

Fenster zur Welt der Unterhaltungselektronik

Einsatz von UPnP in MOST-basierten Systemen

Dieser Artikel beschreibt, wie sich der UPnP-Standard (Universal Plug and Play), der Standardschnittstellen für die Vernetzung von Unterhaltungselektronik- und mobilen Geräten definiert, in einem Fahrzeug-Infotainment-System einsetzen lässt, das auf einem MOST-Netzwerk basiert.

Von Dr. Alexander Leonhardi, Dr. Matthias Stümpfle, Rico Hauke und Thomas Gschwandtner

Fahrzeug-Infotainment-Systeme bieten Unterhaltungs-, Navigations- und Kommunikationsfunktionen nicht nur für den Fahrer eines Fahrzeugs, sondern auch für dessen Passagiere. Vor allem in Premiumfahrzeugen sind Infotainment-Systeme in den letzten Jahren immer wichtiger geworden. Sie stellen umfangreiche Funktionen zur Verfügung, ähnlich denen in stationären und mobilen Unterhaltungselektronikgeräten [1]. Neben den etablierten Infotainment-Funktionen nutzen Fahrer und Passagiere zunehmend auch die Möglichkeit, ihre Lieblingsmusik, die auf mobilen Datenspeichern, mobilen Geräten oder auf einem über ein Kommunikationsnetz zugänglichen Server gespeichert ist, mit ins Auto zu nehmen.

Über ein MOST-Netzwerk können Multimedia-Datenströme, Steuerbefehle und Massendaten zwischen den verschiedenen Geräten eines Fahrzeug-Infotainment-Systems übertragen werden. Neben der robusten optischen Bit-Übertragungsschicht ist die synchrone Datenübertragung, die eine einfache Übertragung von Multimedia-Daten erlaubt, einer der größten Vorteile des MOST-Netzwerks. Zusätzlich defi-

niert MOST Standardschnittstellen für verschiedene automobilspezifische Anwendungen, die bereits erfolgreich in einer ständig wachsenden Zahl von Fahrzeugen eingesetzt werden. Mit der aktuellen Version, die eine Übertragungsrate von 150 Mbit/s bietet, unterstützt MOST auch einen Ethernet-Kanal, der eine einfache Übertragung von IP-Daten ermöglicht.

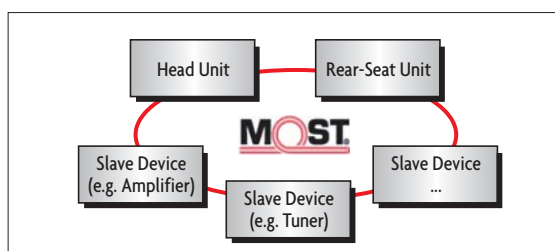
Der UPnP-Standard wurde definiert, um Geräte der Unterhaltungselektronik zu vernetzen [5]. Er beruht auf Internet-Standardprotokollen (z.B. TCP/IP, HTTP), die nicht domänenspezifisch sind. UPnP enthält Mechanismen, um Geräte (Devices) in einem IP-Netzwerk zu finden, Dienste (Services) anzubieten und auf diese zuzugreifen. UPnP ist weit verbreitet, zum Beispiel für die Kommunikation zwischen einem Streaming-Server und einem Streaming-Client oder um einen DSL-Router zu konfigurieren. Basierend auf dem UPnP-Gerätemodell hat die DLNA (Digital Living Network Alliance) Richtlinien definiert, etwa die Medienformate betreffend, welche die Interoperabilität zwischen Unterhaltungselektronikgeräten sicherstellen [3]. Für den Austausch von Multimedia-Daten wurden darüber hinaus in Zusammenarbeit mit dem UPnP-Forum Schnittstellen für verschiedene Unterhaltungsfunktionen sowie Streaming-Protokolle und -Dienste definiert. Wenn in diesem Beitrag von UPnP die Rede ist, sind die Anforderungen der DLNA implizit mit eingeschlossen.

In diesem Artikel werden verschiedene Szenarien diskutiert, wie UPnP in einem MOST-Netzwerk eingesetzt werden kann. In einer ersten Stufe wird UPnP für die Kommunikation zwischen den Front- und Rear-Seat-Einheiten eines Infotainment-Systems verwendet. Dabei werden die entsprechenden Standardprotokolle und Implementierungen aus der Welt der Unterhaltungselektronik mit den einfachen und robusten Übertragungsmechanismen für Multimedia-Daten eines MOST-Netzwerks kombiniert. In einer zweiten Stufe wird UPnP genutzt, um auf externe mobile Mediengeräte zuzugreifen und bestehende MOST-Geräte über eine Gateway-Funktion einzusetzen, die geeignete UPnP-Dienste auf die entsprechenden MOST-Schnittstellen abbildet. Als dritte Stufe wird UPnP schließlich als Standardkommunikationsmechanismus in einem MOST-Netzwerk verwendet.

