INTEGRATED SCIENTIFIC SERVICES
A MEDTECH COMPANY

MEDICAL GRADE BOOTLOADER FÜR ARM MIKROKONTROLLER

Vertraulichkeit – Authentizität – Integrität – Sicherheit

Firmware-Updates im Feld sind eine Herausforderung für viele aktive Medizinprodukte. Auf der Basis von mehreren Implementierungen für anspruchsvolle Anwendungen in der Medizintechnik wurde ein generischer Bootloader-Kern entwickelt, der flexibel in Medizinprodukte-Firmware eingebettet werden kann und eine sichere Verteilung und Installation von solchen Firmware-Updates erlaubt.
Durch den Einsatz des ISS Bootloaders kann in einem Projekt viel Entwicklungszeit gespart und Mehrwert im Bereich des sicheren Remote-Updates geschaffen werden.

Ihr Kontakt

Andreas Müller
Head of Software Development

T +41 32 513 67 83
andreas.mueller@iss-ag.ch

Übersicht

Der ISS Bootloader ist Teil einer Platform für die sichere Verteilung von Software und Firmware auf Medizingeräte. Der Bootloader implementiert die Funktionalität für die Installation, Updates sowie den Schutz der Firmware gegen Veränderung und Reverse Engineering. Der Bootloader führt bei jedem Start des Geräts eine Integritätsprüfung der installierten Firmware durch. Mit ensprechend konzipierter Hardware lässt sich somit das Patientenrisiko durch alterungs-, umweltbedingter sowie böswilliger Modifikation der Firmware minimieren.
Der Bootloader wird vorrangig in Systemen eingesetzt, welche aus einem Host-Rechner (PC) und einem oder mehreren Mikrokontrollern bestehen. Der Bootloader und die jeweilige Applikationsfirmware implementieren eine proprietäre RPC-Schnittstelle über welche sie von einer Applikationssoftware auf dem Host-Rechner gesteuert werden können.

Funktionen und Eigenschaften
Der Bootloader wurde im Hinblick auf folgende Anforderungen entwickelt:
- Installation der Firmware während der Produktion
- Update der Firmware und Bootloader im Feld (In-Application Programming)
- Integritäts- und Authentizitätsprüfung der Firmware beim Start
- Schutz des Geräts gegen die Installation von unautorisierter Firmware
- Schutz gegen die Installation der Firmware auf unautorisierten Geräten
- Schutz gegen Modifikation der Firmware oder des Firmware-Updates
- Schutz gegen Reverse-Engineering der Firmware oder des Firmware-Updates
- Resistenz gegen Fehler / Verhindern des Unbrauchbarwerdens oder zumindest sicherstellen der Wiederherstellbarkeit des Geräts bei fehlgeschlagenen Updates (z.B. bei Stromausfall)
- Resistenz gegenüber Angriffen
- Begrenzung des Schadens eines erfolgreichen Angriffs (z.B. Auslesen von Schlüsseln aus dem Mikrokontroller)

Optional:
- Downgrade Prevention – Verhindern der Installation von älteren Firmware-Versionen
- Gerätespezifische Firmware – Die Firmware ist an die Hardware eines Geräts gebunden (z.B. via Unique Identifier des Mikrokontrollers)
- Production Keys – Sicherstellen, dass Geräte oder Komponenten, welche z.B. auf dem Weg vom Elektronikhersteller zum Assemblierer "vom Laster fallen", nicht verwendet werden können

Entwicklungsstand
ISS bietet eine fertige, lizenzierbare Bootloader-Lösung an, sowie kundenspezifische Anpassungen und Entwicklungen als Dienstleistung.

Verifizierung und Integrationsverifikation
Die ISS AG ist ISO 13485 zertifiziert und der Bootloader wurde konform zu IEC 62304 entwickelt, dokumentiert und verifiziert.
Für die Integration in ein Kundenprodukt werden die Integrationspläne und Protokolle mitgeliefert, die durch den Kunden oder ISS an das jeweilige Produkt adaptiert werden.

