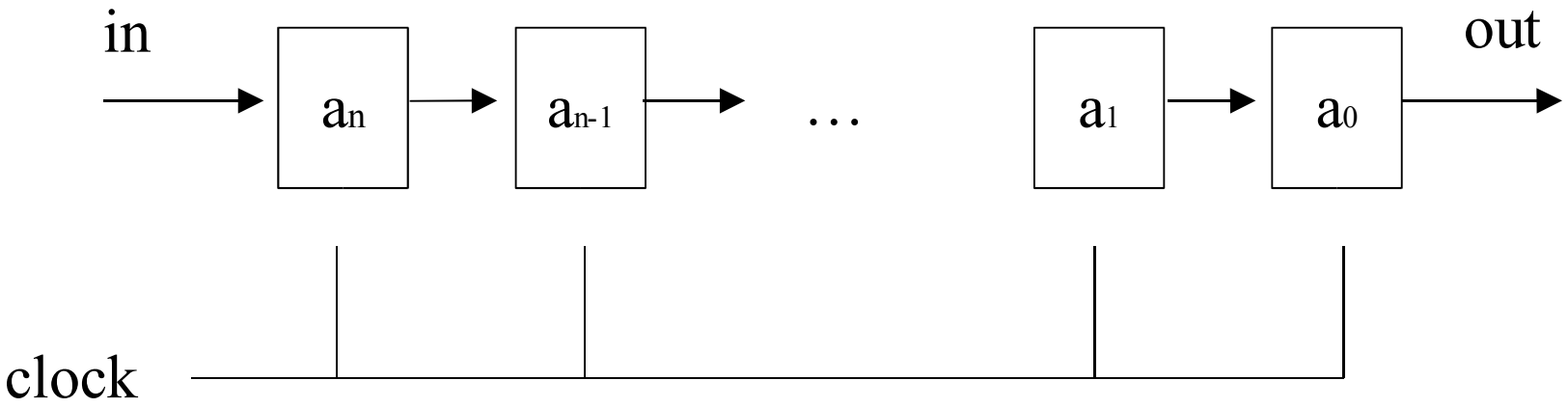
# Registri a scorrimento lineare controllati da clock

Roberto Manicardi

# RSL controllati dal clock



# RSL controllati dal clock

L'idea:

clock di un RSL dipendente da un'altro RSL

- Caratteristiche
  - Non linearita'
  - Potenzialmente generano sequenze piu' complesse (periodo piu' lungo,alta complessita' lineare)

# Importanza dei parametri

Perche' e' importante la lunghezza del periodo:

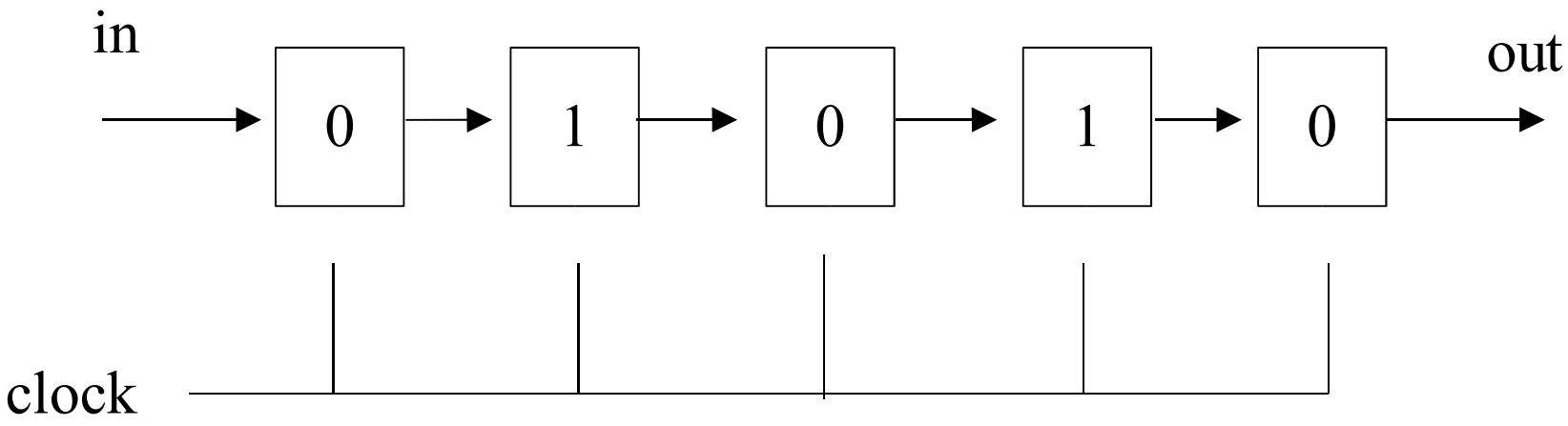
| | | |
|---|---|---|
| Se | 2^32 | lunghezza del periodo |
| e | 1Mb/s | velocita' di trasmissione |
| allora | 8.5 min | tempo dopo il quale si ripete la chiave |

