Embedded Systems

Lab 3: BTnut and Threads

13 & 20 April 2016

Matthias Meyer, Andreas Tretter, Stefan Lippuner, Sebastian Inderst
Goals of Lab 3

- Embedded operating systems
  - Motivation and structure
  - Threads
- Apply knowledge to practice
Embedded OS

- Large variety of requirements and environments for ES

- Modular OS
  - Reducing memory space
  - Timing is controllable
What we learned in Lab 2

- Basic use of timers and interrupts
- Timers are hardware dependent

Example Atmel ATmega128

// configure timer speed
TCCR3B |= 1<< CS30;

Example TI MSP430

// Set the timer A
TACTL = TASSEL_2;

- Rewrite code for each platform?
Device Drivers

- Device drivers handled by tasks
- Example: Timer in BTnut

```c
#include <sys/timer.h>

...  

hTimer = NutTimerStart(3000, _tm_callback, NULL, 0);
```
Threads

- A function that runs concurrently to other functions
- BTnut scheduler
  - Cooperative multithreading
  - Priority based
- Events for inter-thread communication
Threads – Nut/OS

- Create a thread

```c
THREAD(my_thread, arg) { // do something }
...
NutThreadCreate("My Thread", my_thread, 0, 192);
```

- Yielding threads

```c
NutThreadYield()
```

- Set priority

```c
THREAD(my_thread, arg) {
    NutThreadSetPriority(priority);
    // do something
}
```
Embedded Systems Lab 3

Have fun!