Simulation of (7,4) Hamming Code using Text Data

In the next cell, we will define Hamming code D of order 3.

```
D = codes.HammingCode(GF(2),3)
D
```

In the next cell, we define the list of all strings used in this simulation and provide a code to convert strings to its binary format.

In [2]:

```
strings = ' adehimstuxyT90.'
dict1 = {}
for i in xrange(len(strings)):
    dict1[str(strings[i])] = '{0:04b}'.format(i)

key_dict1 = dict1.keys()
val_dict1 = dict1.values()

dict2 = {}
for j in xrange(len(strings)):
    dict2[str(val_dict1[j])] = key_dict1[j]
```

Binaries we obtained previously must be converted to a valid SageMath's vectors data type. In the next cell we define some helpful functions in transforming string to binaries and vectors. The last function is actually a function that help us process, encode, transmit, and decode our data element-wise. This is where the simulation actually happened.

In [3]:

```
def str2bin(string):
    return dict1[string]

def bin2vec(biner):
    return vector(GF(2),[int(k) for k in biner])

def vec2bin(vector):
    return ''.join(str(k) for k in vector)

def bin2str(biner):
    return dict2[biner]

def text2bin(t):
    return ''.join(str2bin(j) for j in t)

def entradec_str(string):
    result = ['','','','','']
    for s in string:
        v = bin2vec(str2bin(s))
        ve = D.encode(v, encoder_name="Systematic")
        ves = vec2bin(ve)
        vt = channel.transmit(ve)
        vts = vec2bin(vt)
        vd = D.decode_to_code(vt, decoder_name="Syndrome")
        vds = vec2bin(vd)
        vm = D.decode_to_message(vt, decoder_name="Syndrome")
        vms = vec2bin(vm)
        m = bin2str(vms)
        result[0] += ves
        result[1] += vts
        result[2] += vds
        result[3] += vms
        result[4] += m
    return result
```

In [4]:

```
text = "This is a dummy text"
for i in [0,1,2,3,4,5,6]:
    channel = channels.StaticErrorRateChannel(D.ambient_space(),i)
    print entradec_str(text)[4]
```

```
This is a dummy text
This is a dummy text
i T0hadi.0 s.0tu9xdT
 ehetteyiT0hsTm9uumy
.mmu0e.tymxy.dex xea
s0eTyt9y TieaTsadhyi
eyxt.xt.0.9muuh.sTis
```

In [ ]: