Seungtack Baek

CE6305-501

Homework4

1. RNS for range []

a) Unrestricted moduli

We start with primes. RNS(41, 37, 31, 29, 23, 19, 17, 13, 11, 7, 5, 3, 2) gives 1.08 times what is needed. The number of bits required is 55 which yields really low efficiency, 0.8% efficiency.

We try to increase efficiency by combining and subtracting…

For this module set, we limit the maximum column length to be 6. So we can try to choose primes/combination of primes or power of primes as close to 64…

We do so by combining 2 and 31 (total column length = 2+5 = 7), which will include 62 (column length 6) while removing 2 and 31. We again, combine 5 and 11 (total column length = 2+5 = 7), which will result in 55 (column length 6). We again combine 3 and 29 for the same logic.

Now we have RNS(62, 57, 55, 41, 37, 29, 23, 17, 13, 7 ). Now we have 0.7 times what we need. So we can add the next closest prime 61, and 59 and remove 7, 13, 17. This will result in RNS(62, 61, 59, 57, 55, 41, 37, 29, 23) with total column length 52 and efficiency of 15.71%.

b) Low cost moduli

We first try to cover this range with the power s of 2 or one less than that. Following the table,

|  |  |  |  |
| --- | --- | --- | --- |
| RNS | Basis | Range | Efficiency |
| RNS(2^3, 2^3-1 , 2^2-1) | 3, 2 | 168 | 0.328125 |
| RNS(2^4, 2^4-1 , 2^3-1) | 4, 3 | 1680 | 0.41015625 |
| RNS(2^5, 2^5-1 , 2^3-1, 2^2-1) | 5, 3, 2 | 20832 | 0.317871094 |
| RNS(2^5, 2^5-1 , 2^4-1, 2^3-1) | 5 ,4, 3 | 104160 | 0.397338867 |
| RNS(2^6, 2^6-1 , 2^5-1) | 6, 5 | 124992 | 0.476806641 |
| RNS(2^7, 2^7-1 , 2^6-1, 2^5-1) | 7, 6, 5 | 31747968 | 0.473081589 |
| RNS(2^8, 2^8-1 , 2^7-1, 2^5-1, 2^3-1) | 8, 7, 5, 3 | 1.8E+09 | 0.418874323 |
| RNS(2^9, 2^9-1 , 2^8-1, 2^7-1, 2^5-1) | 9, 8, 7, 5 | 2.63E+11 | 0.477778525 |
| RNS(2^10, 2^10-1 , 2^9-1, 2^7-1) | 10, 9, 7 | 6.8E+10 | 0.494641297 |
| RNS(2^11, 2^11-1 , 2^10-1, 2^9-1, 2^7-1) | 11, 10, 9, 7 | 1.53E+13 | 0.494399772 |
| RNS(2^12, 2^12-1 , 2^11-1, 2^9-1, 2^7-1) | 12, 11, 9, 7 | 2.23E+15 | 0.494762236 |

RNS(). With basis 12, 11, 9 and 7 will do.

c) if we were to use ROM for the part b), it will need  entries and

they will hold results of widths 12,12,11,9,7, respectively. According to the table given in the notes, ROM delay with column 12 is **8D**.

2. Converting decimal number to RNS,

a) 247 to RNS (16, 15, 7):

247 % 16 = 7

247 % 15 = 7

247 % 7 = 2

* It is **(7, 7, 2)**

b) -118 to RNS (16, 15, 7):

-118 % 16 = 10

-118 % 15 = 2

-118 % 7 = 1

* It is **(10, 2, 1)**

3. Negation

- Negation is done by , so negating (8, 7, 6) in RNS(16, 15, 7),

16 – 8 = 8

15 – 7 = 8

7 – 6 = 1

* So it becomes **(8, 8, 1)**

4. Converting to decimal.

a) (12, 7, 4) in RNS(16, 15, 7) to decimal

Consider.

And, let, with.

Now, subtracting on both side, we have



Then, we can divide both sides by. However, we cannot simply divide in RNS system, so we will multiply RNS number by the multiplicative inverse of .

So, calculating the multiplicative inverse of 7 in RNS(16, 15, 7),

Finding such that

 which is 

So we have



From this we now let and subtract this on both sides, we have

.

Again, we divide both side by , which is equivalent to multiplying by the multiplicative inverse of 15 in RNS(16, 15, 7).

Finding such that

 which is 

So we have



From this we can say that 

So, (12, 7, 4) in RNS(16, 15, 7) is equivalent to (1, 9, 4) in MRS()=MRS(105,7,1).

Converting (1, 9, 4) in MRS(105,7,1),

Value =  = **172**

b) (1, 13, 6) in RNS(16, 15, 7) to decimal

Consider.

And, let, with.

Now, subtracting on both side, we have



Then, we can divide both sides by. However, we cannot simply divide in RNS system, so we will multiply RNS number by the multiplicative inverse of .

So, calculating the multiplicative inverse of 7 in RNS(16, 15, 7),

Finding such that

 which is 

So we have



From this we now let and subtract this on both sides, we have

.

Again, we divide both side by , which is equivalent to multiplying by the multiplicative inverse of 15 in RNS(16, 15, 7).

Finding such that

 which is 

So we have



From this we can say that 

So, (12, 7, 4) in RNS(16, 15, 7) is equivalent to (4,1,6) in MRS()=MRS(105,7,1).

Converting (4,1,6) in MRS(105,7,1),

Value =  = **433**