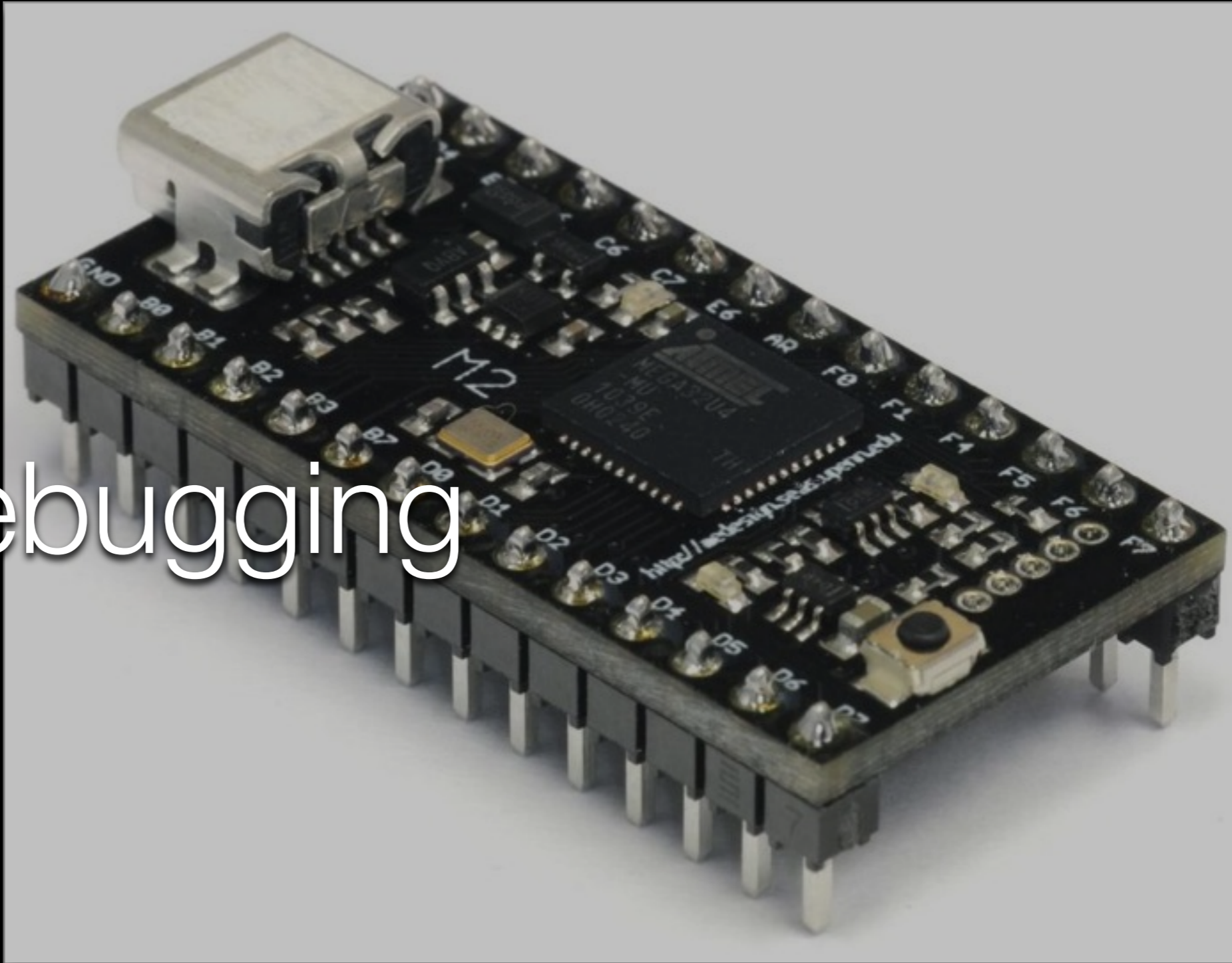


# USB Debugging



mUSB

SOFT  
www.soft

Team 104-2 Suture Final Status Memo-2.docx

data.pdf

data.rtf

eagle

photo-8.pdf

seas681:Documents adrianlievano\$ cd M2/

seas681:M2 adrianlievano\$ ls

Controls\_lect\_notes Striker\_main\_code.c libsaast-v7

Core\_code\_files U line\_follower

Lonebot Wiifi Test

MEAM\_Machine adc\_conversion

seas681:M2 adrianlievano\$ cd wiifi Test

-bash: cd: wiifi: No such file or directory

seas681:M2 adrianlievano\$ ls

Controls\_lect\_notes Striker\_main\_code.c libsaast-v7

Core\_code\_files U line\_follower

Lonebot Wiifi Test

MEAM\_Machine adc\_conversion

seas681:M2 adrianlievano\$ Wiifi\ Test/

-bash: Wiifi Test/: is a directory

seas681:M2 adrianlievano\$ LS

Controls\_lect\_notes Striker\_main\_code.c libsaast-v7

Core\_code\_files U line\_follower

Lonebot Wiifi Test

MEAM\_Machine adc\_conversion

seas681:M2 adrianlievano\$ cd Wiifi\ Test/

seas681:Wiifi Test adrianlievano\$ ls

mac\_instructions.txt ping.hex pong.hex

seas681:Wiifi Test adrianlievano\$ dfu-programmer atmega32u4 erase

seas681:Wiifi Test adrianlievano\$ dfu-programmer atmega32u4 flash pong.hex

Validating...

7734 bytes used (26.97%)

seas681:Wiifi Test adrianlievano\$ ls

mac\_instructions.txt ping.hex pong.hex

seas681:Wiifi Test adrianlievano\$ dfu-programmer atmega32u4 erase

seas681:Wiifi Test adrianlievano\$ dfu-programmer atmega32u4 flash ping.hex

Validating...

7734 bytes used (26.97%)

seas681:Wiifi Test adrianlievano\$ dfu-programmer atmega32u4 erase

seas681:Wiifi Test adrianlievano\$ dfu-programmer atmega32u4 flash pong.hex

Validating...

7734 bytes used (26.97%)

seas681:Wiifi Test adrianlievano\$ make clean

rm -f main.hex main.elf main.o

seas681:Wiifi Test adrianlievano\$ make

avr-gcc -Wall -Os -DF\_CPU=16000000 -mmcu=atmega32u4 -c main.c -o main.o

avr-gcc -Wall -Os -DF\_CPU=16000000 -mmcu=atmega32u4 -o main.elf main.o m\_usb.o m\_imu.o m\_wii.o m\_rf.o m\_bus.o -lm

