

Messen und Schalten im Rechenzentrum: Kostengünstige Sensorknoten mit sicherer Anbindung an offene Netze

Michel Steichen, Dirk Henrici, Paul Müller
{m_steich, henrici, pmueller}@informatik.uni-kl.de

Abstract: Die in Rechenzentren eingesetzten vernetzten Sensorknoten zum Messen und Schalten sind heute in den meisten Fällen unabhängige und kostspielige Endgeräte. Intention dieses Beitrages ist es, ein verbessertes Konzept für solche Sensorknoten vorzustellen. Dabei handelt es sich keinesfalls um eine Nachimplementierung bestehender Lösungen. Vielmehr wurden in dieser Arbeit die Schwächen aktueller Lösungen und Lösungsvorschläge analysiert und darauf basierend eine neue, generische Architektur entworfen und prototypisch implementiert, die auch in anderen Bereichen, wie z.B. dem Assisted Living, einsetzbar ist. Berücksichtigte Anforderungen sind unter anderem Anschaffungskosten, Energieverbrauch im Betrieb, Benutzerfreundlichkeit, Skalierbarkeit, Sicherheit, Zuverlässigkeit und Flexibilität. Um all diese Anforderungen erfüllen zu können, wurden die Aufgaben der Sensorknoten in ein Modell aus mehreren Schichten aufgeteilt.

1 Einleitung

Mit Hilfe von Sensorknoten lassen sich Umgebungszustände überwachen und Aktuatoren steuern. Die Technologie ist zur Gebäudeautomation und damit auch zur Überwachung von Rechenzentren geeignet. Die Aufgaben dieser Geräte sind vielfältig. Passive Datenerfassung, wie beispielsweise die Messung von Temperatur, Luftfeuchtigkeit und Stromverbrauch ermöglichen eine gezielte und schnelle Zustandserfassung der aktuellen Gegebenheiten. Mittels aktiver Komponenten, wie Relais, anderen Aktuatoren oder Displays, besteht die Möglichkeit mit der Umgebung zu interagieren.

In dieser Arbeit werden Sensorknoten vorgestellt, die auf einer flexiblen Schichtenarchitektur [1] basieren. Im Folgenden werden die Gründe und die Absichten erläutert, die die Neuentwicklung solcher Sensorknoten motivieren.

1.1 Motivation und Ziel

Beim genaueren Betrachten der zahlreichen angebotenen Lösungen erkennt man in den meisten Systemen eine Reihe von Schwächen. Aus diesen Erkenntnissen heraus wurden die Anforderungen an das neue System identifiziert.

Ziel der Arbeit war es, ein neues Design für Sensorknoten zu entwickeln, welches Verbes-

serungen in den folgenden Bereichen aufzeigt:

1. Kosten in der Herstellung / im Erwerb
2. Energieverbrauch im Betrieb
3. Benutzerfreundlichkeit bei der Installation und der Konfiguration
4. Skalierbarkeit des Systems
5. Sicherheit beim Zugriff und der Datenübertragung
6. Zuverlässigkeit während des Betriebs

Um sämtlichen Anforderungen gerecht zu werden, war es notwendig, die Funktionalität von Sensorknoten in ein Modell aus mehreren Schichten aufzuteilen. Eine solche Vorgehensweise ermöglicht es, die "Intelligenz" und Komplexität auf der untersten Ebene auf ein Minimum zu reduzieren und die Gesamtaufgabe zweckmäßig auf die verfügbaren Schichten aufzuteilen. Aufbau und Funktion der einzelnen Ebenen werden im dritten Abschnitt behandelt.

1.2 Problemstellung

Der Aufwand bei der Entwicklung von Sensorknoten teilt sich auf die drei Bereiche Hardware, Vernetzung und Software auf.

Hardware: Im Vordergrund steht hier die geeignete Wahl der Bauteile die, unter Berücksichtigung der Anforderungen, kostengünstig und energiesparend sein sollen.

