```
HOST
  ^b0: // start basic block b0
    run_request() : req1          // call req1
  ^b1: // start basic block b1
    angle = recv_cmsg(0)          // receive message
  ^b2 { deadline: [b0 = 0.1*T2] }:
    m = run_routine(angle) : subrt1 // call subrt1
  ^b3:
    return_result(m)              // program result

ROUTINE subrt1                 // Local routine definition
    params: angle              // Argument (in @input[0])
    result: m                  // Result (in @output[0])
  NETQASM_START
    load C0 @input[0]   // load angle
    set Q0 0            // use qubit 0
    rot_y Q0 C0 4       // rotate qubit 0
    meas Q0 M0          // measure qubit 0
    store M0 @output[0] // return outcome
  NETQASM_END

REQUEST req1                 // Request routine definition
    remote_id: alice_id // Node to create EPR pair with
    epr_sck_id: 0       // EPR socket ID
    num_pairs: 1        // Number of pairs to create
    virt_ids: all 1     // Store EPR qubit as virt ID 1
    type: create_keep   // Keep EPR qubit in memory
```