5

a)

$$c = m^e mod\ n$$

$c^{dp} mod\ p \equiv (m^e mod\ n)^{dp}\ mod\ p$	$c^{dq} mod\ q \equiv (m^e mod\ n)^{dq}\ mod\ q$
$let\ x_p = \left(m^{edmod(p-1)} mod\ n\right) mod\ p$	$let\ x_q = \left(m^{edmod(q-1)} mod\ n\right) mod\ q$

If $(ed-1)\ mod\ \phi(n) = 0, ed\ mod\ \phi(n) = ed\ mod(p-1) = ed\ mod\ (q-1) = 1$

$x_p = m\ mod\ n\ mod\ p$ $= x_p\ mod\ p$	$x_q = m\ mod\ n\ mod\ q$ $= x_p\ mod\ q$

$$as\ n = pq\ and\ m < n$$

$x_p\ mod\ p = m\ mod\ n$ $= m$	$x_q\ mod\ q = m\ mod\ n$ $= m$

Since both $x_p\ mod\ p\ and\ x_q\ mod\ q = m.\ x \equiv m.$

Therefore, c is the correct encryption of M

b) The square-and- multiply algorithm has a time complexity of $O(l^3)$ bit operations, from l squarings and l modular multiplications
With the above procedure, one reduces the size of the squarings and multiplications to p and q<l, before combining. The time complexity becomes O($l * p^2 q^2$) which can be shorter than $O(l^3)$ assuming $p * q < l$

c) The adversary can compare between message m and incorrect decryption x' to determine which portion corresponds to $x^q mod\ q$. From there,

$$x_{correct} = x_q mod\ q = c^{dmod(q-1)} mod\ q$$
$$x_c^e = c^{edmod(q-1)mod\ q}$$

Since $ed\ mod(q-1) = 1$,

$$x_c^e = c\ mod\ q$$

He can now determine q and pa from the relation $n = pq$

d) use large capacitors as well in the machine to smooth any power spikes, with diodes to reduce the chance of reverse discharge. In fact, one should use a redundant power supply to prevent the machine from losing power halfway during calculations as well.

Discussed with Sean, Louis, Min Htet, Min Yue