Embedded systems power consumption issues

Text: Chapter 3.7
STM32L100RC Technical Reference Manual, Chapter 5
CPU power consumption

- Most modern CPUs are designed with power consumption in mind to some degree.

- Power vs. energy:
  - heat depends on power consumption;
  - battery life depends on energy consumption.
CMOS power consumption

- **Voltage drops**: power consumption proportional to $V^2$.
- **Toggling**: more activity means more power.
- **Leakage**: basic circuit characteristics; can be eliminated by disconnecting power.
CPU power-saving strategies

- Reduce power supply voltage.
- Run at lower clock frequency.
- Disable function units with control signals when not in use.
- Disconnect parts from power supply when not in use.
C55x low power features

- Parallel execution units---longer idle shutdown times.
- Multiple data widths:
  - 16-bit ALU vs. 40-bit ALU.
- Instruction caches minimizes main memory accesses.
- Power management:
  - Function unit idle detection.
  - Memory idle detection.
  - User-configurable IDLE domains allow programmer control of what hardware is shut down.
Power management styles

- **Static power management**: does not depend on CPU activity.
  - Example: user-activated power-down mode.

- **Dynamic power management**: based on CPU activity.
  - Example: disabling off function units.
Application: PowerPC 603 energy features

- Provides doze, nap, sleep modes.
- **Dynamic power management features:**
  - Uses static logic.
  - Can shut down unused execution units.
  - Cache organized into subarrays to minimize amount of active circuitry.
## PowerPC 603 activity

- Percentage of time units are idle for SPEC integer/floating-point:

<table>
<thead>
<tr>
<th>unit</th>
<th>Specint92</th>
<th>Specfp92</th>
</tr>
</thead>
<tbody>
<tr>
<td>D cache</td>
<td>29%</td>
<td>28%</td>
</tr>
<tr>
<td>I cache</td>
<td>29%</td>
<td>17%</td>
</tr>
<tr>
<td>load/store</td>
<td>35%</td>
<td>17%</td>
</tr>
<tr>
<td>fixed-point</td>
<td>38%</td>
<td>76%</td>
</tr>
<tr>
<td>floating-point</td>
<td>99%</td>
<td>30%</td>
</tr>
<tr>
<td>system register</td>
<td>89%</td>
<td>97%</td>
</tr>
</tbody>
</table>
Power-down costs

- Going into a power-down mode costs:
  - time;
  - energy.
- Must determine if going into mode is worthwhile.
- Can model CPU power states with power state machine.
Application: StrongARM SA-1100 power saving

- Processor takes two supplies:
  - VDD is main 3.3V supply.
  - VDDX is 1.5V.

- Three power modes:
  - Run: normal operation.
  - Idle: stops CPU clock, with logic still powered.
  - Sleep: shuts off most of chip activity; 3 steps, each about 30 μs; wakeup takes > 10 ms.
SA-1100 power state machine

- $P_{\text{run}} = 400$ mW
- $P_{\text{idle}} = 50$ mW
- $P_{\text{sleep}} = 0.16$ mW
STM32L100RC Ultra-low-power MCU

- Designed for low-power applications
- **Low-power modes:**
  - 0.35\(\mu\)A standby mode (3 wakeup pins)
  - 1.3\(\mu\)A standby mode + real-time clock
  - 0.65\(\mu\)A stop mode (16 wakeup lines)
  - 1.5\(\mu\)A stop mode + real-time clock
  - 11\(\mu\)A low-power run mode
  - 238 mA/MHz run mode
  - 8 \(\mu\)s wakeup time
STM32L1xxx performance vs. $V_{DD}$ and $V_{CORE}$

Source: STM32L100RC Reference Manual
STM32L1xx low-power modes

- **Low power run mode:** regulator in low power mode, limited clock frequency, limited # peripherals running
- **Sleep mode:** Cortex-M3 core stopped, peripherals still running
- **Low power sleep mode:** Cortex-M3 core stopped, limited clock frequency, limited # peripherals running, regulator in low power mode, RAM in power down, flash stopped
- **Stop mode:** all clocks stopped, regulator in low power mode
- **Standby mode:** VCORE domain powered off

*Source: STM32L100RC Reference Manual*
## STM32L1xx Low Power Modes

<table>
<thead>
<tr>
<th>Mode name</th>
<th>Entry</th>
<th>Wakeup</th>
<th>Effect on (V_{\text{CORE}}) domain clocks</th>
<th>Effect on (V_{\text{DD}}) domain clocks</th>
<th>Voltage regulator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low power run</td>
<td>LPSDSR and LPRUN bits +</td>
<td>The regulator is</td>
<td>None</td>
<td>None</td>
<td>In low power mode</td>
</tr>
<tr>
<td></td>
<td>Clock setting</td>
<td>forced in Main</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>regulator (1.8 V)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sleep</td>
<td>WFI</td>
<td>Any interrupt</td>
<td>CPU CLK OFF</td>
<td>None</td>
<td>ON</td>
</tr>
<tr>
<td>(Sleep now or Sleep-on-exit)</td>
<td>WFE</td>
<td></td>
<td>no effect on other clocks or analog clock</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>sources</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Low power sleep</td>
<td>LPSDSR bits + WFI</td>
<td>Any interrupt</td>
<td>CPU CLK OFF</td>
<td>None</td>
<td>In low power mode</td>
</tr>
<tr>
<td>(Sleep now or Sleep-on-exit)</td>
<td></td>
<td></td>
<td>no effect on other clocks or analog clock</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>sources</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LPSDSR bits + WFE</td>
<td>Wakeup event</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stop</td>
<td>PDDS, LPSDSR bits + SLEEPDEEP bit + WFI or WFE</td>
<td>Any EXTI line (configured in the EXTI registers, internal and external lines)</td>
<td>All (V_{\text{CORE}}) domain clocks OFF</td>
<td>HSI and HSE and MSI oscillators OFF</td>
<td>ON, in low power mode (depending on PWR_CR)</td>
</tr>
<tr>
<td>Standby</td>
<td>PDDS bit + SLEEPDEEP bit + WFI or WFE</td>
<td>WKUP pin rising edge, RTC alarm (Alarm A or Alarm B), RTC Wakeup event, RTC tamper event, RTC timestamp event, external reset in NRST pin, IWDG reset</td>
<td></td>
<td></td>
<td>OFF</td>
</tr>
</tbody>
</table>

Source: STM32L100RC Reference Manual
Conserve power in run mode

- Reduce system clock frequencies
- Turn off clocks to unused peripherals
- Configure voltage regulator for low-power mode
Sleep mode

- Enter by executing WFI (wait for interrupt) or WFE (wait for event) instruction
  - Can select immediate entry to sleep mode, or entry upon completion of the lowest-priority interrupt handler
  - “Low power sleep mode” if voltage regulator configured for low power mode
- All I/O pins retain their functionality
- Exit sleep mode by any NVIC interrupt request or event
- Wakeup time shortest of the low-power modes
Stop mode

- Based on Cortex-M3 “deepsleep” mode, combined with peripheral clock gating
- All core clocks stopped and oscillators disabled
- Flash memory enters low power mode
- Can also configure other power-saving options
- Enter via WFI/WFE, but with SLEEPDEEP bit and other options set in system control register
- Exit by triggering any EXTI
- Longer recovery time than sleep mode
Standby mode

- Lowest power consumption
- Uses Cortex-3M “deepsleep” mode with voltage regulator disabled
- Core domain powered off, oscillators switched off, SRAM and register contents are lost (except RTC)
- Enter via WFI, with configuration options set in system control register
- Exit via WKUP pin or RTC alarm
- Longer recovery time