Vernetzung: Für die Verbindung von Sensorknoten und Steuerknoten werden Bussysteme oder andere Datennetztechnologien benötigt. Neben der Übertragungsgeschwindigkeit stehen auch hier die Merkmale Energiebedarf und Kosten an vorderster Stelle. Beispielsweise benötigt Ethernet mit ca. 1 Watt pro Port deutlich mehr Energie als Feldbussysteme wie ein CAN-Bus oder Profibus. Darüber hinaus gibt es weitere Kriterien, etwa Quality-of-Service-Eigenschaften.

Software: Der Bereich Software beschäftigt sich mit der Implementierung von Protokollen, Sicherheitsalgorithmen, Dienstprogrammen zu Installations- und Konfigurationszwecken, sowie einer benutzerfreundlichen Steuer- und Benutzungssoftware. Die Firmware der Sensorknoten vermittelt zwischen Sensorhardware, Mikrocontroller und Schnittstellen.

1.3 Anforderungen an die Architektur

Im Allgemeinen sind Sensorknoten darauf ausgelegt, unterschiedliche physikalische Größen, beispielsweise Temperatur, Druck oder Helligkeit, in einen entsprechenden elektrischen Spannungswert oder eine entsprechende Stromstärke umzuwandeln. Diese analoge Eingabe wird dann mit Hilfe von A/D-Wandlern in einen digitalen Wert überführt. Neben analogen Werten gilt es aber auch binäre Zustände zu erfassen.

Neben der Datenerfassung spielt das Interagieren mit der Umwelt eine elementare Rolle. Die Hardware der Sensorknoten muss somit die Funktionalität besitzen, auch aktiv in die Umgebung einzugreifen zu können. In den meisten Fällen begrenzt sich dieses Steuern und Regeln auf das Ein- und Ausschalten von Verbrauchern. In Fällen, wo beispielsweise ein analoger Spannungswert gebraucht wird, können entsprechende D/A-Wandler eingesetzt werden. Displays sollten zur übersichtlichen Anzeige von Betriebszuständen ebenfalls angesteuert werden können.

Ein weiteres relevantes Merkmal der Architektur ist die benutzte Vernetzungstechnologie. Wie schon kurz angedeutet, wird das Sensornetz in diesem Beitrag auf mehrere Schichten aufgeteilt. Um den Anforderungen gerecht zu werden, ist die Hardware auf den untersten Schichten kostengünstig und ressourcensparend gehalten. Hier kommen Feldbusse zum Einsatz, auf höheren Schichten hingegen Ethernet. Im Rahmen der für diesen Beitrag durchgeführten Implementierung werden CAN-Busse als Feldbusse verwendet. Jedoch können prinzipiell zahlreiche unterschiedliche Bussysteme und auch drahtlose Übertragungstechniken zum Einsatz kommen.

Was die Eingaben angeht, so sollten die Sensorknoten Wert- und Zustandsänderungen eigenständig weitergeben. Das Sensornetz kann somit seine Umgebung aktiv überwachen. Ein permanentes Abfragen der aktuellen Werte und Zustände, sogenanntes "polling", wird überflüssig. Besonders interessant ist die Frage, wann und wie oft Änderungen weitergeschickt werden. Bei binären Eingängen ist das Versenden einer Nachricht bei jedem Flankenwechsel ausreichend. Optional kann dazu noch die Zeit gemessen werden, die zwischen zwei Zustandsänderungen verstrichen ist. Die Reaktion auf analoge Eingabewerte erweist sich als komplizierter. Allgegenwärtiges Rauschen erzeugt kontinuierliche Schwankungen bei den Messergebnissen. Diese unerwünschten und chaotischen Änderungen können durch Tiefpässe geglättet werden. Falls dies nicht anwendbar ist, besteht die Möglichkeit, nur in periodischen Zeitabständen eine Wertänderung weiterzuleiten.

2 Aktuelle Lösungen und Stand der Technik

Es gibt eine Vielzahl von Konzepten und Produkten auf dem Markt, die auf unterschiedliche Anwendungsbereiche ausgerichtet sind und die oben identifizierten Anforderungen in unterschiedlichem Maße erfüllen. Im Folgenden sind einige Beispiele dargestellt, um das verfügbare Spektrum aufzuzeigen.

