

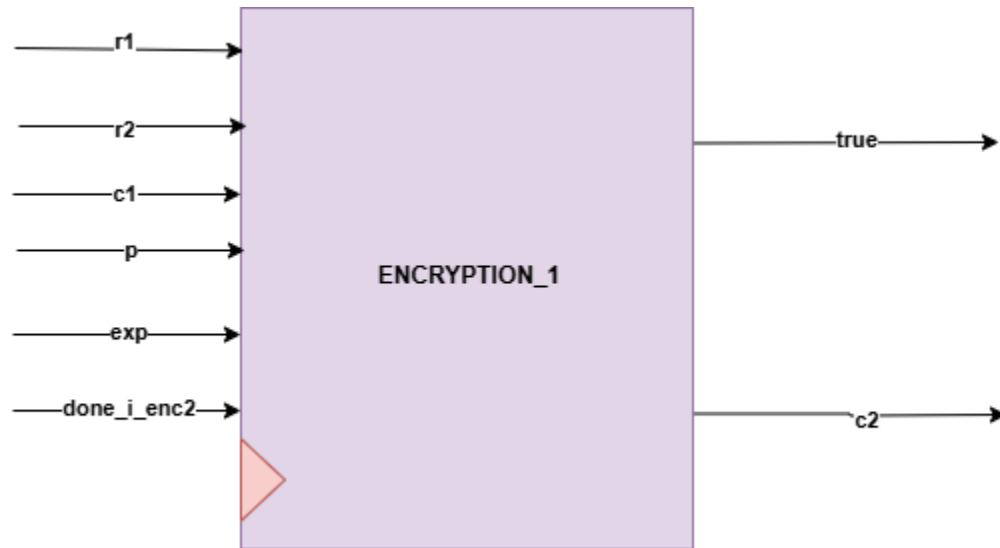
## ENCRYPTION\_1

### Introduction

This block contain 3 blocks ( CLC\_K , CHECK\_1 , CLC\_C2 ), which calculate required key then decrypt by this key c1 input to get R2` then check if received R2 = R2` .

### Design and Implementation:

Block Diagram



### Interfaces

| Signals     | Width  | Interface           | Description                           |
|-------------|--------|---------------------|---------------------------------------|
| R1          | INPUT  | U0_CLC_R1           | $R1 = (g^x) \bmod p$                  |
| R2          | INPUT  | U0_CLC_R2           | $R2 = (g^y) \bmod p$                  |
| C1          | INPUT  | U0_ENCRYPTION_R2    | $C1 = K, \text{ExOR } R2$             |
| exp         | INPUT  | U2_exponentiation_r | $(g^x)$                               |
| P           | INPUT  | TOP MODULE INPUT    | The prime number p must be very large |
| Done_i_enc2 | INPUT  | U2_exponentiation_r | Start flag                            |
| true        | OUTPUT | U0_CONTROLKER       | Flag for first check was correct      |
| C2          | OUTPUT | U0_CHECK_2          | $C2 = E(K, R1)$                       |