Euclid's Extended Algorithm gives us the multiplicative inverse of e, such that:

ed ≡ 1 (mod phi(n))

But we don’t use phi(n) to encrypt and decrypt. We use n. So we need to prove the same for n:

ed ≡ 1 (mod n) – does this hold true?

Because we have found n by multiplying two primes p and q, phi is found by

1. phi(p\*q) ≡ (p-1)(q-1)

So:

1. ed ≡ 1 (mod (p-1)(q-1))

All modular congruences can be expressed as the following equality:

1. a ≡ x mod r

becomes:

1. a = xk + r

So:

1. ed ≡ 1 (mod (p-1)(q-1))

becomes:

1. ed = (p-1)(q-1)h + 1

or

1. ed - 1 = h(p-1)(q-1)

To check that med ≡ m (mod pq), it is sufficient to check that med ≡ m (mod p) and med ≡ m (mod q) separately. So for p:

If med is a multiple of p, then m = 0. 0 multiplied by anything is 0, so 0ed ≡ 0 (mod p)

If med !≡ 0:

Pull an m out of the right hand side: med = m(ed - 1)m

Substitute ed-1 for h(p-1)(q-1) as per equation (7) above: m(ed - 1)m = mh(p-1)(q-1)m

mh(p-1)(q-1)m = (mp – 1)h(q-1)m

Substitute Fermat’s Little Theorem, which is: mp – 1 ≡ 1 (mod p)

(mp – 1)h(q - 1)m ≡ 1h(q - 1)m (mod p)

1 raised to anything is 1, so 1h(q - 1)m (mod p) ≡ m (mod p)

So med ≡ m (mod p)

Do this for q as well, and you prove med ≡ m (mod q)

If p and q are coprime (which they will be since we are using prime numbers), and a ≡ b (mod m) and a ≡ b (mod n) then a ≡ b (mod nm).

So if med ≡ m (mod p) and med ≡ m (mod q) then med ≡ m (mod pq)

pq = n, so:

med ≡ m (mod n)

So if n is made up of primes p and q, and apart from the special case where m is 0, med ≡ m (mod n) and med ≡ m (mod phi(n)), if Fermat’s Little Theorem holds. In the special case of m = 0, the relationship still holds because 0 multiplied by anything is 0.