Synthetic and simplified view of the global simulation process, with lengths computations CPU::getNextTransaction Cmd::Prepare until masterNextTransaction (advance to next transaction-command) calls busMaster::accessGranted? update task's command with it calls Bus::schedule calls Bus::calcStartTimeLength reduces vlen Cmd::prepareNextTransaction delays start produce new transaction vlen: from TML source & progress all granted? rtime: end task's previous transaction yes update cmd's transaction with it if last master update len returns 0 (calcLength on all buses) returns nextTransaction Channel::testRead/testWrite may reduce vlen w.r.t Simulator initialisation channel's constraints and state prepare all tasks schedule all CPUs up to date transaction w.r.t. → Simulator loop application's constraints search next step transaction selects transaction CPU::schedule calls CPU::addTransaction? selects transaction last master? decides allocated timeslice yes / no ⊥ then calls execute (archive) increments master CPU::calcStartTimeLength (getNextMaster()) calls prepare reduces vlen, computes len returns yes returns no sets start and pnties no ves/ sets first _masterNextTransaction calls CPU::schedule (using getFirstMaster()) -specific to data channel transactions data flow —specific to other transactions call —common to all transactions ➤ control flow — result