E.2 Invariancces on 12 bits

In order to compute all invariances on 12 bits we need first to create a text file, we name it 'IOquestion12.txt', of this format:

```
/LZSD=17,14,12,1,3,8,28,32,36,P=27,30,34,26,35,31,13,33,6,23,7,22,20,9,24,29,11,4,2,25,16,18,15,19,5,21,10,
a = i25
b = i26
c = i27
d = i28
e = i29
f = i30
g = i31
h = i32
i = i33
j = i34
k = i35
l = i36

K = S1
L = S2

i25 = o26
i26 = o27
i27 = o28
i29 = o30
i30 = o31
i31 = o32
i33 = o34
i34 = o35
i35 = o36
i28 = F + Z + o25 + o32
i32 = F + Z + o29
i36 = F + o33

Z = Z(L,c,f,j,b,k)
Z = Z1
```

Figure E.1: IOquestion12.txt

We use Python3 and Sagemath 8.0, so before running the files we need to execute the commands below:

```
pip install itertools
pip install sys
pip install compiler
```

Step 1: We run mongen.py by:

```
python mongen.py IOquestion12.txt
```

which takes as input the text file 'IOquestion12.txt', and it will create two separate text files, called 'draft1.txt' and 'draft2.txt'. 'draft1.txt' contains all possible monomials for the variables $\{a, b, c, \ldots, k, l\}$ and 'draft2.txt' contains all the resulting polynomials after one round, for each possible monomial in 'draft1.txt'.

Step 2: We then run mongen.ipynb, a Sagemath file, which calculates all resulting polynomials by removing the parentheses from 'draft2.txt'. It will then create a text file, called 'IOquestion12.all_monomials.txt', which contains two columns of data. The first column contains all the possible monomials from 'draft1.txt' and the second column contains the corresponding polynomials after one round.

Step 3: When the new text file is created, we run ax64.exe as follows:

```
ax64.exe 41012 "IOmonomials.temp.txt" "IOquestion12.all_monomials.txt"
```

to create a text file, called "IOmonomials.temp.txt", which contains the XOR of the two columns.

Step 4: As soon as "IOmonomials.temp.txt" is created, we run replacebooleanfunction.ipynb, a Sagemath file, which will substitute F,L and Z, do the calculations for each polynomial and write the result in the text file called 'IOmonomials.temp2.txt'.

Step 5: Finally, we use ax64.exe again by executing the command below:

```
ax64.exe 41013 "IOmonomials.temp.rewritten.txt" "IOmonomials.temp2.txt"
```

A text file, called 'Kernel_abcd.txt' will be created, which will contain all the invariances for the specific long term key.