# Car Security System

## Using combinational logic only

***Three inputs:*** Main switch (M), Vibration/Motion Sensor (V), Door Sensor (D)
***Output:*** Car Alarm (A)

**Truth Table**

| Inputs | | | Output |
|---|---|---|---|
| **M** | **V** | **D** | **A** |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

**Boolean Expression**

$A = M\bar{D}V + MD\bar{V} + MDV$

**Boolean algebraic simplification**

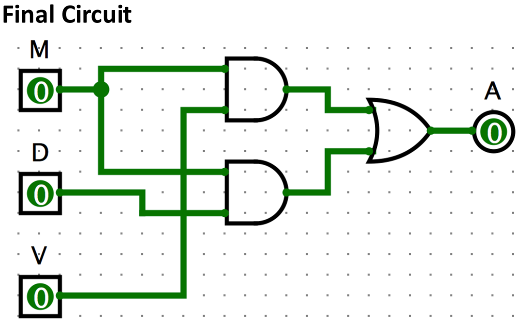
$A = M\bar{D}V + MD\bar{V} + MDV$
$A = M(\bar{D}V + D\bar{V} + DV)$ by distributive theorem, *xy+xz = x(y+z)*
$A = M(\bar{D}V + D\bar{V} + DV + DV)$ by idempotent theorem, *x + x = x*
$A = M(V(D + \bar{D}) + D(V + \bar{V}))$ by distributive theorem
$A = M(V(1) + D(1))$ by inverse theorem *x+x' = 1*
$A = M(V + D) = MV + MD$

**Karnaugh map based simplification**

| | | **D, V** | | | |
|---|---|---|---|---|---|
| | | 00 | 01 | 11 | 10 |
| **M** | 0 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 1 | 1 |

M V + M D

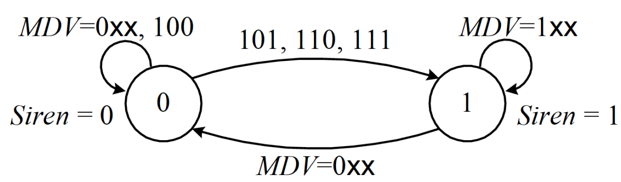
**Final Circuit**



## FSM based variant

The problem with a combinational circuit is that once the alarm is triggered, by let's say opening the door, the alarm can be turned off immediately by closing the door again. However, what we want is that once the alarm is triggered, it should remain on even after closing the door, and the only way to turn it off is to turn off the master switch. This requirement suggests that we need a sequential circuit instead where the output is dependent on not only the current input switch settings but also on the current state of the alarm.

**Step 1: State Diagram**




**Step 2: Next State Table**

| Present State | Inputs | | | Next State |
|---|---|---|---|---|
| ***Q*** | ***M*** | ***D*** | ***V*** | ***$Q_{next}$*** |
| 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 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

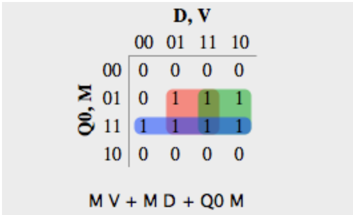
**Step3: Implementation Table**

Given the present state and a desired next state, what will be the input(s) to the flip-flop(s) used in the FSM? To answer that question, we use an implementation table. Here we are going to use a D flip-flop, which is a transparent latch i.e. $Q_{next} = D_0$. Hence the implementation table will be the same the next state table (step 2) but next state ($Q_{next}$) replaced with flip-flop input ($D_0$). If we use other flip-flops (SR, JK, and T), we will have to use their respective excitation tables to come up with the implementation table.

| Present State | Inputs | | | Flip-flop implementation |
|---|---|---|---|---|
| ***$Q_0$*** | ***M*** | ***D*** | ***V*** | ***$D_0$*** |
| 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 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

**Step 4: Excitation Equation(s)**

Karnaugh map based simplification



| Q0, M \ D, V | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 0 | 0 | 0 |
| 01 | 0 | 1 | 1 | 1 |
| 11 | 1 | 1 | 1 | 1 |
| 10 | 0 | 0 | 0 | 0 |

M V + M D + Q0 M

$D_0 = MV + MD + Q_0M$

**Step 5: Final Circuit**