Subtlety:

Alternatively I was thinking about what would happen with two pieces in the first row, and then if it would be able to tell which two those were, since this amounts to a bunch of circuits in parallel and then series, we lose information except for number of pieces in place. To get it to work for both scenarios we then need the two +/- as two test leads instead, and the TST connected to +5v for instance.

This will cause a the system to know where the short begins and ends, if we have a block of pieces. Hence we can consider the system by potentially switching between these types of tests, or using them in tandem with external circuitry. By switching between the two measuring types we can obtain both the size and position of the blocks of pieces, we obtain them in total because each leads to a unique resistance.

for example: _ 1 _ _ 1 _ _ _

yields by the first measurement: two steps in the resistance or change thereof, which amounts to a non-unique placement by a single reading along x, but yields the total number of pieces and their combination uniquely, meaning on off on on off on on on or a cyclic permutation.

Then by the second measurement we get their positions by the two other out leads, which supply an infimum and supremum on their locations uniquely. If we had a key piece (i.e. the king) we could calibrate in both position and arrangement.

Hence we obtain two rules for location measurement…

1.) get arrangement and number and lose information of index or dis-placement (arrangement and size unique by earlier arguments).

2) get arrangement position by key piece, or assumption of circuit. (ends determined by +5v and 0), another calibration. with the key piece we may be able to use 1) alone...

Together these will determine whether any combination of pieces are present or not and as such determine a unique bit string for each configuration on the board uniquely.

The ramification is that we need to switch between sensing styles and have potentially at least 1 pot command per line if we can switch between reading methods, this makes for 8 pot commands.

Since it has 16 i/o we are down to 8 i/o for the controlling of the rest of its functionality.

Servos will each need 1 i/o (forward backwards) and the servo needs 1 i/o, so that is 3 more, and we are down to 5. reset/start/off toggle switches amount to two more, which yields 3 open. The sides may only need one i/o as well, for 2 more taken, leaving 1 free i/o port at the end. This can be used for communications with the computer, which uses all our i/o.

-Paris