Example: Ideal 2-bit Block Cipher

With 2-bits, there are 4 possible plaintext inputs and 24 different possible permutations of ciphertext output (i.e. 24 possible keys).

P	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12
00	00	00	00	00	00	00	01	01	10	10	11	11
01	01	01	10	10	11	11	00	00	00	00	00	00
10	10	11	01	11	01	10	10	11	01	11	01	10
11	11	10	11	01	10	01	11	10	11	01	10	01

P	K13	K14	K15	K16	K17	K18	K19	K20	K21	K22	K23	K24
00	01	01	10	10	11	11	01	01	10	10	11	11
01	10	11	01	11	01	10	10	11	01	11	01	10
10	00	00	00	00	00	00	11	10	11	01	10	01
11	11	10	11	01	10	01	00	00	00	00	00	00

The arrangement above may differ (e.g. K1 could be 00,01,11,10; or K1 could be 00,10,01,11; or ...). Therefore for the Sender to tell the Receiver the mapping that is being used (i.e. the key), then the Sender must send that exact mapping to the Receiver. For example, if Sender choose to encrypt P using mapping to K8, then Sender must tell Receiver that K8 is:

01
00
11
10

In effect, the Sender must send those 8 bits to the Receiver. Then the Receiver will know how to perform the decryption, as follows:

C	P
01	00
00	01
11	10
10	11

That is, if the Receiver receivers ciphertext '11', then they will know to decrypt to '10'.

Generalising, with a n-bit block cipher, there are 2^n possible plaintexts and $2^n!$ possible mappings to ciphertexts, or $2^n!$ keys. A key must specify the precise mapping being used. One way to do this requires a key of $n.2^n$ bits in length. For large values of n (e.g. 64 bits), the key becomes too large. To overcome this limitation (and the limitation that with small values of n the cipher is easy to break), the Feistel structure was proposed. It allows smaller keys, but maintains security by applying multiple rounds.