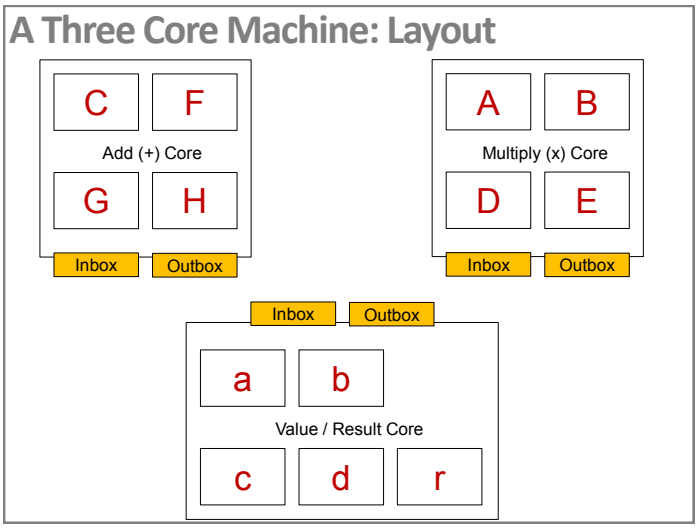
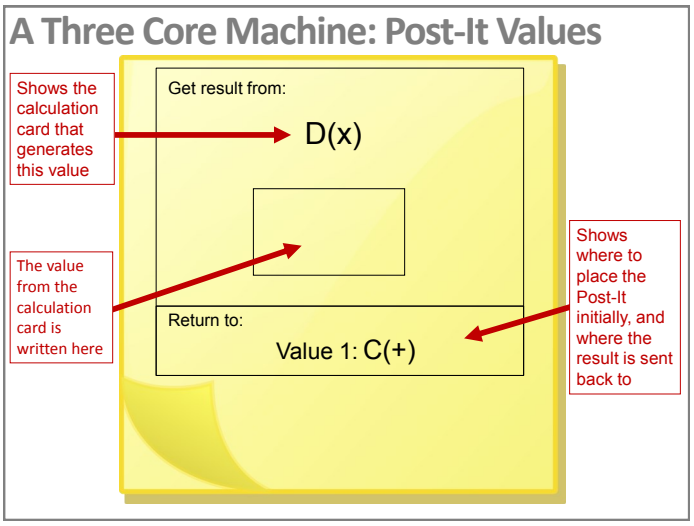
1 Problem, Many Cores

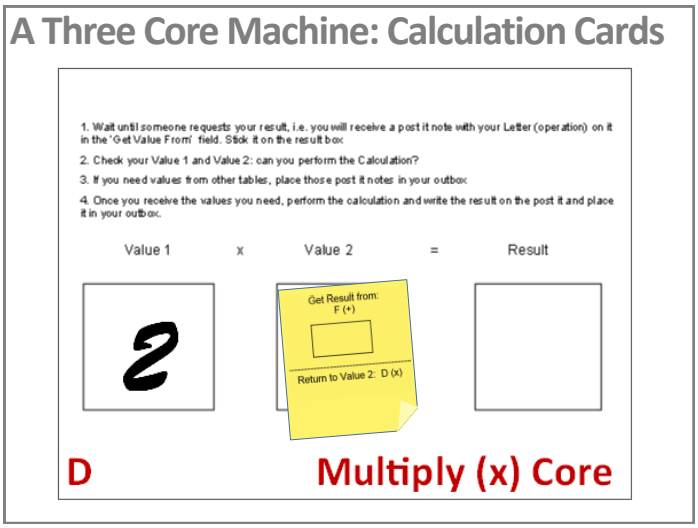
In essence, this activity involves pupils collaborating to calculate the result of a lengthy equation. You might ask students to calculate the result individually and then consider how much less mental work is involved for them distributing the task between machines. Working in teams of at least five, three children represent 3 specialist processors at 3 separate tables, handling values, addition or multiplication. The other two children act as the network, passing the values around the tables. Other numbers can be catered for by doubling up.

$$4xdx((a+1)x(b+2)+(2x(3+c)))$$



Setting up the desks is important. There are 3 tables which represent specialist processors. One handles values, one addition and the other multiplication. Each requires something to act as an input box and output box. On the tables are placed the Calculation Cards. There are 4 cards each for the x desk and the + desk. They are identified by the details at the top. Each has space for two values and a result. On some, a value may already be present. The other values are 'delivered' across the network on post-it notes. The details (an example is shown on the right) need to be written onto the post-it notes before the activity. The sheet included can be photocopied onto labels to make that task quicker. Each value card, or post-it says where to get the value from, has a space for the value to be written, and states where to return it to.



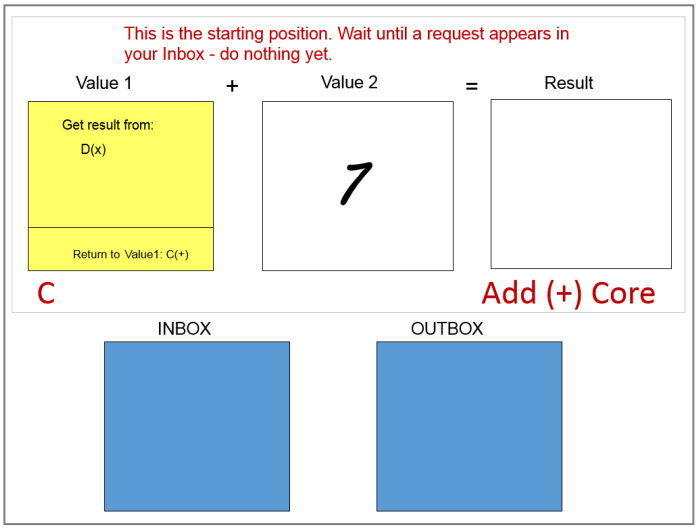


The calculation cards on the + and x desks are part complete. The post-its (without the value) are placed in the appropriate place on the cards. The pupils in charge of these WAIT until a post-it arrives in their inbox requesting a result from a card. To complete that result may require them requesting (putting a post-it in their outbox) a value from somewhere else. The pupils who act as the network carriers constantly scan outboxes and deliver any post-its to the relevant inbox.

