WHITEPAPER

Enabling IP Voice Applications Using uCMK6x

Overview and Design Considerations

Arcturus
We are Edge.

Table of Contents

Part-1 Overview of Voice/ IP Audio Applications 1

Two-way Voice (VoIP) 2

SIP - Session Initiation Protocol 2

RTP – Real-time Transport Protocol (Transmission of IP Audio) 2

IP Audio Distribution 2

Voice Control Interface 3

Part -2 Creating Voice/ IP Audio Devices 3

Encoding and Decoding Audio for IP Transmission 3

Creating High Quality Duplex Audio Systems 4

Enclosure Design 4

Audio Intelligibility and Psychoacoustic Enhancements 4

Part-3 Implementation using the uCMK6x 5

VoIP (PIP Mode) 5

IP Audio Distribution 6

uCMK6x Hardware Design 6

NXP K60 Microcontroller 7

Configuration and Maintenance 7

Operation and Control 8

Unsupervised / Autonomous Mode 8

Host Controlled via UART 8

Host Control via Network 9

Host Controlled / Supervised with Peripheral Integration 9

Operation and Control using Dedicated I/O Signal Interface (Autonomous) 9

Operation, Control Using Mbarx Private Cloud Model (Supervised and Controlled) 10

Mbarx Virtual Control Panel (VCP) Reference Implementation 11

Summary 11

Part-1 Overview of Voice/ IP Audio Applications

The proliferation of network connectivity combined with an insatiable appetite for content and communication has resulted in media over IP (internet protocol) becoming the biggest consumer of bandwidth globally. This demand has prompted IP network providers and equipment vendors to become media mature. Applications today can take advantage of the benefits packet switching has, while leading-edge projects, such as webRTC, promise a whole new voice and video experience.

Packet switched media benefits from improved distribution, relatively low-cost transmission and low signal loss over long hauls. The media itself can include telephony and voice communication, on-demand video content or two-way video. Streaming video on demand has its own set of data format, interoperability and content management issues, however the non-real-time, unidirectional nature and suitability for buffering offers flexibility in the way this service is delivered. In contrast, real-time, two-way voice or video is more susceptible to interruption, packet loss and jitter. In applications where hands-free full-duplex audio is required, dropping several frames can result in misunderstanding utterances in a conversation, thus overall audio intelligibility should always be paramount.

This whitepaper will address real-time voice interfaces and two-way voice communication with a focus on applications in building systems, healthcare and transportation industries. The table below describes a cross-section of applications with the perspective of how these devices can work together with and integrated system using other connected elements such as the Internet of Things (IoT).

Mode of Voice Application	Application Segment	Integrated Systems
Two-way voice	Security and building systems	Door entrance access control, voice/video door bells integrated with door-lock controls, NFC, RFID or access card sensors.
	Home monitoring	Two-way voice alarms integrated with mesh sensors; glass break, motion, smoke alarm sense, smoke, temperature detection.
	Healthcare	Patient care, nurse call, assisted living and senior care systems – integrated with biometric sensing, cloud-based patient records, activity tracking and safety.
	Transportation	Public safety, passenger assistance communication integrated with on-board safety buttons, vehicle location tracking.
One-to-many audio distribution	Security and building systems	PA paging and IP speakers integrated with fire and safety equipment for automated evacuation. School intercoms integrated with bells and time tracking. Pre-recorded announcements integrated with access control systems for automated lock-down procedures.
	Healthcare	PA paging and IP speakers for site and zone “code blue” notifications, staff alerts, integrated with corridor lamps, emergency call buttons, medical equipment.
	Transportation	PA paging and IP speakers for transit and airport

		operations. Automated boarding and stop announcements. Vehicle tracking and real-time passenger information. Driver, vehicle, central office communications.

Two-way Voice (VoIP)

In VoIP, Session Initiation Protocol (SIP) is used as the signalling to setup, change and tear down the media streams, the media streams themselves are transmitted independently using RTP (Real-time Transport Protocol).

SIP - Session Initiation Protocol

The SIP stack consists of a User Agent *Client* (UAC) and User Agent *Server* (UAS), capable of supporting direct P2P (peer-to-peer) connections between end-points. In most real-world applications, direct P2P is generally limited to simple setups with a limited number of fixed IP addressed devices. Instead, most deployments favour an architecture consisting of a central registrar and proxy service. These infrastructure elements support authentication, maintain location information and proxy signalling for many end-points. Infrastructure removes the need for each end-point to track the location of all other peers. Typically, the registrar and proxy are provided together as part of SIP server or IP-PBX, the capabilities of which can vary widely from stateless proxy/registrars to superclass softswitches. Using SIP infrastructure provides better overall system routing, call handling, reporting and security.