rm -f main.hex

avr-objcopy -j .text -j .data -O ihex main.elf main.hex

seas681:Wiifi Test adrianlievano\$ ls

mUSB

Demo

# mUSB Files

- ✦ Download from the MEAM wiki
  - ✦ m\_usb.h & m\_usb.c

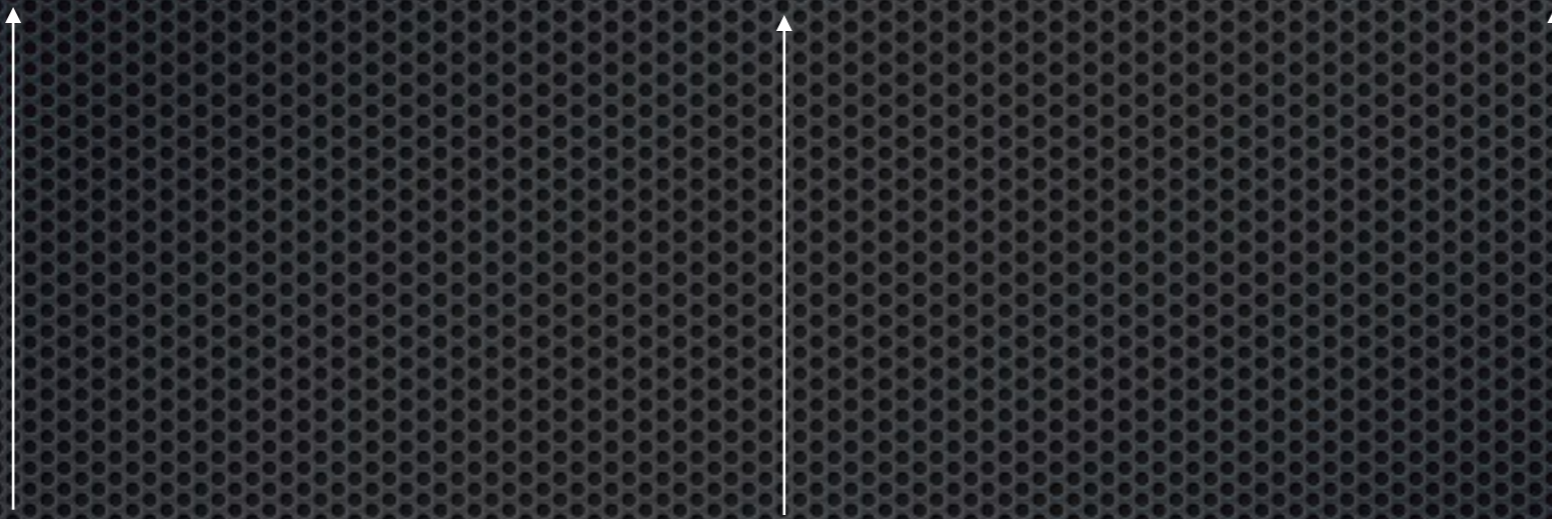


**#include in  
main.c**

**Add to your  
project**

# Process Flow

- Initialization —> receive data —> transmit data(via TX/RX)