SCADA

SCADA [2] steht für "Supervisory Control and Data Acquisition". Dabei handelt es sich um ein reines Konzept, welches der Überwachung und Steuerung von Industrieanlagen dient. Anwendungen von SCADA liegen im Bereich der Regelung unter Berücksichtigung von Echtzeitfähigkeit. Die Technologie ist in mehrere Schichten aufgeteilt. Die Sensoren und Aktuatoren werden von Fernbedienungsterminals (RTU) oder speicherprogrammierbaren Steuerungen (SPS) kontrolliert. Mit Hilfe einer Benutzerschnittstelle in den oberen Schichten können die erfassten Daten visualisiert werden, sowie das Verhalten des Systems gesteuert werden.

Da es sich bei SCADA um ein Konzept handelt und keine abgeschlossene Technologie, gibt es eine Reihe von Herstellern, die ihre eigenen proprietären Interpretationen von SCADA anbieten. Oft kommen hier schon auf unterster Ebene komplexe Techniken wie Ethernet und TCP/IP zum Einsatz.

LCN

LCN [3] (Local Control Network) ist ein proprietäres und universelles Gebäudeleitsystem. Die einzelnen Sensorknoten haben frei benutzbare Ein- und Ausgänge. Die Kommunikation der Knoten erfolgt über ein Bussystem. Ein bestehendes Netz lässt sich jederzeit leicht erweitern und mit Hilfe einer PC-Software steuern.

Nachteil dieser Technologie ist die erforderliche Verdrahtung und Installation durch einen qualifizierten Elektroinstallateur. Die Unterstützung von drahtlosen Komponenten würde das Nachrüsten in alten Gebäuden und Häusern wesentlich vereinfachen.

BTNode

Bei BTNode [4] handelt es sich um eine reine drahtlose Sensornetz-Plattform. Die von der ETH Zürich entwickelten Sensorknoten können autonome Ad-Hoc-Netzwerke untereinander aufbauen. Die Schnittstellen der Ein- und Ausgänge sind sehr generisch gehalten, und die Kommunikation erfolgt über Bluetooth.

Trotz der interessanten und leistungsfähigen Funktionen, die diese Knoten besitzen, erfüllen sie in mehreren Punkten keinesfalls die in diesem Beitrag angestrebten Anforderungen. BTNodes bestehen aus kostspieliger Hardware und sind leistungsfähige Endgeräte mit anderer Zielsetzung.

Im nächsten Kapitel folgt ein Einblick in die praktische Umsetzung eines Sensornetzes, welches die vorgestellten Anforderungen erfüllt.

3 Umsetzung der Anforderungen

Basierend auf den Designentscheidungen wird in diesem Abschnitt das Konzept der Schichtenarchitektur erklärt. Anschließend werden die Anforderungen in Bezug auf die Umsetzung auf das Schichtenmodell nochmals detaillierter formuliert.

3.1 Die Schichtenarchitektur

Auf den ersten Blick scheint ein Modell aus mehreren Schichten die Entwicklung eines Sensorknotensystems unnötig komplizierter zu machen. Doch durch eine solche Struktur vermeidet man genau die Probleme, die bei vielen Lösungen immer wieder auftauchen. Abbildung 1 gibt einen ersten Überblick über dieses Schichtenmodell.

In den Endgeräten, d.h. den Sensorknoten selbst, sollte möglichst wenig Funktionalität liegen, damit sie einfach und kostengünstig sind. Die "Intelligenz" des Systems wird auf

