A Appendix

A.1 RSP Statistics

If we know that RSP which is in our registers synchronizes after *n* steps, then we have limited choice of patterns of our registers and therefore we have some gain which can be expressed in bits (we do not have to guess 64 bits of registers, but 64-*k*, where *k* is our gain.) These gains for particular classes of RSP are summarized in Table 1.

Table 1. Gains for different RSP lengths

RSP#	2	3	4	5	6	7	8	9
gain	8	13	14.54	16.93	19.31	21.45	23.63	25.80

Table 2. Some properties of RSP5

length of RSP	# of variables	matrix rank	# of cases
12	13	7	4
12	13	8	4
13	13	7	2
13	13	8	4
13	14	8	8
14	14	8	6
15	15	9	2

RSP5s Resynchronization occurs after 5 step and we get $5+5-1=9$ different equations. On average (see Table 2), we get 12.6418 bits from the pattern, the system has 13.3582 variables, 9 equations and its rank is 7.71642. There are 30 RSP5s. Some of the RSPs are shorter, some are longer. So we consider also *fixed length* RSP: we add some number of bits to each RSP considered in order to get the same length of each pattern (15 in the case of RSP5.) Then we get 134 RSP5s of length 15. Probability of getting RSP5 is $\frac{134}{2^{15}} = 0.00408936$. If we guess all 15 bits of fixed length RSP5, then after solving the system of equations we have to guess missing A $13.3582 - 7.71642$ bits. But from our 2^{15} RSPs about $2^{9-7.71642}$ are rejected without further investigating, so in fact we are guessing only $(\log 134) - 9 + 7.71642$ bits for normalized RSP. Our gain is equal to

$$15 + 13.3582 - (13.3582 - 7.71642) - \big((\log 134) - 9 + 7.71642\big) = \\ = 15 + 9 - \log 134 = 16.93$$

bits.

Table 3. Some properties of RSP6

length of RSP	# of variables	matrix rank	# of cases
14	15	8	4
14	15	9	12
14	15	10	4
15	15	8	2
15	15	9	8
15	15	10	4
15	16	9	12
15	16	10	16
16	16	9	8
16	16	10	16
16	17	10	12
17	17	10	12
18	18	11	2

RSP6s There are 11 equations. On average (for details see Table 3), we get 14.8523 bits from the pattern, the system has 15.569 variables and its rank is 9.34383; there are 112 RSP6s of variable length and 826 RSP6s of length 18; probability of getting RSP6 is $\frac{826}{2^{18}} = 0.00315094$. The gained bits we calculate just as for RSP5: $18+11-\log 826 = 19.31$bits.

Table 4. Some properties of RSP7

length of RSP	# of variables	matrix rank	# of cases
16	17	9	4
16	17	10	20
16	17	11	28
16	17	12	4
17	17	9	2
17	17	10	12
17	17	11	24
17	17	12	4
17	18	10	16
17	18	11	68
17	18	12	32
18	18	10	10
18	18	11	52
18	18	12	28
18	19	11	36
18	19	12	36
19	19	11	26
19	19	12	40
19	20	12	16
20	20	12	20
21	21	13	2

RSP7s There are 13 equations. On average (details in Table 4), we get 17.0481 bits from the pattern, the system has 17.7645 variables and its rank is 10.9993; there are 480 RSP7s of variable length and 5986 RSP7s of length 21; probability of getting RSP7 is $\frac{5986}{2^{21}} = 0.00285435$. Gain: $21+13-\log 5986 = 21.45$ bits.

Table 5. Some properties of RSP8

length of RSP	# of variables	matrix rank	# of cases
18	19	10	4
18	19	11	28
18	19	12	68
18	19	13	44
18	19	14	4
19	19	10	2
19	19	11	16
19	19	12	52
19	19	13	40
19	19	14	4
19	20	11	20
19	20	12	136
19	20	13	180
19	20	14	64
20	20	11	12
20	20	12	96

length of RSP	# of variables	matrix rank	# of cases
20	20	13	164
20	20	14	60
20	21	12	60
20	21	13	228
20	21	14	100
21	20	14	8
21	21	12	40
21	21	13	200
21	21	14	96
21	22	13	80
21	22	14	64
22	22	13	66
22	22	14	80
22	23	14	20
23	23	14	30
24	24	15	2

RSP8s There are 15 equations. On average, we get 19.2876 bits from the pattern (see Table 5), the system has 19.9923 variables and its rank is 12.6763; there are 2068 RSP8s of variable length and 42070 RSP8s of length 24; probability of getting RSP8 is $\frac{42070}{2^{24}} = 0.00250757$. Gain equals $24 + 15 - \log 42070 = 23.63$ bits.

RSP9s There are 17 equations. On average (see Table 6), we get 21.5097 bits from the pattern, the system has 22.2072 variables and its rank is 14.3567; there are 8992 RSP9s of variable length and 301182 RSP9s of length 27; probability of getting RSP9 is $\frac{301182}{2^{27}} = 0.00224398$. Gain is $27 + 17 - \log 301182 = 25.80$ bits.