RTP – Real-time Transport Protocol (Transmission of IP Audio)

Media is transmitted over IP using the Real-time Transport Protocol, this protocol supports headers that identify a minimum set of media stream characteristics (sequence number, source identifier, timestamp). The role of RTP is to provide a low-latency stream of small packets (generally 20mS) to minimize encode/decode latency and interruption due to packet loss. RTP relies on secondary protocols (most commonly SIP) to control the session, negotiate type of encoding and jitter buffer facilities used to reassemble packet sequence at the destination. RTP is UDP, sent via best effort, low latency and is non-deterministic.

IP Audio Distribution

IP audio distribution requires the transmission of one audio source to many receivers (one-to-many). In this architecture receivers are effectively passive; communication is one-way from the transmitter to the receiver, which makes it possible to exploit the efficiency of multicast networking. Since multicast uses only one stream this improves synchronization and scalability as the number of receivers do not impact the resources of the transmitter or network bandwidth. There is no defined standard for PA paging session control, as such, several approaches that have been used. Some systems opt for no control protocol, requiring the receiver to play any received audio data from any source. This approach provides no authentication of the media stream and no way to clear channels. Some PBX systems opt to use unicast SIP notifications to each end-point, either inviting them to a unicast VoIP conference where audio is mixed to all participants or to provide the location of the multicast stream. In either case, processing resource needs to be considered when contacting many participants, this creates latency network overhead, signalling complexity and scalability limitations. Overall, the best approach is a subscribe/notify architecture using a multicast control protocol. In this architecture, each listener is subscribed to a common command channel. Notifications on this channel contain event information, session credentials and the multicast address of the RTP media stream. Each listening device can decode the credentials, determine if the notification is applicable to a group they are subscribed to and selectively listen to the media stream.

Voice Control Interface

Voice control relies on underpinning speaker independent speech recognition; this can be done natively on a host processor, or remotely by using an upstream server. Since good quality, large vocabulary speech recognition requires moderate resources (CPU / RAM) native implementations are not suitable for microcontrollers.

In a hosted model the end-point is responsible for detecting and encoding the analog voice input, packaging the data and transmitting it to the server. The server is responsible for processing the audio data, decoding speech and performing any post-processing required to improve context dependency and overall quality/confidence. The audio data is generally transmitted to the server using a typical HTTP post method with the server responding to the end-point in plain text, typically using a container format such as JSON (Java Script Object Notation).

Voice interfaces on popular smart phones and search engines use hosted implementations even though the device resources are capable of processing speaker independent speech. This is due to a desire to:

- Data mine user interaction and understand context better
- Post-process utterances to improve quality, patch up commands, re-order words or disambiguate
- Interface to search or knowledge engines
- Learn unique characteristics about individual users to build better interaction
- Rack up data usage to keep the carrier happy

Hosted speech applications are reliant on a low-latency Internet connection.

Developing voice control applications requires consideration of human factors, such as how to trigger the recognition engine, and how to apply heuristics to disambiguate language and interpret context.

Speech recognition relies on coarticulatory features in human voice; in contrast, human speech has evolved in a way where rudimentary words, such as "yes", "no", "on" "off" require the least articulatory effort. Thus, better recognition confidence can be obtained by designing interaction models that prompt for more complex responses. Similarly, improved user experience can be achieved by understanding context (e.g.: if a light is already on the predictability of a user will asking to turn it "on" again, is lower than "off"). Disambiguation also plays a role; phonetically, "on" and "one" are very similar; however, "turning a lamp one" is an unlikely voice command.

A separate white paper is available on speaker independent speech recognition for voice controlled applications.

Part -2 Creating Voice/ IP Audio Devices

Encoding and Decoding Audio for IP Transmission

Analog audio encode is handled by the host processor communicating using a16-bit PCM interface to the physical hardware codec. A software vocoder is used to encode the PCM data from the codec, packetize it and pass it to the IP stack. Decode uses the reverse method. Depending on the capability of the processor, DSP resources or instructions may be available to accelerate processing.

