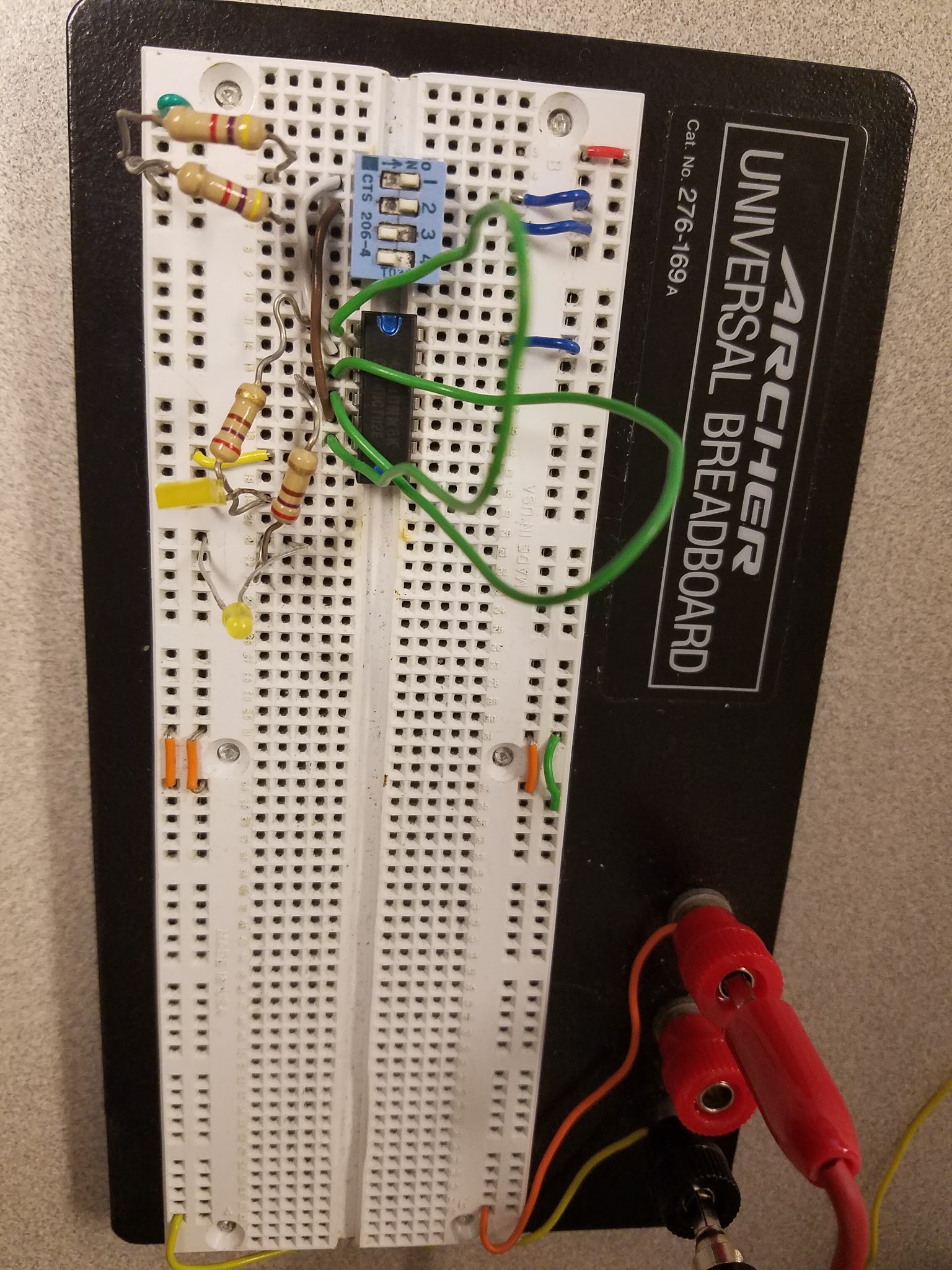
1.



