Schlussbericht

Secure, Mobile visual sensor networks ArchiTecture

SMART

Förderprogramm	ARTEMIS-2008-1
Förderer	BMBF (83,3 %) ARTEMIS-JU (16,7 %)
Förderkennzeichen	01IS09001C
Projektlaufzeit	01.03.2009 - 31.12.2012
Stand	14.05.2013
Autoren	Oliver Stecklina Frank Vater Stephan Kornemann Prof. Dr. Peter Langendörfer
Zuwendungsempfänger	IHP GmbH

Das diesem Bericht zugrundeliegende Vorhaben wurde durch Mittel des Bundesministeriums für Bildung und Forschung (BMBF) unter dem Förderkennzeichen 01IS09001C gefördert. Die inhaltliche Verantwortung dieses Berichtes liegt bei den Autoren.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1. Kurze Darstellung

1.1. Aufgabenstellung

Das Ziel des Verbundprojektes SMART war die Entwicklung einer neuen Klasse von Sensorknoten, deren Einsatzmöglichkeiten die von bisher verfügbaren Systemen weit überschreitet. Mit dieser Klasse von drahtlosen Sensorknoten sollen neue Anwendungsgebiete erschlossen und damit die Wettbewerbsfähigkeiten Europas verbessert werden.

Die zu entwickelnden drahtlosen Sensorknoten zeichnen sich insbesondere dadurch aus, dass sie sich an verändernde Umgebungsbedingungen flexibel anpassen können und damit ein höheres Maß an Funktionalität bieten. Sie bieten Funktionen für eine sichere und energieeffiziente Übertragung von Sensor- und Videodaten. Mittels einer neuartigen Software-Architektur sollen das Rekonfigurieren und der Aufbau von robusten Netzwerkstrukturen vereinfacht werden.

Für die Umsetzung des Systems waren zwei alternative Varianten vorgesehen. Zum einen sollen auf FPGAs basierende Systeme und zum anderen anwendungsspezifische Prozessoren zum Einsatz kommen. Beim FPGA sollen durch die Verwendung einer partiellen Rekonfiguration die Konfigurationszeit und der Energieverbrauch signifikant reduziert werden. Der ASIP soll durch die Verwendung von anwendungsspezifischen Befehlssätzen an die sich verändernden Umgebungsbedingungen angepasst werden. Zum Validieren der Ansätze sollen Sensorknoten in beiden Varianten implementiert werden.

Das IHP ist wesentlich für die Fabrikation des RASIP verantwortlich. Hierfür soll die im IHP vorhandene Fertigungstechnik genutzt werden. Darüber hinaus verfügt das IHP über IPs, die im Rahmen des Projektes im FPGA als auch im RASIP verwendet werden können. Hierbei handelt es sich um IPs für AES, SHA1, eine Elliptic Curve (EC) Unit zur Punktmultiplikation und einen 16-bit-Mikroprozessor.

1.1. Vorhabensvoraussetzungen

Das IHP ist ein Forschungsinstitut der Leibniz-Gemeinschaft und beschäftigte im Jahr 2009 ca. 220 Mitarbeiter aus zwanzig verschiedenen Nationen (heute ca. 300 Mitarbeiter). Der Schwerpunkt der Forschung am IHP liegt in der Entwicklung von Hochfrequenzschaltungen und deren Anwendung. Das Institut ist in die vier Abteilungen Materialforschung, Technologie, Schaltungsentwurf und System Design gegliedert und verfolgt erfolgreich einen vertikalen Forschungsansatz, in dem die einzelnen Abteilungen durch die Nutzung von Synergien effizienter und schneller Forschungsthemen bearbeiten können.

Der Reinraum des IHP verfügt über zwei Pilotlinien, die über die Multiprojektwafer auch für Kleinstserien genutzt werden können. Auf den Linien können ICs in der IHP-eigenen SiGe-BiCMOS-Technologie mit Strukturgrößen von 0,13 µm und 0,25 µm gefertigt werden. Darüber hinaus ist in 0,25 µm die Fertigung von nichtflüchtigen Speichern (Flash) möglich, so dass Low-Power-Mikrocontroller mit integriertem Datenspeicher gefertigt werden können.

Die Abteilung System Design verfügt zu Beginn des SMART-Projektes bereits über mehrjährige Erfahrungen beim Design bei der Fertigung von digitalen Schaltungen. So wurden im Rahmen von anderen Forschungsprojekten (Basuma, TSN, TANDEM) bereits erfolgreich Mikroprozessoren (MIPS,

Leon2) gefertigt. Zusätzlich wurden IP-Cores für AES, SHA-1 und ECC entwickelt und im Rahmen von wissenschaftlichen Publikationen präsentiert.

Durch eine enge Zusammenarbeit mit Universitäten und Fachhochschulen in Berlin und Brandenburg bestehen sehr gute Möglichkeiten zur wissenschaftlichen Verwertung der Ergebnisse in Forschung und Lehre.

1.2. Planung und Ablauf des Vorhabens

Gemäß den Vorgaben aus dem Projekthandbuch war das IHP primär für die Fertigung des RASIP, die Untersuchung und Implementierung von Anti-Tamper-Mechanismen und die Bereitstellung von Crypto-Funktionen verantwortlich. Darüber hinaus war eine enge Zusammenarbeit mit den Partnern (Lippert, UPM, Nanosens und TSI) zur Erstellung der Sensorknotenhardware vorgesehen.

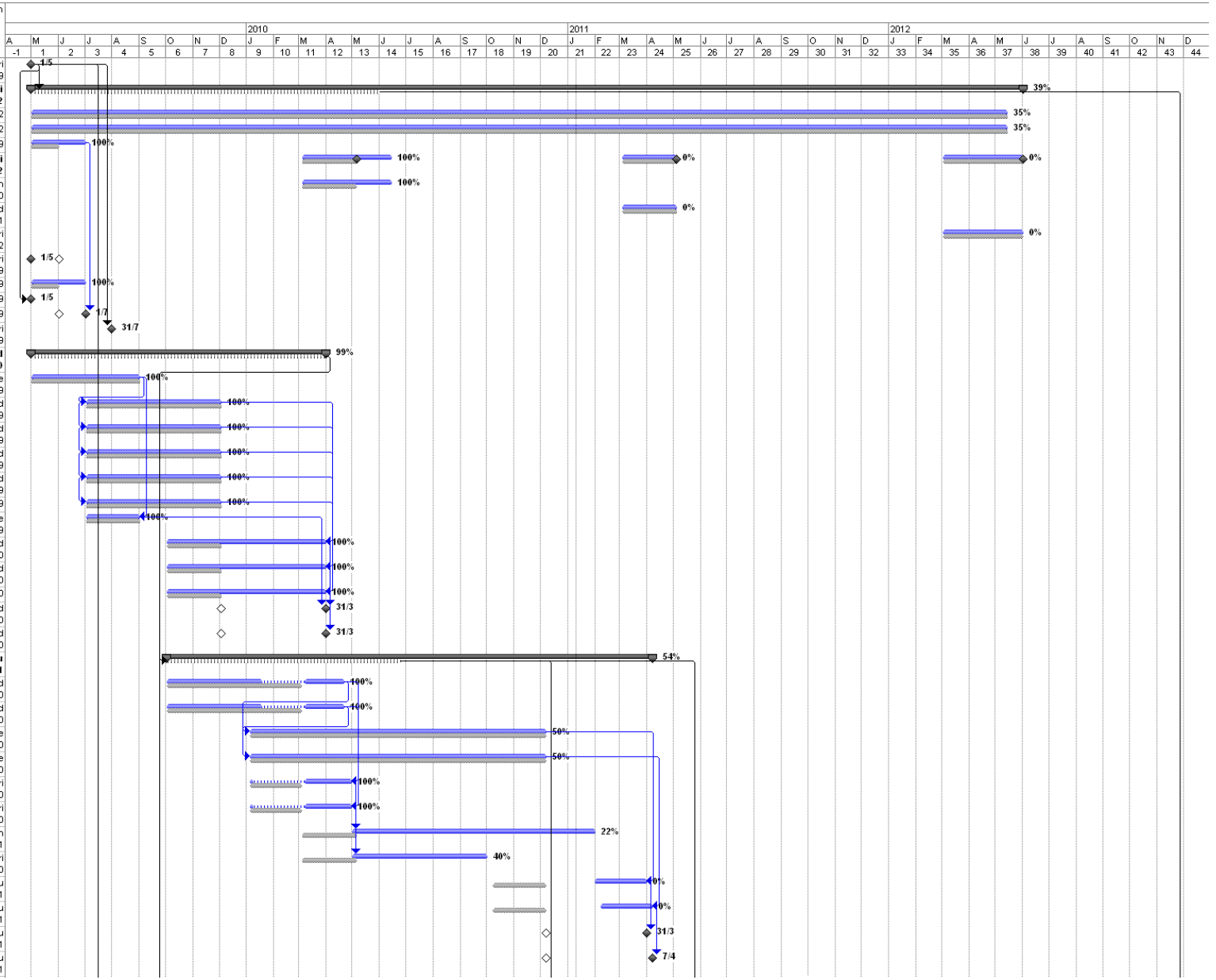
Bezogen auf die einzelnen Arbeitspakete verteilten sich die Aufgaben wie folgt:

- **WP1**: Planung und Koordinierung der eigenen Projektaufgaben. Mitwirkung am Technical Board.
- **WP2**: Untersuchung und Erarbeitung von möglichen Anwendungsszenarien sowie deren Anforderungen für die Umsetzung des geplanten Sensorknotens. Planung und Entwurf der High-Level-Architektur des Sensorknotens.
- **WP3**: Unterstützung bei der Entwicklung der Low-Power Sensing und Data Compression Architektur.
- **WP4**: Erweiterung der vorhandenen Crypto-Einheiten zur Reduzierung des Energieverbrauchs und der Nutzbarkeit in rekonfigurierbaren Hardware-Blöcken. Entwicklung von Anti-Tamper-Mechanismen zum Schutz gegen physische Angriffe und Side-Channel-Attacken.
- **WP5**: Fertigung des RASIP und eines konfigurierbaren Sensorknotens.
- **WP6**: Mitwirken an der Integration und Validierung der Ergebnisse durch Spezifikation der System- und Leistungsparameter und Anpassung der Systemumgebung an die IHP-spezifischen Parameter.
- **WP7**: Publikation von Forschungsergebnissen und Präsentation auf Konferenzen und Fachausstellungen.

