Let $R'(x)$ be the polynomial representation of the *magic* value of `0x1c2d19ed` (as in the paper). Straight division on both counts, on an augmented message (last 4 bytes are 0). k is the message length in bits, $n = \deg G(x) = 32$.

Prefixing: Prefixing the message with 32 1's.

$$\begin{aligned} M'(x) &= x^n M(x) + x^k x^n I(x) \\ &\equiv \underbrace{x^n M(x)}_{\deg = n+k-1} + \underbrace{x^k R'(x)}_{\deg = n+k-1} \pmod{G(x)}. \end{aligned}$$

That is, initializing the CRC register to all ones is **equivalent** to XOR-ing the 32 MSb of the message, $M(x)$, with the *magic* value, and then computing the remainder of that. I've verified it by software. I.e. the CRC = MAGIC XOR 32 MSb, and then start division.

Adding: Complementing the the first 32 MSb of the message.

$$\begin{aligned} M'(x) &= x^n M(x) + x^k I(x) \\ &= x^n M(x) + x^{-n} x^k x^n I(x) \\ &\equiv x^n M(x) + x^{-n} x^k R'(x) \pmod{G(x)} \\ &= x^n M(x) + x^{k-n} R'(x) \\ &= \underbrace{x^n M(x)}_{\deg = n+k-1} + \underbrace{\sum_{i=\max(0, n-k)}^{n-1} r'_i x^{k-n+i}}_{\deg = k-1}. \end{aligned}$$

Now, this means that complementing the 32 MSb of the message is **equivalent** to XOR-ing the second batch of 32 MSb[1] of the message with the *magic* value, and then computing the remainder of that. I.e. the CRC = 32 MSb, but the next 32 bits of the message have been XOR-ed with the magic value.

[1]Or as much as is available, if the message length is less than n ($n = 32$).