Perche' e' importante la complessita' lineare:

(Berlekamp-Massey)

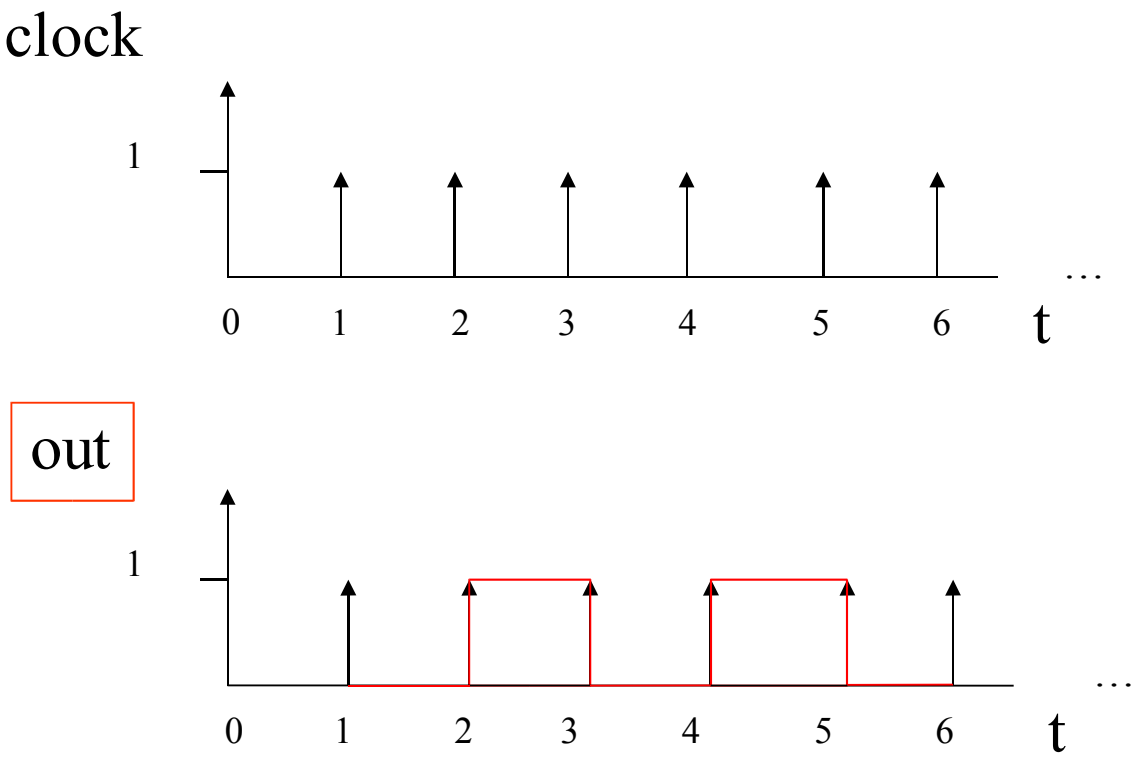
Se L(u) = k scopro la chiave con 2*k bit del messaggio in chiaro