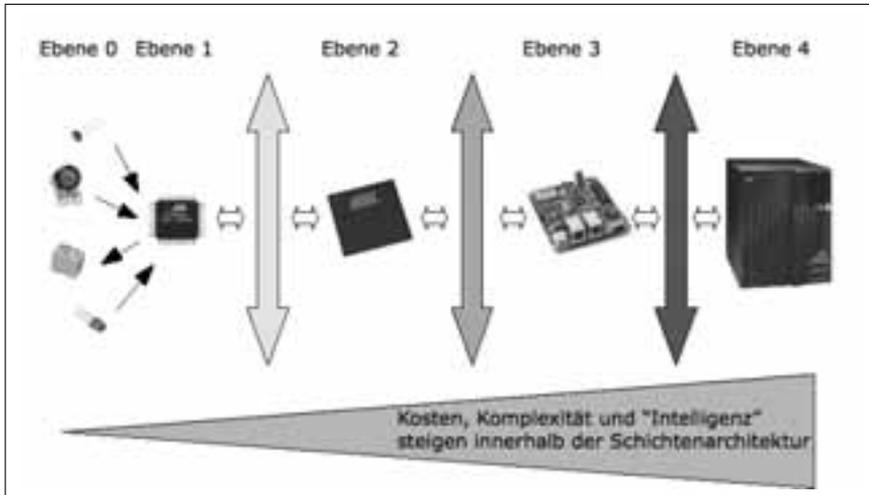


Abbildung 1: Schichtenarchitektur

die Hardware und Software der einzelnen Schichten aufgeteilt. Unter Berücksichtigung der Kosten und des Energiebedarfs gibt es ein Gefälle von den höheren Schichten aus zu den niedrigeren Schichten hin. Die unteren Ebenen enthalten jeweils nur soviel "Intelligenz", wie unbedingt erforderlich ist. Somit befinden sich auf den Ebenen null und eins auch die günstigsten Komponenten des Sensornetzes. Ebene eins hat nur die Aufgabe, die Sensoren und Aktuatoren zu bedienen. Die gesammelten Information werden dabei so schnell wie möglich an die nächste Schicht weitergegeben.

Dieser Ansatz weist einige Vorteile auf. Soll eine Erweiterung des Netzes vorgenommen werden, so reicht es aus, neue Komponenten in den ersten beiden Schichten einzufügen. Die neu eingefügten Komponenten gliedern sich problemlos in das bestehende Netz ein und können von den darüber liegenden Schichten ohne großen Aufwand zusätzlich bedient werden. Damit erreicht man nicht nur eine Verringerung der Anschaffungskosten sondern zentralisiert und vereinfacht die Verwaltung und verringert die benötigte Hardware und die Komplexität.

Dies lässt sich am besten anhand eines Beispiels erläutern. In einem installierten Sensorknotennetz wird ein zusätzlicher Temperatursensor benötigt. Bei vielen konventionellen Sensorknoten geschieht eine solche Erweiterung durch den Kauf eines neuen Sensorknotens. Da es sich hierbei um komplexe selbstständige Endgeräte handelt, erwirbt der Kunde zusätzlich zum Sensor ein komplettes eingebettetes System, welches beispielsweise über einen Ethernet-Port und die Rechenleistung eines kleinen Webservers verfügt. Durch diesen zusätzlichen, nicht erforderlichen Ballast erhöhen sich die Kosten deutlich. Hier sind Anschaffungskosten der Komponente, Energieverbrauch und Kosten für benötigte Switchports zusammen zu rechnen. Um Denial-of-Service-Angriffe zu verhindern, müssen die Geräte vom Internet abgetrennt betrieben werden. Die Komplexität der Software kann aufwändige Updates erfordern.

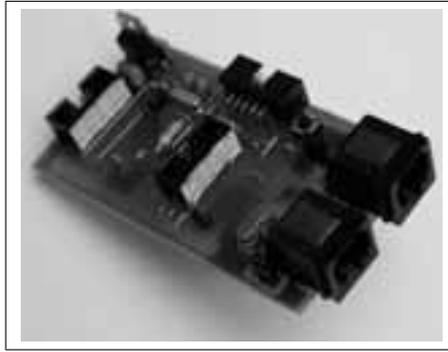


Abbildung 2: Sensorknoten

Durch das Schichtenmodell vermeidet man diese Kosten und Probleme. Der Bedarf eines zusätzlichen Temperatursensors reduziert sich auf die Anschaffung eines Sensorknotens auf der ersten Ebene. Dieser lässt sich benutzerfreundlich in das bestehende Netz einfügen. Die Kosten reduzieren sich auf ein Minimum, ebenso der Energiebedarf. Der neue Sensorknoten teilt sich nun, zusammen mit den Knoten seiner Ebene, das Rechenpotential der höheren Schichten, die kompliziertere Aufgaben bewältigen können. Somit fällt diese Lösung viel billiger und umweltfreundlicher aus.

