3GPP TSG RAN WG1 meeting #5 ***TSGR1#5(99)548***
June 1-4 1999, Cheju, Korea

Source: **Philips**

Title: **Modified RACH for TDD**

Agenda item:

Document for: **Discussion**

Summary

This document makes some proposals for increasing the RACH payload in TDD mode. Depending on comments received, an updated version may be submitted to the WG1 meeting #5.

Discussion

The current TDD specification includes the RACH bursts shown in Figures 1 and 2. Each occupies half a slot. The mid-amble duration is 512 chips to allow good channel estimation in worst case channels. Each data part contains 336 chips giving a total of 21 symbols with a spreading factor of 16. It has been suggested that up to 8 codes can be used to distinguish different access attempts.

For comparison the normal traffic bursts are shown in Figures 3 and 4.

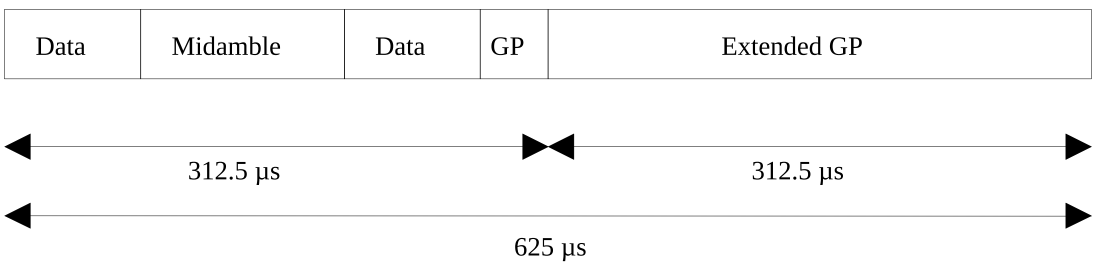

Figure 1: Access burst 1

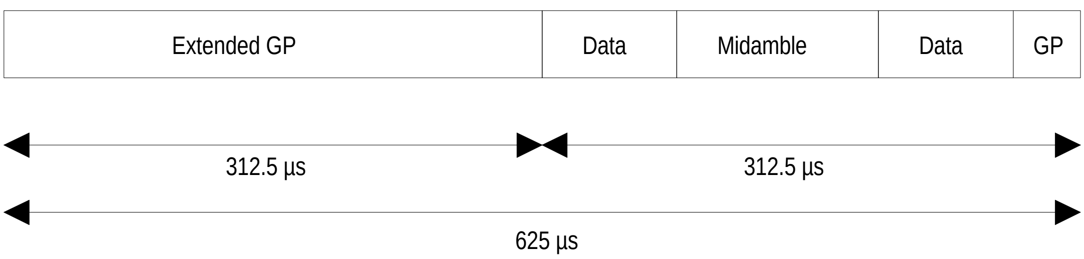

Figure 2: Access burst 2

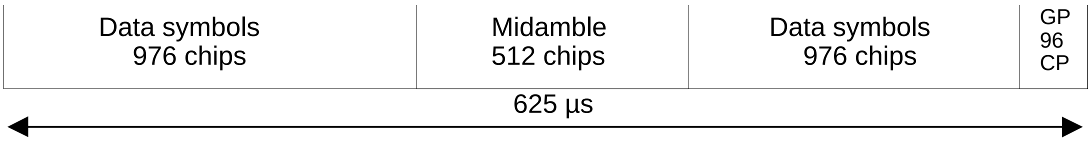

Figure 3: Burst structure of the burst type 1

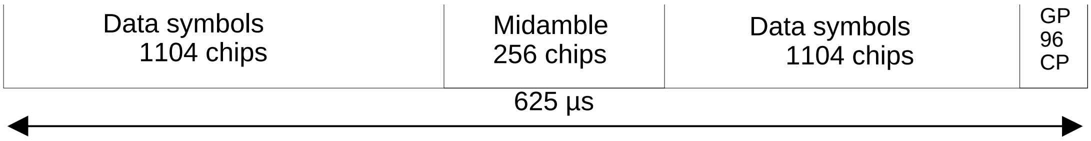

Figure 4: Burst structure of the burst type 2

It is recognized in WG1 and WG2 that the TDD RACH payload using these access bursts may be insufficient for initial access. However, some accesses might require a smaller payload (e.g. if the UE is already in connected mode). Some possibilities for increasing the payload are mentioned below:

