For C_3 and C_4, the carry generation, propagation and annihilation behavior of both cases are identical. Hence we only discuss $C(\tau) = C_3$. In this case, (**??**) is modified to give (1). We have $Y_{[\tau+1]} = Y_{[\tau]}$ as the appending digit $y_{\tau+\delta+1} = 0$. Thus D only depends on the number of leading non-zero digits of $Y_{[\tau]}$.

$$P_{[\tau+1]} = r^{-\delta+1}(x_{\tau+\delta+1}Y_{[\tau+1]}) = r^{-\delta+1}(x_{\tau+\delta+1}Y_{[\tau]}) \qquad (1)$$

For the first stage where $\tau = -\delta$, substituting this into (1) yields $P_{[-\delta+1]} = r^{-\delta+1}(x_1 y_1)$. Therefore the value in (**??**) can only be obtained when $C(-\delta) = C_2$, otherwise $d(-\delta) = 0$.

For the other stages where $\tau > -\delta$, there are 2 possible situations for $\tau' = \tau - 1$as stated below:

- If $y_{\tau'+\delta+1} \neq 0$, then D equals to the maximum word-length of $Y_{[\tau'+1]}$. Thus we have $d(\tau) = d(\tau')_{max} - 1$.
- If $y_{\tau'+\delta+1} = 0$, then $Y_{[\tau']} = Y_{[\tau'-1]}$. This is a recursive process and these two judgements will be performed again for $\tau'' = \tau' - 1$. It terminates when the first judgement is satisfied or .