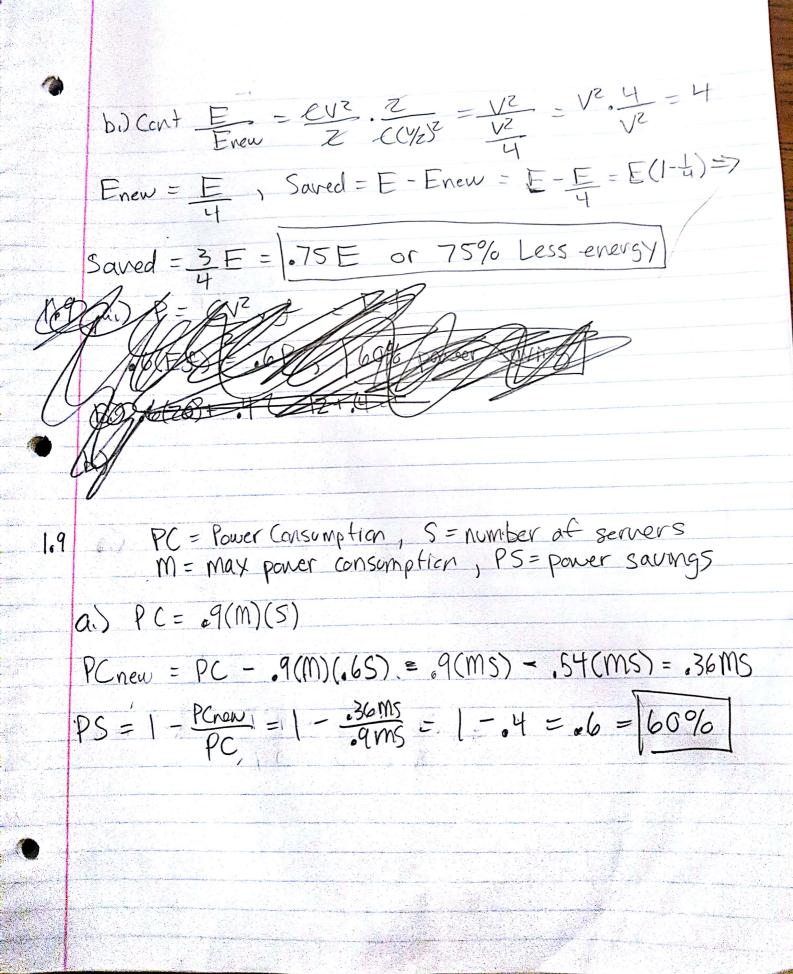
H1 Arch

- 1. a) False, increasing performace usually decreases evergy efficiency not the other way around
 - b.) True, a system needs all of its parts to function probably properly, so the failure point will always be the weakest link in the chain. This is why designing redundencies are good.
 - Ci) False, Amolahl's Law is very useful con for for finding latency and/or bottleneck points. It can also be used to show speedup in not only serial programs, but parallel ones as well.
 - d) False, Clock speed is only one aspect of performance. There are many other factors you must take into account, in order to minimize exec time
 - e.) True, you can have 1 million comes available but if the software is serial, it want make much of a difference
- Z. (1.8) a) $E = \frac{1}{2}$ (apacitive Load $\times V^2 = \frac{1}{2}CV^2$

E = Eexec + Eidle, since computation takes 1/2 the amount of time to run, the energ would be 50%

b.) With Energy, f dozesn't matter, so $E = \frac{CV^2}{2}$ $F_{noi} = \frac{C(K)^2}{2} - \frac{1}{2}$ cont on next page



