Issue:-

at Tx, $d[0] \rightarrow \boxed{\text{quantizer}} \rightarrow dq[0]$

at Rx

$dq[0] \rightarrow \bigcirc$ — $m[0]$

$\hat{m}[0] \leftarrow \boxed{\text{predictor}}$

$$d[0] = m[0] - \hat{m}[0]$$

But predictor at Tx, takes $m[0]$ & finds $\hat{m}[1]$

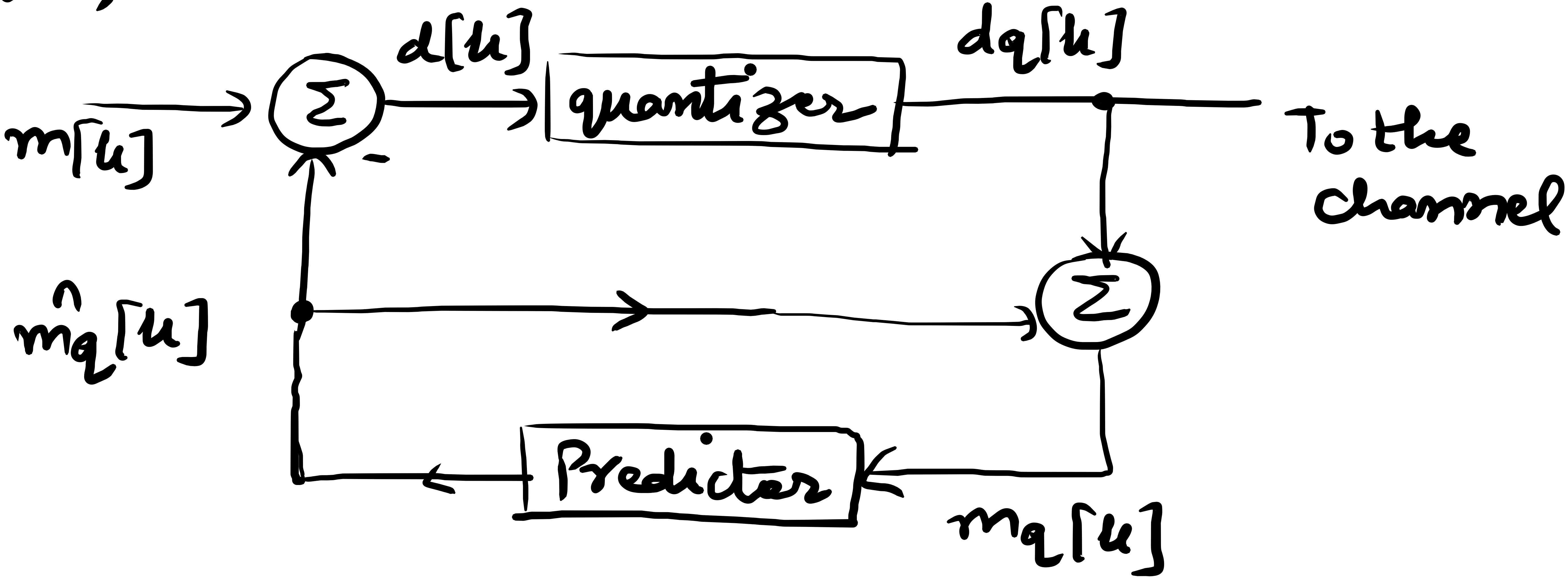
$$\bar{m}[0] = \hat{m}[0] + dq[0]$$

$$= \hat{m}[0] + d[0] + q[0]$$

$$\bar{m}[0] = m[0] + q[0]$$

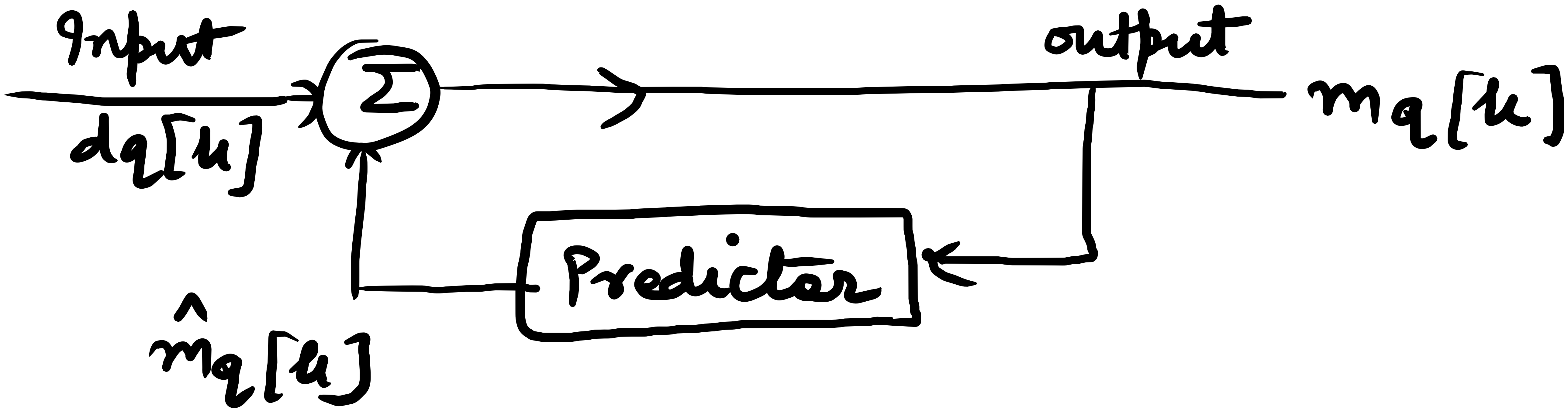
now, predictor stores $\bar{m}[0]$ as the correct value of $m[0]$. Next based on $\bar{m}[0]$, it finds $\hat{m}[1]$

