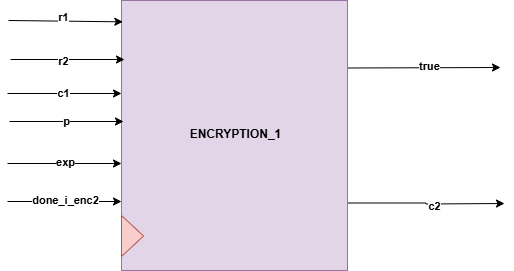
**ENCRYPTION\_1**

**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) mod p |
| R2 | INPUT | U0\_CLC\_R2 | R2 = (g^y) mod p |
| C1 | INPUT | U0\_ENCRYPTION\_R2 | C1 = K, 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) |