Solutions 6

Exercise 6.1.

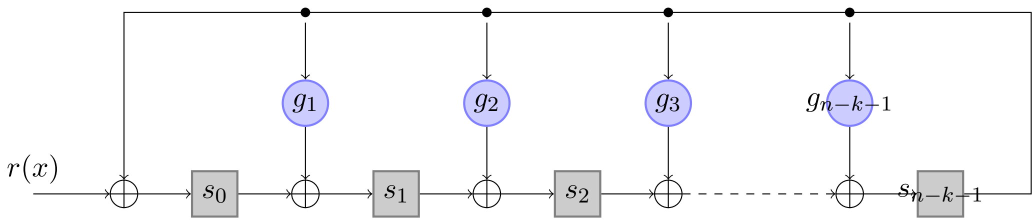
1. We first observe that $m(x) = x^{n-k}u(x) - b(x)$ is divisible by $g(x)$ so m belongs to $\mathcal{C}$. We see also that $x^{n-k}u(x)$ amounts to shift the bits of u to the rightmost positions in m, besides $b(x)$ has degree $< \deg g = n-k$. So the bits of b are zeros on the last k positions and do not interfer with $x^{n-k}u(x)$.

2. Let us denote $\beta^{(j)}(x)$ the content of the registers at step j. At step 1, the registers contain $\beta^{(1)}(x) = u_{k-1}g(x)$. Now to go to from a step $j-1$ to the step j, every bit is shifted to the right, which amounts to multiply $\beta(x)$ by x, and the last bit, corresponding to $\beta_{n-k-1}x^{n-k-1}$ give rise to an additional $\beta_{n-k-1}(g_0 + \cdots + g_{n-k-1}x^{n-k-1})$. In other words, the feedback loops amounts to reduce mod $x^{n-k} - (g_0 + \cdots + g_{n-k-1}x^{n-k-1})$. On the other hand, we input $u_{k-j}(g_0 + \cdots + g_{n-k-1}x^{n-k-1})$ at step j, so the register becomes $\beta^{(j)}(x) = x\beta^{(j-1)}(x) + u_{k-j}x^{n-k} \mod g(x)$. So by induction, $\beta^{(k)}(x) = x^{n-k}u(x) \mod g(x) = b(x)$ as claimed.

3. Let $v = (v_0, \ldots, v_{n-1})$ be a codeword. We know that the check equations of the code can be written in terms of the coefficients of h as $\sum_{i=0}^{k} h_i v_{n-i-j} = 0$ (see last exercise sheet). Thus, since $h_k = 1$,
$$\forall 1 \leq j \leq n-k, \quad v_{n-k-j} = \sum_{i=0}^{k-1} h_i v_{n-i-j}$$
which is exactly what the circuit computes.

4. Using $g(x) = x^3 + x + 1$ requires 3 registers and 2 xors. Using $h(x) = 1 + x + x^2 + x^4$ requires 4 registers and 2 xors. Notice on $\mathbb{F}_2$, multiplication by 0 mean that there is no connection, while multiplication by 1 mean that there is a connection.

5. We use a circuit that is analogous to the first one. This time, the entry is on the left.



Exercise 6.2.

1. We must have $\omega^{13} = 1$ and $\omega^r \neq 1$ for every $r < 13$. In particular, ω must be a root of $x^{13} - 1$ that lives in the smallest splitting field of this polynomial. Let $q := 3, n := 13$ and consider the smallest integer m such that n divides $q^m - 1$. For our choices, we will have $m = 3$. The degree of the smallest splitting field of $x^n - 1$ must be m (i.e., 3), as we need $x^n - 1$ divide $x^{q^m-1} - 1$ but not $x^{q^s-1} - 1$ for any $s < m$.

2. First we note that for every α, the minimal polynomial of α and α^3 are the same over $\mathbb{F}_3$. So the minimal polynomial of the elements on each of the following lines are the same:

$$\omega^0 = 1$$
$$\omega, \omega^3, \omega^9, \omega^{27} = \omega$$
$$\omega^2, \omega^6, \omega^{18} = \omega^5, \omega^{15} = \omega^2$$
$$\omega^4, \omega^{12}, \omega^{36} = \omega^{10}, \omega^{30} = \omega^4$$
$$\omega^7, \omega^{21} = \omega^8, \omega^{24} = \omega^{11}, \omega^{33} = \omega^7$$

So we only need to list g_0, g_1, g_2, g_4, g_7. Each one of these is the minimial polynomial of the powers of ω indicated below:

$$\begin{array}{ll} g_0: & 0 \\ g_1: & 1,3,9 \\ g_2: & 2,5,6 \\ g_4: & 4,10,12 \\ g_7: & 7,8,11 \end{array}$$

In particular the degrees of g_0, g_1, g_2, g_4, g_7 are $1, 3, 3, 3, 3$, respectively. As the dimension of the code needs to be 6, the generator polynomial of the code must pick two minimal polynomials of degree 3 and the one with degree 1. Moreover, as the distance of the code needs to be $2 * 2 + 1 = 5$, we can in particular pick g_0, g_1, g_2 so as to have $\omega^0, \omega^1, \omega^2, \omega^3$ as roots of the generator polynomial, and thus, achieve a distance of 5 by the BCH bound. Thus, letting $g(x)$ denote the generator polynomial, we will have $g(x) = g_0(x)g_1(x)g_2(x)$.