Im folgenden Abschnitt wird die praktische Umsetzung der einzelnen Schichten in Bezug auf die identifizierten Anforderungen dargestellt.

3.2 Die Realisierung eines Prototypen

Im Rahmen einer Diplomarbeit [5] an der Technischen Universität Kaiserslautern wurde das geschilderte Schichtenmodell detailliert ausgearbeitet. Unter Berücksichtigung der Anforderungen wurde ein Prototyp entwickelt, der nun zur Klimaüberwachung im Rechenzentrum eingesetzt wird.

Sensoren/Aktuatoren und Sensorknoten: Ebenen null und eins

Die Sensorknoten, ein Beispiel ist in Abbildung 2 dargestellt, sind sowohl aus Sicht der Herstellung als auch der benötigten Betriebsenergie kosteneffizient. Bei der Auswahl der Mikroprozessoren und Controller wurde auf einen niedrigen Anschaffungspreis, eine gute Verfügbarkeit sowie eine stromsparende Ausführung geachtet. Die Wahl fiel dabei auf den 8-Bit Mikrocontroller ATmega8 von Atmel¹. Die Hardwareressourcen des ATmegas sind mit 8 KB Flash-Programmspeicher und 1 KB RAM-Arbeitsspeicher recht bescheiden aber vollkommen ausreichend. Eine Taktfrequenz von 4 MHz bildet einen guten Kompromiss zwischen Energiesparen und Rechengeschwindigkeit. Zudem ist das Bauteil gut erhältlich und liegt in der Preisklasse der Ein-Euro- μ C.

Die Datenübertragung auf Ebene eins erfolgt über einen CAN-Bus. Dieser Feldbus zeich-

¹<http://www.atmel.com>

net sich durch seine Robustheit, Priorisierungsmöglichkeiten, sowie einem geringen Strombedarf aus. Die Firma Microchip² vertreibt einen preiswerten CAN-Controller unter dem Namen MCP 2515. Mittels SPI-Interface lässt er sich leicht an den ATmega anschließen. Eine weitere energiesparende Maßnahme ist die Möglichkeit, den Mikrocontroller und den Controller während Rechenpausen in einen so genannten Schlafmodus zu versetzen. Die Stromaufnahme kann in dem Fall um den Faktor eintausend gesenkt werden.

Ganz nach dem Prinzip der Benutzerfreundlichkeit erfolgt der Anschluss der Sensoren auf flexible und generische Art und Weise. In den Sensorknoten sind die zahlreichen Ein- und Ausgänge des μC in Form von sechs analogen Eingängen, einem analogen Ausgang, sowie sieben digitalen Ein- und Ausgängen herausgeführt. Der Anschluss von Sensoren und Aktuatoren kann somit direkt oder mit Hilfe eines Aufsteckmoduls für die Anpassung von Spannungspegeln oder das Erlangen einer galvanischen Trennung erfolgen. Die Kommunikation zwischen Ebene null und Ebene eins basiert auf Spannungs-/Stromsignalen und digitalen logischen Zuständen. Wie viele und welche E/A benutzt werden, ist frei wählbar und vielfältig kombinierbar. Die gute Skalierbarkeit der Sensorknoten bleibt sowohl innerhalb eines Knotens (beispielsweise bei Benutzung sämtlicher E/A) als auch bei der Kaskadierung mehrerer Sensorknoten an einem CAN-Bus jederzeit erhalten. Durch einfaches Plug-and-Play können bis zu 30 Sensorknoten an einem CAN-Bus betrieben werden.

Trotz der geringen Rechenleistung und eingeschränkter Hardwareressourcen werden Sicherheitsaspekte bei der Datenübertragung berücksichtigt. Sämtliche Knoten authentisieren sich, und alle Nachrichten erhalten zur Integritätssicherung einen eindeutigen Fingerabdruck. Somit wird es potenziellen Angreifern erschwert, falsche Nachrichten in das System einzuschleusen bzw. durch so genannte Replay-Angriffe abgehörte Szenarien nachzuahmen. Bewusst wurde auf dieser Ebene auf rechenintensive Verschlüsselungsverfahren verzichtet, um schnelle Reaktionszeiten des Systems zu garantieren und die Hardwareressourcen zu minimieren. Da der CAN-Bus nur lokal Einsatz findet, ist Vertraulichkeit der Datenübertragung nicht notwendig.