Functions describing these on the wiki:  
MEAM410/510 USB Comm

## initialize

`m_usb_init()`

initialize the USB subsystem

`char m_usb_isconnected()`

confirm recognition by PC

## receive data

`char m_usb_rx_char()`

receive a character, -1 error

`unsigned char m_usb_rx_available()`

number of bytes in buffer

`m_usb_rx_flush()`

flush the receive buffer

## transmit data

`char m_usb_tx_char(char c)`

transmit an 8-bit char, -1 error

`m_usb_tx_hex(unsigned int i)`

transmit an int as hex

`m_usb_tx_int(int i)`

transmit an int as decimal

`m_usb_tx_uint(unsigned int i)`

transmit an uns. int as decimal

`m_usb_tx_long(long i)`

transmit a long as decimal

`m_usb_tx_string(char* s)`

transmit a char string

`m_usb_tx_flush()`

flush the transmit buffer

## Hints:

### **Find the serial object:**

```
#("ls/dev/tty.*")
```

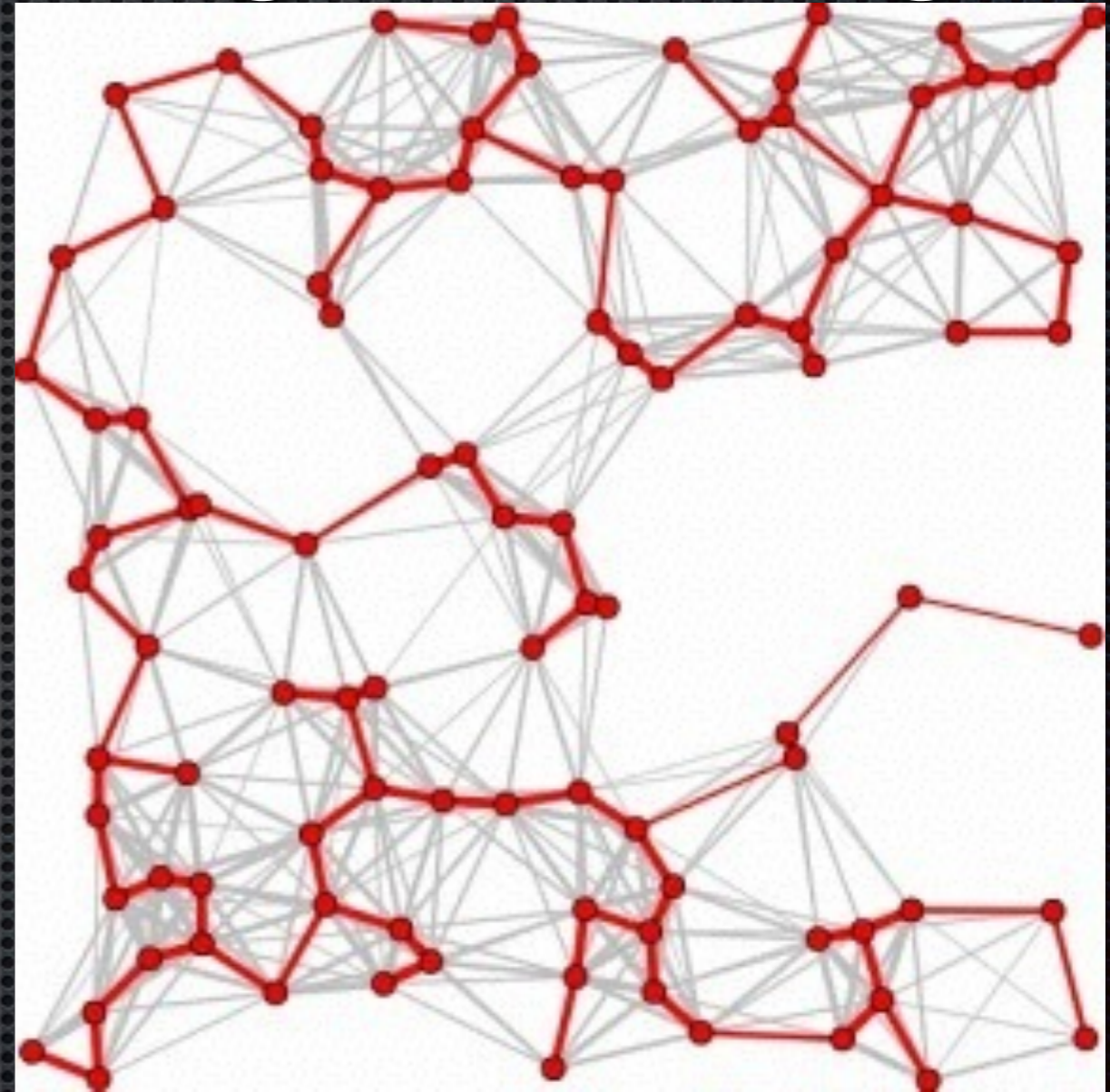
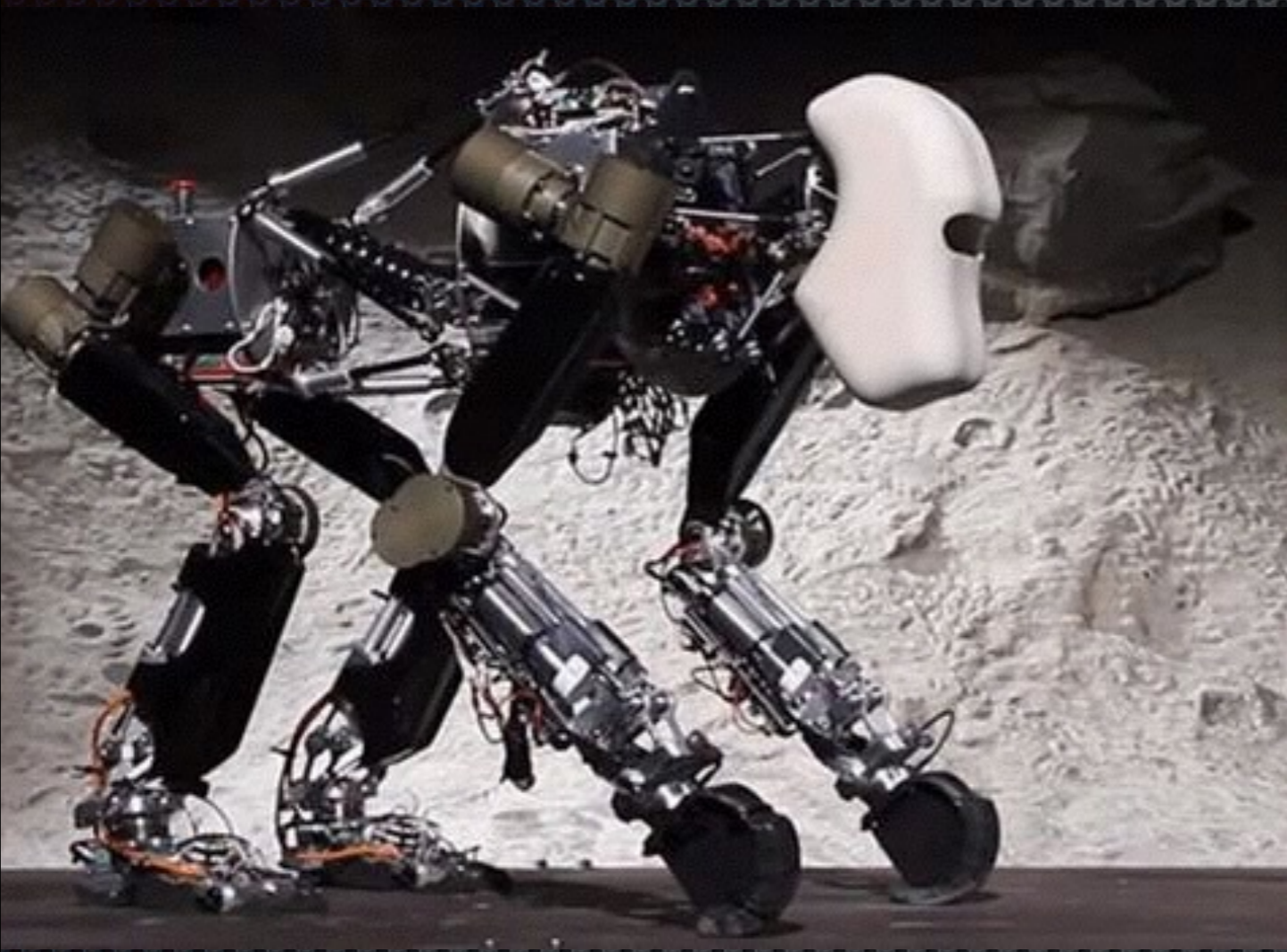
### **Start the session:**

```
screen /dev/tty.usbmodem### —> execute in  
terminal
```