Verwendung von UPnP in einem MOST-basierten Infotainment-System

Dieser Abschnitt beschreibt die oben genannten drei Stufen, um UPnP in MOST-basierten Infotainment-Systemen einzuführen. Dabei soll eine schrittweise Einführung ermöglicht werden, um die Rückwärtskompatibilität sicherzustellen und die Stärken beider Standards zu erhalten. Zu diesem Zweck werden bestehende Schnittstellen und Implementierungen

Bild 1. Beispiel für ein typisches MOST-Netzwerk.



nur möglichst geringfügig modifiziert, selbst wenn dies Leistungseinbußen zur Folge hat – z.B. wenn UPnP-Standardprotokolle eingesetzt werden, obwohl diese nicht für hohe Übertragungsraten in eingebetteten Systemen optimiert wurden.

Für die Diskussion wird das folgende Modell eines Infotainment-Systems angenommen (Bild 1): Eine Head-Unit (HU) stellt die Nutzerschnittstelle zum Fahrer sowie die Rechenleistung für die entsprechenden Anwendungen zur Verfügung. Ähnlich der HU stellt eine Rear-Seat-Unit (RU) die Schnittstelle und die Anwendungen für die hinteren Passagiere zur Verfügung. HU und RU sind über ein MOST-Netzwerk verbunden und können auf verschiedene Slave-Geräte wie einen Verstärker oder einen Digitalradio-Tuner zugreifen.

Auf Basis dieses Systemmodells sind die folgenden drei Stufen zur Einführung von UPnP möglich:

► Stufe 1 (Bild 2): HU und RU implementieren jeweils einen UPnP-Stack und realisieren entsprechende Funktionen durch die Kommunikation über geeignete UPnP-Mechanismen. Die RU kann z.B. UPnP-Standardmechanismen verwenden, um Multimedia-Daten, die auf der HU gespeichert sind, zu durchsuchen und abzuspielen, wobei die HU als UPnP-AV-Media-Server fungiert. Die dafür notwendigen UPnP-Befehle werden übertragen über TCP/IP und den MOST-Ethernet-Kanal, den MOST150 zur Verfügung stellt. Zur Übertragung der Multimedia-Daten bei der A/V-Wiedergabe werden die UPnP-Mechanismen so erweitert, dass die UPnP-Streaming-Kanäle auf die synchronen (Audio-) oder die isochronen (Video-)Kanäle des MOST-Netzwerks abgebildet werden. Zusätzlich stellt die HU einen Media-Renderer als Proxy für den Zugriff auf den externen MOST-Verstärker zur Verfügung.

► Stufe 2 (Bild 3): Anwendungen auf HU und RU werden vollständig in einer UPnP-Umgebung implementiert. Zu diesem Zweck müssen entweder die HU oder die RU Proxy-Funktionen für die MOST-Slave-Geräte zur Verfügung stellen. Weil Standard-UPnP-Dienste häufig nicht die Funktionen bieten, die für eine Fahrzeugumgebung oder deren spezielle Anforderungen an die Benutzerinteraktion not-

wendig sind, müssen neue UPnP-Dienste entsprechend zu den MOST-Blöcken der Slave-Geräte definiert werden. Unter Verwendung dieser Dienste können dann beispielsweise externe mobile Geräte, wenn sie UPnP unterstützen, auf die Funktionen des Fahrzeug-Infotainment-Systems zugreifen, beispielsweise um Audio-Daten über Verstärker des Infotainment-Systems auszugeben. In diesem Zusammenhang müssen allerdings noch entsprechende Sicherheitsmechanismen ausführlich untersucht werden.

► Stufe 3 (Bild 4): Schließlich soll UPnP als allgemeiner Mechanismus für die Kommunikation innerhalb eines Fahrzeug-Infotainment-Systems eingesetzt werden. Alle Anwendungen werden in einer UPnP-Umgebung ausgeführt und alle Befehle werden auf Basis von UPnP-Mechanismen übertragen. Dazu müssen auch die MOST-Slave-Geräte einen UPnP-Stack implementieren und Dienste gemäß den für die vorherige Stufe definierten Schnittstellen zur Verfügung stellen. In Ausnahmefällen wird der Zugriff jedoch weiterhin mittels eines Proxy-Dienstes auf der HU erfolgen, zum Beispiel bei einem externen Verstärker mit geringer Rechenleistung.