# RSL controllati dal clock



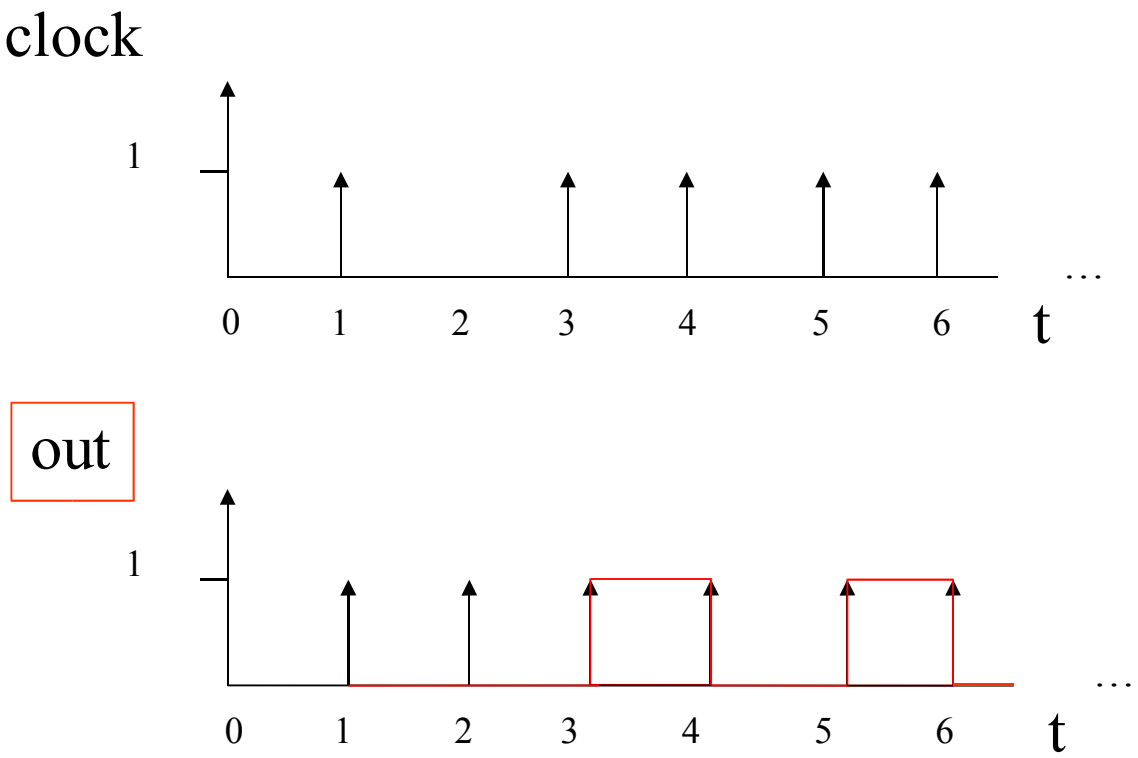

out = 0 , 1 , 0 , 1 , 0 , ….

# RSL controllati dal clock




out = 0 , 1 , 0 , 1 , 0 , ….

