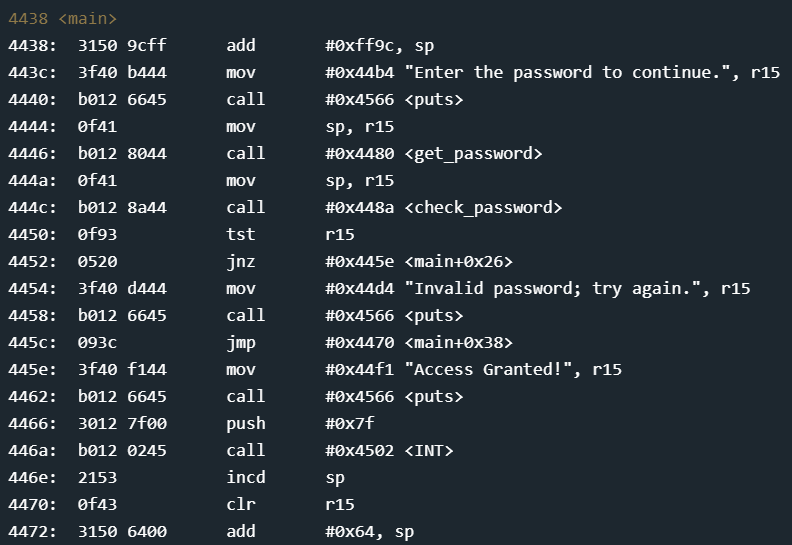
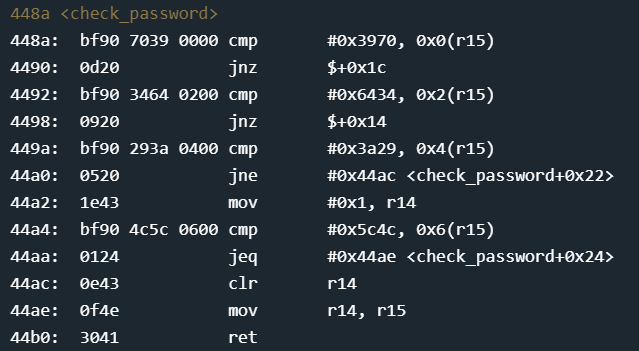
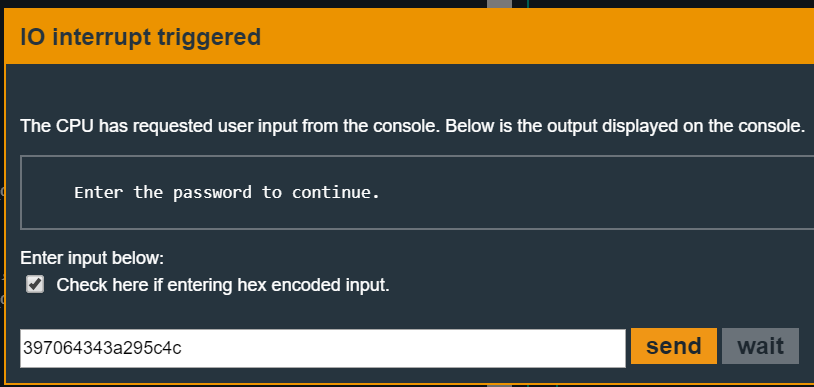
As always, we start out by toggling minimal mode and checking out <Main>:

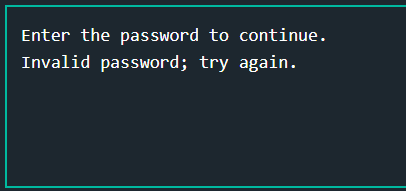


We can see here that we have a check\_password function. Let’s go check it out:

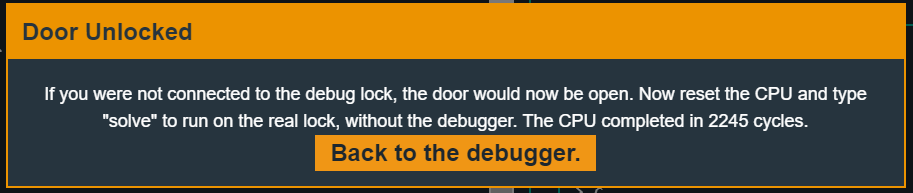


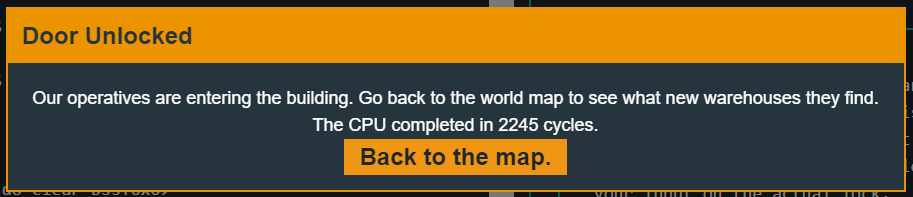
We can see that we are comparing different values to multiple locations of memory in sequence. We can make a reasonable guess that r15 is the location where the user given password is stored and that the values we are checking against are the password. Looking at the hex this gives us 0x397064343a295c4c which, when converted to ascii, is [ 9pd4:)\L ] without the []. Let’s try entering this in and see where it gets us:





Well then... what could have gone wrong? Looking at some documentation for the MSP430 we find that the MSP430 stores values in little endian format. This means that we need to switch the values around to match accordingly: 0x70393464293a4c5c or [ p94d):L\ ] again without the []. Let’s give it a try:





Overall, very straight forward using the basic techniques introduced in some of the challenges I have done.