IOT GATEWAY WITH ALEXA

LEO GAO
APAC MARKET

MAR 2017
Agenda

• LS1012A Introduction

• Wireless Speaker Solution

• Cloud IoT Gateway Solution
LS1012A Block Diagram

- Single ARMv8 64-bit Cortex-A53 processor
  - 1840 DMIPS / 2240 Coremark @ 800MHz
  - NEON Co-processor and DP FPU
  - 256 KB L2 cache with ECC
- Memory Controller
  - DDR3L up to 1000 MHz
  - 16-bit data bus, 1 chip select
- High Speed Interconnect
  - 1x PCI Express Gen2
  - 1x SATA Gen3
  - 1x USB 3.0 w/PHY
  - 1x USB 2.0 w/ULPI
- Ethernet Packet Accelerator
  - 2x GbE (2.5G or 1G)
- Datapath
  - Packet Acceleration Engine (PPFE)
  - Security acceleration engine (SEC)
  - 2x SD 3.0/SDIO/eMMC
  - QSPI, 1x SPI, 2x UART, 2x I2C
  - 2x I2S, 5x SAI
- Secure Boot, Trust Architecture, ARM TrustZone
- Advanced Power Management
- Package: 9.6x9.6mm, routable in 4-layers
Performance starts with the core
- First 64-bit ARM Cortex-A53 core to be offered in a sub-10x10 mm package, delivering over 2,000 CoreMark® of performance at 1W (typical) for outstanding performance at exceptionally low power utilization
- Best in class 2.5 CoreMark / mW ratio

Broadest range of peripheral and I/O features in the sub-$10 ASP price range
- Only product in its class to offer Packet Acceleration for IP forwarding and NAS, delivering outstanding packet throughput for this power/package envelope
- Trust and Security acceleration enables root of trust and high performance encryption consistent with much higher cost microprocessors
- First in its class to offer 64-bit support for battery powered mobile applications and performance efficiency
- Only 1W 64-bit processor to combine USB 3.0 with integrated PHY, PCIe, 2.5 Gigabit Ethernet and SATA3 on a single SoC to enable lower system-level costs
- Enables low-cost, 4-layer board level designs together with high system level integration to support ultra-small form factor systems
Wireless Speaker Solution

- Based on ARMv8 64bit LS1012A
- Wireless speaker with connections:
  - WiFi
  - DLNA(UPnP)
  - Bluetooth
- Voice assistant
- Multi-speaker synchronization
- Support OpenWRT
- Support secure boot
- Target market: smart home
Wireless Speaker Reference Design
Wireless Speaker

Multi speakers based on DLNA(UPnP) Protocol
play music independently
Smart Voice Assistant + Wireless Speaker Ref Design
Voice Assistant

How’s the weather today?

Microphone Array

Voice Sample

Voice Awake

LS1012A

WIFI Data Transfer

Voice Synthesize

Speech Recognition Cloud Platform

Today is cloudy to sunny…
Multi Speakers Synchronous Playback

Master Role

LS1012A Based WIFI speaker

LS1012A Based WIFI speaker

LS1012A Based WIFI speaker
UPnP Player- QQ Music Player Screenshot
Cloud IoT Gateway Solution

• WIFI Gateway based on NXP LS1012A
• IoT Gateway with rich connections:
  • WiFi
  • Bluetooth
  • Zigbee/Thread
  • Zigbee
  • NFC

• Support IoT clouds:
  • IBM Bluemix(MQTT), Node-Red
  • Alibaba Aliyun(Alink)

• Support OpenWRT, Docker/LXC, Secure Boot
• Target Market: Smart Home, Industry 4.0
IOT Gateway by LS1012A

Wireless speaker

Smart Lock

Wi-Fi Security Cameras (up to 8)

IoT Gateway (LS1012A)

Rich IoT Ecosystem

Node-RED

IoT Gateway (LS1012A)
Smart Home

IoT devices of Smart home connected through wireless gateway
Smart Home NFC Commissioning

Devices authenticated through NFC can be added in wireless network
Smart Control

Devices connected with IoT clouds through MQTT/Alink protocol

1. Node1 -> node2 -> node3 -> node4
2. Node5 -> node3 -> node2 -> node1
3. Node1 -> node2 -> node3 -> node6
Smart Control

Thread/6LoWPAN
Zigbee

Figure 1. Thread Communication Stack
Industry 4.0 Application

- Big Data
- Autonomous Robot
- Simulation
- System Integration
- Internet of Things
- Additive Manufacturing
- Argument Reality
- CRM
- HMI
- Sensor
- Monitor
- Instrument
Cloud-based IoT Fog Computing Platform

- IBM IoT SDK
- Alibaba IoT SDK
- Google IoT SDK
- Azure IoT SDK
- Private IoT SDK
- AWS IoT SDK
- Greengrass

