Embedded systems are becoming increasingly more complex as customers demand more functionality, more environmentally friendly and cheaper to operate systems. This has led to a need for more powerful processing and greater network connectivity including to the cloud. Avionics are no exception. This has led to two key challenges from a real-time perspective: multi-core, and the need for robustness and resilience. Multi-core means the ability to accurately predict the timing characteristics is difficult. Too much pessimism wastes resources. Optimism could lead to requirements not being met. At the same time developing all the system to the same highest standard is expensive. However, there is a recognition that some functional service can be dropped as long as the loss of service is bounded, and in some situations (e.g. fault handling) extra functionality is needed for a short time.

There are two distinctly different approaches to timing analysis: static and measurement-based. Modern processors have led to a general recognition that static analysis cannot provide an accurate Worst-Case Execution Times (WCET). Rolls-Royce and The University of York have developed a completely automated test-case generator (called TACO) that reliably delivers the WCET. The technique uses simulated annealing and structural coverage. The WCET is obtained with a software task running atomically in isolation. In practice other tasks will interfere, and with multi-core there is the additional contention caused by shared resources. University of York's Forecast-Based Interference (FBI) analysis systematically tests the system to first determine which shared resources lead to the greatest interference and then uses a deep-learning neural network to build an interference model. This work needs significant refinement to demonstrate the reliability of the model and the portability of the analysis.

Traditional approaches to scheduling strive to guarantee all tasks meet their deadlines all the time. More recently Mixed-Criticality Scheduling (MCS) has been researched. With MCS, when a high-criticality task exceeds its WCET, then it is granted more execution time and a mode change occurs. After the mode change, some lower-criticality tasks are not released or completed if they have already been released. More recently we have recognized that even the lower-criticality tasks need guaranteed service levels and that even though the *normal* WCET may provide a safe upper, the designer may not wish to include all situations, e.g. long execution times handling exceptional circumstances such as faults. Our work needs further refinement to understand what service level requirements are needed by the system and how statistical guarantees can be provided.

A final challenge is task allocation. Most work in this area deals with independent tasks assuming systems do not change, there is a single reliable WCET, and all tasks have to meet their deadlines. Significant research is needed to deal with functionally-dependent tasks, tasks with uncertainties over their WCET or with multiple WCETs, to support composability, to minimize interferences and overheads, and give lower-criticality tasks the best possible service. This paper will introduce the timing-related research needed to support systems engineering approaches, and changes needed to how systems are specified and modelled.