3. Let $E(x) := x^3 + a_2x^2 + a_1x + a_0$. If $E(x)$ is reducible then it must have a factor of degree 1, i.e., either x, $x-1$, or $x+1$. We want to eliminate these possibilities. We can ensure that x is not a factor by letting $a_0 \neq 0$. If $x - 1$ is a factor of $E(x)$, then we must have $E(1) = 0$, i.e., $1 + a_0 + a_1 + a_2 = 0$. Similarly, if $x + 1$ is a factor of $E(x)$, we must have $a_0 + a_2 = 1 + a_1$. We can ensure these conditions hold by letting $a_0 := 1, a_1 := -1, a_2 := 0$, and obtain $E(x) = x^3 - x + 1$, which is irreducible over $\mathbb{F}_3$.

4. The element α is a primitive element of $\mathbb{F}_{3^3}$, and thus it is a primitive 26th root of unity. As ω must be a primitive 13th root of unity, we can take $\omega := \alpha^2$. Now we take α as a root of the polynomial $E(x)$ above. The table below shows various powers of α, and confirms that α has order 26:

i	α^i
0	1
1	α
2	α^2
3	$\alpha - 1$
4	$\alpha^2 - \alpha$
5	$-\alpha^2 + \alpha - 1$
6	$\alpha^2 + \alpha + 1$
7	$\alpha^2 - \alpha - 1$
8	$-\alpha^2 - 1$
9	$\alpha + 1$
10	$\alpha^2 + \alpha$
11	$\alpha^2 + \alpha - 1$
12	$\alpha^2 - 1$
13	-1
$13 + j$	$-\alpha^j$

Thus, we can write the minimal polynomials as follows:

$$
\begin{aligned}
g_0 &= (x - \alpha^0) = x - 1 \\
g_1 &= (x - \alpha^2)(x - \alpha^6)(x - \alpha^{18}) = x^3 + x^2 + x - 1 \\
g_2 &= (x - \alpha^4)(x - \alpha^{12})(x - \alpha^{10}) = x^3 + x^2 - 1 \\
g_4 &= (x - \alpha^8)(x - \alpha^{24})(x - \alpha^{20}) = x^3 - x^2 - x - 1 \\
g_7 &= (x - \alpha^{14})(x - \alpha^{16})(x - \alpha^{22}) = x^3 - x - 1
\end{aligned}
$$

5. Using the previous parts, we can conclude that the code is generated by

$$g(x) = g_0(x)g_1(x)g_2(x) = (x-1)(x^3+x^2+x-1)(x^3+x^2-1) = x^7+x^6-x^3+x^2-x-1.$$

6. Let $y = (y_0, \ldots, y_{12})$ and $y(x) := \sum_i y_i x^i$ so we have $y(x) = -x + x^5$, and consider the *error-locating polynomial* $e(x) = a_1 x^{i_1} + a_2 x^{i_2}$ where i_1 and i_2 are the error positions and a_1 and a_2 are error values. Let $X := \omega^{i_1}$ and $Y := \omega^{i_2}$, so we want to know X and Y. We have that $y(x) = e(x)$ for $x = \omega^i, i = 0, 1, 2, 3, 5, 6, 9$. So

$$S_0 := y(\omega^0) = a_1 + a_2 = 0 \Rightarrow a_1 = -a_2.$$

$$S_1 := a_1(X - Y) = y(\omega) = \alpha^{10} - \alpha^2 = \alpha.$$

$$S_2 := a_1(X^2 - Y^2) = y(\omega^2) = \omega^{10} - \omega^2 = -\alpha^7 - \alpha^4 = \alpha^2 - \alpha + 1 = -\alpha^5.$$

Thus $X+Y = S_2/S_1 = -\alpha^4$. We may without loss of generality assume that $X-Y = \alpha$ (if $a_1 = -1$, this will only changed the order of X and Y). So,

$$X = ((X+Y) + (X-Y))/2 = \alpha^2 + \alpha = \alpha^{10} = \omega^5,$$

$$Y = ((X+Y) - (X-Y))/2 = -\alpha^4 - \alpha^{10} = \alpha^2 = \omega,$$

so we conclude that the errors are at positions 1 and 5. Now from $S_1 = a_1(X-Y) = \alpha$, we obtain that $a_1 = 1$, so the error value at the position corresponding to X (i.e., 5) is 1 and the error value at the other position is -1. We can use this to decode the received word to its nearest neighbor, i.e., $(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$.