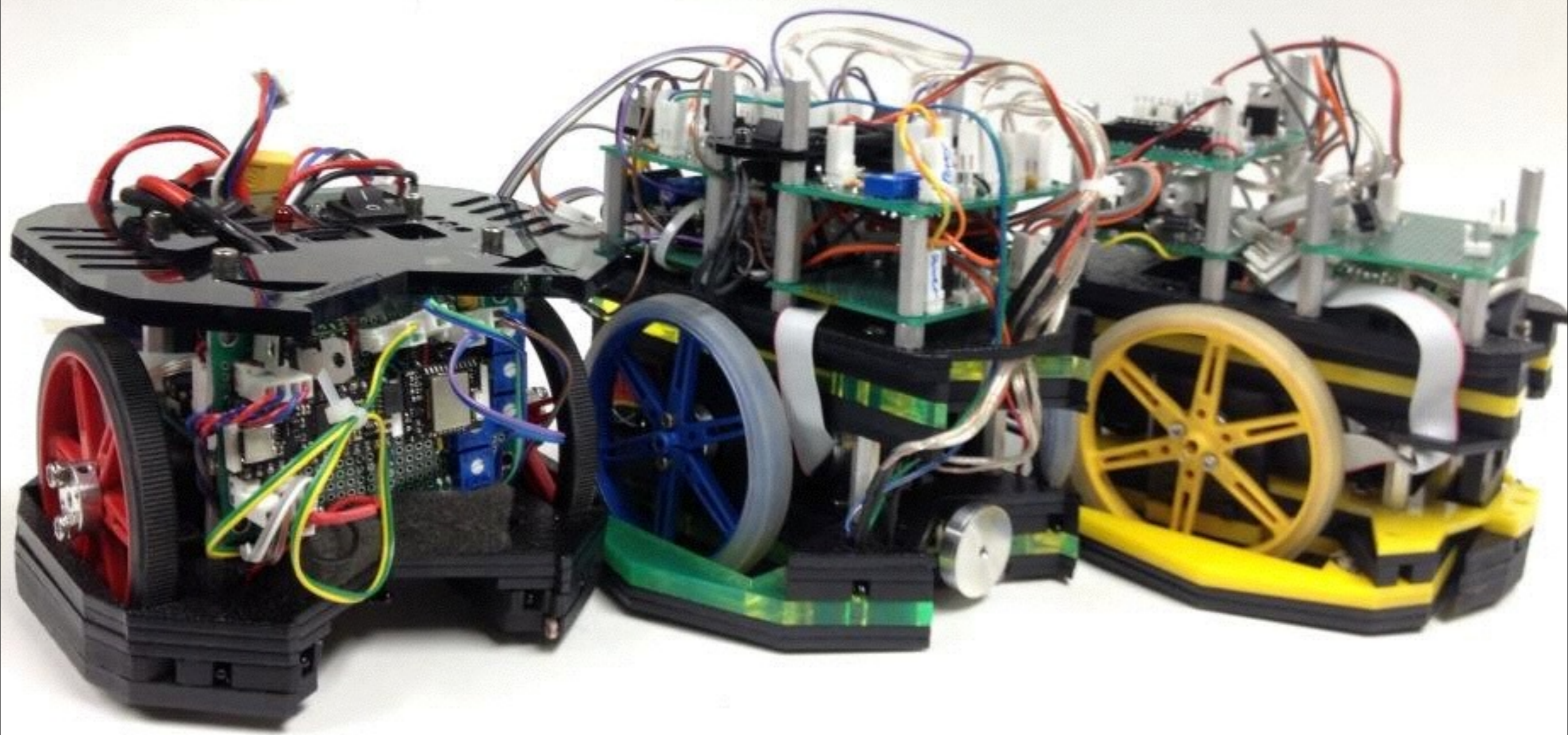
### **End the Session:**

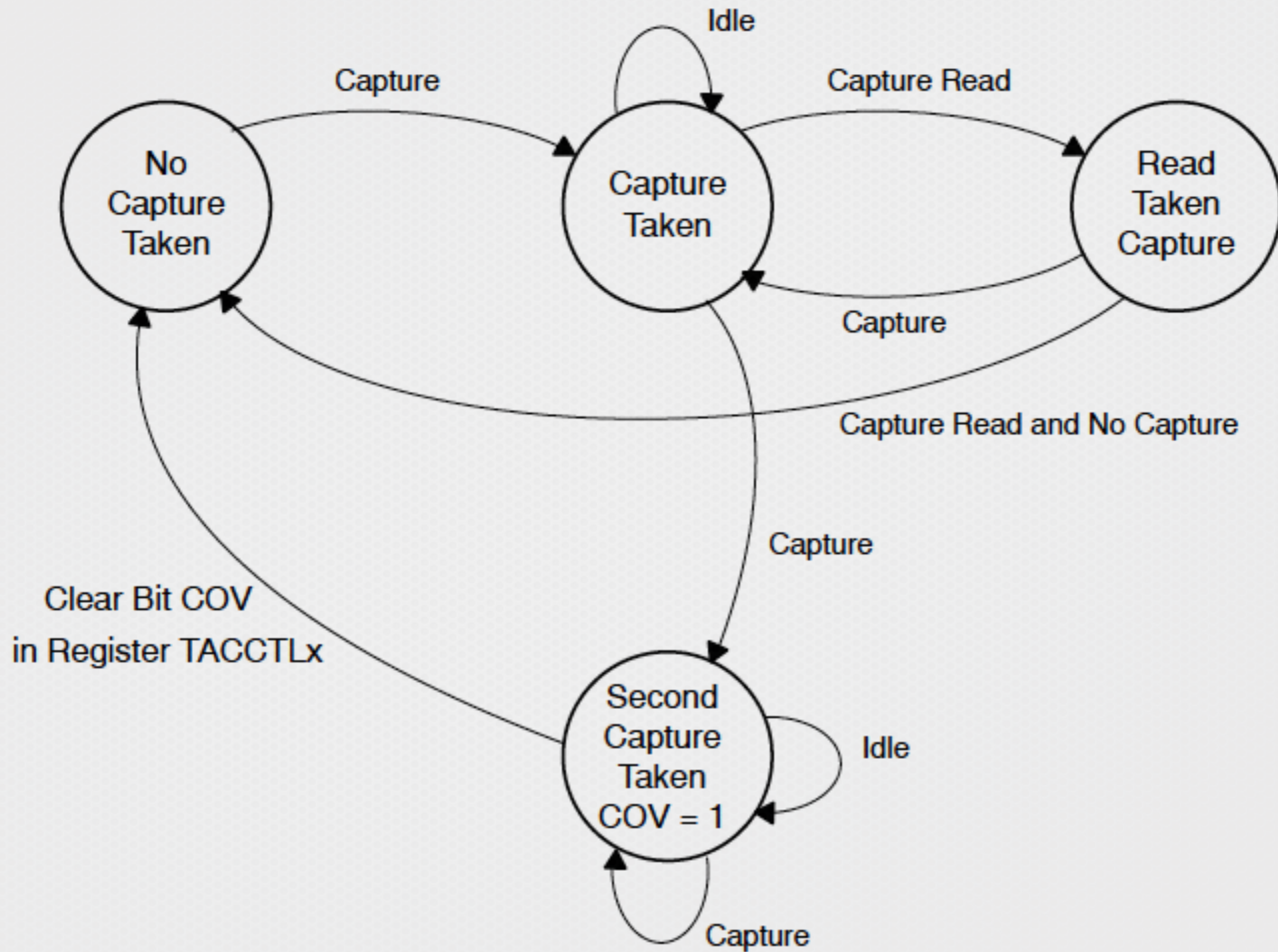
```
press Ctrl-A then Ctrl-\
```

# State Machine Programming











# C: switch/case syntax