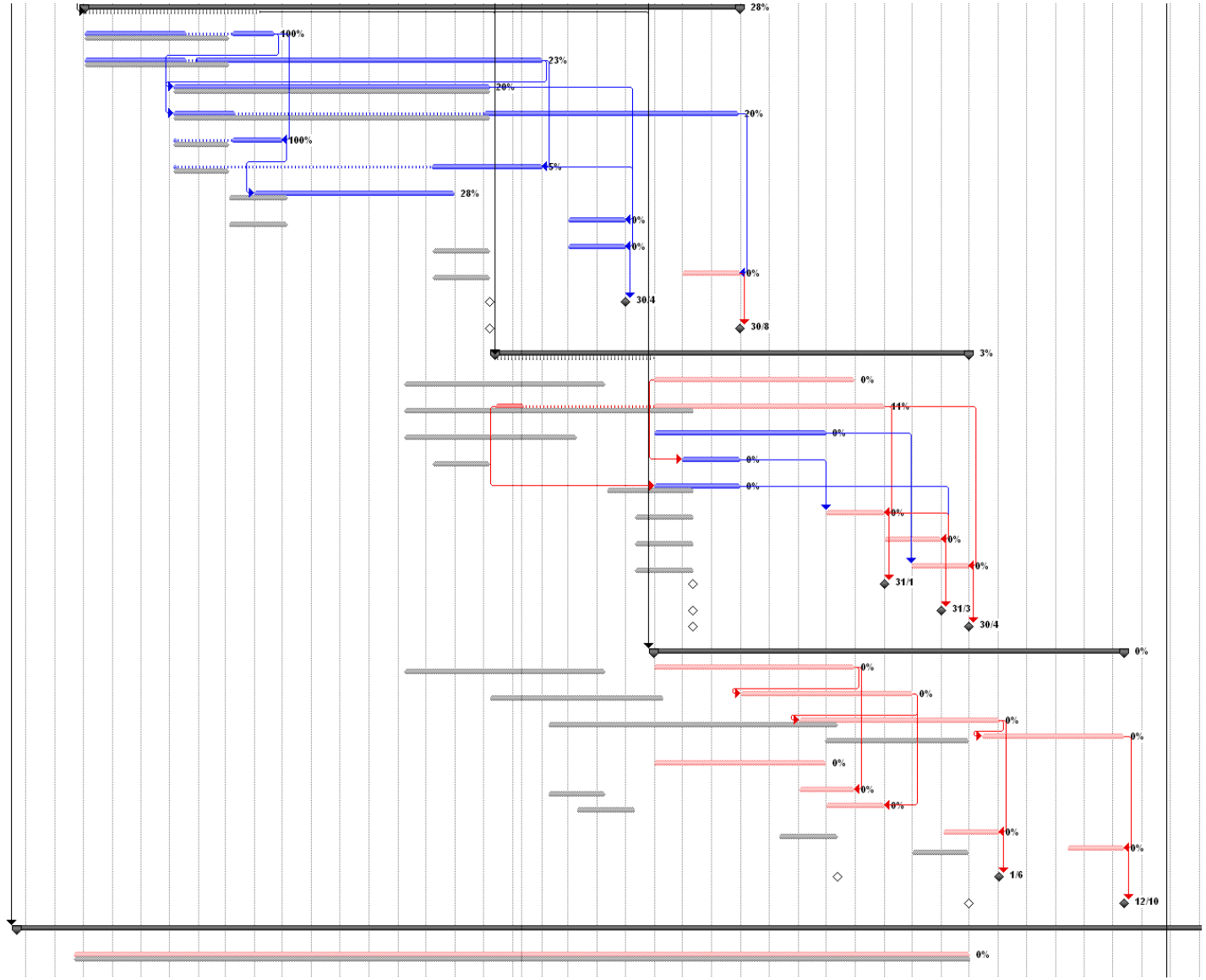
Durch die Verzögerungen bei der Förderung von einzelnen europäischen Partnern konnte der ursprüngliche Zeitplan, siehe Abbildung 1, nicht umgesetzt werden. Bereits frühzeitig war eine 6-monatige Verzögerung vorhanden, die im Laufe des Projektes nicht aufgeholt werden konnte. Zum Erreichen der gesetzten Projektziele wurde aus diesem Grund eine Verlängerung des Projektzeitraumes bis zum Ende des Jahres 2012 beantragt und vom Projektträger bewilligt. Entsprechend der Projektverlängerung wurden Aufgaben später bekommen, so dass Abhängigkeiten zwischen den einzelnen Aufgaben eingehalten und die Verlängerung kostenneutral durchgeführt werden konnte.

Durch den vorzeitigen Ausstieg von Lippert Ende 2011 musste die Fertigung der Sensorknoten durch Projektpartner übernommen werden, so dass die Budget- und Zeitplanung entsprechend angepasst werden musste. Dies führte zu einer zusätzlichen Verzögerung bei der Fertigung der Sensorknoten, so dass die Arbeiten am WP6 erst sehr spät bzw. auf alternativen Hardwarekomponenten durchgeführt werden musste.

ID	WP-TASK-DELIV-MILEST. No.	Task Name	Duration	Start	Finish
1		PROJECT START	0 mo	Fri 1/5/09	Fri 1/5/09
2	**WP1**	**Project Coordination & Management**	36,64 mo	**Fri 1/5/09**	**Fri 1/6/12**
3	T1.1	Project Administration	36 mo	Fri 1/5/09	Wed 4/5/12
4	T1.2	Project Quality Assurance	36 mo	Fri 1/5/09	Wed 4/5/12
5	D1.1	Project Handbook	2 mo	Fri 1/5/09	Fri 1/7/09
6	**D1.2.X**	**Periodic Progress Report**	26,64 mo	**Fri 5/3/10**	**Fri 1/6/12**
7	D1.2.1	Periodic Progress Report 1	3,27 mo	Fri 5/3/10	Mon 4/6/10
8	D1.2.2	Periodic Progress Report 2	2 mo	Fri 4/3/11	Wed 4/5/11
9	D1.2.3	Periodic Progress Report 3	3 mo	Fri 2/3/12	Fri 1/6/12
10	D1.3	SMART e-mail reflector and FTP	0 mo	Fri 1/5/09	Fri 1/5/09
11	D1.4	SMART WEB site	2 mo	Fri 1/5/09	Fri 1/7/09
12	M1.1	Kick-off meeting	0 mo	Fri 1/5/09	Fri 1/5/09
13	M1.2	Project handbook	0 mo	Fri 1/7/09	Fri 1/7/09
14	M1.3	Project cost, time and scope baseline	0 mo	Fri 31/7/09	Fri 31/7/09
15	**WP2**	**Requirements Capture & Specifications**	10,86 mo	**Fri 1/5/09**	**Wed 1/3/10**
16	T2.1	Identification of WSN high-level requirements and	4 mo	Fri 1/5/09	Tue 1/9/09
17	T2.2	Power Requirements capture	5 mo	Thu 2/7/09	Wed 1/12/09
18	T2.3	Security Requirements capture	5 mo	Thu 2/7/09	Wed 1/12/09
19	T2.4	Video Requirements capture	5 mo	Thu 2/7/09	Wed 1/12/09
20	T2.5	Reconfigurability and programmability requirements	5 mo	Thu 2/7/09	Wed 1/12/09
21	T2.6	High-level node architecture	5 mo	Thu 2/7/09	Wed 1/12/09
22	D2.1	Set of representative WSN application Domains	2 mo	Thu 2/7/09	Tue 1/9/09
23	D2.2	Power, Programming and Reconfigurability	5,86 mo	Fri 2/10/09	Wed 1/3/10
24	D2.3	Security and Video Requirements	5,86 mo	Fri 2/10/09	Wed 1/3/10
25	D2.4	High-level node architecture	5,86 mo	Fri 2/10/09	Wed 1/3/10
26	M2.1	Applications and System Requirements	0 mo	Wed 2/12/09	Wed 1/3/10
27	M2.2	Node's Complete Architecture	0 mo	Wed 2/12/09	Wed 1/3/10
28	**WP3**	**Power-aware Reconfiguration and Data-Compression**	17,95 mo	**Fri 2/10/09**	**Thu 7/4/11**
29	T3.1	Design of sensing and reconfiguration mechanisms	5 mo	Fri 2/10/09	Wed 1/4/10
30	T3.2	Design of data compression subsystems	5 mo	Fri 2/10/09	Wed 1/4/10
31	T3.3	Implementation of real-time partial reconfiguration and	11 mo	Mon 4/1/10	Tue 1/12/10
32	T3.4	Implementation of hardware-based data	11 mo	Mon 4/1/10	Tue 1/12/10
33	D3.1	Micro-Architecture of Reconfiguration and Sensing	2 mo	Mon 4/1/10	Fri 4/4/10
34	D3.2	Micro-Architecture of Data compression systems	2 mo	Mon 4/1/10	Fri 4/4/10
35	D3.3	Initial Version of Reconfiguration and Sensing	8,93 mo	Fri 30/4/10	Mon 1/1/11
36	D3.4	Initial Version of Data Compression Systems	5,02 mo	Fri 30/4/10	Fri 1/10/10
37	D3.5	Implementation of Reconfiguration and Sensing	2 mo	Mon 31/1/11	Thu 1/3/11
38	D3.6	Implementation of Data compression systems	2 mo	Mon 7/2/11	Thu 7/4/11
39	M3.1	Fully Implemented Reconfiguration and Sensing	0 mo	Thu 31/3/11	Thu 1/3/11
40	M3.2	Fully Implemented Data Compression Systems	0 mo	Thu 7/4/11	Thu 7/4/11