Voice communication primarily uses standardized ITU vocoders to encode / decode audio streams and maintain compatibility across provider networks. The most common of the telephony vocoders is G.711 Pulse Code Modulation (PCM), which is a 16-bit, narrow-band vocoder with a frequency response of 300–3400Hz, using 8KHz sample rates. In telephony, G.711 is considered toll quality, uncompressed speech. A single stream of G.711 uses 64Kbps of data, it is a low-latency / low-complexity algorithm that can use look-up tables and does not require DSP acceleration.

In recent years, concern over bandwidth has yielded to a desire to increase audio fidelity, resulting in the popularity of wide-band vocoders such as G.722. G.722 is a sub-band adaptive differential pulse code modulation (SB-ADPCM) vocoder that uses16KHz sample rates to achieve a frequency response of 50-7000Hz, while still using 64Kpbs of data, per audio stream. Recent advances including the OPUS codec are intended to address both narrow and wide-band applications under a single framework. For applications such as hosted speech recognition higher sample rates will improve recognition accuracy.

Those unfamiliar with IP based audio may defer to familiar digital audio codecs such as MP3 or wave formats. These are generally not suitable for real-time packet-based audio transmission due to encode/decode latency, interoperability with existing infrastructure and susceptibility to break down under less-than-ideal transport conditions.

Creating High Quality Duplex Audio Systems

Echo is the most common psychoacoustic factor for full-duplex communication systems. It does not reflect a technical limitation as overall audio quality maybe excellent. Echo is instead a human perception problem that can disrupt thought process and require considerable effort to overcome in common conversation. There are two typical types of echo, line echo and acoustic echo.

Line echo is introduced in 4-wire to a 2-wire hybrid bridge applications, such as a traditional Plain Old Telephone Service (POTS) circuit. It is a result of the transmitted audio signal reflecting off the far-end hybrid bridge into the return audio path. This produces a clear, pronounced echo with a fixed delay. Resolving line echo requires a Line Echo Canceller (LEC). Good results can be achieved.

Speakerphone applications suffer from acoustic echo introduced by acoustic coupling between the speaker and the microphone. This may be caused by reflected signal or through vibration in the physical enclosure. The echo length may be dynamic as conditions in a room change. Resolving acoustic echo requires an Acoustic Echo Canceller (AEC). Good AEC implementations have algorithms to effectively adapt to dynamic echo variances; however, applications that require two-way, hands-free voice, require considerable attention to physical design before a benefit from audio intelligibility or psychoacoustic enhancements will be perceived.

Enclosure Design

The most important single factor in creating a high quality full-duplex speakerphone/intercom device is enclosure design. A whitepaper is available from Microsemi® - below are some of the recommendations on how to minimize acoustic coupling contained in the whitepaper:

- Place microphone and speaker at opposite ends of the enclosure
- Isolate speaker into a separate physical cavity
- Mount speaker to enclosure using a vibration dampening foam rubber or silicone.
- Ensure speaker and microphone are not placed on the same three-dimensional plane, ideally orient the microphone 90 degrees perpendicular to the speaker
- House the microphone in a rubber grommet with no air gaps
- Do not physically mount microphone directly to circuit board or enclosure
- Minimize any enclosure grill surface over the microphone, ideally having a separate acoustically isolated cover.

Audio Intelligibility and Psychoacoustic Enhancements

A list of common algorithmic enhancements is provided below; however, it cannot be understated that a good acoustic design not only improves perceived audio quality, but also improves the performance gains of the enhancement algorithms.

Enhancement	Purpose	Tuning
Noise Reduction	Reduces background equipment sound such as HVAC or engine noise.	Does not need tuning
AEC – Acoustic Echo Canceller	Reduces return path echo caused by the speaker output detected by microphone input.	Needs to be tuned on a per-enclosure basis. Configurable setting help adjust tuning. Levels need to be set and EQ'd
AGC – Auto Gain Control	Dynamically adjusts levels based on ambient background audio level.	Base ranges and limits need to be set.
Beam Forming	Uses more than one microphone source to steer selective audio signals.	Microphone array needs to be calibrated.
Far-Field Mic Pickup	Isolates voice signal from distances as far as 3 meters, includes de-reverb and pick up auto gain control.	Makes use of several tuneable algorithms.

