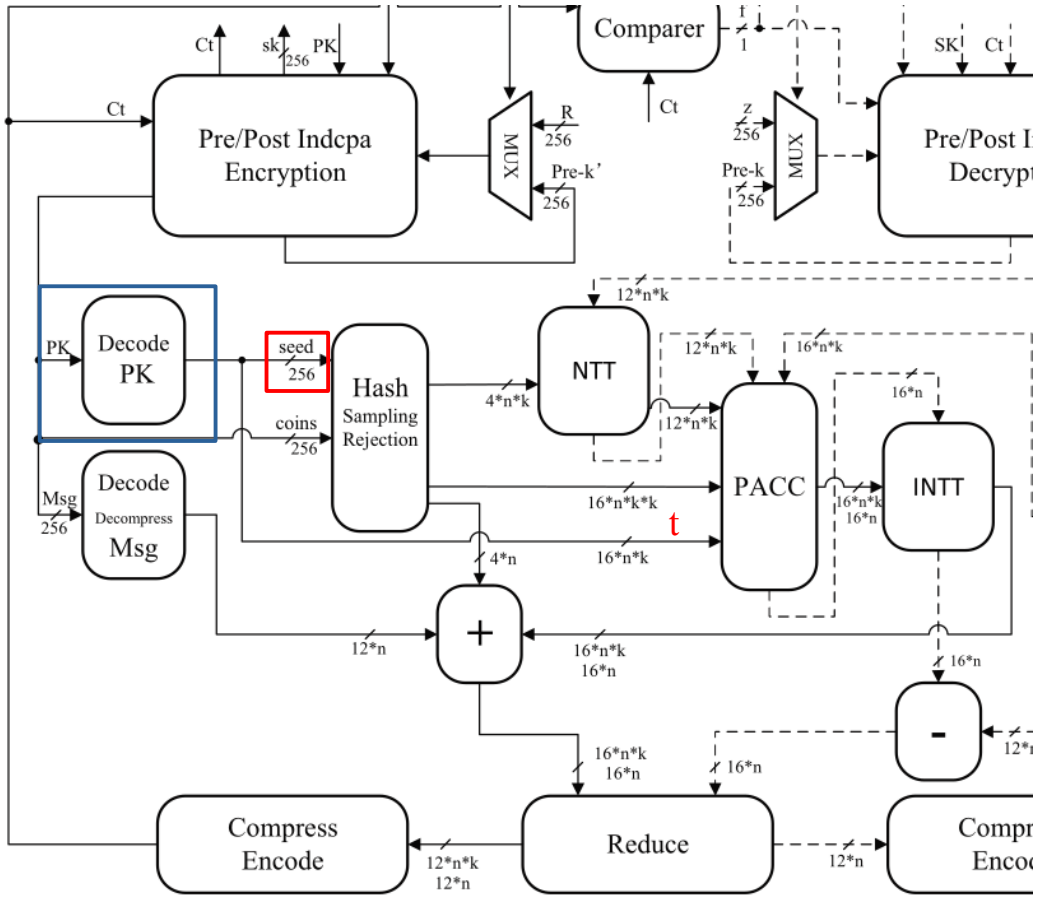


Decode PK

Split public key into two part

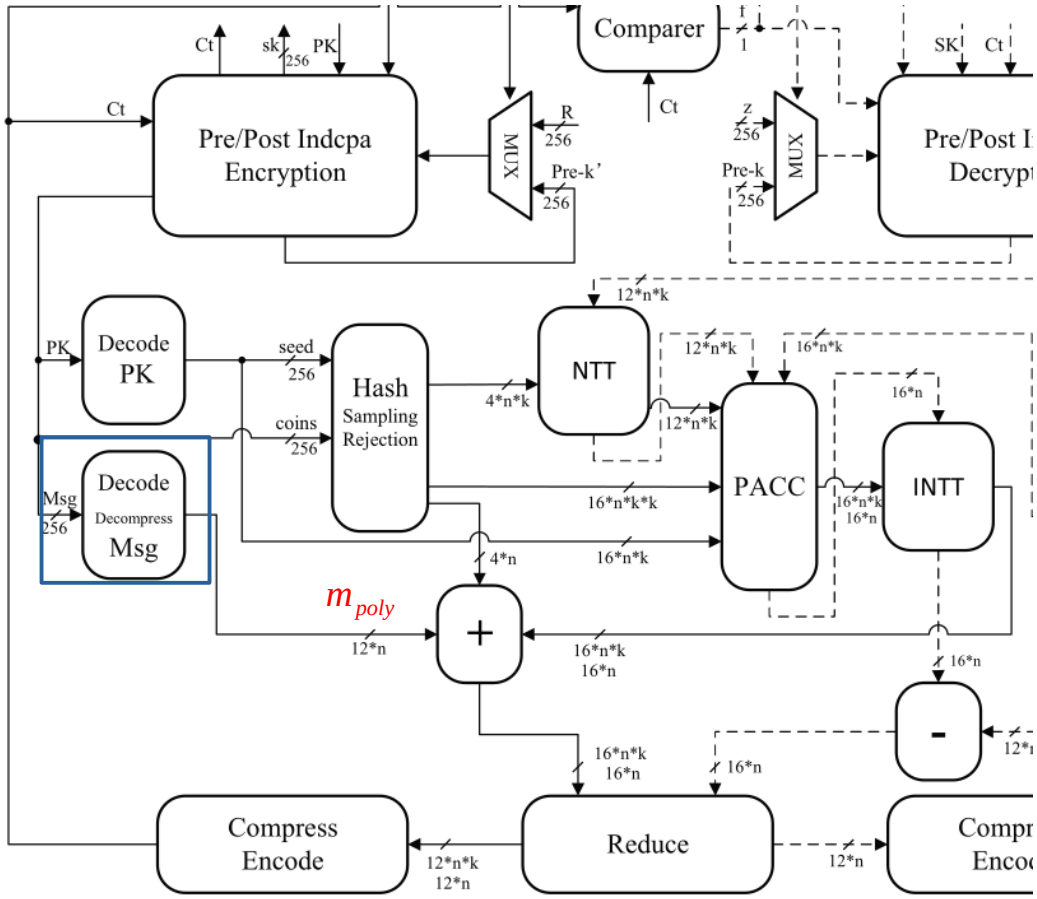
Input : public keys $PK=(\rho,\hat{t})$

$$\rho=\{0,1\}^{256},\hat{t}=\begin{pmatrix}R_q\\R_q\\R_q\end{pmatrix}$$

Output

1) Seed = ρ
2) $\hat{t}$ Increase coef size 12→16bits

$$\rho=\{0,1\}^{256},\hat{t}=\begin{pmatrix}P\\P\\P\end{pmatrix}$$



Decode Msg

Input : plain text message

$$m=\{0,1\}^{256}$$

q=3329

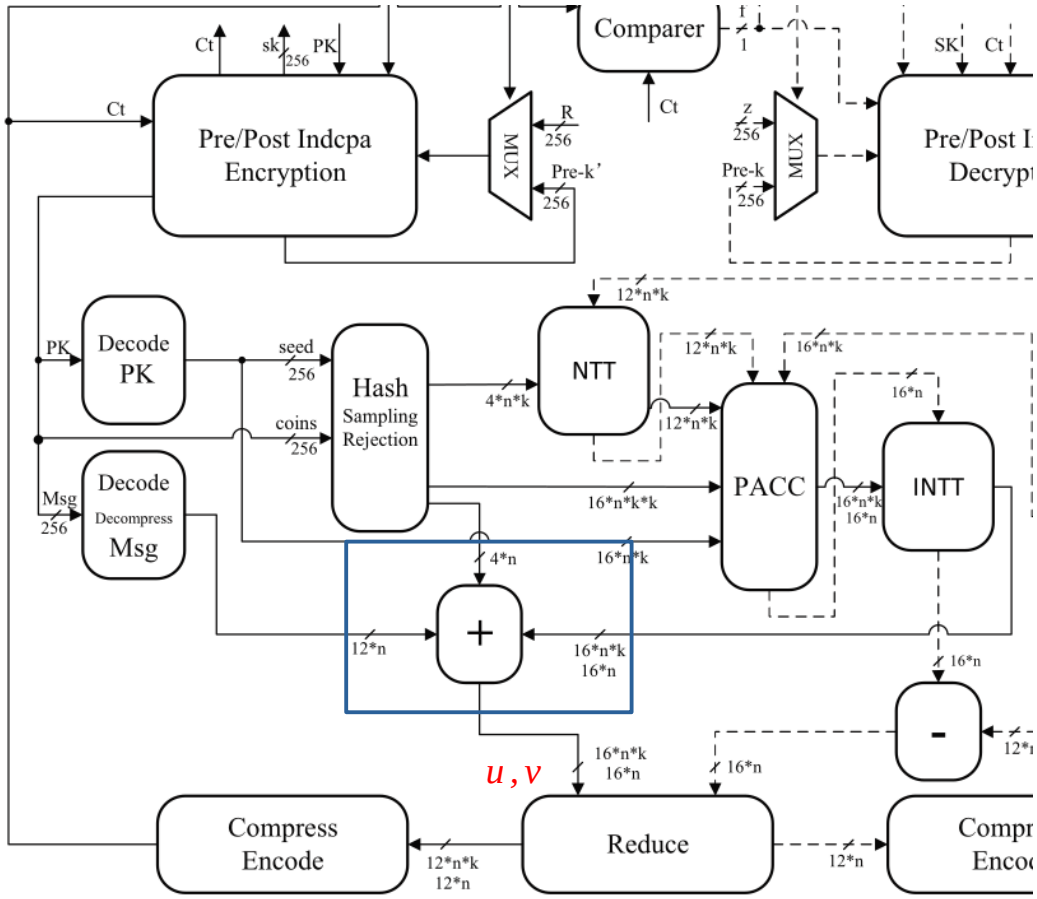
Calculate

$$m_{poly}=rounded(q/2)m=1665\,m$$

For every bit covert: $0 \rightarrow 0$, $1 \rightarrow 1665$

Output : m_{poly} (polynomial form R_q)

$$m_{poly}=R_q$$



Addition Module (+)

Compute polynomial addition

Input

1) x, y
2) m_{poly}
3) e_1, e_2

$$x = \begin{pmatrix} P \\ P \\ P \end{pmatrix},\ y = P$$

$$m_{poly} = R_q \quad e_1 = \begin{pmatrix} S_\eta \\ S_\eta \\ S_\eta \end{pmatrix},\ e_2 = S_\eta$$

Output

1) $u = x + e_1$
2) $v = y + e_2 + m_{poly}$

$$u = \begin{pmatrix} P \\ P \\ P \end{pmatrix},\ v = P$$

Decode SK

Decode Encapsulation key then Transpose encryption key

Input : Private key

$SK = (\hat{s}, PK, pre-k, coin)$

$$\hat{s} = \begin{pmatrix} S_\eta \\ S_\eta \\ S_\eta \end{pmatrix} \quad \hat{t} = \begin{pmatrix} R_q \\ R_q \\ R_q \end{pmatrix}$$

$$pre-k, coin, \rho = \{0,1\}^{256}$$

Output : decryption key $\hat{s}^T$ in polynomial form

$$\hat{s}^T = (P \quad P \quad P)$$