- Container Engine
- Protocol Adaptor
- Data processing
- Data filter

- Xenomai
- IEEE 1588
- TSN
- OP-TEE
- SEC

- Bluetooth
- Thread
- ZigBee
- ...
Fog Computing Service Orchestration

Cloud

Swarm/Kubernetes

Edge Node Master

Edge Node 1

Edge Node 2

Edge Node N
Cloud-based Development and Deployment

- Git Repos: Store source code
- Jenkins: Compile code
- Image builder: Build Docker image
- Image Verification: Verify Docker images
- Image Repo: Store Docker images

Push Docker image to Fog computing
Business Benefits

- Lead the decentralized datacenter trend
- Extend Cloud computing from data center to IoT fog computing platform to leverage silicon high performance capability
- Computing close to data to support real-time, actionable analytics and decisions
- Filter data and minimize the cost of transmitting IoT data to the cloud
- Operate with intermittent connections to cloud
- Containerize to support different cloud providers
- Containerize computing functionality for easy cloud-based deployment
- Cloud-based code building, packing, testing, deployment and monitor
Cloud-based IoT **Gateway** Platform on layerscape

**NXP Control Layer Application**
- Alink IoT SDK
- Alibaba IoT SDK
- AWS IoT SDK
- IBM Blumix SDK
- Baidu Speech SDK
- Amazon Alexa SDK
- MQTT Broker
- NFC Stack App
- Thread/BLE Control App
- Zigbee Control Bridge
- Mp3 Decoder
- AWS Greegrass

**User Space Application**
- TCP/IP Protocol
- NFC Driver
- KW41 Driver
- Zigbee Driver
- Speaker Driver

**Kernel**
- Hardware

**Cloud Service**
Overview of IOT Gateway with Alexa

The Alexa IoT Gateway demo contains:

- **LS1012ARDB**: The IoT gateway, post customer’s voice to Alexa voice cloud and get the commands from IoT cloud, then control the smart device.
- **Alexa Voice Service (AVS) API**: A Service allows everybody to integrate Alexa’s built-in voice capabilities into a connected products.
- **The Smart Home Skill API**: A service that understands the voice commands and converts them to *directives* (JSON messages) that are sent to smart home skills.
- **AWS Lambda**: A compute service offered by Amazon Web Services (AWS) that hosts the smart home skill code, which is called a *skill adapter*.
- **Smart Home Skill**: Code and configuration that interpret directives and send messages to IoT cloud.
- **AWS IoT Device Cloud**: The cloud environment provided that controls and manages the customer’s cloud-enabled devices.
- **Smart Devices**: The devices to be controlled by Alexa via cloud.
IOT Gateway with Alexa
Alexa IoT Gateway Demo

```python
def handleDiscovery(context, event):
    # Gateway logic
```

**control_light**

- **SmartHome**
- **Language**: English (U.S.)

**Configuration**

- **Language**: English (U.S.)

**Test**

- **ApplicationId**: amzn1.ask.skill.28a7cb65-3f4e-4080-b645-028ecf937d19

**Service Endpoint Type**

- **AWS Lambda ARN (Amazon Resource Name)**

**Privacy & Compliance**

- Name: control_light

**Publishing Information**

- Name: control_light

Example payload:

```json
payload=json.dumps({'state': {'desired': {'power': 'value'}}})
```
Alexa IoT Gateway Demo workflow

- A customer who has previously added devices or scenes to their device cloud says, “Alexa, turn on the kitchen light” to Alexa-enabled device(Ls1012ardb).
- The Alexa-enabled Ls1012ardb hears this instruction and sends it to the Alexa service for interpretation.
- The Alexa Smart Home Skill API interprets the action as "turn on" and the device name as "kitchen light". It composes a message to send to the skill adapter that controls the kitchen light. This message is called a directive. The directive includes:
  - The action (turn on)
  - The device identifier (an ID representing the device that the customer named "kitchen light")
  - Information authenticating the customer
- The skill adapter receives and parses the request for the action, the device identifier and authentication details. It uses this information to communicate with the device cloud. The skill adapter generates a message to the customer's device cloud, and tells the kitchen light to turn on.
- The device cloud gets the message and send the “turn on” command to Ls1012ardb Iot gateway.
- Ls1012ardb turns on the kitchen light.
- The skill adapter sends a response back to the Smart Home Skill API indicating whether it was successful. Alexa uses this response to determine the appropriate response to the customer. For example, Alexa might say, “OK” to indicate the requested is complete. (not shown in diagram)
Wireless speaker and IOT gateway demo

https://nxp1-my.sharepoint.com/personal/white_weng_nxp_com/_layouts/15/guestaccess.aspx?guestaccesstoken=K7MfJmFh9eouFhgBQ36DXQ1cLpjbpzrZEOO1wls1yk0%3d&folderid=2_174d6a34345d0453799cdd64e9256696e&rev=1