# RSL controllati dal clock

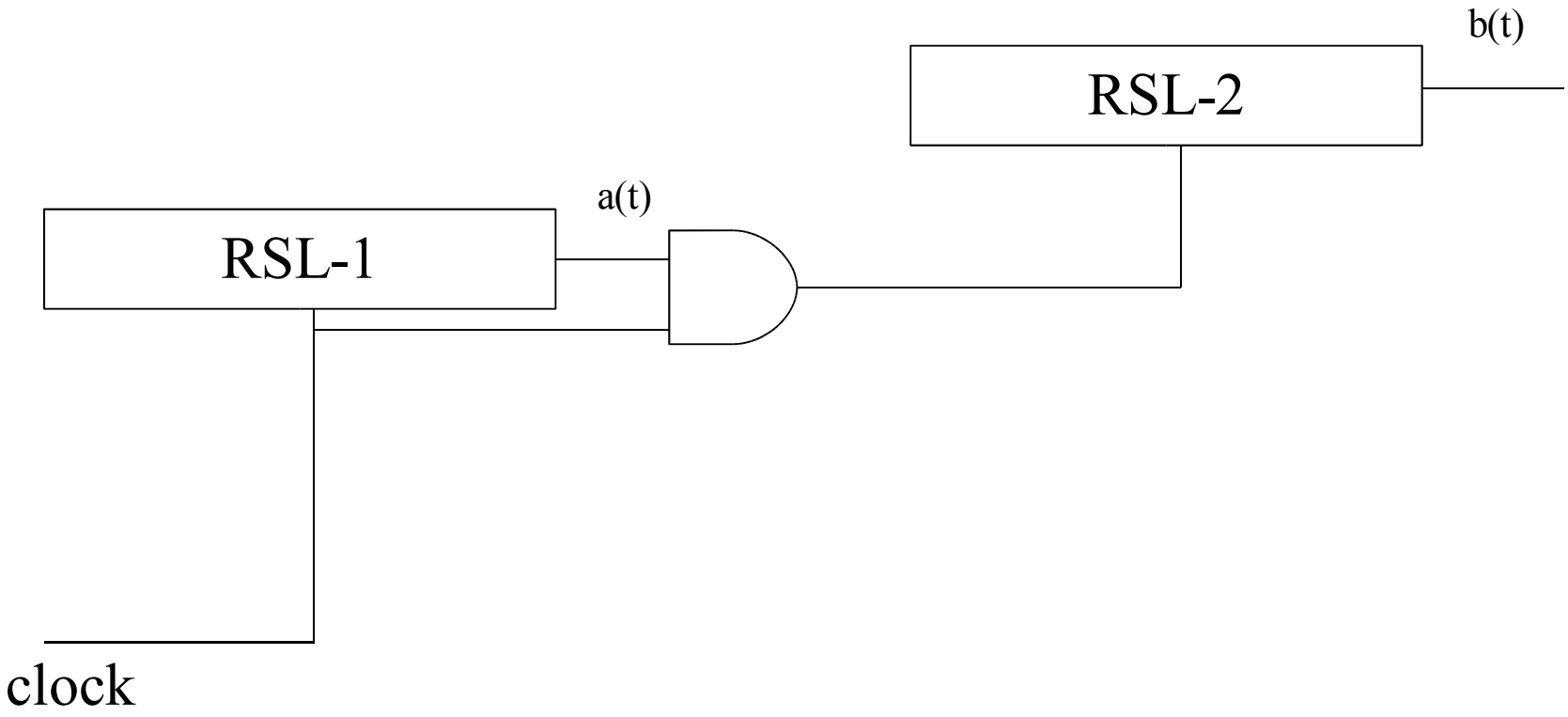



out = 0 , 0 , 1 , 0 , 1 , 0 , …

# Generatori controllati da clock

- Stop-and-go generator
  - T. Beth , F .Piper (1984)
- Cascade generator
  - D. Gollmann (1985)
- Shrinking generator
  - D. Coppersmith , H. Krawczyk , Y. Mansour (1994)

# Generatore stop-and-go



# Generatore stop-and-go : Periodo

Se

$P_1$ periodo di RSL-1

$P_2$ periodo di RSL-2

Qual'e' il periodo totale $P_t$?

Dipende!

# Generatore stop-and-go : Periodo

**w**: n. di 1 nella sequenza generata da RSL-1

a(t) = 0 1 1 0 0 1 0 1 0 0
w = 4

condizione <u>necessaria</u>:

se $(w,P_2) = 1$ allora

$$P_t = P_1 * P_2$$

altrimenti non si puo' dire nulla

# Generatore stop-and-go : Periodo

Esempio:

$s(t) = \{ s_0\ s_1\ s_2 \}$ sequenza generante

$c(t) = \{ 1\ 0\ 1\ 0\ 1 \}$ sequenza che regola il clock

$u(t) = \{ s_0\ s_0\ s_1\ s_1\ s_2 \}$ sequenza in output

$P_1 = P_s = 3$

$P_2 = P_c = 5$

$w = 3$

$P_t = 15$ ?

no, $P_t = 5$ !