Technische Details

Unterstützte Hardware-Platformen

Als erstes werden Modelle der STM32F4xx (Cortex-M4) Familie von STMicroelectronics unterstützt und die Unterstützung dann, bei Bedarf, auf weitere Kontrollerfamilien desselben Herstellers ausgeweitet. Geplant ist die Unterstützung von Mikrokontrollerfamilien, welche auf ARMv7-M, ARMv7e-M und eventuell ARMv6-M Architekturen basieren und die benötigten Hardware-Ressourcen zur Verfügung stellen (RAM, ROM). Aktuell laufen Aktivitäten zur Unterstützung von Texas Instruments C2000™ Mikrokontrollern.

Hardware-Anforderungen

On-Chip Flash-Memory: Es werden vier unabhängige Regionen im Flash-Speicher benötigt; die Regionen müssen unabhängig voneinander lösch- und beschreibbar sein und genügend Platz für die Bootloader-Komponenten und Applikationsfirmware bieten.
Dies sind mindestens:

Region 1	4kB (Prebootloader)
Region 2	100kB (Bootloader Primär) oder 16…32kB für Varianten mit angepasster Kryptographie
Region 3	100kB (Bootloader Sekundär) oder 16…32kB für Varianten mit angepasster Kryptographie
Region 4	Gegeben durch Applikationsfirmware
On-Chip SRAM	32kB RAM und mehr (abhängig vom Funktionsumfang)

Bootloader Update

Nebst der Applikationsfirmware können auch der Bootloader und die kryptographischen Schlüssel sicher im Feld aktualisiert werden. Damit das Gerät selbst bei einem fehlgeschlagenen Update weiterhin verwendbar oder immerhin wiederherstellbar ist, existieren auf dem Target-System zwei Kopien des Bootloaders. Ein Bootloader kann immer nur die andere Kopie überschreiben, womit im Fehlerfall ein Fallback garantiert ist.

Asymmetrische Kryptographie

Durch die Verwendung von asymmetrischer Kryptographie für die Ver- und Entschlüsselung der Firmware sowie Generierung und Verifikation der digitalen Signaturen können Schlüssel, welche von einem Angreifer aus einem Gerät extrahiert wurden, nicht direkt für weitere Angriffe auf andere Geräte oder das Gesamtsystem verwendet werden. Ein Schlüssel welcher z.B. für die Verifikation einer Signatur im Speicher des Gerätes liegt, kann nicht zur Generierung von gültigen Signaturen verwendet werden.

Schnittstellen

Die Funktionen des Bootloaders können über ein proprietäres RPC-Protokoll (Remote Procedure Call) verwendet werden. Dasselbe Protokoll kann ebenfalls in der Applikationsfirmware eingesetzt werden, was eine optimale Integration von Bootloader und Applikation erlaubt. Softwarebibliotheken für C, C# und C++ stehen bei Bedarf zur Verfügung.

Das RPC-Protokoll kann über die meisten verfügbaren elektronischen Schnittstellen eingesetzt werden und kann für verschiedene Topologien konfiguriert werden:

- Point to Point – Ein Host (Master) und ein Mikrokontroller (Slave): Master → Slave 1
- Daisy Chain – Ein Host (Master) und mehrere in Serie geschaltete Mikrokontroller (Slaves): Master → Slave 1 → Slave 2 → …
- Stern – Ein Host (Master) mit mehreren physikalischen Point-to-Point Verbindungen
- Netzwerk – Ein Host (Master) mit mehreren logischen Point-to-Point Verbindungen, z.B. über ein bestehendes TCP/IP Netzwerk, einen I2C oder SPI Bus mit mehreren adressierbaren Slaves.

Einschränkungen

Da der Bootloader in den meisten Fällen, gleich wie die Applikationsfirmware selbst, im normalen Programmspeicher abgelegt wird, wird empfohlen diesen zusätzlich durch Hardware-Massnahmen gegen böswillige Modifikation (z.B. über JTAG) zu schützen. Die meisten Mikrokontroller enthalten bereits von Haus aus verschiedene Möglichkeiten dazu.