41	**WP4**	**Security and Video Subsystems Design and**	22,64 mo	**Fri 2/10/09**	**Tue 0/8/11**
42	T4.1	Design of low-power security systems	5 mo	Fri 2/10/09	Wed 1/4/10
43	T4.2	Design of Video Compression sub-systems	15,41 mo	Fri 2/10/09	Mon 1/1/11
44	T4.3	Implementing resistant to side-channel attack security	11 mo	Mon 4/1/10	Tue 1/2/10
45	T4.4	Implementing Video Compression submodules	11 mo	Mon 4/1/10	Mon 9/8/11
46	D4.1	Micro-Architecture of Encryption and Authentication	2 mo	Mon 4/1/10	Fri 0/4/10
47	D4.2	Micro-Architecture of Video compression systems	4 mo	Mon 4/1/10	Mon 1/1/11
48	D4.3	Initial Version of Encryption and Authentication Systems	6,95 mo	Wed 31/3/10	Sat /10/10
49	D4.4	Initial Version of Video Compression Systems	2 mo	Tue 1/3/11	Sat 0/4/11
50	D4.5	Implementation of Encryption and Authentication Systems	2 mo	Tue 1/3/11	Sat 0/4/11
51	D4.6	Implementation of Video compression systems	2 mo	Thu 30/6/11	Tue 0/8/11
52	M4.1	Fully Implemented Encryption and Authentication Systems	0 mo	Sat 30/4/11	Sat 0/4/11
53	M4.2	Fully Implemented Video Compression System	0 mo	Tue 30/8/11	Tue 0/8/11
54	**WP5**	**RASIP and Node Implementation and**	16,38 mo	**Mon 3/12/10**	**Mon 0/4/12**
55	T5.1	Middleware, Configuration and Management tools	7 mo	Tue 31/5/11	Fri /12/11
56	T5.2	Design and implementation of RASIP	9 mo	Mon 3/12/10	Tue 1/1/12
57	T5.3	Design and implementation of the hardware subsystem of	6 mo	Tue 31/5/11	Wed /11/11
58	D5.1	Initial Version of SMART Middleware, Configuration	2 mo	Thu 30/6/11	Tue 0/8/11
59	D5.2	Preliminary Version of HDL and Synthesis Report for	3 mo	Tue 31/5/11	Tue 0/8/11
60	D5.3	SMART Middleware, Configuration and	2 mo	Thu 1/12/11	Tue 1/1/12
61	D5.4	SMART RASIP RTL/Silicon Chip	2 mo	Tue 31/1/12	Sat 1/3/12
62	D5.5	SMART hardware node	2 mo	29/2/12	0/4/12
63	M5.1	Fully Implemented Middleware Configuration and	0 mo	Tue 31/1/12	Tue 1/1/12
64	M5.2	Fully Implemented RASIP	0 mo	31/3/12	1/3/12
65	M5.3	Fully implemented SMART node	0 mo	Mon 30/4/12	Mon 0/4/12
66	**WP6**	**Integration & Validation**	16,32 mo	**31/5/11**	**/10/12**
67	T6.1	Trials Specification & Set-up	7 mo	Tue 31/5/11	Fri /12/11
68	T6.2	Identification of the testing environment	6 mo	Wed 31/8/11	Thu 1/3/12
69	T6.3	SMART integration	7 mo	1/11/11	1/6/12
70	T6.4	Validation of the SMART platform	5 mo	Mon 14/5/12	Fri /10/12
71	T6.5	Initial SMART Integration	6 mo	Tue 31/5/11	Wed /11/11
72	D6.1	Trials specification & Setup	2 mo	1/11/11	1/2/11
73	D6.2	Test plan and demo scenarios for SMART platform	2 mo	Thu 1/12/11	Tue 1/1/12
74	D6.3	Integrated SMART platform	2 mo	3/4/12	1/6/12
75	D6.4	Validation of the SMART platform	2 mo	Tue 14/8/12	Fri /10/12
76	M6.1	Fully Integrated SMART platform	0 mo	Fri 1/6/12	Fri 1/6/12
77	M6.2	Complete evaluation accomplished	0 mo	Fri 2/10/12	Fri /10/12
78	**WP7**	**Exploitation & Dissemination**	44,82 mo	**Wed 22/7/09**	**Wed 1/5/13**
79	T7.1	Market Assessment & Analysis	31 mo	Mon 21/9/09	Tue 1/5/12



80	T7.2	Exploitation Plans and contributions to standards	29 mo	Mon 3/11/09	Wed 2/5/12
81	T7.3	Dissemination & Workshops	33 mo	22/7/09	2/5/12
82	**D7.1**	**Assessment and analysis of the ad-hoc &**	31,59 mo	**Fri 5/3/10**	**Thu 1/11/12**
83		Periodic Progress Report 1	8 mo	Fri 5/3/10	Fri 1/1/10
84		Periodic Progress Report 2	2 mo	Tue 6/9/11	Fri 1/11/11
85		Periodic Progress Report 3	2 mo	Mon 3/9/12	Thu 1/11/12
86	**D7.2**	**Strategic directions and scheduling for the**	25,64 mo	**Mon 6/9/10**	**Thu 1/11/12**
87		Periodic Progress Report 1	7,86 mo	Mon 6/9/10	Wed 4/5/11
88		Periodic Progress Report 2	2 mo	Tue 6/9/11	Fri 1/11/11
89		Periodic Progress Report 3	2 mo	Mon 3/9/12	Thu 1/11/12
90	**D7.3**	**Dissemination strategy**	29,73 mo	**3/5/10**	**1/11/12**
91		Periodic Progress Report 1	7,95 mo	Mon 3/5/10	Sat 1/1/11
92		Periodic Progress Report 2	2 mo	Tue 6/9/11	Fri 1/11/11
93		Periodic Progress Report 3	2 mo	Mon 3/9/12	Thu 1/11/12
94	M7.1	Dissemination and use plan	0 mo	1/5/13	1/5/13
95		**PROJECT COMPLETION**	0 mo	1/12/12	1/12/12