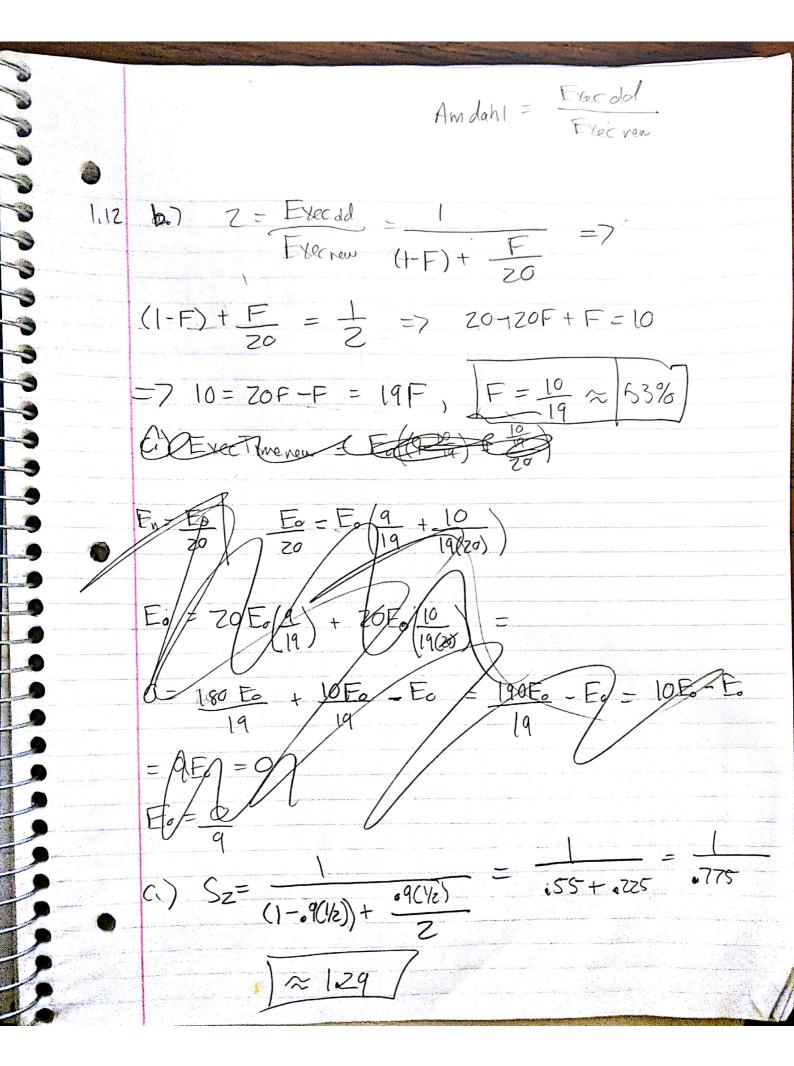
1.9 PC = Power Consumption,
$$M = max pc$$
, $S = \# servers$, $PS = Power SavmsS$

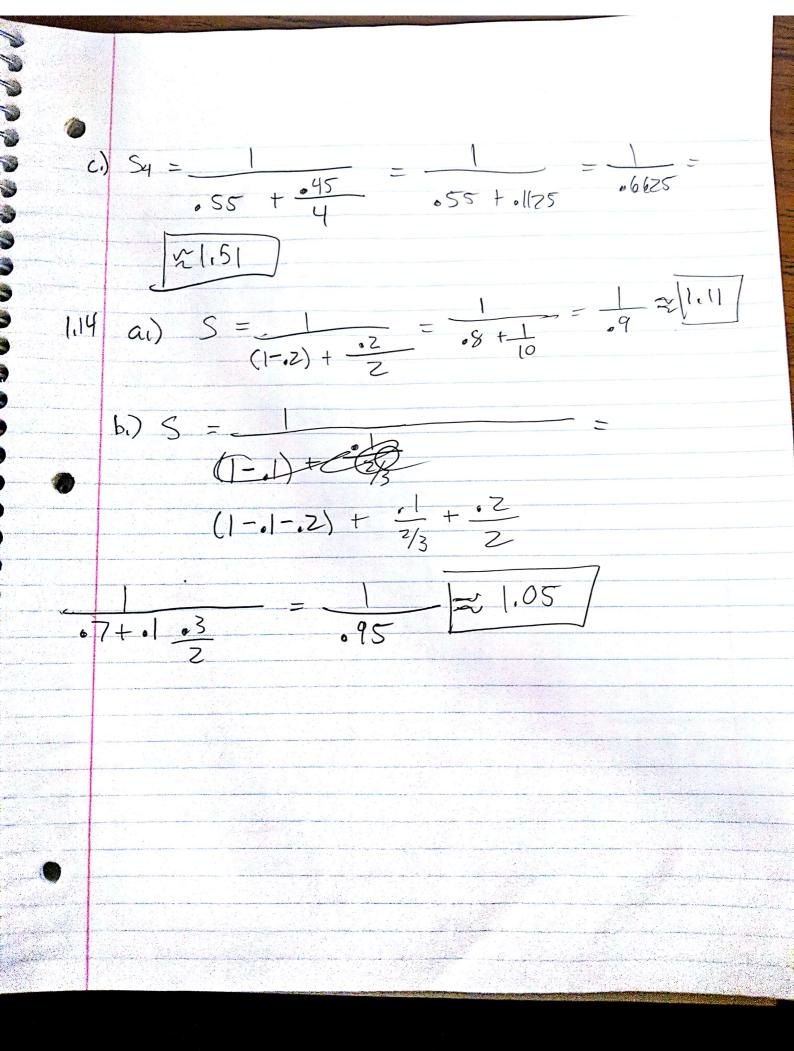
and $PC = .9MS$, $PC = .9(m)(1-.6)S = .36 MS$

$$PS = 1 - \frac{PC_{New}}{PC} = 1 - \frac{.36}{.9} = 1 - .4 = .6 = 60\%$$

P(rew =
$$((1-.2)V)^2((1-.4)f) = ((.8V)^2(.6f) = .384 CV^2f = .384Ef$$

Z





S = Speedup from bi)

Co)
$$F0\% = \frac{2}{2}S = \frac{2}{2} \times 1.05 = 0.105 = 10.15\%$$

Cote $\% = \frac{1}{24}S = \frac{3}{2} \times 1.05 = 0.1575 = 15.75\%$

1.16 ai) $S = \frac{1}{100} =$