Part-3 Implementation using the uCMK6x

The uCMK6x is an MCU based platform that supports various modes of IP voice communication. It is suitable for full-duplex, point-to-point, distributed IP audio applications. All firmware resides in the MCU with no requirement for external memory. The MCU is connected to an audio subsystem part that supports a-to-d, d-to-a codec functions as well as audio intelligibility / psychoacoustic functions. The platform can be operated autonomously or can be host controlled by using dedicated I/O signals or using the built-in Mbarx host-protocol over serial or Ethernet. No code development on the MCU is required.

The uCMK6x makes use of firmware loads to support operating modes. By uploading firmware it is possible to change operating mode, allowing hardware reuse across multiple applications.

Type	Firmware Load	Description	Duplex Audio	Multicast Audio	Capability Summary
VoIP	PIP	Push-to-Call Intercom	✓		Push-to-call, intercom
VoIP	PIP-P	Push-to-Call Intercom	✓		Push-to-call, intercom with extended features
IP Audio Distribution	PAS	PA System Receiver		✓	Multicast (MCPG) receiver
IP Audio Distribution	PAT	PA System Transmitter		✓	Multicast (MCPG) transmitter
VoIP	PTT	Push-to-Talk multicast		✓	Walkie-talkie or PTT radio communication
VoIP+IP Audio Distribution	MSF	Push-to-Call and PA	✓	✓	Hybrid duplex intercoms with PA.

VoIP (PIP Mode)

VoIP capability is enabled with PIP mode firmware which supports RFC compliant SIP signalling and RTP media using G.711a, G.711u, G.722 vocoders. It is suitable for telephony, speakerphone and intercom applications. Audio intelligibility and psychoacoustic enhancements are supported using an audio subsystem. These enhancements include Acoustic Echo Cancellation (AEC), noise reduction, and may be extended to support Automatic Gain Control (AGC) beam-forming and far-field microphone pickup.

The SIP stack supports all key SIP methods and features including SIP infrastructure and P2P modes, Call originate / call receive, caller-ID, call hold and retrieve, refer (call transfer) and RFC2833 RTP DTMF event method.

An internal decode function is available to detect a specified DTMF signal and assign activate an output. This makes the solution ideal for the creation of door-entry systems.

IP Audio Distribution

IP audio distribution applications are relatively straight-forward when compared with two-way full duplex intercoms; however, architecture and network considerations require planning. The uCMK6x uses the MCPG (Multicast Paging) protocol to create one-to-many distributed audio systems. The stack use a subscribe/notify model that relies on multicast networking to distribute both control and audio communication.

For network applications that span across multiple routers and switches, special consideration must be applied when using multicast networking:

- Devices listening for multicast transmissions use IGMP (Internet Group Management Protocol) to communicate with routers to ensure data is transmitted through to devices subscribed to the multicast groups.
- Multicast transmissions use a TTL (Time To Live) counter that is decremented with each router hop traversed.

It is important to ensure the network a) supports IGMP and multicast appropriately b) architecture is sufficiently understood and TTLs set appropriately for messages to traverse to all destinations.

The subscribe/notify architecture uses a generic command packet broadcast on a pre-defined multicast address and port to provide broadcast, start, stop and continue control information. These messages contain the broadcast group number, the multicast RTP audio stream information as well as priority and other credentials. The architecture can support up to 100 concurrent groups. Each listening device can subscribe (listen) to any combination of groups and will use a last-in-wins method to arbitrate between transmissions that could occur concurrently.

The example below illustrates how this implementation can be used in a building system environment to assign multiple, hierarchical groups to represent the whole building (group 0), each floor (groups 1,2 and 3) as well as individual units (groups 10..18).

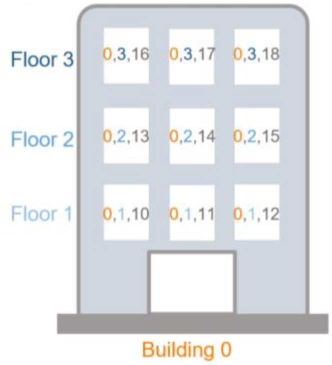

Figure 1: PA Paging in Building System

For applications that require broadcast transmission to a remote network, a multicast rebroadcaster solution is available that allows a unicast SIP VoIP call to be used as a method to connect to a remote location, then rebroadcast the VoIP call audio as a MCPG transmission inside the remote network.

uCMK6x Hardware Design

The Arcturus uCMK6x design implements the required subsystems, transceivers and power supply to create a flexible, cost effective board level solution. The design also implements an 802.3af compliant PoE power device (PD) subsystem, a class-D amplifier and exposes most MCU peripheral functions to one of two 60-pin header connectors to provide a modular design, or for expansion. The board implements 8 loop closure pairs and 8 loop closure detectors for connection to external equipment, buttons or sensors, these signals include transient voltage suppression or optical isolation. Additional I/O is provided on alternate MCU pins.