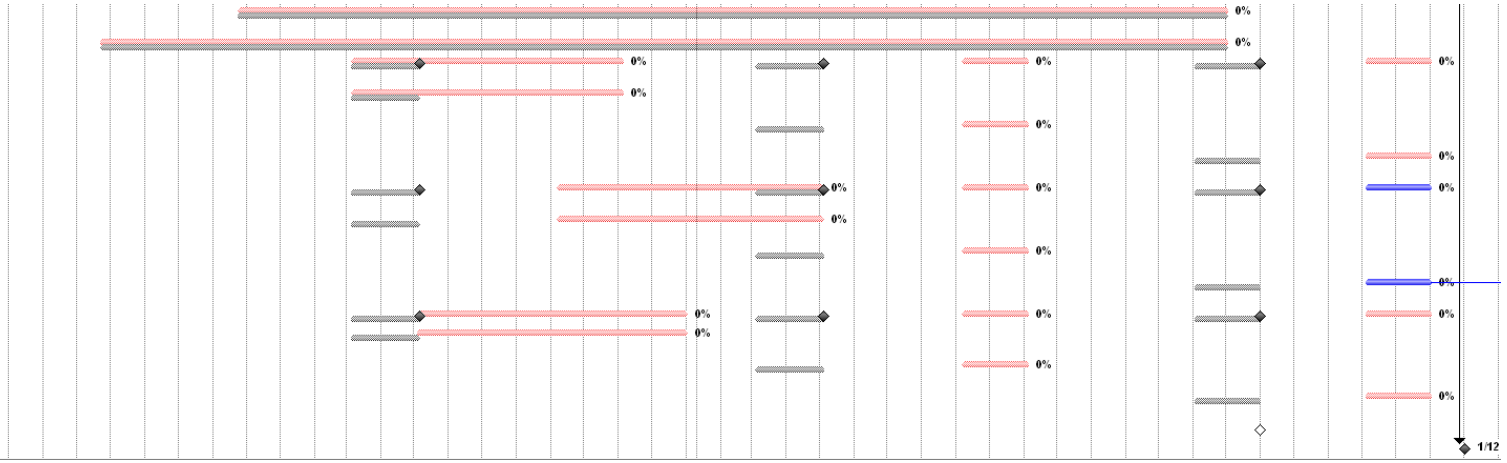


Abbildung 1: Zeitliche Planung des Vorhabens (Gantt-Chart) - ©SMART-Konsortium

1.3. Wissenschaftlicher und technischer Stand

Zum Zeitpunkt der Beantragung des Projektes SMART existierten bereits umfangreiche Forschungsergebnisse im Bereich der drahtlosen Sensornetze. Bestandteil dieser Ergebnisse sind eine Vielzahl von kommerziell verfügbaren Sensorknoten. Hiervon sind insbesondere die Knoten TelosB und MICAz von Bedeutung. Sie haben in der wissenschaftlichen Gemeinde eine weite Verbreitung und Akzeptanz gefunden und sind auf einen möglichst niedrigen Energieverbrauch ausgerichtet. Darüber hinaus sind der iMote von Intel und der SunSPOT von Sun Microsystems, jetzt Oracle, zu nennen. Diese Sensorknoten unterscheiden sich von den zuvor genannten durch den verwendeten Microcontroller und deren Energieverbrauch. Während TelosB und MICAz auf einen minimalen Energieverbrauch ausgerichtet sind und nur eine schwache Rechenleistungen erbringen, sind iMote und SunSPOT mit leistungsstärkeren Controllern ausgestattet. Für eine Realzeit-Kompression von Video- und Sensordaten sowie die Verschlüsselung des Datenverkehrs ist die Rechenleistung von TelosB und MICAz jedoch nicht ausreichend. iMote und SunSPOT weisen dagegen einen höheren Energieverbrauch auf, der für einen längeren Einsatz, wie er für SMART angestrebt wird, ungeeignet ist.

Außer in Sensorknoten werden Microcontroller in einer Vielzahl von technischen Geräten eingesetzt. Dementsprechend umfangreich ist die Menge der verfügbaren Typen. Hierbei werden die verschiedensten Leistungsklassen abgedeckt. Aus Kosten- und Effizienzgründen ist deren Feature-Set jedoch meist anwendungsspezifisch. Dynamische Anpassungen an sich verändernde Umgebungsbedingungen sind somit nicht möglich. Außerdem waren hardwarebasiert Crypto-Einheiten, wie sie für eine sichere Kommunikation benötigt werden, zum Zeitpunkt der Projektbeantragung nur sehr eingeschränkt verfügbar.

Die Kompression von Videodaten in Echtzeit ist auf Standardprozessoren und insbesondere auf Microcontrollern bis heute nicht möglich. Insbesondere moderne Verfahren mit einer hohen Kompressionsrate, wie z. B. H.264, sind komplex und bei einer Ausführung in Software sehr rechenintensiv. Eine Umsetzung dieser Algorithmen in Hardware ist für eine Echtzeitverarbeitung dringend notwendig. Zum damaligen Zeitpunkt existierten wenige ASICs, die hierzu in der Lage waren. Eine Implementierung für Low-Power-Geräte oder für rekonfigurierbare Hardware war nicht verfügbar.

Eine dynamische Anpassung von Hardware an sich verändernde Umgebungsbedingungen ist durch die Verwendung von konfigurierbaren Arrays, wie sie in CPLDs und FPGAs eingesetzt werden, seit den 90er Jahren des letzten Jahrhunderts möglich. Hierbei war es jedoch notwendig, das gesamte System neu zu konfigurieren. Eine derartige Rekonfiguration ist zeit- und energieintensiv. Darüber hinaus ist hierfür ein zusätzlicher Controller notwendig, was die Komplexität des Systems weiter erhöht. In ASICs war eine Systemanpassung durch den Einsatz von VLIW-Prozessoren mit komplexen Befehlsdekodierwerken möglich, z. B. Transmeta Crusoe. Diese Systeme waren jedoch aufgrund ihres Energieverbrauches für den Einsatz in Sensorknoten ungeeignet.

1.4. Zusammenarbeit mit anderen Stellen

Das SMART-Konsortium setzt sich aus elf Partnern aus fünf verschiedenen Nationen zusammen. In der ersten Phase des Projektes wurden Anwendungsszenarien und die Anforderungen an den SMART-Sensorknoten gemeinsam erarbeitet. Anschließend erfolgte die Umsetzung der einzelnen Komponenten jeweils bei den Projektpartnern oder in enger Kooperation einzelnen Partner.

Das IHP war für die Fabrikation des RASIP und das Design des Sensorknotens zuständig. Hierzu wurde in der zweiten Phase des Projektes eng mit der TU Braunschweig und Lippert zusammengearbeitet. Die TUBS wurde bei der Implementierung des HiveCore in der IHP-eigenen Technologie unterstützt. Darüber hinaus wurde eine Schnittstelle für den HiveCore und den IHP430x implementiert. Nach dem vorzeitigen Ausstieg von TUBS und Lippert und dem damit verbundenen Ende der RASIP-Entwicklungen hat das IHP in enger Zusammenarbeit mit UPM die Umsetzung der FPGA-Variante vorangetrieben.

In der dritten und letzten Projektphase wurden in gemeinsamen Integrationsmeetings die Entwicklungen zusammengeführt und der Aufbau des Demonstrators durchgeführt.

Bereits frühzeitig hat das IHP zusammen mit der UPM einzelne Projektergebnisse auf Konferenzen, Ausstellungen und in Journals präsentiert. So wurde z. B. gemeinsam mit der UPM auf der Secon 2009 ein rekonfigurierbares ECC-Modul präsentiert.

2. Verwendung der Zuwendung und erzielte Ergebnisse

2.1. Erzielte Ergebnisse aus den einzelnen Arbeitspaketen

2.1.1. Workpackage 1 - Project Coordination & Management

Im WP 1 sind alle Arbeiten zum Projektmanagement zusammengefasst. Das IHP hat hierzu Zuarbeiten zu den Deliverables 1.3.x geliefert.

2.1.2. Workpackage 2 - Requirements Capture & Specifications

Das WP2 befasste sich mit der Analyse und der Definition von Anforderungen für einen konfigurierbaren Sensorknoten. Die Ergebnisse sind anschließend in die WPs 3 – 5 eingeflossen.

Aufgabe 2.1 (Identification of WSN high-level requirements and application selection): Das IHP hat eine Analyse von Anwendungsszenarien für den SMART-Sensorknoten durchgeführt. Hierbei wurden zwei Szenarien, Monitoring of Industrial plants und Body Area Networks, genauer betrachtet und beschrieben. Die Ergebnisse sind im Deliverable 2.1 zusammengefasst.

Aufgabe 2.2 (Power requirements capture): Um den Energiebedarf des SMART-Sensorknotens genauer bestimmen zu können, führte das IHP am IHPNode Messungen des Stromverbrauchs durch. Der IHPNode verfügt ebenfalls über einen 16-bit-Microcontroller und über ein IEEE 802.15.4 kompatiblen Radio-Transceiver und ist damit ähnlich der geplanten SMART-Plattform. Die Ergebnisse der Messungen wurden im Deliverable 2.2 veröffentlicht.

Aufgabe 2.3 (Security requirements capture): Ausgehend von den Beschreibungen von möglichen Anwendungsszenarien wurde eine Analyse der erforderlichen Sicherheitsmechanismen für den SMART-Sensorknoten durchgeführt. Eine Zusammenfassung der Ergebnisse ist im Deliverable 2.3 zu finden.

Aufgabe 2.4 (Video requirements capture): Analog zur Analyse der Sicherheitsmechanismen wurde eine Untersuchung der Videofunktionen durchgeführt. Hierbei wurden durch das IHP insbesondere die notwendigen Schnittstellen zur Übertragung der Daten betrachtet. Die Ergebnisse sind in das Deliverable 2.4 eingeflossen.

Aufgabe 2.5 (Reconfigurability und programmability requirements capture): Das IHP verfügte zum Zeitpunkt des Projektestarts bereits über HDL-Implementierung für die Crypto-Algorithmen AES, SHA1 und ECC. Im Rahmen von SMART wurden diese Implementierungen zunächst hinsichtlich ihrer Eignung für ein rekonfigurierbares Design genauer untersucht. Darüber hinaus wurden die Anforderungen für das Design des IHP430x verfeinert. Hierbei wurde insbesondere eine Auswahl der notwendigen Peripherien analysiert. Die Ergebnisse der Untersuchungen wurden im Deliverable 2.3 veröffentlicht.

Aufgabe 2.6 (High-level node architecture): Ausgehend von den Anforderungen an den SMART-Sensorknoten und insbesondere an die RASIP-Variante hat das IHP an der Definition der High-Level-Architektur der SMART-Sensorknotenvarianten mitgewirkt. Die Ergebnisse sind in das Deliverable 2.4 eingeflossen.