1. Reducing the size of the mid-amble to 256. This would degrade performance in channels with large delay spread. Also the payload increase is probably not sufficient.
2. Reducing the spreading factor (e.g. from 16 to 8) for the data part. Assuming that with a spreading factor of 8 up to 4 codes could be detected, this would reduce the maximum number of simultaneous accesses to 4. It might still be allowed to assign up to 8 codes to avoid collisions. However, there would still be some reduction of RACH capacity.
3. Use a full data burst (i.e. burst type 1). This would halve the number of time/code slots, increasing collision probability and reducing RACH capacity.
4. Concatenate two RACH access attempts in successive half-slots. This would provide increased payload for initial access. If a smaller payload were sufficient only one RACH burst would be used.

Proposal

Option 4 seems most promising and is elaborated in more detail.

Depending on the size of the payload, either “single” or “double” RACH mode is selected. The single mode is substantially the same as for current proposals. The “double RACH” access would use both half slots in the RACH slot, as shown in Figure 5. The payload would be divided between the two bursts.

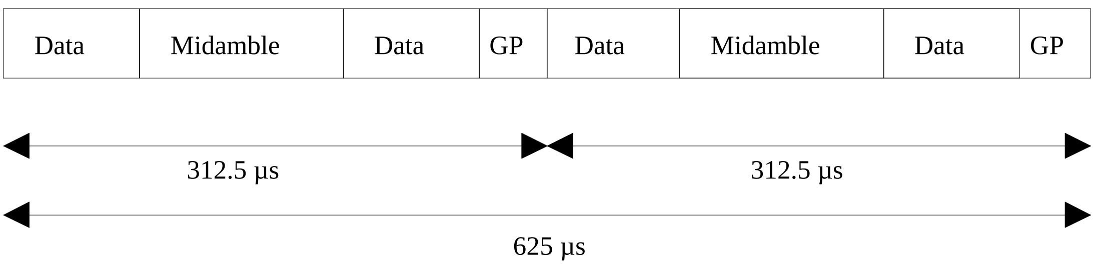

Figure 5: Double RACH burst structure

Some signaling bits in the data part would be needed to identify which RACH mode was currently being used. This could also be done using the same method as for TFCI (Transport Format Combination Indicator). It would also be necessary to correctly associate the RACH bursts between the two half slots. The pair of codes used could be indicated by signalling. Rather better, the same codes could be used in both half slots (which would minimize the number of signalling bits).

The acknowledgement of successful reception could possibly be separate for each half slot. In this case if either half-slot was received with errors then this would be separately indicated on the FACH. If only one was in error, only that one would need to be re-transmitted. However, it would probably be simpler to acknowledge the complete access attempt, where both half-slots are re-transmitted in the case of an error. This latter approach also appears to give a more consistent interface between protocol layers. It would also allow interleaving of the data between the half slots and only require one CRC.

The format of the second half slot could be different to the first one. For example, a CRC might be placed in only one of them. Any signaling information might be placed in the first half slot.

Clearly, if all access attempts required “double RACH” payload, the RACH capacity in access attempts per second would be reduced by about a factor of two compared with “single RACH”. However, much of this “lost” capacity is regained if a significant number of accesses require a lower payload and can use the

Conclusion

The “Double RACH” method proposed here should be considered for inclusion in the TDD specification. However, the usefulness of this scheme compared with using a full slot RACH burst will depend on the RACH traffic model (e.g. proportion of RACH attempts requiring large payload). More work may be needed on the details.