The uCMK6x is available in three forms:

- A development kit for evolution and prototype
- A System Solutions Board (SSB) suitable for direct field installation
- A module that can be connected to a custom host board using 2x 2 row 60 pin header connectors.

Various parts population options are available with both the system solutions board and the module.

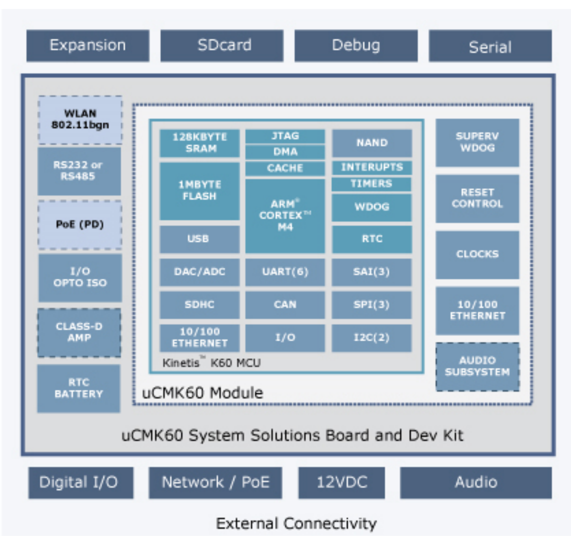

Figure 2: uCMK6x-VoIP Board and Module

The system consists of two key hardware semiconductor parts, the host MCU (A NXP K60 microcontroller) and an audio subsystem component.

NXP K60 Microcontroller

The K60 part includes all necessary flash and RAM internal to the MCU package with no need for external memory. The MCU's peripherals support Ethernet MAC, serial communications, external device interconnects and synchronous audio interface for connection to the audio subsystem using the I2S.

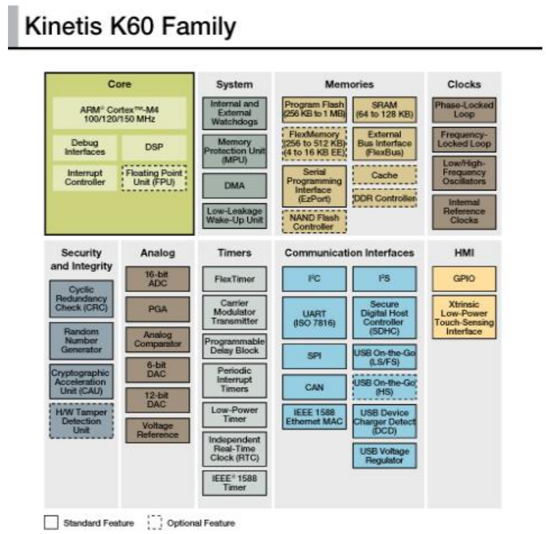

Figure 3: NXP K60 Microcontroller

Configuration and Maintenance

The uCMK6x makes use of the Mbarx-System Manager tool for configuration and maintenance. System Manager is a desktop GUI utility available for Windows® or Mac® that detects devices and simplifies management work flow through an easy to use interface. A free evaluation copy is available for download from the support site. The evaluation version is fully functional and can detect up to 5 devices on the network, the licensed version can detect unlimited devices. System Manager supports configuration, firmware management including system wide update deployments and configuration templates.

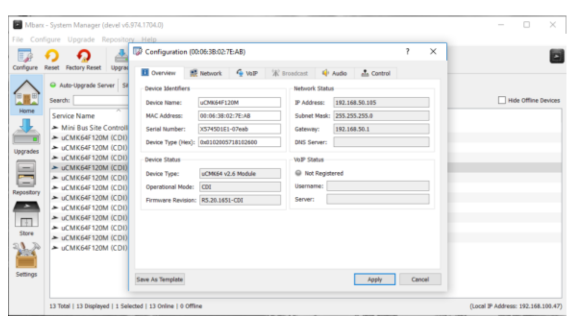

Figure 4: Mbarx System Manager

System Manager is part of an ecosystem of Mbarx Secure IoT tools, endpoints and gateways.