# Generatore stop-and-go : complessita' lineare

- $L(u) = (2^{(L(RSL\text{-}1))} - 1) * L(RSL\text{-}2)$

$L(RSL\text{-}1)$ = complessita' lineare di RSL-1

$L(RSL\text{-}2)$ = complessita' lineare di RSL-2

L(u) = complessita' lineare della sequenza in uscita

# Crittanalisi dei generatori controllati da clock

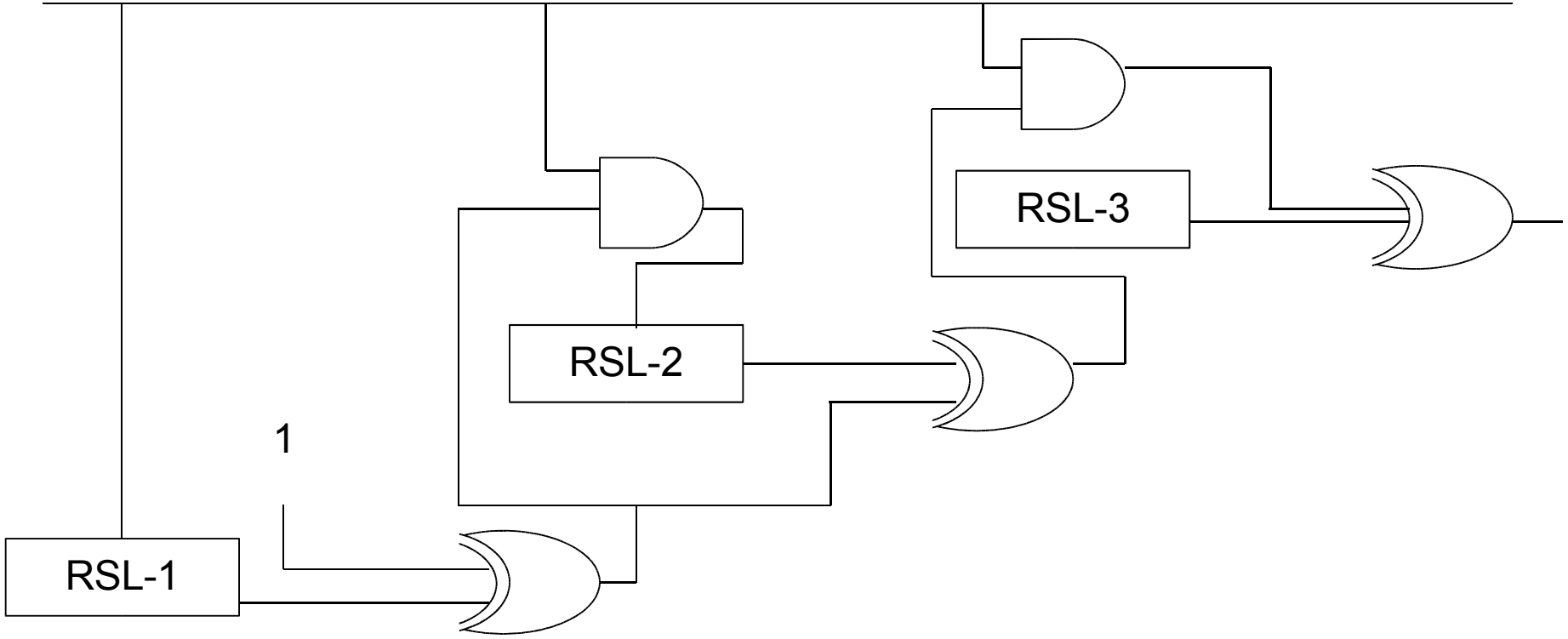
- Attacchi a clock controllati (golic' e o'conner), estensioni dell'attacco a correlazione:
  - Embedded (esaustivo)
  - Probabilistic

Metodi per individuare la differenza tra la sequenza di output e quella del generatore se cloccato regolarmente

# Generatore cascade

- Semplici da realizzare
- Generano sequenze con
  - Periodo lungo
  - Grande complessita' lineare
  - Buone proprieta' statistiche
- Risolve alcuni problemi di debolezza statistica degli stop-and-go

# Generatore cascade



# Generatore cascade

- Periodo
  - Considerazioni simili allo stop-and-go
- Complessita' lineare
  - n*(2^(n)-1)^(k-1) con k numero e n lunghezza degli RSL
- Attacco Lock-in
  - si ricostruisce l'input dell'ultimo RSL nella cascata,poi l'input del penultimo, etc… fino al messaggio in chiaro

# Generatore shrinking

Due RSL S e C che generano rispettivamente s(t) e c(t)

$s(t) = a_1\ a_2\ a_3\ a_4\ a_5\ a_6\ \ldots$

$c(t) = 1\ 0\ 1\ 1\ 0\ 1\ \ldots$

$z(t) = a_1\ a_3\ a_4\ a_6\ \ldots$

z(t) e' ottenuta come sottosequenza della sequenza pseudocasuale generata s(t)

- Un problema: z(t) non e' sincronizzata con il clock!

# Generatore shrinking - caratteristiche

- Periodo
  - n = |S|  m=|C|
  - Se S e C periodo massimo
  - Se (Ps,Pc)=1
  - Allora $P_t$= Ps*2^(m-1)=(2^n-1) *2^(m-1)
- Complessita' lineare
  - Sotto le condizioni precedenti

    n*2^(m-2) < L(u) < n*2^(m-1)

# Generatore shrinking - crittoanalisi

- Non funzionano gli attacchi classici ( analisi delle funzioni booleane, correlazione)
- Esistono dei metodi alternativi
  - Ricerca del seme di C (richiede conoscenza di S)
  - Attacco sulla complessita' lineare (esaustivo)

# Conclusioni

- GSM
  - A5