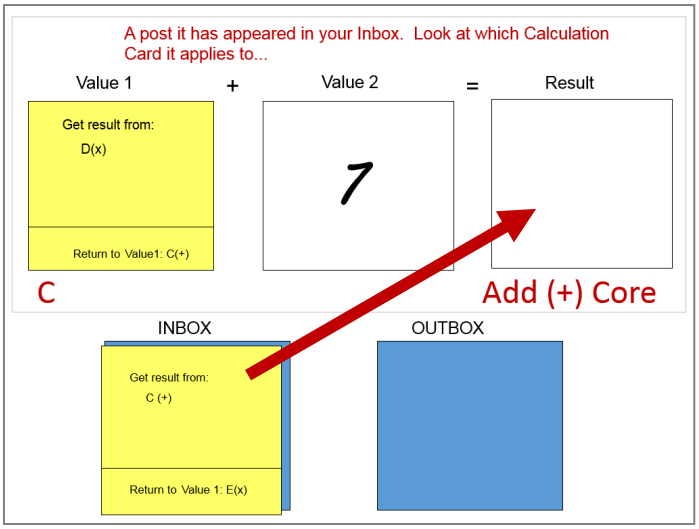
The Value / Result table starts the machine running. They have a set of value cards with space to accept a post-it. Two sets are provided so the activity can be run twice with different values, firstly a, b, c and d being 1, 2, 3 and 4 respectively. A second run would use 5, 6, 7 and 8. You could obviously create more and once the pupils are happy with the mechanism calculations can be done quickly.

The activity starts by the Value table putting the Result post-it into the outbox.....

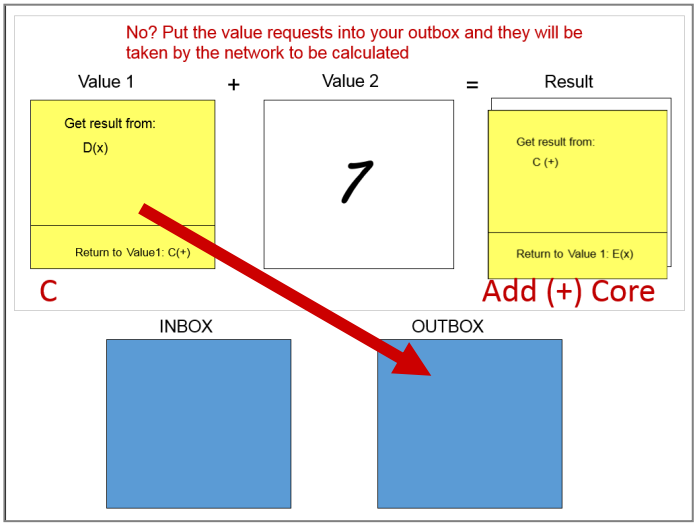
The following is a step by step example of a part of the process, looking at the Calculation Card C, on the (+) desk. The slide on the right shows the starting position. The pupil will wait until a request for the result from card C appears in their inbox.



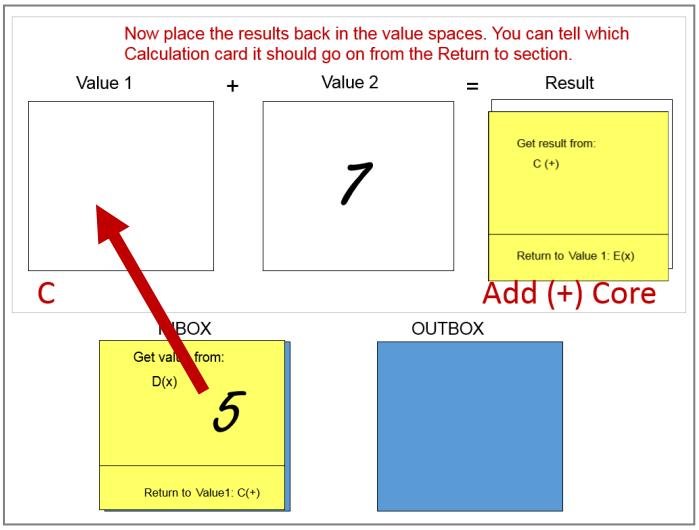
When a post-it appears on the (+) desk inbox. The pupil checks to see which Calculation Card it applies to. In this case, it is Card C, so the post-it is moved to its correct position, in the Result box (below).



The calculation cannot be completed because Value 1 isn't present, so that post-it is placed in the outbox, ready to be transported to the x table. Here it will be placed in the Result box of Card D. When Card D is able to calculate the result, the value is written in and it will be placed in their outbox. From there it will be returned by the network carriers to the + desk inbox (below). The network carriers know where to return it to because the post-it has a return address on it. The pupil in charge of the + desk can now place it in Value 1. They are now in a position to complete the calculation and the value is written onto the Result post-it.



That in turn is placed in the + desk outbox to await delivery to the return address indicated. The activity continues with values moving around until the result is finally returned to the Value desk.



Although this may seem a little complicated, a quick run through, following these prompts should make it clear. Once complete the students can reflect on the benefits and challenges posed by multi core technology.