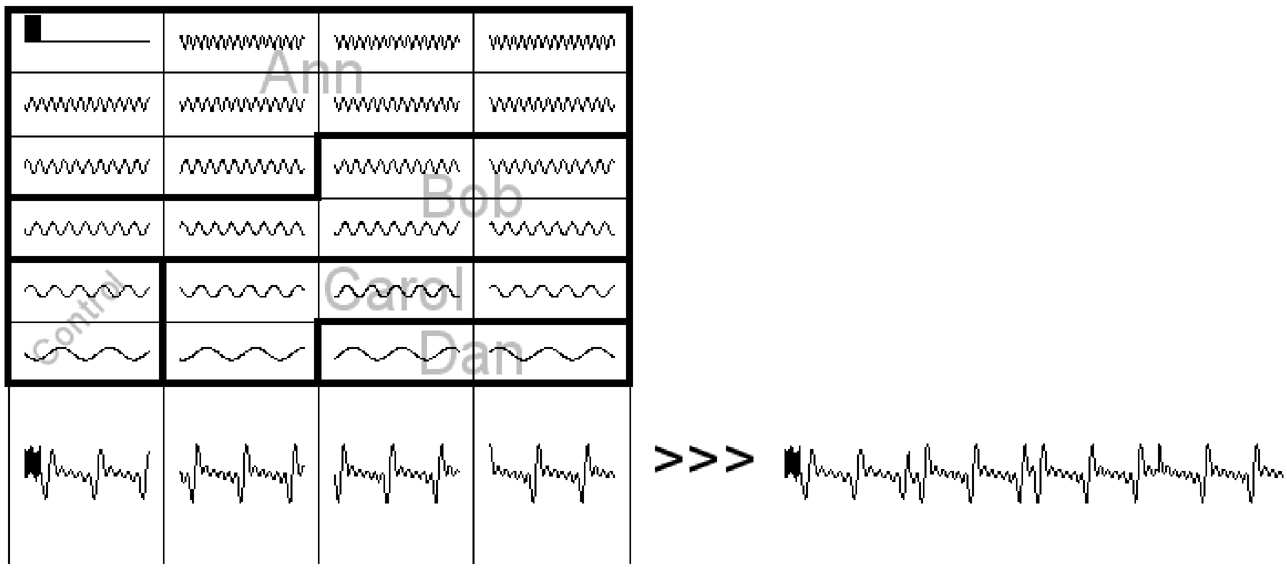
UNDERSTANDING THE LTE DOWNLINK : PART 2

Let's continue with our very simplified LTE downlink model, which involves starting with an empty table, assigning different areas of the table to different users, filling in the areas with data intended for each user, and converting that data to a squiggle to be transmitted via radio to the user's cell phones. Each table produces 0.5 milliseconds of squiggle.

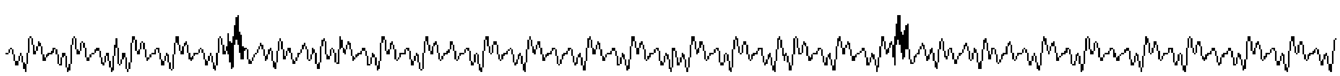
The cell phone reverses the process. If you were paying attention in part 1, you might have noticed a problem. The radio is transmitting a continuous squiggle, which the cell phone has to break up into 0.5 millisecond chunks.   Here's a sample squiggle.   Where do you draw the dividing lines for the chunks?

Well, there's nothing in this squiggle to tell the cell phone where the breaks should be. In theory we could use synchronized clocks – the cell tower and cell phone using absolute time points to agree as to where the breaks are, but in practice such a scheme would never work.   Keeping the two clocks in sync would just be too hard.

In LTE the solution is to put special marker squiggles in the transmitted stream. When we're building our table we steal a table entry away from data carrying duty and dedicate it to carrying a distinctive mark.   In my toy table, I'm going to use the upper left table entry to hold the marker. Our modified table (at the squiggle stage) looks like this:

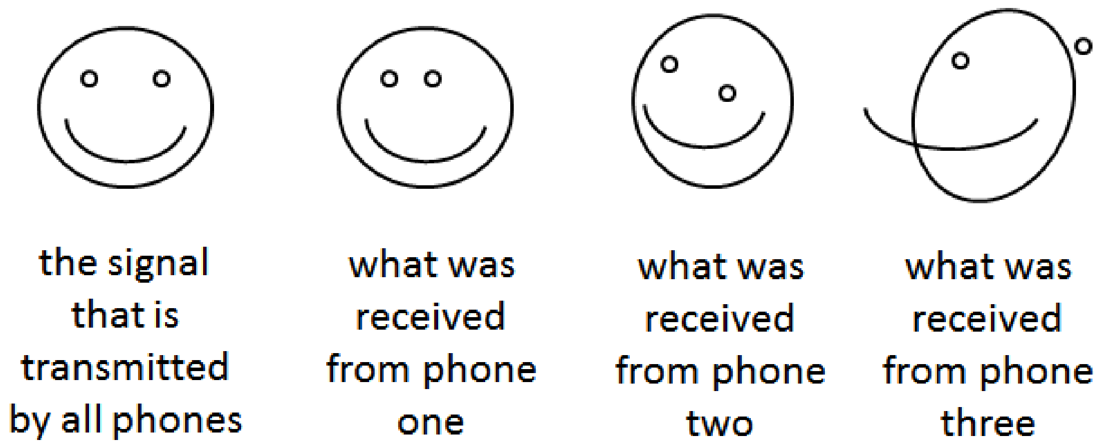


Note that the marker does not represent any combination of bits – no string of ones and zeroes. It's just a unique shape that is easy to spot. Here's what our radio transmission looks like now:



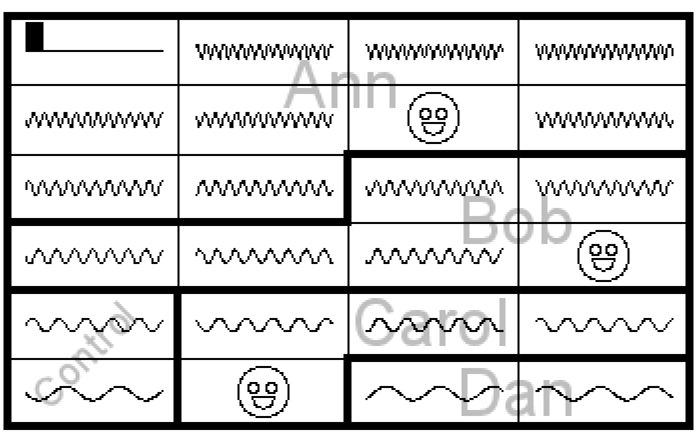
I'll wager you can now tell where the cuts should be.

Another point was made in part 1 – the transmitter has to decide how much data to place into each table entry. Should it put 2 bits, or 4 bits, or 6 bits? How in the world can it know what a reasonable choice is? The answer is reference squiggles. When the cell phone sends data to the cell tower it puts well known, predefined squiggles in certain predefined spots. The cell tower receives those squiggles and compares what it actually got to what it knows it was supposed to get. From the difference the cell tower can guestimate what radio conditions are like between the cell tower & the cell phone, and choose the number of bits for transmission accordingly. Take a look at the following:



The leftmost figure represents what was transmitted. If what the cell tower receives is like sample 1 then radio conditions are good, and it might decide to try for 6 bits per entry. If the received squiggle looks like sample 2 then conditions aren't so great – maybe put in 4 bits per entry. And if something like sample 3 is received then conditions are horrible and you better stick to 2 bits per entry – or maybe not bother trying to send any data at all to that user.

Now we've been talking about the cell phone sending reference squiggles to the cell tower, but of course the cell phone also wants to know what radio conditions are like, so to be nice the cell tower will also use up a few table entries for reference squiggles. Our modified table, with marker squiggle and reference squiggle now looks like this:



I'm using a smiley face to represent the reference squiggle. I'm not going to try to sum the columns, since I really don't know how to add in a smiley face to the mix!

Note 1: You'll note that the table now holds less data – the marker and reference squiggles are overhead, reducing the number of table entries available for user data. In a real LTE table the percentage is not negligible – it can be on the order of 15%. But it's a price that must be paid if we're going to use radio resources intelligently.

Note 2: In real LTE the marker and reference squiggles are most definitely not spikes and smiley faces! They are something called "Zadoff-Chou Sequences". Now a Zadoff-Chou sequence is something that is easy for an electronic circuit to recognize, but it wouldn't mean anything to you or me if we were to see one – hence my diagrammatic use of spikes and smiley faces.

Note 3: As you've no doubt intuited, in the table the columns correspond to time. What do the rows correspond to? Frequency. The upper rows correspond to a higher transmission frequency than the lower rows. The reference signals (smiley faces) are scattered throughout the table because radio conditions are constantly changing, not only with respect to time, but also with respect to frequency.