In sämtlichen Ebenen kommt der "Keyed-Hash Message Authentication Code" (HMAC [6]) zum Einsatz. Zur Berechnung der benötigten Hashfunktion wird SHA-1 [7] benutzt. Eine fortlaufende 16-Bit breite Transaktionsnummer sowie ein vier Byte langer Fingerabdruck schützen das System vor Angreifern. Bedingt durch das einfach gehaltene und schichttransparente Protokoll besteht die Möglichkeit, dass sich ab einem gewissen Zeitpunkt eine Kombination aus Transaktionsnummer und Fingerabdruck für eine gleiche Nachricht wiederholen könnte. Um diesen potenziellen Schwachpunkt zu beseitigen, erfolgt die Kommunikation in Sitzungen, wobei die Sitzungsschlüssel in regelmäßigen Zeitabständen aktualisiert werden.

Die Zuverlässigkeit des Systems auf Ebene eins basiert auf den ausgefeilten Fehlererkennungsalgorithmen des CAN-Standards. Erzeugt ein CAN-Knoten permanent falsche und ungültige Nachrichten, wird er automatisch vom CAN-Bus ausgeklinkt. Des Weiteren ist CAN eine kollisionsfreie Übertragungstechnologie. Somit gehen keine Nachrichten verloren.

²<http://www.microchip.com>

Masterknoten: Ebene zwei

Der Masterknoten auf Ebene zwei nutzt den leistungsfähigeren Mikrocontroller ATmega644 von Atmel. Mit 64 KB Flash-Speicher und 4 KB Arbeitsspeicher erfüllt er die an ihn gestellten Aufgaben.

Der Hauptaufgabenbereich des Masterknotens besteht darin, die Nachrichten zwischen den Ebenen eins und drei bidirektional weiterzureichen. Auf der einen Seite hängt der Masterknoten am CAN-Bus und an der anderen Seite erfolgt die Datenübertragung über Ethernet. Im Prinzip fungiert er als Vermittler zwischen den Nachrichten vom CAN-Bus und den Ethernet-Paketen. Um den Overhead bei den viel größeren Ethernet-Paketen so gering wie möglich zu halten, werden mehrere CAN-Nachrichten in einen Ethernet-Frame gepackt. Erst wenn das Paket voll ist, ein interner Timeout abgelaufen ist oder eine Nachricht hoher Priorität verschickt werden muss, wird das Paket versandt. Um zeitkritische Nachrichten nicht zu verzögern, ist es möglich, wichtige Sensordaten oder Steuerbefehle mit einem Prioritäts-Bit zu kennzeichnen. Diese Nachrichten werden dann so schnell wie möglich weitergereicht. Die Sicherheit auf dieser Ebene wird ebenfalls, wie schon weiter oben erklärt, mit HMAC und SHA-1 erreicht.

Bei dem CAN-Controller handelt es sich um den schon erwähnten MCP 2515. Als Ethernet-Controller kommt der ENC 28J60 von Microchip zum Einsatz. Mit einem zusätzlichen PoE-Controller kann die Stromversorgung auf dieser Ebene über Power-over-Ethernet erfolgen. Dies minimiert den Verkabelungsaufwand.

Ein weiteres Merkmal ist die Berücksichtigung des Aspekts der Verfügbarkeit im Systemdesign. Bei Ausfall höherer Schichten oder ihrer Anbindung, z.B. bei Ausfall des Ethernet-Netzes, übernimmt ein einfaches Notfallprogramm wichtige Teilaufgaben des Systems und stellt somit den Erhalt einer minimalen Funktionalität sicher. Um das Schichtenmodell nicht zu verletzen, erfolgt die Implementierung dieses Notfallprogrammes als unabhängige Softwarekomponente innerhalb des Masterknotens. Sobald die Ethernet-Verbindung ausfällt, kümmert sich dieses Notfallprogramm um einen minimalen Teilaufgabenbereich der höheren Schicht. Im Rechenzentrum könnte das zum Beispiel bedeuten, dass auch bei Ausfall des Ethernet-Netzes eine existenzielle Reaktion, z.B. das Herunterfahren von Servern bei Überschreitung eines maximalen Temperaturwertes, erhalten bleibt.