2.1.3. Workpackage 3 - Power-aware Reconfiguration and Data-Compression Systems

Das WP3 befasste sich mit dem Entwurf und der Implementierung der Sensor- und Rekonfigurationsmechanismen für den SMART-Sensorknoten. Das IHP hat hierbei das Design der RASIP-Variante verfeinert und die bereits vorhandenen Crypto-Algorithmen angepasst.

Aufgabe 3.1 (Design of sensing and reconfiguration mechanisms): Basierend auf der High-Level-Architektur des SMART-Sensorknotens wurden durch das IHP die Definitionen des RASIP-Interfaces und der Peripherien verfeinert und endgültig festgelegt. Die Ergebnisse sind im Deliverable 3.1 zusammengefasst.

Aufgabe 3.3 (Implementierung of real-time partial reconfiguration and sensing mechanisms): Nach der Analyse der am IHP vorhandenen Crypto-Algorithmen hinsichtlich ihrer Eignung für eine partielle Rekonfiguration erfolgte die notwendige Anpassung der Implementierungen des AES und des ECC.

2.1.4. Workpackage 4 - Security and Video Subsystems Design and Implementation

Im WP4 werden die Algorithmen für die Verschlüsselung, Kompression und die Verarbeitung der Videodaten erstellt. Darüber hinaus werden in diesem WP die Low-Power-Fähigkeiten des Mikrocontrollers und die Anti-Tamper-Mechanismen umgesetzt. Das IHP konzentrierte sich hierbei insbesondere auf das Low-Power-Design des IHP430x und die Implementierung der Anti-Tamper-Mechanismen.

Aufgabe 4.1 (Design of low power security systems): Für die Untersuchung der Low-Power-Fähigkeiten des SMART-Sensorknotens wurde eine Software-Implementierung der ECC für den MSP430 erstellt. Diese Implementierung wurde anschließend auf dem IHPNode vermessen, um den Energieverbrauch charakterisieren und eine Software-Hardware-Partitionierung durchführen zu können. Außerdem wurde für die FPGA-Variante des SMART-Sensorknotens die Implementierung der ECC auf den Xilinx FPGA portiert. In einer weiteren Teilaufgabe wurde ein Design für das Clock- und das Power-Gating des IHP430x erstellt. Das Clock-Gating wurde im IHP430x V4 erfolgreich umgesetzt.

Aufgabe 4.3 (Implementing resistant to side-channel attack security sub-systems): Das IHP befasste sich in der Aufgabe 4.3 mit dem Design und der Implementierung der Anti-Tamper-Mechanismen. Zur Evaluierung wurden die Mechanismen als Anti-Tamper-IC gefertigt. Hierbei wurden die

Versionen V1 und V2 erstellt, wobei in der Version V2 die aus der Evaluierung der Version 1 gewonnenen Erkenntnisse eingearbeitet wurden.

2.1.5. Workpackage 5 - RASIP and Node Implementation and Software Tools Development

Das WP5 umfasst die Arbeiten zur SMART-Middleware, dem RASIP und der Sensorknotenplattform. Das IHP war in diesem WP wesentlich an der Umsetzung des RASIP und der Plattform beteiligt.

Aufgabe 5.2 (Design and implementation of RASIP): Die Umsetzung des RASIP umfasste das Design und die Fertigung des IHP430x und des HiveCores. Der HiveCore wurde im Wesentlichen von der TUBS entwickelt. Das IHP leistete hierbei Unterstützung bei der Anpassung des Designs an die IHP-Technologie. Zusätzlich wurde die Entwicklung des IHP430x vom IHP verantwortet. Hierbei wurden die notwendigen Peripherien für den Microcontroller angepasst und die Ansteuerung des Flash-Speichers umgesetzt. Die Ergebnisse der Arbeit sind in den IHP430x V4 eingeflossen und konnten im Rahmen des Projektes erfolgreich gefertigt und getestet werden. Darüber hinaus hat das IHP eine Portierung des IHP430x auf den Xilinx FPGA durchgeführt, so dass dieser auch in der FPGA-Variante des SMART-Sensorknotens verwendet werden konnte.

Aufgabe 5.3 (Design and implementation of the hardware subsystem of the SMART node): Für die Definition der RASIP-Variante des SMART-Sensorknotens hat das IHP intensiv mit Lippert zusammengearbeitet. Nach dem vorzeitigen Ausstieg von Lippert und dem damit verbundenen Abbruch der Entwicklung der Plattform hat sich das IHP auf die Entwicklung von Modulen für den IHPStack konzentriert. Hierbei wurden Module für den Anti-Tamper-IC und IHP430x gefertigt, so dass der Ausstieg von Lippert teilweise kompensiert werden konnte.

2.1.6. Workpackage 6 - Integration & Validation

Das WP6 befasst sich mit der Integration und dem Test der SMART-Plattform. Nach den Verzögerungen im Ablauf des Projektes und insbesondere bei der Fertigung der Sensorknoten konnten die Arbeiten des WPs nur teilweise ausgeführt werden.

Aufgabe 6.3 (SMART integration): Die Zusammenführung der Komponenten von SMART wurde im Rahmen des Projektes in drei Integrationsmeetings durchgeführt, an denen das IHP jeweils teilnahm. So wurde im ersten Meeting vordringlich der Integrationspfad spezifiziert und der IHP430x erstmals auf der FPGA-Variante ausgeführt. Im zweiten Integrationsmeeting folgte die vollständige Integration des IHP430x auf dem UPM HiReCookie und initiale Tests der SMART-Middleware. Im letzten Integrationsmeeting wurden die Arbeiten für die Abschlusspräsentation abgeschlossen. Darüber hinaus hat das IHP im Rahmen dieses WP eine Portierung des Sensorknotenbetriebssystems TinyOS auf den IHP430x durchgeführt.

2.1.7. Workpackage 7 - Exploitation & Dissemination

Das WP7 umfasst alle Arbeiten zur Verwertung der Ergebnisse aus dem Projekt SMART. Hierzu zählen Marktanalysen, das Schreiben der Verwertungspläne und Besuche von Konferenzen und Messen.

Aufgabe 7.2 (Exploitation Plans and contributions to standards): Umfasst die Arbeiten des IHP zur Ausarbeitung des Verwertungsplans.

Aufgabe 7.3 (Dissemination & Workshops): Bereits frühzeitig im Projekt konnte auf der Secon 2009 ein Demonstrator für eine konfigurierbare ECC-Implementierung präsentiert werden. Der

Demonstrator wurde außerdem auf dem Artemis/ITEA2 Co-Summit 2009 präsentiert. Darüber hinaus wurden durch das IHP und Kooperation mit Projektpartnern verschiedene Konferenz- und Journalbeiträge veröffentlicht.

2.2. Technische Beschreibung der Ergebnisse

Das IHP war maßgeblich an der Realisierung des Reconfigurable Application Specific Instructionset Processor (RASIP) beteiligt. Der RASIP sollte als eine Variante des SMART-Sensorknotens gefertigt werden und im Gegensatz zum FPGA-Ansatz über einen konfigurierbaren Befehlssatz verfügen. Hierzu sollte, wie in Abbildung 2 dargestellt, ein Microcontroller-Kern mit dem Hive Core, einem rekonfigurierbaren 32-bit-VLIW-Prozessor, kombiniert werden. Ziel war die Integration beider Komponenten in ein Die. Darüber hinaus sollten diese durch Anti-Tamper-Mechanismen geschützt werden.

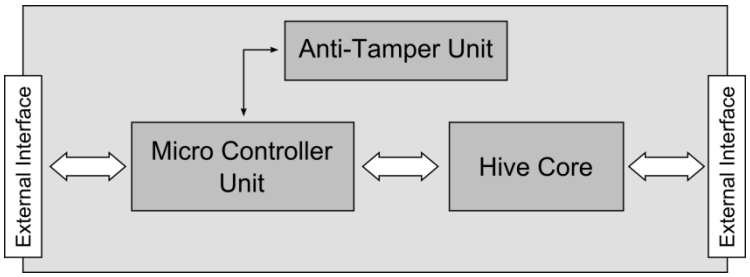


Abbildung 2: Blockschaltbild des RASIP

Die Arbeiten zur Entwicklung und zur Fertigung des RASIPs erstreckten sich über die Arbeitspakete WP3, WP4 und WP5. Um bereits frühzeitig erste Ergebnisse für eine Evaluierung zu erhalten, wurden die einzelnen Komponenten zunächst als separate ICs entwickelt und gefertigt. Nach einer erfolgreichen Evaluierung der einzelnen Komponenten sollte diese abschließend als Single-Chip gefertigt werden.

Im Folgenden werden die Ergebnisse seitens des IHPs der einzelnen Komponenten/ICs im Detail erläutert. Anschließend werden die SMART-Sensorknoten vorgestellt und im letzten Abschnitt werden die bei den Reviews präsentierten Resultate kurz beschrieben.

2.2.1. IHP430x

Der IHP430x ist ein 16-bit-Microcontroller für sicherheitskritische Applikation. Abbildung 3 zeigt ein Blockschaltbild des IHP430x der Version 4. Der Controller besteht aus dem asynchronen Prozessorkern IPMS430X, welcher vom Fraunhofer IPMS entwickelt wurde, und einem typischen Satz an Peripherie: zwei Timer, ein Analog-Digital-Converter (ADC), zwei Serial Peripheral Interface (SPI) Master, drei General Purpose IO (GPIO) Ports und zwei Universal Asynchronous Receiver and Transmitter (UART). Darüber hinaus ist der IC mit 16-kB-RAM und 64-kB-Flash-Speicher ausgestattet. Der Flash-Speicher wird zum Speichern des Programmtextes verwendet, während der RAM für die Laufzeitwerte (Variablen) verwendet wird.