At Tx,



$m[k]$
Σ
$d[k]$
quantizer
$dq[k]$
To the channel
Σ
$\hat{m}_q[k]$
Predictor
$m_q[k]$

At Rx,



Input
$dq[k]$
Σ
output
$m_q[k]$
Predictor
$\hat{m}_q[k]$

$d[k] = m[k] - \hat{m}_q(k]$

$\hat{} \rightarrow$ estimate

$m_q[k] = \hat{m}_q[k] + d_q[k]$

$= m[k] - d[k] + d_q[k]$

$= m[k] + q[k]$

$m_q[k] = m[k] + q[k]$

$d_q[k] = d[k] + q[k]$

Issue (contd.):- Instead of past samples $m[k-1]$, $m[k-2]\ldots$, as well as $d[k]$, we have their quantized versions $m_q[k-1], m_q[k-2], \ldots$. Hence, we cannot determine $\hat{m}[k]$.

We can det. $\hat{m}_q[k]$ (estimate of the quantized sample $m_q[k]$) in terms of the quantized samples $m_q[k-1], m_q[k-2], \ldots$; so an $\uparrow$ in error in reconstruction.

<u>Strategy</u>:- Determine $\hat{m}_q[k]$ (estimate of $m_q[k]$ instead of $m[k]$) at the Tx. also from quantized samples $m_q[k-1], m_q[k-2], \ldots$

$$Tx \rightarrow \text{diff} \rightarrow m[k] - \hat{m}_q[k] \text{ via PCM.}$$

$$d[k] = m[k] - \hat{m}_q[k]$$

↳ quantize it to yield:- $d_q[k] = d[k] + q[k]$

Predictor O/P is fedback to its input so that predictor I/P $m_q[k]$ is

$$m_q[k] = \hat{m}_q[k] + d_q[k] = m[k] + q[k]$$

We are able to receive the desired signal $m[n]$ + quantizn. noise $q[n]$, which is associated with difference signal $d[n]$ & is generally much smaller than $m[n]$.

Received samples $m_q[n]$ are decoded & passed through a LPF for D/A conversion.

TDM – time division multiplexing $\rightarrow$ $1/4000$

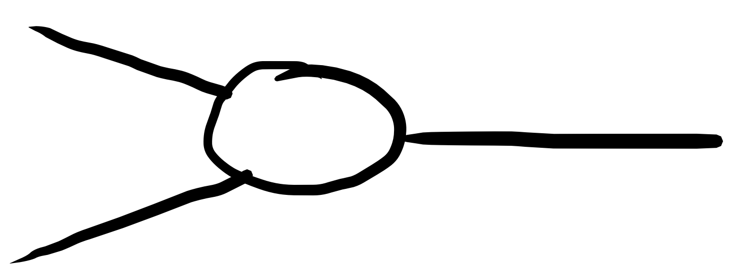
$b_1 \; b_2 \; b_3 \; b_4$

$a_1 \; a_2 \; a_3 \; a_4 \; a_5 \; a_6 \; a_7 \; a_8$

$\rightarrow 1/8000$

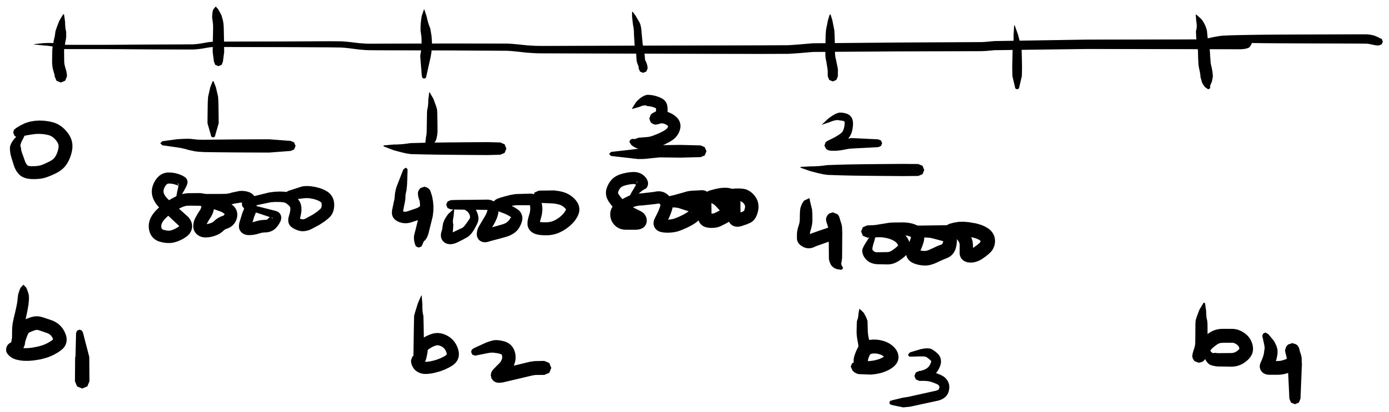
$S_1 \rightarrow 4$ Kbps

$S_2 \rightarrow 8$ Kbps



$b_1 \; a_1 \; a_2 \; b_2 \; a_3 \; a_4 \; b_3 \; a_5 \; a_6 \; b_4 \ldots$ Interleaved bit sequence

$0 \quad 1/8000 \quad 2/8000 \quad 1/4000$



not possible this way!

respecting the timing information

$$\frac{1}{4} + \frac{1}{8} = \frac{3}{8}$$