Wi-Fi Clock Radio

Kyle Stewart

Greg Bray

Introduction & Motivation

- Waking up is a painful experience and people want to alleviate pain
- People love music and love to have access to large collections (iPod)
- Market only has one viable wi-fi radio solution (price tag = $400)
  - Not sensible considering dedicated nature of alarm clock

Introduction & Motivation

- Client Functional Description
  - Touch screen graphical user interface
  - Linux OS
  - Embedded x86 processor
  - Wireless a/b that supports encryption
  - Use HTTP on TCP/IP for audio transport
  - Embedded streaming media player in Alarm Clock software

Introduction & Motivation

- Server Functional Description
  - Use Firefly media server (Open Source)
    - Streaming audio using HTTP on TCP/IP
    - on-the-fly transcoding between different audio formats (OGG, FLAC, MP3, AAC, and WMA)
    - Web-based configuration
    - Integrates with iTunes library and playlists
  - Customize code base to work with client software
  - Create web UI for configuring Alarm Clock

Introduction & Motivation

- Demonstration
  - Alarm clock successfully boots into OS and loads GUI
  - User can set time/date/alarm time using touchscreen interface
  - User can browse playlist on network machine and select playlist for alarm
  - Alarm “goes off” and plays a song from the user playlist stored on remote machine

Project Tasks

- Client
  - Obtain touch screen hardware
  - Install OS (Linux)
  - Load drivers (Wi-Fi, touch screen, Audio)
  - Configure for Wi-Fi network
  - Develop Client Software for Clock Radio
- Server
  - Setup standard media server (FireFly)
  - Customize code base to work with client software
- Integration Testing
- User Documentation

Specific Task Interfaces

- Client HW/SW interface (Linux, Drivers)
  - Touch screen input
  - WiFi networking
  - Audio output to speakers
- Client/Server interface (HTTP, Firefly API)
  - Use HTTP for audio transport
  - Web based configuration of media server
- Server/Music collection (Firefly, iTunes)
  - Songs (OGG, FLAC, MP3, AAC, or WMA format)
  - Playlists (M3U or iTunes format)

Testing & Integration

- Test each component individually.
- Get system operational early!
- Identify bugs, fix and repeat!
- KISS Principle: it is only an alarm clock!
- Use standard protocols and open source software.
  - Don’t reinvent the wheel!

Schedule & Milestones

- August
  - Collect all hardware components
  - Finish planning all interface requirements
- September
  - First Client/Server interoperability check
  - Lower level API implemented
- October
  - Connect GUI to lower level API
  - Begin testing

Risk Assessment

- EarthLCD Supply (low)
- Documentation support for TC-10 (high)
- Driver support for Linux (medium)
- Touch screen calibration (medium)
- Hard Drive (medium)
- Client/Server Integration (medium)
- Hardware/Software integration (low)

Bill of Materials

- Touch Screen/Embedded System
  - TC-10 from EarthLCD.com
  - $259
- Low profile wireless PCI card
  - Novatech NV-926W from LogicSupply.com
  - $28
- 1GB CompactFlash Card
  - From NewEgg.com
  - $15.99

Vendor List

- Primary
  - EarthLCD.com
  - LogicSupply.com
  - NewEgg.com
- Secondary
  - TigerDirect.com (Flash)
  - www.ncix.com (Wireless card)

Conclusion

An alarm clock that will let you choose what you want to hear when you first wake up in the morning will not only make sure that you get to school or work on time, but it will also make sure that you are in a better mood when you get there.