Table 6. Some properties of RSP9

length of RSP	# of variables	matrix rank	# of cases
20	21	11	4
20	21	12	36
20	21	13	124
20	21	14	180
20	21	15	60
20	21	16	4
21	21	11	2
21	21	12	20
21	21	13	88
21	21	14	148
21	21	15	56
21	21	16	4
21	22	12	24
21	22	13	220
21	22	14	576
21	22	15	396
21	22	16	136
22	22	12	14
22	22	13	148
22	22	14	464
22	22	15	360
22	22	16	144
22	23	13	88
22	23	14	604
22	23	15	732
22	23	16	284
23	22	16	40
23	23	13	56
23	23	14	468
23	23	15	700
23	23	16	280
23	24	14	188
23	24	15	560
23	24	16	224
24	23	16	36
24	24	14	134
24	24	15	560
24	24	16	240
24	25	15	144
24	25	16	100
25	25	15	138
25	25	16	140
25	26	16	24
26	26	16	42
27	27	17	2

A.2 Attack Complexity in Details

Phase 1 We just wait for the frame where output of a "good" and a "faulty" sequence resynchronize after 5 to 8 steps. From Table 1 in [2] we see that chances for such an event are $0.34+0.28+0.25+0.22=1.09$ percent. If we observe resynchronization after 9 steps, then we have a chance of about 45% that none of RSPk, $k \in \{5,6,7,8,9\}$ occurred. This is why such a very long RSP is not well suited for cryptanalysis ("very long" i.e. of length close to natural boundaries of the registers.) Chances that we have an RSPk for $k \in \{5,6,7,8,9\}$, if output synchronizes after 5 steps are

$$60.61\% + 23.33\% + 3.76\% + 5.11\% + 3.03\% = 95.84\%.$$

Analogous numbers for output resynchronization after 6,7,8 steps are 91.95%, 86.18%, 81.18%. Thus chances that we have one of RSP$\{5,6,7,8,9\}$, if we observe output resynchronization after 5,6,7,8 steps are about 90% which seems reasonable:

$$\frac{95.84 \cdot 0.34 + 91.95 \cdot 0.28 + 86.18 \cdot 0.25 + 81.18 \cdot 0.22}{0.34+0.28+0.25+0.22} = 89.67\%.$$

Phase 2 We list all possible patterns (of course if output re-synchronizes after step e.g. 7, then we consider RSP7s, RSP8s and RSP9s only; obviously we must exclude RSP5s and RSP6s) and then for each pattern we solve (partially precomputed) system of linear equations related to a given pattern. The exact numbers of patterns are given above in Appendix A.1. During this second phase we have to guess some number of bits (the difference between the number of equations and the rank of the system.) During this second phase more than 70% of patterns are excluded (each RSPk pattern has about 2^{r-2k-1} chances of passing this phase, r is the rank of the system, $2k-1$ is the number of equations in the system for RSPk.)

Phase 3 This phase closely resembles attack presented in [1]. We need 64 linearly independent equations in unknowns representing the bits contained in the registers at the moment when the fault is injected. Most of the equations constructed are of the form

$$unknown = value$$

or get translated into this form during guessing some other bits. Therefore solving such systems of linear equations will demand only few tens of binary additions.

As it was mentioned in the overview of the attack, in Phase 3 we gradually guess the values of unknown bits needed for the clocking mechanism, emulate a move of the system with the values guessed and construct a linear equation with current rightmost bits of the registers and the output bit.

Now we estimate how many such equations we have to inspect. Note that the numbers obtained below for an average pattern have to be multiplied by the number of patterns.

Suppose that the pattern considered has length p and that it is RSPk, for $k = 5 \ldots 9$. So each register has on average $p/3$ bits in the clocking window and to the left of it with the values indicated by the pattern. It turns out that for patterns of length p the average number of unknowns in system of linear equations constructed for the pattern is $p + 0.7$ (compare Appendix A.1.) So for an average pattern we have equations that define $(p + 0.7)/3$ rightmost bits.

Before starting further computations we emulate the work of the system for k steps. So far we have about $2p + 0.7$ equations (or, in this case, known bits.) So we still need $64 - 2p - 0.7$ additional equations. As it was observed in [1], not all 64 bit content of three registers may be the successor of some other state. In fact $\frac{3}{8}$ states have no predecessors, and they can be filtered out by additional linear equations. So the number of possible states is $2^{64} \cdot \frac{5}{8} = 2^{63.32}$ and now we lack on average only $62.62 - 2p$ equations.