Additional Mbarx products are available for remote site management using a Site Controller. A demonstration of this capability is built into System Manager tool, allowing secure management of a remote Arcturus lab. Mbarx Operations Controller gateway stack is available for interactive workflow systems that require users, groups, notifications and device operational control. It is ideal for developing applications such as patient care systems or building entrance access controllers.

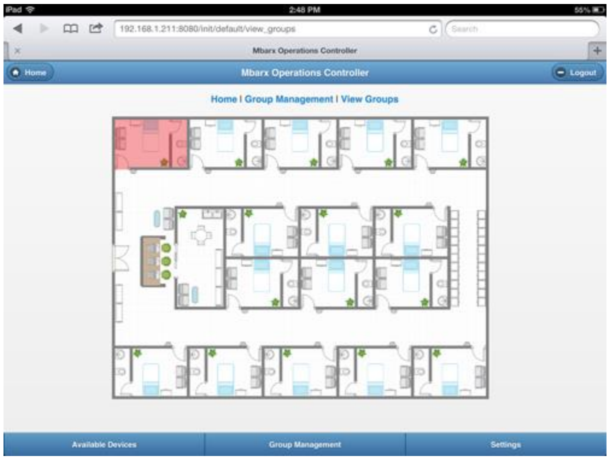

Figure 5: Example HTML5 Floor Plan Layout with Alarm Notification as Seen On iPad®

Operation and Control

The uCMK6x supports several operational and control paradigms, depending on the level of system integration required. Each paradigm relies on physical inputs to trigger events and outputs to report status notification. Additional system integration can be accomplished using the Mbarx protocol over a dedicated serial UART or TCP/IP socket connection.

Mode	Operation	Host Application	Peripheral Connectivity
Autonomous	Dedicated functions assigned to input and output signals	None	I/O
Autonomous with Supervision	Dedicated functions assigned to input and output signals	Yes, passive monitoring only, via UART or Network	I/O
Controlled (UART or Network)	Inputs are reported to controlling application (host) by software notification – controlling application interprets and responds. Outputs are controlled by application.	Yes, active control of the device. Device notifies hot application of events and expects instruction from application.	I/O Network to UART Pass-through commands

Unsupervised / Autonomous Mode

Autonomous mode provides an I/O interface where dedicated input signals trigger events and outputs notify states, using this mode it is possible to develop simple applications quickly. No supervising application or host control is required – the I/O interface support stateless operation.

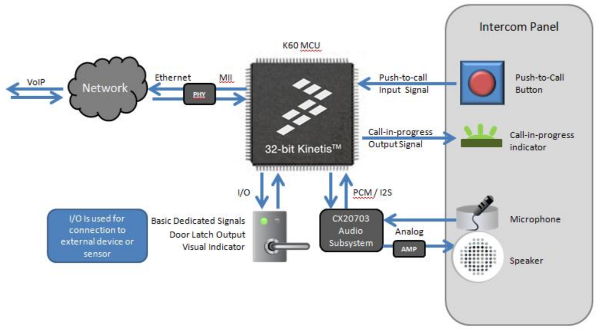

Figure 6: Autonomous Operation

Host Controlled via UART

For applications that require the uCMK6x to operate as a subsystem, it is possible to use the Mbarx host protocol to integrate using the UART to TCP/IP socket connection. This method allows for passive supervision or active control of the uCMK6x and I/O subset.

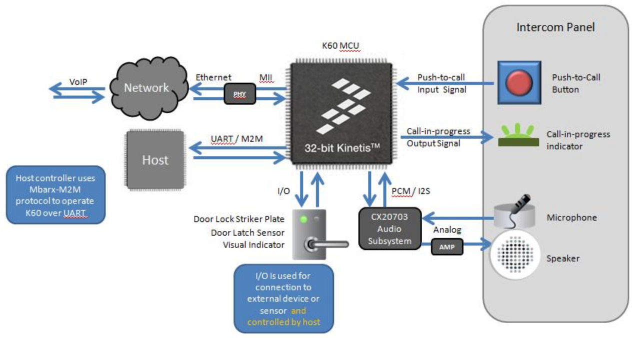

Figure 7: Host Control via UART

Host Control via Network

With applications that require a network of devices to have system-level integration it is possible to supervise and/or control the uCMK6x remotely using the Mbarx protocol over a network connection. This is suitable for private-cloud M2M models. This method allows for passive supervision or active control of the uCMK6x and I/O subset.