Gebäudeknoten, Gateways und Anwendungen: Ebenen drei und vier

Als Gebäudeknoten kann ein ganz konventionelles, stromsparendes eingebettetes System benutzt werden. Auf diesem kleinen Rechner läuft ein kompaktes Betriebssystem mit einem geeigneten Protokollstapel. Der Gebäudeknoten ist für die Verbindung der Ebenen zwei und vier verantwortlich. Eine auf dem System installierte Middleware ermöglicht es, mit Hilfe von geeigneten Steuerbefehlen Daten aus dem Sensornetz zu erlangen und gewünschte Aktionen auszuführen. Beispielsweise kann das Linux-Betriebssystem zum Einsatz kommen. Sehr hohe Zuverlässigkeit kann durch die doppelte Ausführung dieses Gebäudeknotens erreicht werden. Bei Ausfall einer Einheit bleibt das System dank der anderen Einheit voll funktionsfähig.

Noch eine Ebene höher ist der Steuerungsdienst angesiedelt. Er läuft im Leitstand eines Campus. Auf dieser Ebene können Dienste als Schnittstelle zu unterschiedlichsten Proto-

kollen und anderen Anwendungen betrieben werden, z.B. Webservices oder SNMP. Auf dem gleichen oder anderen Rechnern können graphische Benutzeroberflächen und komplexe Steuerungen implementiert werden. Auch eine Anbindung an das Internet kann erfolgen. Als Verschlüsselungsverfahren kann hier das klassische SSL/TLS eingesetzt werden.

Weitere Informationen zu den Möglichkeiten und der Architektur der Ebenen drei und höher finden sich in [1] und der Diplomarbeit [8]. Das nächste Kapitel befasst sich mit der Bewertung des umgesetzten Versuchnetzes und den somit gesammelten Erfahrungen.

4 Erfahrungen mit Prototypen

Zum Testen und Sammeln von Erfahrungen wurde ein Testaufbau der unteren Ebenen des Netzwerkes bestehend aus drei Sensorknoten und einem Masterknoten vorgenommen. Es zeigte sich, dass der Aufbau wie geplant funktioniert und die Anforderungen gut erfüllt werden.

Zur Erhöhung der Praxistauglichkeit wurden nach Abschluss der Entwicklung des ersten Prototypen noch folgende Verbesserungen bei der Sensorknoten-Hardware durchgeführt: Mittels eines Umschalters ist es nun möglich, die Stromversorgung der einzelnen Knoten ein- und auszuschalten. Somit können die Knoten während der Installation und Konfiguration bequem nacheinander aktiviert werden, da ein gleichzeitiges Aktivieren mehrerer neuer Knoten nicht erlaubt ist. Um die Fehlersuche zu vereinfachen, wurden jeweils zwei Status-LEDs angebracht. Diese ermöglichen eine schnelle Rückmeldung über den Betriebszustand (Installationsmodus, Konfigurationsmodus oder Betriebsmodus) des jeweiligen Sensorknotens. Zur Spannungsreglung wurden anfangs Regler vom Typ 7805 eingesetzt. Wegen deren hohen Verlustleistung werden in der neuen Hardware-Revision ausschließlich "Low-Drop"-Spannungsregler verwendet. Dadurch konnte die Spannungsversorgung des Netzes um 2 V gesenkt werden. Ein neues doppelseitiges Layout erleichtert die Bestückung der Platinen. Zusätzlich erfüllen nun auch sämtliche Hardware-Komponenten die RoHS-Richtlinie.