2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **R** | **S** | **previous Q** | **New Q** | **New Q'** |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 |

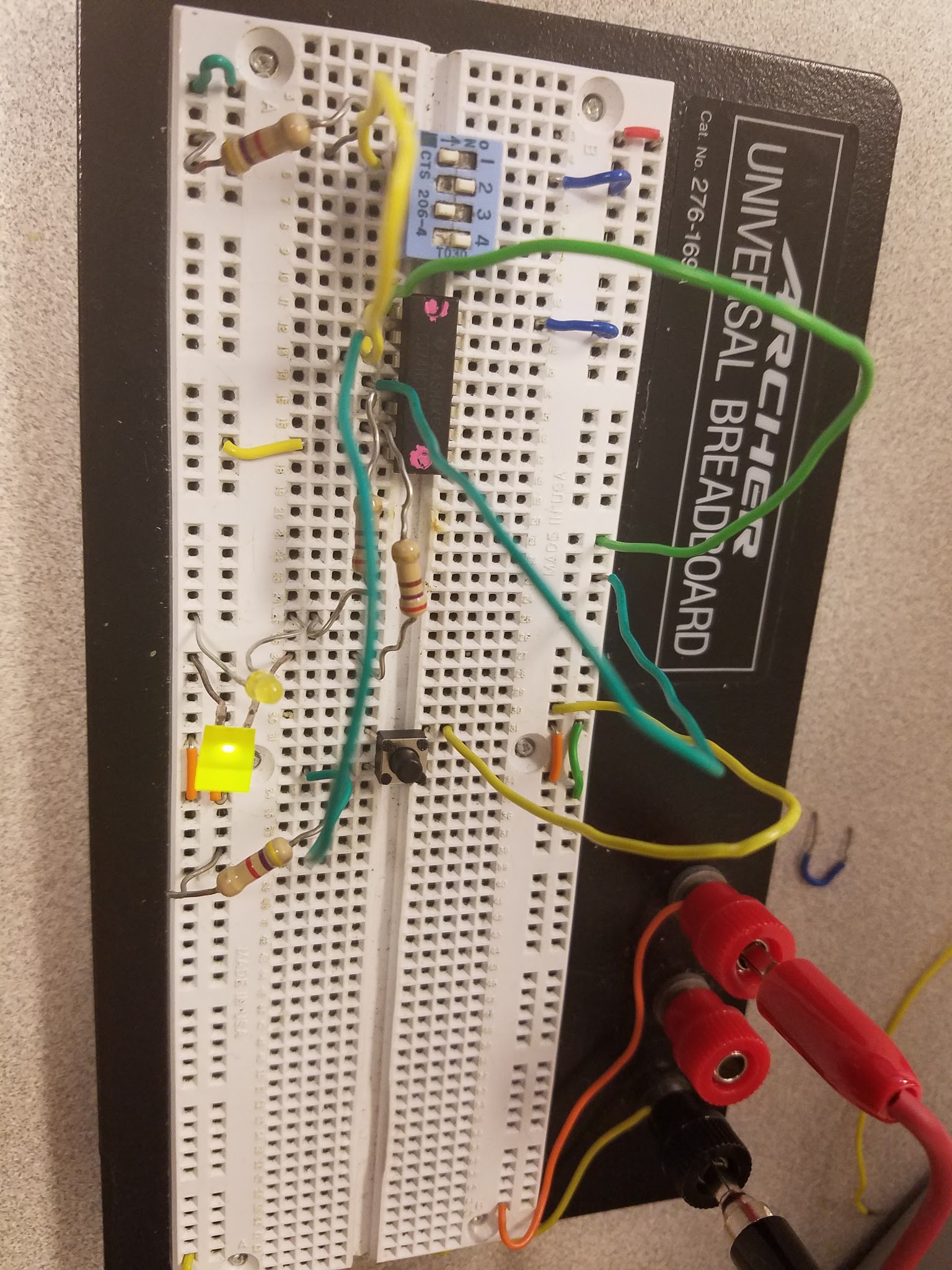
3. Both of the outputs are 0

4. When r=1 and s=1 the new q value outputs a 0. This is considered a restricted state, This occurs due to the nature of the nor gate and the sr latch circuit. The set reset functionality is violated. When any 1 is passed into a NOR gate it will output a zero since Q, the output of R, is hooked to the input of S and vice versa, passing a 1 into both r and s will always results in the output of q and ~q being 0.

