```
404         /*
405          * I2C Slave RX States
406          */
407
408         case slaveAddressReceived: {
409             uart0_puts("Entered state 0x60");
410             mpI2CRegs->I2CONSET = 0x04;
411             clearSIFlag();
412             break;
413         }
414
415         case slaveDataReceived: {
416             uart0_puts("Entered state 0x80");
417             if (isFirst80) { //Register number is received
418                 isFirst80 = false;
419                 mTransaction.firstReg = mpI2CRegs->I2DAT;
420             }
421             else {
422 //                if ((mTransaction.firstReg - *mTransaction.pMasterData) +
    write_counter + 1 <= mTransaction.trxSize){
423 //                    *(mTransaction.pMasterData + mTransaction.firstReg +
    write_counter++) = mpI2CRegs->I2DAT;
424 //                }
425 //                else {
426 //                    uart0_puts("buffsploit prevented");
427 //                }
428                 *(mTransaction.pMasterData + mTransaction.firstReg +
    write_counter++) = mpI2CRegs->I2DAT;
429             }
430             clearSIFlag();
431             break;
432         }
433
434         case slaveStoporRptStartRecv: {
435             uart0_puts("Entered state 0xA0");
436             isFirst80 = true;
437             write_counter = 0;
438             mpI2CRegs->I2CONSET = 0x04;
439             clearSIFlag();
440             break;
441         }
442
443         /*
444          * I2C Slave TX States
445          */
446         case slaveDataSend: {
447             uart0_puts("Entered State 0xA8");
```

```
448             if (read_counter + 1 <= mTransaction.trxSize){
449                 mpI2CRegs->I2DAT = *(mTransaction.pMasterData +
    mTransaction.firstReg + read_counter++);
450             }
451             else {
452                 uart0_puts("Read too far");
453             }
454             mpI2CRegs->I2CONSET = 0x04;
455             clearSIFlag();
456             break;
457         }
458 
459         case dataAckedByMaster: {
460             uart0_puts("Entered State 0xB8");
461             if (read_counter + 1 <= mTransaction.trxSize){
462                 mpI2CRegs->I2DAT = *(mTransaction.pMasterData +
    mTransaction.firstReg + read_counter++);
463             }
464             mpI2CRegs->I2CONSET = 0x04;
465             clearSIFlag();
466             break;
467         }
468 
469         case masterNackRecv: {
470             uart0_puts("Entered State 0xC0");
471             read_counter = 0;
472             mpI2CRegs->I2CONSET = 0x04;
473             clearSIFlag();
474             break;
475         }
```