Durch die modulare Programmierung konnten die Anforderungen Schritt für Schritt umgesetzt und einzeln getestet werden. Beispielsweise wurde empirisch ermittelt, dass das Authentifizierungsverfahren in den Sensorknoten die Reaktionszeit des Netzes nicht beeinträchtigt. Von besonderem Interesse waren auch die Untersuchungen des nebenläufigen Datenverkehrs. Um die Monopolisierung des Busses durch einen einzelnen Teilnehmer zu vermeiden, wurden Verhaltensregeln entwickelt und evaluiert. Der Masterknoten erhält die höchste Priorität und erhält somit Vorrang für seine Nachrichten. Niedrigere Prioritäten werden von den angeschlossenen Sensorknoten genutzt. Die gewählte Busgeschwindigkeit ermöglicht die Übertragung von durchschnittlich 1000 Nachrichten pro Sekunde. Das Sendepotenzial eines Sensorknotens liegt hingegen bei maximal 30 Nachrichten pro Sekunde. Das heißt, dass auch im Worst-Case zwischen zwei Nachrichten desselben Knotens die anderen 29 Teilnehmer jeweils auch eine Nachricht verschicken können.

Auf der Ethernet-Seite wurde die Belastungsgrenze des Masterknotens erprobt. Der Mas-

terknoten puffert bis zu 150 CAN-Nachrichten. Ist der Puffer voll, werden die ältesten Nachrichten überschrieben. Während den Testphasen wurde der Master absichtlich mit Paketen geflutet. Die Ethernet-Kommunikation war dadurch erwartungsgemäß schwer beeinträchtigt (dies ist systembedingt), jedoch erwiesen sich der Masterknoten und das darunter liegende CAN-Netzwerk als absolut stabil. Nach Einstellen des Denial-of-Service-Angriffes funktionierte das Netzwerk sofort wieder fehlerfrei.

5 Zusammenfassung

Dieser Beitrag beschäftigte sich mit der Verwirklichung eines Sensorknotennetzes zum Messen, Steuern und Schalten. Zu Beginn wurde eine Vielzahl von Anforderungen identifiziert. Dazu gehören Energieverbrauch und Anschaffungskosten genauso wie vielfältige andere Punkte wie Sicherheit, Skalierbarkeit, Zuverlässigkeit und letztendlich auch Benutzerfreundlichkeit in Installation und Betrieb. Diese Anforderungen können nur durch eine Schichtenarchitektur zufriedenstellend umgesetzt werden. Durch das Schichtenmodell konnte die Komplexität und "Intelligenz" möglichst weit weg von unteren Ebenen heraus in Richtung höherer Ebenen verlagert werden.

Es wurde ein Prototyp entwickelt, der sich hervorragend zum Messen und Schalten in Rechenzentren einsetzen lässt. Der Prototyp zeigt, dass die identifizierten Anforderungen auch praktisch in ein funktionierendes Produkt umgesetzt werden können. Das Ergebnis ist ein kostengünstiges, benutzerfreundliches, zuverlässiges, generisches und sicheres Sensorknotennetz, das flexibel an unterschiedlichste Anforderungen anpassbar ist.

Literatur

- [1] Dirk Henrici, Patric de Waha, Paul Müller: Bridging the Gap Between Pervasive Devices and Global Networks. International Symposium on Collaborative Technologies and Systems, Workshop on Distributed Collaborative Sensor Networks, 2008
- [2] SCADA. URL: de.wikipedia.org/wiki/Supervisory_Control_and_Data_Acquisition (abgerufen 22.01.2009)
- [3] LCN. URL: <http://www.lcn.de/> (abgerufen 22.01.2009)
- [4] BTNode. URL: <http://www.btnode.ethz.ch/> (Stand 2007; abgerufen 22.01.2009)
- [5] Michel Steichen: Entwicklung kosteneffizienter Sensorknoten mit sicherer Anbindung an offene Netze. Diplomarbeit, TU Kaiserslautern, 2009
- [6] H. Krawczyk, M. Bellare, R. Canetti: RFC 2104 - HMAC: Keyed-Hashing for Message Authentication
- [7] D. Eastlake, P. Jones: RFC3174 - US Secure Hash Algorithm 1 (SHA1)
- [8] Patric de Waha: Sichere und verlässliche Kommunikation zwischen Low-Cost-Devices und PCs. Diplomarbeit, TU Kaiserslautern, 2007