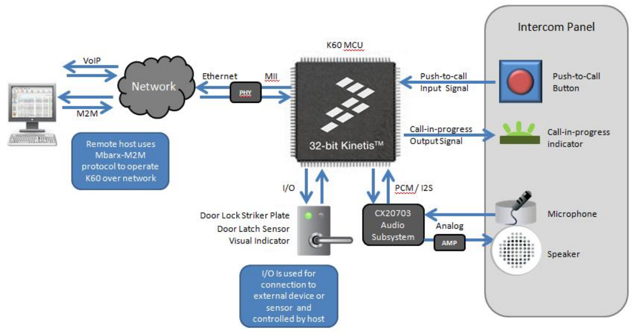

Figure 8: Host Control via Network Socket

Host Controlled / Supervised with Peripheral Integration

For applications that require integration with a serial peripheral device such as sensor, segment LCD or other automation equipment it is possible to pass commands from the network to the peripheral device using a pass-through command mode. This allows for better overall system integration and can reduce wiring and other costs.

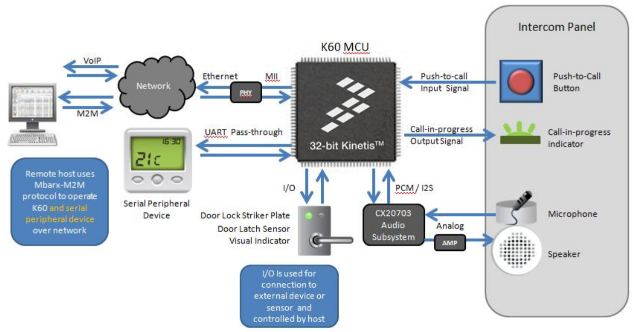

Figure 9: Peripheral Device Integration with Remote Host Connectivity

Operation and Control using Dedicated I/O Signal Interface (Autonomous)

The uCMK6x supports a broad set of dedicated signals that allow the MCU to be treated much like an ASIC (Application Specific Integrated Circuit). Using the MCU in this way allows dedicated inputs to trigger events and outputs to present status information. The specific I/O functions are defined for each firmware load and are contained in the uCMK6x–Signal Descriptions.pdf, available from the dedicated support site.

The table below represents PIP firmware signals. Using these signals it is possible to create stateless, stand-alone VoIP system of up to 10 configured devices, communicating using infrastructure or via P2P without the need for additional SIP servers. This has specific application in small-scale building entry intercom systems, vehicle communications or hospice health care environments.

Board Signal	Signal Name/Desc	Board Signal	Signal Name/Desc
OUT1	Registered / Ready	IN1	Push-to-Call (PTC) Input 1 / Answer
OUT2	Call-in-Progress / Ringing	IN2	Terminate
OUT3	Network Ready	IN3	Speaker Volume Up

OUT4	Alarm	IN4	Speaker Volume Down
OUT5	External Amp Enable	IN5	MIC Mute toggle
OUT6	Application Connected	IN6	SW Reset
OUT7	Spare	IN7	Spare
OUT8	Spare	IN8	Spare
OUT9	Hook State (off hook)	IN9	DND toggle
OUT10	RING (dedicated)	IN10	Call forward enable toggle (reserved)
OUT11	MIC mute enabled	IN11	Speaker mute toggle
OUT12	DND enabled	IN12	Answer call (dedicated)
OUT13	Speaker mute enabled	IN13	Factory reset (dedicated)
OUT14	Call forward enabled (reserved)	IN14	PTC Input 2
OUT15	PTC Output 2	IN15	PTC Input 3
OUT16	PTC Output 3	IN16	PTC Input4
OUT17	PTC Output 4	IN17	PTC Input 5
OUT18	PTC Output 5	IN18	PTC Input 6
OUT19	PTC Output 6	IN19	PTC Input 7
OUT20	PTC Output 7	IN20	PTC Input 8
OUT21	PTC Output 8	IN21	PTC Input 9
OUT22	PTC Output 9	IN22	PTC Input 10
OUT23	PTC Output 10		
OUT24	Message Waiting Indicator (MWI) (reserved)		
OUT25	External Door Lock Signal (DTMF)		

Operation, Control Using Mbarx Private Cloud Model (Supervised and Controlled)

The uCMK6x Mbarx protocol stack is comprised of the ASD operational protocol and an MDNS based service discovery announcement. Using the Mbarx stack it is possible to detect, connect, configure and operate devices as well as obtain status information in real-time. The combination of MDNS and ASD makes it possible to create a central, system-level application on a host server – generally referred to as an IoT model.