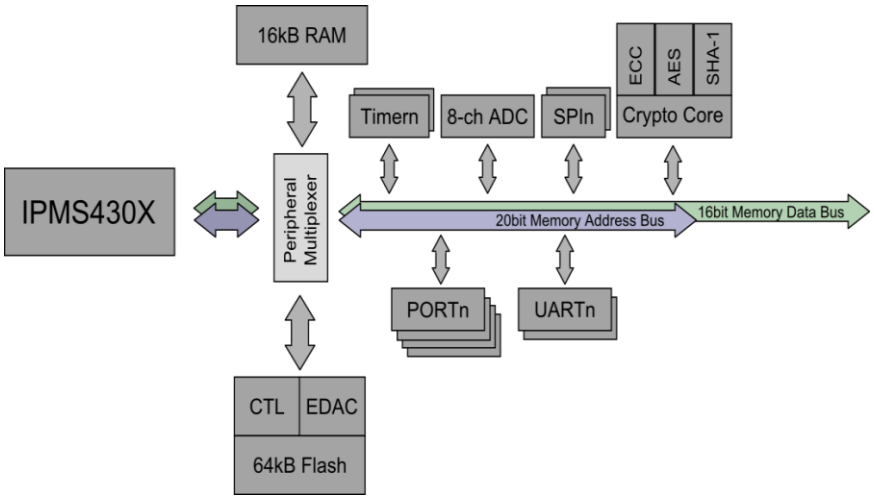


Abbildung 3: Blockschaltbild des IHP430x V4.

Der Controller ist vollständig binär-kompatibel zum MSP430x von Texas Instruments. Damit ist es möglich, die Programmwerkzeuge (Compiler, Linker, Debugger) von Texas Instruments als auch den GNU Toolchain für die Microcontroller zu verwenden. Der IHP430x kann ebenfalls einen 20-bit-Adressraum verwalten, so dass 1 MB adressiert werden können.

Während der Projektlaufzeit wurden vier Versionen des IHP430x gefertigt. Tabelle 1 zeigt eine Übersicht der einzelnen Versionen des IHP430x. Für den SMART-Sensorknoten wurde die Version 4 als separater IC gefertigt. Er verfügte über die notwendigen Peripherien und einen internen Flash-Speicher, wie sie entsprechend den Anforderungen an den SMART-Sensorknoten benötigt werden.

Version	Technology	RAM	Flash Memory	Peripherals	Status
V1	250 nm	16 kB	-	SPI, Timer 16-bit, 4x GPIO, BB CoP	Success
V2	250 nm	8 kB	-	2x UART, SPI, Timer 16-bit, 4x GPIO, BB CoP, Crypto (SHA1, AES, ECC, RSA)	Success
V3	130 nm	16 kB	-	2x UART, Timer 16-bit, SPI, 3x GPIO, BB CoP, Crypto (SHA1, AES, ECC, RSA), MAC	Failed
V4	250 nm	16 kB	64 kB	2x UART, 2x SPI, 8-ch 12-bit ADC, 2x Timer 16-bit, 3x GPIO, Crypto (SHA1, AES, ECC)	Success

Tabelle 1: Versionen und Funktionsumfang des IHP430x

Der IHP430x der Version 4 wurde in der 0,25-µm-CMOS-Technologie des IHP gefertigt. Der IC benötigt insgesamt, inklusive der Pads, eine Chipfläche von 24,7 mm². Das Layout der ICs ist in Abbildung 4 dargestellt. In der Darstellung gut erkennbar ist der Flash-Speicher (rechts), der 20 % der Chipfläche belegt, der RAM (unten links) mit 13 % der Fläche und der ADC 6,9 % (oben links). Die verbleibende Fläche wird von den Peripherien (3,5 %), den Cryptoblocks (8,5 %) und dem Prozessorkern (1,2 %) belegt.

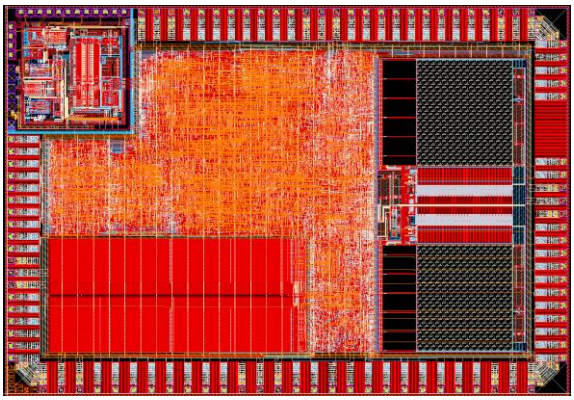


Abbildung 4: Layout des IHP430x mit integriertem Flash und ADC

Der IC wurde zunächst in einem 128-pin-PQFP-Gehäuse verpackt. Hierbei wurden 71 digital, 26 analog und 20 Stromversorgungsleitungen nach außen gelegt. Aufgrund der zu geringen Anzahl an IO-Pins stehen bei diesem IC der MAB und der MDB zum Anschluss des Hive Cores nicht zur Verfügung, stattdessen war für den SMART-Sensorknoten eine 144-Pin-BGA-Version vorgesehen. Bei dieser Version konnte der MAB und MDB nach außen gelegt werden und außerdem erschweren BGA-Packages den physischen Zugriff auf die Pins, so dass ein zusätzlicher Schutz gegen Angriffe geboten werden kann. Auf die Fertigung dieses ICs wurde jedoch aufgrund der Budget-Situation des Projektes seitens des IHPs und des Abbruchs der Fertigung des Hive-Core-ICs bisher verzichtet.

Der IHP430x Version 4 wurde während des Designs für eine Taktfrequenz von bis zu 20 MHz spezifiziert. Messungen am gefertigten IC haben gezeigt, dass die Maximalfrequenz des Chips bei 11,4 MHz liegt. Die Differenz zwischen den beiden Frequenzen wird durch die minimalen Lesezeiten des Flash-Speichers bestimmt, welche bei der Evaluierung des Designs des Kerns nicht berücksichtigt werden konnte. Der gemessene Stromverbrauch des IHP430x ist in Tabelle 2 zusammengefasst. Er setzt sich aus dem Verbrauch des Kerns, des ADCs und Pads zusammen, da diese jeweils mit unabhängigen bzw. verschiedenen Spannungen versorgt werden.

Frequency	Core 2.5V (mA)	ADC 2.5V (mA)	Pad 3.3V (mA)	Consumption (mW)
1 MHz	2.5	1.6	1.4	14.87
10 MHz	14.0	1.6	9.5	70.35

Tabelle 2: Stromverbrauch des IHP430x V4 bei 1 und 10 MHz

Befindet sich der IC im Idle-Zustand, wobei der Systemtakt aktiv sowie der Flash und der ADC aktiviert sind, beträgt der gemessene Strombedarf weniger als 4 mA. Im Standby-Zustand, wenn Systemtakt, Flash und ADC deaktiviert sind, verbraucht der IC weniger als 1 mA.

Flash-IC mit Controller-Logik

Für den Einsatz des IPMS430-Prozessorkerns in einem autark agierenden Low-Power-Sensorknoten war es notwendig, diesen mit der IHP-eigenen Flash-Technologie zu kombinieren. Zum Zeitpunkt des Projektstarts wurden bereits erfolgreich Flash-Speicher in der 0,25-µm-CMOS-Technologie des IHP gefertigt. Hierbei handelte es sich jedoch lediglich um die Flash-Zellen mit einem minimalen digitalen Interface. Für eine Nutzung in einem Sensorknoten musste diese um einen Flash-Controller und um eine Fehlerkorrektur erweitert werden.

Die Entwicklung des Flash-Controllers und der Fehlerkorrektur erfolgt ebenfalls in einem separaten IC, der mit bereits gefertigten Versionen des IHP430x kombiniert werden konnten. Abbildung 5 zeigt den Flash-IC mit zwei 32-kB-Flash-Speicherblöcken, die über den Flash-Controller mit dem IHP430x verbunden werden können.

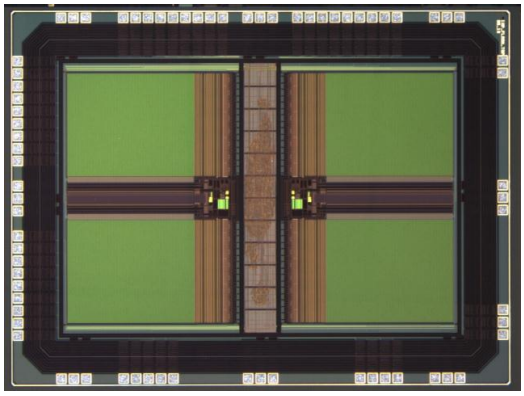


Abbildung 5: IHP Flash-IC mit zwei 32-kB-Flash-Blöcken und Controller-Logik

Neben der erfolgreichen Evaluierung des Flash-Controllers und der IHP-Flash-Zellen konnte mit dem IC auch die Funktionsfähigkeit des externen Interfaces des MAB und des MDB nachgewiesen werden. Wie bereits zuvor beschrieben, war dieses Interface für die Anbindung des Hive Cores an den IHP430x in der Mehr-Chip-Lösung vorgesehen.