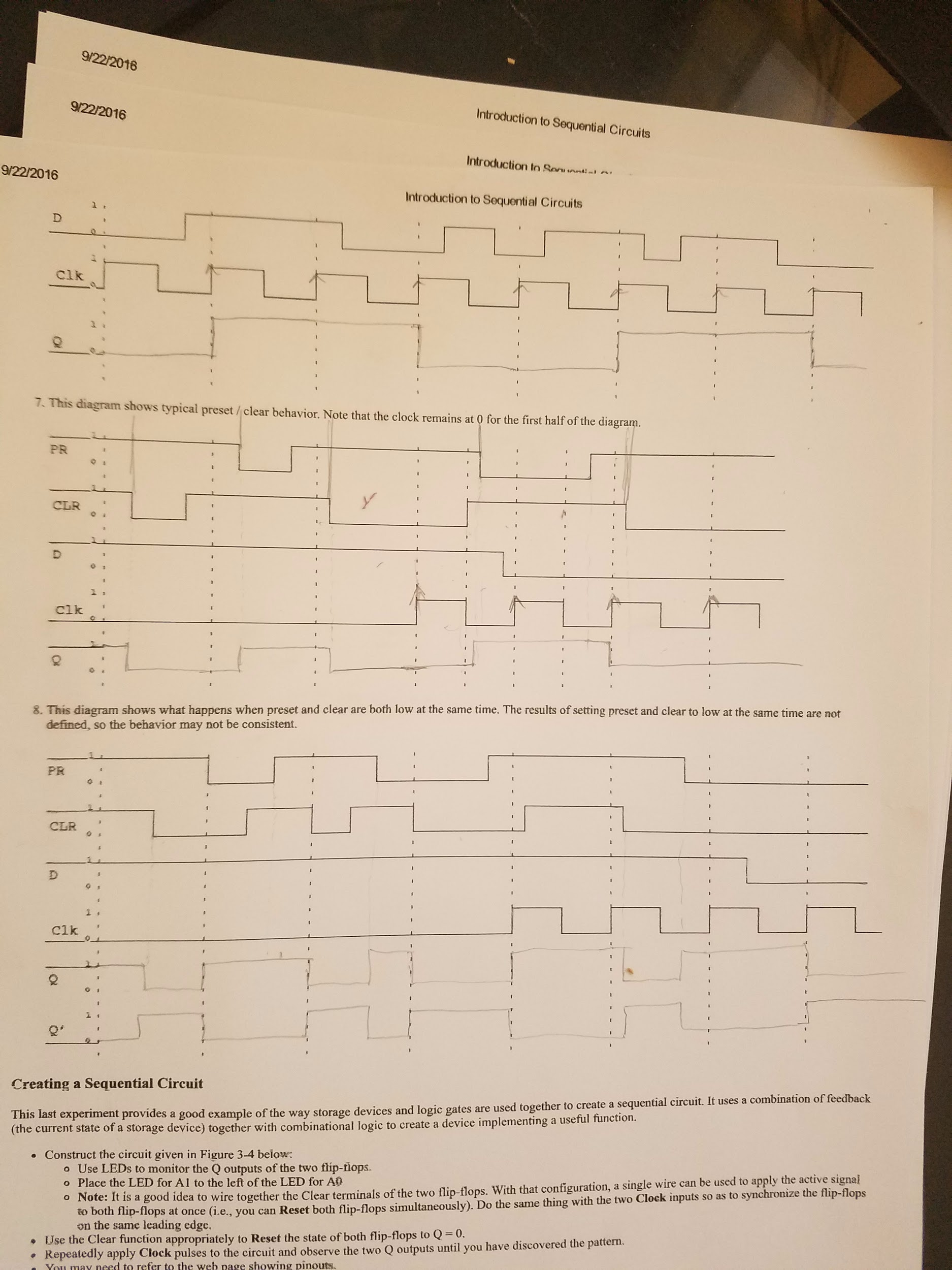
5. If the latch was in a 00 state and both inputs changed to 1 at the same time the latch can enter a metastable state, which can last significantly longer than the delay time of the gates. The outputs either drift towards their final state, or oscillate before setting on the final state.

Normally one input becomes one, and the loop propagates through both gates until the stable condition is met. If the other input becomes one while the propagation is still taking place, the second input starts to propagate as well and there is no telling which one will take precedence. Sometimes neither win and you enter the restricted state. If from 00 state one input changes to 0 and the second changes to 0 before the propagation of the first has settled the two changes are racing for priority, this is a race condition.

d flip flop



#6,7,8



9

|  |  |  |  |
| --- | --- | --- | --- |
| A1 Pre | A0 Pre | A1 Next | A0 Next |
| 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |

10. This circuit is a counter