After k steps we have unknown bits on the rightmost positions and on the positions to the left of clocking window. The bits of resynchronization pattern are located immediately to the right of the clocking window. Now we gradually guess bits on clocking positions, but no more than we have unknown bits on the rightmost positions. Note that there are 33 positions to the right of the clocking window and p of them contain the bits of the resynchronization pattern. Hence in this step we have to guess about $33 - p$ bits, clock the system and obtain about $\frac{4}{3}\frac{33-p}{3}$ linear equations for our unknown rightmost bits (one equation for each move.) One can easily see that these equations are linearly independent – it follows from the fact that the equation describing move i contains two or three unknowns that have not occurred in the equations related to moves 1 through $i - 1$. Now we lack only about

$$62.62 - 2p - (33 - p) - \frac{4}{3} \cdot \frac{33 - p}{3} = 14.95 - 0.56p$$

equations.

At this moment the state of our registers is such that a few rightmost positions are known (they contain bits from resynchronization pattern.) So we guess further bits approaching the clocking window and check their consistency with the output generated. This helps to filter out some of the guesses for the clocking positions. We may estimate the number of cases obtained as follows. Think of a tree with all valid options for possible values of bits in the clocking window. If we have to decide upon the next move, with probability $\frac{3}{4}$ we have to fetch only 2 bits (since one of the registers is not clocking) and with probability $\frac{1}{4}$ we fetch new 3 bits to the clocking window. So the average number of possibilities to consider is $\frac{3}{4} \cdot 4 + \frac{1}{4} \cdot 8 = 5$. However, on average about half the options are rejected since the clocking they imply would lead to the output bit inconsistent with the output really occurring (recall that we have rightmost bits of registers in this case – the bits already guessed and the bits from resynchronization pattern, so we may compute these output bits.) Thus each node of the tree has 2.5 children on average. A tree corresponding to h moves of the system (i.e. of depth h) gives us about $\frac{3}{4}h$ bits for every register, so the depth h required is about

$$\frac{4}{3} \cdot \frac{14.95 - 0.56p}{3} = 6.64 - 0.25p\,.$$

Since an average node in the tree has 2.5 valid children, we have about

$$2.5^{6.64-0.25p} = 2^{8.76-0.33p}$$

leaves in the tree. Earlier we guessed about $33 - p$ bits, so the number of systems of linear equations that have to be solved is on average

$$2^{33-p} \cdot 2^{8.76-0.33p} = 2^{41.76-1.33p} \qquad (1)$$

For typical values of p the above formula leads to the following values:

RSPk	average length p	number of equations
5	12.64	$2^{24.95}$
6	14.85	$2^{22.01}$
7	17.05	$2^{19.08}$
8	19.29	$2^{16.10}$
9	21.51	$2^{13.15}$

Suppose that we are considering a given RSPk pattern, let r be the rank of linear equation connected with it (from phase 2 of the attack) and let p be the length of the pattern. Then solution of this equation gives us about $2^{p+0.7-r}$ options ($p+0.7$ being the average number of equations in such a system, see Appendix A.1), so together with (1) the number of possibilities is

$$2^{42.46-r-0.33p}$$

But we have $2k-1$ equations in Phase 2 and the rank is r so about 2^{r-2k+1} of all sequences will not contradict the system of equation. In other words, probability that this pattern will pass to the Phase 3 is about 2^{r-2k+1}. So the expected number of equations derived from this pattern and considered in Phase 3 equals

$$2^{43.46-2k-0.33p} \qquad (2)$$

If we sum the above formula on the whole set of our RSPk patterns, we will have the estimation of complexity of the attack. Once again - this is the average number of linear systems considered in Phase 3.

For $k = 5$ we have 8 patterns of length 12, 14 patterns of length 13, 6 patterns of length 14 and 2 patterns of length 15 (see Appendix A.1.) So, taking formula 2 we have that for all RSP5 our attack will consider in Phase 3 on average about

$$8 \cdot 2^{29.5} + 14 \cdot 2^{29.17} + 6 \cdot 2^{28.84} + 2 \cdot 2^{28.51} = 2^{34.08}$$

systems of equations. For RSP6, RSP7, RSP8 and RSP9 we obtain about $2^{33.21}$, $2^{32.38}$, $2^{31.90}$ and $2^{31.26}$cases, respectively. The whole exhaustive search through all RSPk for $k \in \{5,6,7,8,9\}$ will take on average

$$2^{34.08} + 2^{33.21} + 2^{32.38} + 2^{31.90} + 2^{31.26} = 2^{35.23}$$

systems of linear equations. But in the attack we stop searching when we find the solution, so in fact our average complexity (in number of linear equations in Phase 3) is about half of the above number, that is

$$2^{34.23}$$

Note that this is valid when output synchronizes after step 5; if output synchronizes after for example step 6, then we skip the RSP5 part and the number of cases to be considered is smaller.

References

1. Jovan Dj. Golič, *Cryptanalysis of Alleged A5 Stream Cipher*, Eurocrypt'97, LNCS 1233, Springer, 1997, pp. 239–255
2. Marcin Gomułkiewicz, Mirosław Kutyłowski, Heinrich Theodor Vierhaus and Paweł Wlaź, *Synchronization Fault Cryptanalysis for Breaking A5/1*, Proceedings of WEA'2005