Conclusion : On vient de voir que l'utilisation de PRNG seul est faible pour la sécurité ; d'où on fait appel à CS-PRNG (i.e. Cryptographiquement secure PRNG) : si on a $b_1 \dots b_n$, il est difficile calculatoirement de calculer les bits $b_{n+1}, b_{n+2} \dots$

La question est donc :

Comment construire des cryptosystèmes à longueur variable pratique ? On a 3 propositions :

(1) S-cryptosystème optimaux par le software :

peu d'instructions CPU par calculer un bit d'une S-clé

(2) S-cryptosystème facile par le hardware :

Un exemple connu suffisamment est l'utilisation de LSFR (Linear-Shift Feed-Back Register)

(3) S-cryptosystème basé sur le B-cryptage.

(par ex : CFB, OFM, ... voir 5 modes opératoires, après avoir étudié le DES ou le AES).