Software Toolchain

Für die Evaluierung des IHP430x war die Entwicklung von Softwarewerkzeugen notwendig. So wurde mit dem HSE430 eine Simulationsumgebung geschaffen, die eine Zyklen-akkurate Simulation von Programmen für den IHP430x erlaubt. Außerdem wurden Werkzeuge zum Programmieren der ICs entwickelt.

Hybrid Simulation Environment 430 (HSE430)

Das HSE430 ist eine auf dem MSP430sim basierende Simulationsumgebung für den IHP430x. Der MSP430sim ist ein frei verfügbarer Java-basierter Simulator für MSP430-Programme und wurde am Swedish Institute of Technologie (SIT) entwickelt.

Für eine zeitnahe Simulation von am IHP entwickelten Peripherien wurde das Speicherinterface des MSP430sim dahingehend erweitert, dass es die Anbindung von externen Simulationsumgebungen erlaubt. So wurde eine Schnittstelle zu einer SystemC-Umgebung umgesetzt. Damit können Peripherien hardwarenah in der Hardware Description Language (HDL) SystemC beschrieben und mit dem MSP430sim evaluiert werden.

Die Simulationsumgebung wurde als Publikation auf der DDECS 2011 „*14th Symposium on Design and Diagnostics of Electronic Circuits and Systems*“ in Cottbus, Deutschland präsentiert.

Build Chain

Die Software Toolchain des IHP430x basiert im Wesentlichen auf den gleichen Werkzeugen wie für den MSP430 von Texas Instruments (TI). Der Programmcode kann sowohl mit dem TI Code Composer Studio als auch mit der GNU Toolchain erzeugt werden. Anpassungen sind lediglich bei der Nutzung der Programmierschnittstelle notwendig. Während der MSP430 JTAG verwendet, nutzt der IPMS430-Prozessorkern eine I²C-Schnittstelle. Beide Schnittstellen sind zueinander inkompatibel.

Das Fraunhofer IPMS stellt für die Nutzung ihres Prozessorkerns eine kommerzielle Programmierumgebung zur Verfügung. Das IHP verfügt über die notwendigen Lizenzen zur Nutzung des Werkzeuges. Da das Programm nur für *Microsoft Windows© (32-bit)* zur Verfügung stand und der IHP430x auch von anderen Projektpartnern und in Folgeprojekten verwendet werden soll, wurde mit dem `ihp430x-gdbproxy` ein Programm zur Nutzung der GNU Debug Tools implementiert. Hiermit kann der IHP430x über den GNU gdb unter Linux als auch MS Windows programmiert und debuggt werden. Darüber hinaus wurden Werkzeuge entwickelt, um den IHP-eigenen Flash-Speicher über die I²C-Schnittstelle zu beschreiben.

Crypto-Einheiten

Nachdem die Fertigung des Hive Core ICs nicht mehr ausgeführt werden konnte, wurde der IHP430x V4 mit der am IHP verfügbaren Crypto-Einheit ausgestattet. Diese beinhaltet einen 128-bit AES Encryption Unit, einen SHA1 Hash und einen ECC-Multiplizier. Die Crypto-Einheit ist über den MAB und den MDB an den Prozessorkern angebunden und kann mittels des Memory Mapped Interface verwendet werden. Obwohl diese Einheiten nicht rekonfigurierbar sind, bieten sie einen minimalen Funktionsumfang, um die im SMART-Projekt angestrebten Sicherheitsmechanismen energieeffizient auszuführen.

2.2.2. Hive Core

Bei dem Hive Core handelt es sich um einen 32-bit-VLIW-Prozessor mit einem konfigurierbaren Befehlssatz. Das Konfigurieren des Prozessors erfolgt in einem semi-automatischen Prozess mit den Tools von Silicon Hive. Hierbei handelt es sich um ein kommerzielles Produkt, für dessen Nutzung die TU Braunschweig über die notwendigen Lizenzen verfügt. Für die Fertigung des ICs musste der erzeugte IC mit Design-Parametern des IHP synthetisiert werden. Hierfür stellt das IHP eine Design-Bibliothek zur Verfügung, die mit den Werkzeugen von Synopsys verwendet werden kann.

Zur Anbindung des Hive Cores an den 16-bit IHP430x wurde in Zusammenarbeit mit der TUBS ein Wrapper definiert, der zunächst in den Hive Core IC integriert werden sollte. Der Zugriff auf den Hive Core durch den IHP430x erfolgt Memory Mapped. Hierzu wurde ein Registersatz definiert, der vom IHP430x wie eine Peripherie genutzt werden kann.

Die physische Anbindung des Hive Cores an den IHP430x erfolgt über den Memory Address Bus (MAB) und den Memory Data Bus (MDB). Hierzu wurden diese zunächst über Pads aus dem IHP430x herausgeführt, so dass in einer Mehr-Chip-Lösung der Hive Core IC über ein PCB angebunden werden kann. In der Single-Chip-Lösung entfallen lediglich die Pads, so dass die Lösung ohne Portierungs-aufwand übertragen werden kann.

2.2.3. Anti-Tamper Mechanismen

Sensorknoten, die z. B. zur Überwachung von Umweltinformationen eingesetzt werden, sind durch ihre freie Platzierung oftmals anfällig gegen Diebstahl oder mechanische Manipulationen. Zum Schutz des eigenen IPs und zum Sicherstellen einer vertrauenswürdigen Funktion werden Mechanismen zum Manipulationsschutz und Auslesen von Systeminformationen eingesetzt.

Die Entwicklung eines manipulationssicheren und vor unbefugtem Auslesen sicheren Sensorknotens war auch Bestandteil des SMART-Projektvorhabens. Hierbei sollte insbesondere der RASIP mit Anti-Tamper-Mechanismen ausgestattet werden, die derartige Angriffe auf den Knoten verhindern oder zumindest erkennen können.

Für die Umsetzung der Anti-Tamper-Mechanismen wurde eine schrittweise Entwicklung geplant. Im ersten Schritt wurden verschiedene, bekannte Mechanismen aufgelistet und hinsichtlich ihrer Umsetzbarkeit in der IHP-eigenen Technologie untersucht. Die Ergebnisse wurden im Deliverable 4.1 präsentiert. Insgesamt wurden fünf Mechanismen ausgewählt. Im folgenden Schritt wurden drei dieser Mechanismen umsetzt und in Simulationen evaluiert. Nach der erfolgreichen Evaluierung in der Simulation wurden die Mechanismen in einem separaten Anti-Tamper-IC, siehe Abbildung 6, in der IHP-eigenen Technologie umgesetzt. Nach einer erfolgreichen Evaluierung war abschließend die Integration in den RASIP geplant.

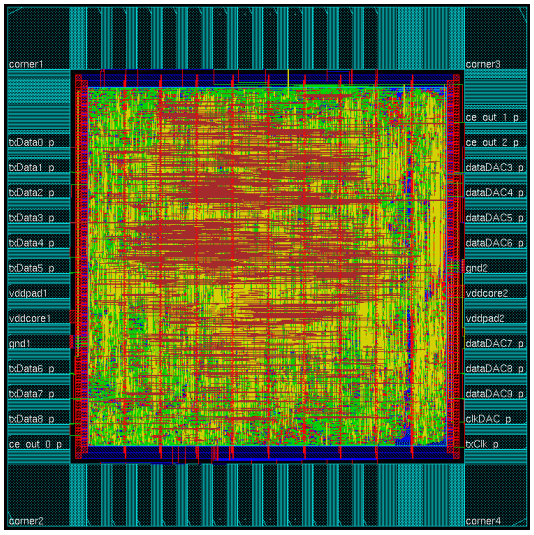


Abbildung 6: Anti-Tamper-IC mit Systemtakt- und Temperaturüberwachung und OTP-Memory zum Schutz der Scan-Chain.

Im Folgenden werden die Mechanismen im Detail vorgestellt und die Ergebnisse aus der Simulation und der Evaluierung des ICs aufgeführt. Im Rahmen des Projektes wurden die Ergebnisse im Deliverable 5.4 präsentiert.

Systemtaktüberwachung

Ein Sensorknoten ist, wie die meisten digitalen Systeme, mit einer Vielzahl von internen Speichern ausgestattet, die verschiedene Informationen enthalten. Darunter sind oftmals auch Informationen, die das System lediglich nur intern verarbeitet und deren Bekanntwerden verhindert werden soll. Hierzu zählen z. B. kryptographische Schlüssel. Mittels einer Power Analyse ist es jedoch möglich, diese Daten aus dem IC ohne einen softwaretechnischen Zugriff auszulesen. Hierzu wird lediglich der Stromverbrauch des ICs bei bestimmten Operationen gemessen. Da digitale Systeme bei der Verarbeitung einer ‚1' einen anderen Stromverbrauch als bei der Verarbeitung einer ‚0' haben, kann hiermit auf den Inhalt des Speichers und damit auf den Schlüssel selbst geschlossen werden. Dieses Verfahren ist insbesondere dann erfolgreich, wenn der Systemtakt möglichst gering ist, da in diesem Fall die logischen Zustände über einen möglichst langen Zeitraum anliegen. Aus diesem Grund ist die Überwachung des Systemtaktes ein geeignetes Mittel, um derartige Angriffe zu verhindern oder zumindest zu erschweren.