■ Einführung von UPnP in einer MOST-Umgebung

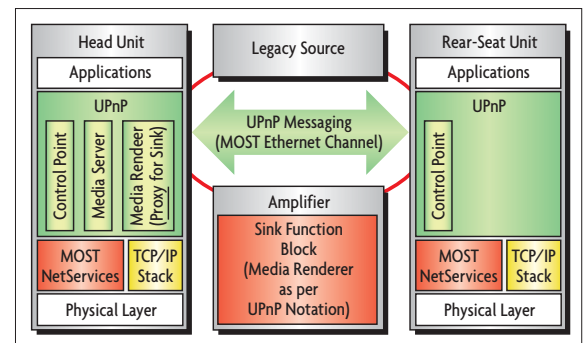
In diesem Abschnitt werden grundlegende Aspekte betrachtet, die zur Realisierung der oben beschriebenen Konzepte erforderlich sind. Eine ähnliche konzeptionelle Betrachtung wurde in der Vergangenheit bereits für das Thema „Integration eines MOST-Netzwerks in den HAVi-Standard (Home Audio Video Interoperability)“ durchgeführt [2, 6].

Systemarchitektur

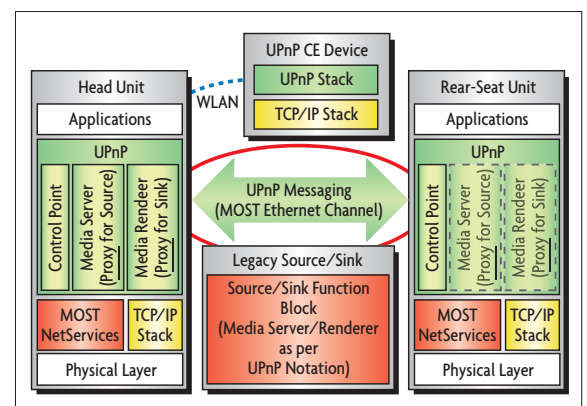
Die Systemarchitektur für UPnP in einer MOST-Umgebung erfordert eine Kommunikation auf der Grundlage von TCP/IP. In einem MOST-System werden HU und RU (und für Stufe 3 auch die Slave-Geräte) einen TCP/IP-Stack für den MOST-Ethernet-Kanal implementieren [4]. UPnP-Befehle für die Erkennung (Discovery), Beschreibung (Description) und Ansteuerung

(Control, Eventing) von Geräten und Diensten werden über diesen Kanal übertragen.

Sowohl HU- als auch RU-Anwendungen implementieren einen Control-Point, um auf Dienste im UPnP-Netz zugreifen zu können. Während des Startvorgangs des MOST-Systems werden die vorhandenen Geräte mit Hilfe der regulären MOST-Verwaltungsmechanismen erkannt. Für die Stufen 1 und 2 werden danach die jeweiligen Proxys auf der HU (oder der RU) initialisiert und bieten die Funktionen der Geräte – basierend auf deren FBlöcken – in Form entsprechender UPnP-Dienste an. So dient beispielsweise der Content Directory Service, der durch einen Media-Server bereitgestellt wird, dazu, Anfragen eines Control-Points bezüglich des Inhalts eines Media-Servers zu beantworten. Ein Proxy ist hierbei dafür verantwortlich, UPnP-Anfragen, die an ein Gerät geschickt werden sollen, in die entsprechenden MOST-Befehle umzusetzen und Mitteilungen von MOST-Geräten in entsprechende UPnP-Ereignisse zu übersetzen.



! Bild 2. Stufe 1 – UPnP für die Kommunikation zwischen HU und RU.



! Bild 3. Stufe 2 – UPnP als Programmierumgebung und für den Zugriff auf mobile Geräte.

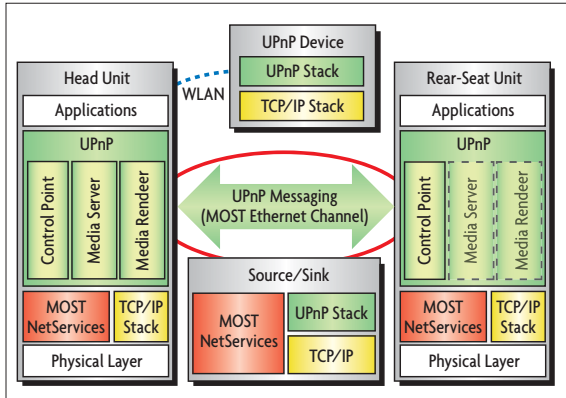


Bild 4. Stufe 3 – UPnP als allgemeiner Kommunikationsmechanismus für ein MOST-System.

Für eine Kommunikation zwischen Geräten ist es erforderlich, dass jeder Control-Point die MOST-Ressourcen kennt, um mit Hilfe dieses Wissens beispielsweise entsprechende Streaming-Verbindungen herstellen zu können.

