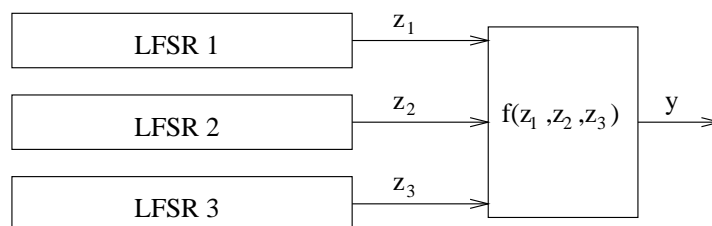


4. Übungsblatt
Kryptographie I (SS 2006)

Stefan Lucks, Emin Islam Tatlı

Besprechung: Mittwoch, 24. Mai 2006

Aufgabe 1 Betrachte die PZBG, die wie folgt aufgebaut sind:



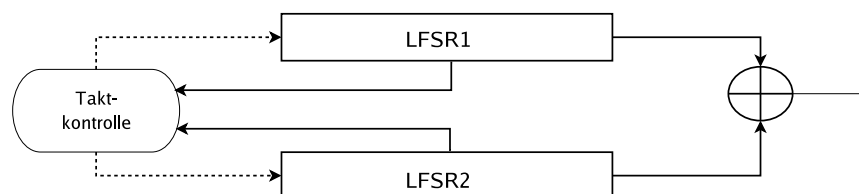
Dabei kannst Du davon ausgehen, dass die Feedback-Polynome bekannt sind und dass es sich um maximale LFSR mit teilerfremden Perioden handelt. Die folgenden Kombinationsfunktionen definieren drei solche Generatoren:

- a) $f(z_1, z_2, z_3) = z_1 \oplus z_2 \oplus z_3$
- b) $f(z_1, z_2, z_3) = z_1 \wedge z_2 \wedge z_3$
- c) $f(z_1, z_2, z_3) = z_1 \oplus (z_2 \vee z_3)$

Gib für jeden dieser Generatoren einen möglichst effizienten Angriff an und schätze dessen Laufzeit grob ab.

Bem.: Solche Generatoren heißen *Kombinations-Generatoren*.

Aufgabe 2 Betrachte den folgenden PZBG mit Taktkontrolle.



Die *mittleren* Bits m_1 und m_2 der LFSR dienen als Input für die Taktkontrollfunktion $t: \{0,1\}^2 \rightarrow \{0,1\}^2$.

Deren Verhalten ist:

$$t(m_1, m_2) = \begin{cases} (1, 1) & \text{falls } m_1 = m_2 \\ (m_1, m_2) & \text{sonst} \end{cases}$$

d.h. wenn $(m_1, m_2) = (1, 1)$ ist, beide Register getaktet, sonst ist genau eines der beiden Bits $m_i = 1$, und das zugehörige LFSR getaktet.

- a) Wie viele Register sollten im Durchschnitt getaktet werden?
- b) Wie häufig wird jedes Register im Durchschnitt getaktet?
- c) Gib einen Angriffsbaum an. Welche Werte kann man in dem Baum raten?

Aufgabe 3 RC4 ist ein Pseudozufallsbitgenerator, der in weit verbreiteten Sicherheitsprotokollen wie SSL, SSH und WEP verwendet wird. Die Struktur von RC4 ist in <http://en.wikipedia.org/wiki/Rc4> beschrieben.

- a) Implementiere den RC4 PZBG Algorithmus in einer Programmiersprache Deiner Wahl.
- b) Erzeuge mit dem Schlüssel "KRYPTO" (in Hex: $0x4B525950544F$) 1000 Schlüsselstrombytes, und gib die letzten 10 von ihnen an.