```
✓ INITIAL
                                                                                                                                                                COMMS READY
            payload = new uint8 t[RPMSG MESSAGE SIZE];
                                                                                                                                       //two beeps a second(PRU 1 BLUE LED);
            resource table wrapper = new Resource Table Wrapper();
                                                                                                                                       // Initialize the RPMsg transport structure
            //one beep a second(PRU 1 BLUE LED);
                                                                                                                                       pru rpmsg init(&transport,
           // Clear the status of all interrupts
                                                                                                                                         &resource_table_wrapper -> to_resourceTable -> rpmsg_vring0,
            CT_INTC.SECR0 = 0xFFFFFFF;
                                                                                                                                         &resource table wrapper -> to resourceTable -> rpmsg vring1,
           CT INTC.SECR1 = 0xFFFFFFF;
                                                                                                                                          IRO PRU1 TO ARM.
                                                                                                     (*status & VIRTIO CONFIG S DRIVER OK)
            // Allow OCP master port access by the PRU so the PRU can read external memories
                                                                                                                                          IRQ ARM TO PRU1);
           CT CFG.SYSCFG bit.STANDBY INIT = 0;
                                                                                                                                       //two beeps a second(PRU 1 GREEN LED);
            // Clear the status of the PRU-ICSS system event that the ARM will use to 'kick' us
                                                                                                                                                                                                   On Entry
            //CT_INTC.SICR_bit.STATUS_CLR_INDEX = IRQ_ARM_TO_PRU0;
                                                                                                                                                                                                    On Exit
            // Make sure the Linux drivers are ready for RPMsg communication
            status = & resource_table_wrapper -> to_resourceTable -> rpmsg vdev.status;
                                                                                           On Entry
                                                                                                                                                                SEND_TO_ARM
                                                                                                                                           //Struct has been received, now it is turned into a
                                                                                                                                           string, and send via RPMsg
                                 pru_rpmsg_channel(RPMSG_NS_CREATE, &transport, RPMSG_CHAN_DRV, RX_CHAN_DESC, RPMSG_PORT) == PRU_RPMSG_SUCCESS
                                                                                                                                           get_struct ((char *) payload, dynamic_Buffer);
                                                                                                                                           five beeps a second(PRU 1 BLUE LED);
          CHANNEL_CREATED
                                            pru_rpmsg_send(&transport, RPMSG_PORT, RPMSG_PRU_PORT, payload, len) == PRU_RPMSG_SUCCESS
                                                                                                                                           len = 24;//strlen((char*) payload);
                                                                                                                                                                                               On Entry
             CT_INTC.SECRO_bit.ENA_STATUS_31_0 & (1 <<
             IRQ_XFR_PRU0_TO_PRU1)
                                                                                                                                                          RECEIVED
                                             CHECK_IRG
                                                                                                            //This state receives structs, then goes to "SEND TO ARM" state for RPMSG send back to Linux.
                                                                                                            //Should arrive here due to receiving IRQ: IRQ XFR PRU0 TO PRU1
CT INTC.SICR = IRQ XFR PRU0 TO PRU1;
                                           //Clear IRQ
R31 = IRO XFR PRU1 TO PRU0 + PRU IRO out strobe; // signal sent to PRU0 to immediately receive
                                                                                                            //From check IRQ state:
                                                                                                            CT_INTC.SICR = IRQ_XFR_PRU0_T0_PRU1;
                                                                                                                                                       //Clear IRQ
                                                                                                On Entry
                                                                                                            R31 = IRO XFR_PRU1_TO_PRU0 + PRU_IRO_out_strobe; // signal sent to PRU0 to immediately receive
                                                                                                            //end IRQ state
                                                                                                 On Exit
                                                                                                            //while (!(CT_INTC.SECR0_bit.ENA_STATUS_31_0 & (1 << IRQ_XFR_PRU0_T0_PRU1)));</pre>
                                                                                                            Comms PRU ucFSM local Buffer;
                                                                                                            CT INTC.SICR = IRQ XFR PRU0 TO PRU1;
                                                                                                                                                       //Clear IRO
                                          CT_INTC.SECRO_bit.ENA_STATUS_31_0 & (1 << IRQ_XFR_PRU0_TO_PRU1)
                                                                                                                                                                                                              On Entry
                                                                                                            // if ( CT_INTC.SECR0_bit.ENA_STATUS_31_0 & (1 << IRQ_XFR_PRU0_T0_PRU1)) {
                                                                                                                                                                                                              On Exit
```