The MDNS service announcement uses a multicast broadcast message to announce device credentials on the network for other devices or services to use. The uCMK6x announces name, firmware name, version, location information, other related tools, (such as the Mbarx System Manager) and can make use of the MDNS service announcement to detect devices on the network and aggregate information into a database or for presentation. It also provides the location (IP address) of the device in order for System Manager to create a point-to-point connection for configuration and maintenance.

The Mbarx ASD protocol is a simple text-based method of passively supervising or actively controlling the uCMK6x. It can be operated over a TCP/IP socket connection or via UART. The protocol supports call control, remote operation of outputs, audio controls and event notifications – including inputs. It also supports configuration of user account, services, networking, and administrative settings. The ASD protocol supports an Ethernet to UART pass-through. This makes it possible to interconnect the uCMK6x to external host or slave devices, wireless sensors or other equipment. Full ASD protocol documentation is provided on the dedicated support site.

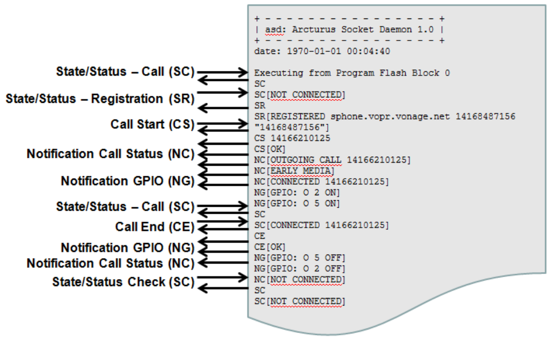

Figure 10: Mbarx Protocol and Actions

Mbarx Virtual Control Panel (VCP) Reference Implementation

A Python and QT reference implementation is available for Windows that creates a GUI Virtual Control Panel interface to a uCMK6x device by using the ASD protocol over a network connection. The interface demonstrates all operational aspects of the ASD protocol including call control, I/O control and event notifications. The implementation is available at no cost from the dedicated support site and can be modified. The reference implementation provides a rudimentary example of how a system-level application could be used to control or operate one or more devices within an IoT private cloud model.

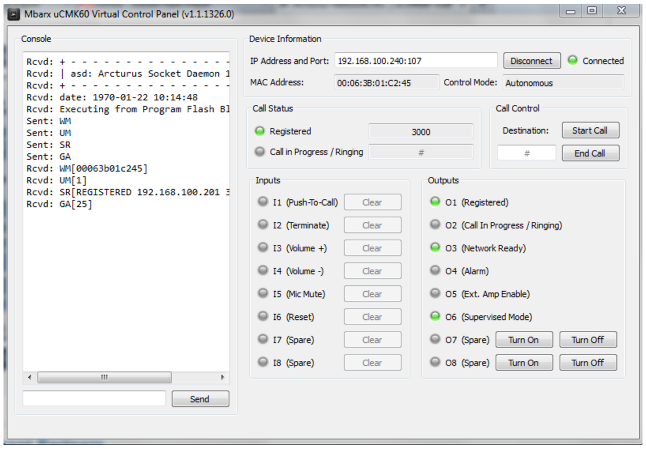

Figure 11: Mbarx Virtual Control Panel Reference Implementation

Summary

As IoT applications proliferate they present new and exciting ways to collaborate, communicate and interact, both from human and machine perspectives.

IP audio is a versatile method to provide human interaction; communication and control. With proper implementation and consideration for acoustics, environment and human factors, audio implementations can provide a more flexible user experience in a more economical way that video, touch screens and other traditional methods.

Arcturus Networks Inc.

701 Evans Ave. – Suite 300
Toronto, ON
M9C 1A3
CANADA

Toll Free North America: 1.866.733.8647
Tel: +1 416.621.0125

https://ArcturusNetworks.com

arcsales@arcturusnetworks.com

Arcturus
We are Edge.