Verwaltung von Datenströmen

Damit der Einsatz von UPnP in einer MOST-Umgebung möglich wird, ist es erforderlich, dass die Verwaltungsfunktionen von UPnP synchrone und isochrone Kanäle eines MOST-Netzwerks unterstützen, um A/V-Datenströme übertragen zu können. Aus diesem Grund muss die Ressourcen-Beschreibung (Resource Description) in UPnP erweitert werden, damit synchrone bzw. isochrone MOST-Kanäle und deren Parameter (z.B. die Sampling-Rate) verwendet werden können. Um einen Kanal im MOST-Netzwerk zu identifizieren, wird z.B. die MOST-Verbindungskennung (Connection Label) im Parameter ConnectionId der jeweiligen URI (Uniform Resource Identifier) übertragen.

Ein UPnP-Control-Point in einem MOST-System verwendet entsprechende MOST-Ressourcen, sobald eine Einigung bezüglich der Protokolle und Formate erfolgt ist. Erst dann ist die Herstellung einer Verbindung zwischen einem Media-Server und Media-Renderer möglich. Der Media-Server übersetzt beispielsweise die entsprechende PrepareForConnection-Anfrage in die passende Allocate-Nachricht auf MOST und gibt die MOST-Verbindungskennung als Teil der ConnectionId zurück. Wird später eine Anfrage mit dieser ConnectionId an den Media-Renderer übermittelt, so ist dieser im Stande, eine MOST-Verbindung

mit dieser Verbindungskennung herzustellen. Um den Kanal wieder abzubauen, verschicken Media-Server und Media-Renderer bei Eintreffen der ConnectionComplete-Anfrage des Control-Points die entsprechenden Disconnect- und DeAllocate-Befehle auf dem MOST-Netzwerk.

Zur Integration von Geräten, die UPnP-Streaming via HTTP unterstützen – etwa wenn ein mobiles Gerät den Verstärker des Fahrzeugsystems zur Audioausgabe verwendet – müssen die Media-Renderer und Media-Server in der Lage sein, HTTP-Streaming-Daten zu empfangen und zu versenden. In diesem Fall ist der Media-Renderer-Proxy für den MOST-Verstärker auf der HU dafür verantwortlich, die HTTP-Streaming-Daten in ein Format zu übersetzen, das nach Übermittlung via MOST-Kanal vom Verstärker ausgegeben wer-

den kann. Der dazu notwendige Kanal wird aufgebaut, sobald die entsprechende PrepareForConnection-Anfrage empfangen wird. sj

Literatur + Links

- [1] Leonhardi, A.: *Networking Vehicle Infotainment Systems with MOST. 1st International MOST Conference and Exhibition. Stuttgart 2009.*
- [2] Leonhardi, A.; Gschwandtner, T.; Simons, M.; Vollmer, V.; de Boer, G.: *Networking In-Vehicle Entertainment Devices with HAVi, in Advanced Microsystems for Automotive Applications 2004. P. 295 – 312. ISBN 3-540-20586-1.*
- [3] www.dlna.org
- [4] www.mostcooperation.com
- [5] www.upnp.org
- [6] Vollmer, V.; Simons, M.; Leonhardi, A.; de Boer, G.: *Architekturkomponenten für zukünftige Multimediastysteme im Fahrzeug. Konferenzbericht zur 24. Tagung Elektronik im Kfz. Haus der Technik. Essen 2004.*



Dr. Alexander Leonhardi

ist in der Entwicklung Elektrik/Elektronik-Telematik der Daimler AG für die Systemarchitektur der Infotainment-Systeme verantwortlich. Er vertritt die Daimler AG im Technical Coordination Committee der MOST Cooperation. alexander.leonhardi@daimler.com



Dr. Matthias Stümpfle

leitet die Abteilung Telematik-Systemarchitektur und -Plattformen in der Vorentwicklung der Daimler AG. Er ist Mitglied des Steering Committees der MOST Cooperation. matthias.stuempfle@daimler.com



Rico Hauke

ist bei der Daimler AG für die Vernetzungsarchitektur der Infotainment-Systeme der C- und E-Klasse zuständig. Er vertritt die Daimler AG in der Working Group Device Architecture der MOST Cooperation. rico.hauke@daimler.com



Thomas Gschwandtner

arbeitet in der Vorentwicklung der Daimler AG in der Abteilung Telematik-Systemarchitekturen und -Plattformen an den Themen Software-Architektur und Vernetzung. Er vertritt die Daimler AG in der Working Group Device Architecture der MOST Cooperation. thomas.gschwandtner@daimler.com