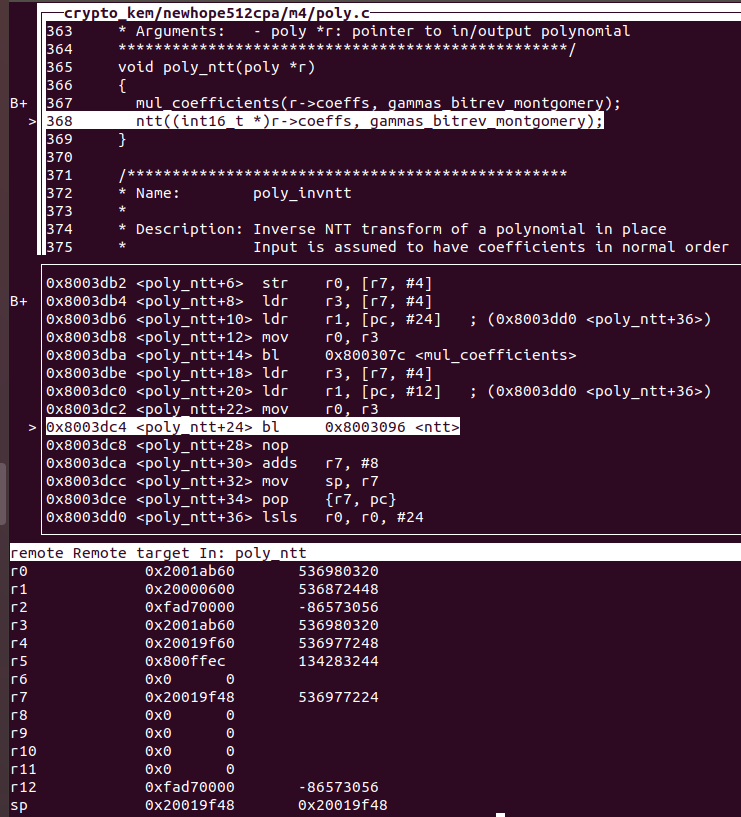
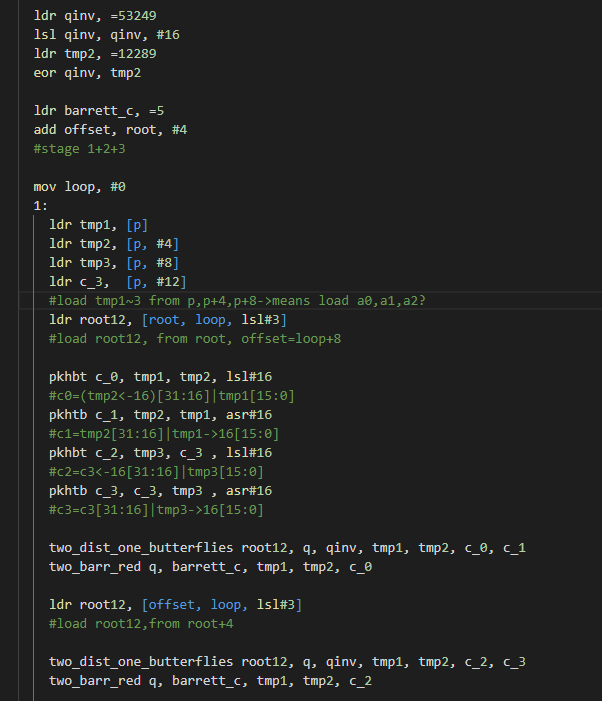
I have to say that debugging is really exhausting but interesting.



So, as for r1, which is 0x20000600, is initialized before the function: asm\_ntt512 by:

ldr r1, [pc, #12]

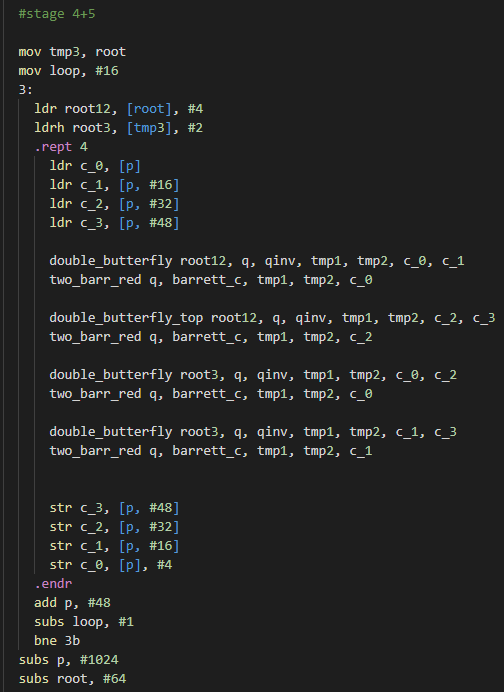
So you mean that if we inject the fault to skip this instruction, it is probably that we will get 0 when there is an instruction like: ldr r7, [r1].



I add some comment during debugging. So can you tell me that if I understand correctly?

The function is ntt (a,omega), so the address of a is in p and address of omega is in root?

If correctly, when the memory address in r1 lead to 0, the root12 will be always 0 too.



While, in “stage 4+5”, I think it doesn’t use “loop” to modify the “root12”. So I couldn’t figure out how it changes the Twiddle in these stage.

The other question is, in stage “1+2+3”, it only load [p],[p+4],[p+8],[p+12], and I guess it load a[0],a[1]…

But in “stage4+5”, it only load [p], [p+16], [p+32], [p+48], does this mean that load a[4], a[8], a[16]?

What’s about other coefficient like a[6]?