In den Anti-Tamper-IC wurde ein Clock Watchdog integriert, der, wie in Abbildung 7 dargestellt, aus einem Ring-Oszillator, einem Counter-Register und einer Comparator-Logik besteht. Zusätzlich ist die Einheit mit dem System-Reset verbunden.

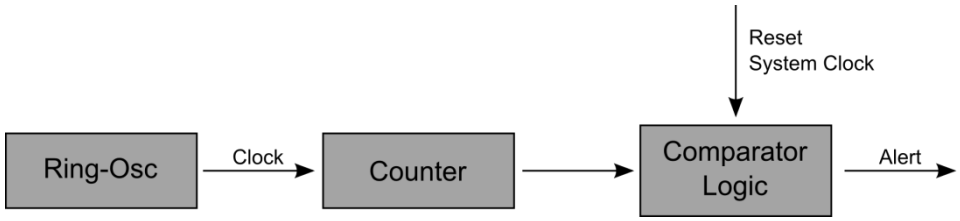


Abbildung 7: Clock-Watchdog mit eigenem Ring-Oszillator und Comparator Logik

Der Ring-Oszillator generiert einen schnellen internen Takt, der im Counter-Register gezählt wird. Die Comparator-Logik vergleicht den internen Takt mit dem Systemtakt. Weichen beide zu stark voneinander ab, deutet dies auf eine Veränderung des Systemtaktes hin. In diesem Fall wird ein Alarmsignal generiert, das von der CPU verarbeitet werden muss.

Temperaturüberwachung

Neben dem Verändern des Systemtaktes bietet insbesondere das Verändern der Umgebungstemperatur eines ICs Möglichkeiten, dessen Eigenschaft derart zu verändern, dass Speicher leichter ausgelesen werden können. Aus diesem Grund wurde für den Anti-Tamper-IC eine Schaltung entwickelt, die eine Überwachung der IC-Temperatur ermöglicht.

Bei der Temperaturüberwachung wurden ausschließlich digitale Schaltungselemente verwendet, so dass dessen Fertigung sehr einfach ist und direkt in die Schaltung integriert werden kann. Das Modul besteht, wie in Abbildung 8 dargestellt, aus zwei Delay-Lines, die über einen Controller gesteuert und mittels einer Compare-Logik ausgewertet werden.

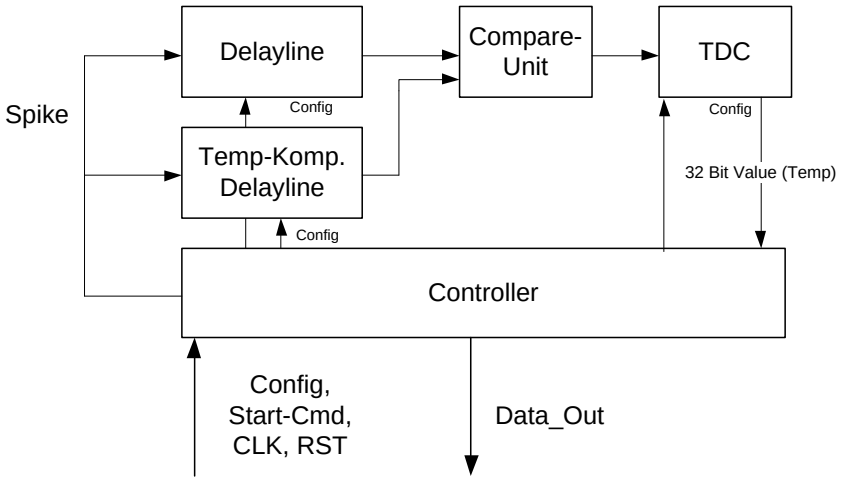


Abbildung 8: Blockdiagramm der Temperaturüberwachung

Den Kern der Temperaturüberwachung bilden die beiden Delay-Lines. Sie bestehen aus einer Kette von Invertern, wobei die Schaltzeit der einen Kette temperaturabhängig ist und die andere nicht. Zur Messung der Temperatur werden beide Ketten über einen kurzen Puls angeregt. In der Compare-Unit werden beide Signale zusammengeführt. Je nach Temperatur ist der entstandene Puls länger oder kürzer. Im Time-to-Digital Converter (TDC) wird die Länge des Pulses ermittelt.

Um eine möglichst genaue Messung realisieren zu können, müssen die Ketten hinreichend lang sein. Aus diesem Grund wurde das Layout der Zellen per Hand optimiert. Abbildung 9 zeigt das Layout des

temperaturkompensierten Puffers. Jede Kette besteht aus 10 Delay-Elementen mit den Abmaßen 226 x 46 µm. Durch die reguläre Struktur des Puffers lassen sich diese leicht zu einer Kette verbinden.

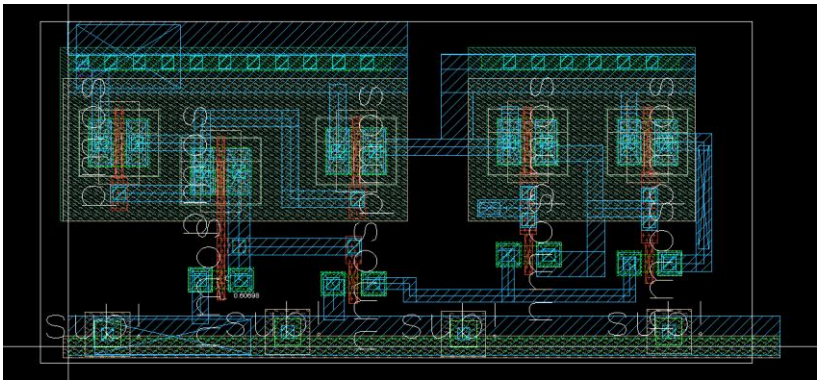


Abbildung 9: Layout des temperaturkompensierten Puffers

Der TDC besteht aus einer Delay-Line und einem Counter Register. Die Delay-Line wird durch den Pulse aus der Compare-Unit angeregt und in einer Schleife in die interne Delay-Line zurückgeführt. Eine solche Delay-Line hat die Eigenschaft, dass der Puls am Ausgang immer ein wenig kürzer ist als das eingespeiste Signal. Mittels des Counter-Registers wird die Anzahl der Runden gezählt, bis der Puls vollständig „verschwunden ist“. Die Anzahl der Runden erlaubt nun einen Rückschluss über die Temperatur des ICs.

Scan-Chain-Schutz

Für den Test und die Verifikation der Funktionen eines ICs werden diese mittels einer Scan Chain ausgestattet. Hierbei handelt es sich um einen Datenpfad durch alle Register, der letztendlich auf einen Debug-Port herausgelegt wird. Da diese Funktion auch nach Test des ICs erhalten bleibt, können hierüber zu einem späteren Zeitpunkt alle Daten des Systems ausgelesen werden.

Zum Schutz der Scan Chain beinhaltet der Anti-Tamper-IC einen One-Time Programmable (OTP) Speicher. Dieser beinhaltet genau ein Bit, das genau einmal beschrieben werden kann und mit dem Ausgang der Scan Chain über ein Und-Gatter kombiniert wird. Der OTP enthält nach der Fabrikation des ICs eine Eins, so dass der Ausgang der Scan Chain durch das Und-Gatter nicht verändert wird. Wenn die Tests abgeschlossen sind, wird der OTP-Speicher auf eine Null umgesetzt, so dass das Ergebnis am Debug-Port immer gleich Null ist und damit ein Auslesen der Daten über die Scan Chain nicht mehr möglich ist.

2.2.4. Sensorknoten

Ein wesentliches Ziel von SMART war die Entwicklung eines flexiblen Sensorknotens, in dem die FPGA- als auch die RASIP-Variante umgesetzt werden kann. Mit dem Ausscheiden von Lippert und dem damit verbundenen Verlust der zugehörigen Ressourcen war die Umsetzung eines einheitlichen Sensorknotens nicht mehr möglich. Um die Projektziele trotzdem erreichen zu können, wurde der von UPM und vom IHP entwickelte Sensorknoten für SMART angepasst bzw. weiterentwickelt. Damit können sowohl Kosten als auch Ressourcen eingespart werden, so dass sowohl die FPGA- als auch die RASIP-Variante im Rahmen des Projektes umgesetzt werden konnte.

Obwohl der UPM-Sensorknoten (HiReCookie) und der IHP-Knoten (IHPStack) den gleichen Designansatz verfolgen, sind ihre Bestandteile nicht zueinander kompatibel, so dass kein einheitlicher SMART-Sensorknoten erstellt wurde.

IHPStack

Der IHPStack, siehe Abbildung 10, ist ein Sensorknoten, der Rahmen des Projektes IQlevel entstand und sich durch seine modulare Bauweise auszeichnet. Er wird aus mehreren Modulen zusammengesetzt, die über einen einheitlichen Steckverbinder, *Mote Component Interconnect* (MCI), verbunden sind. Der MCI hat eine fest vorgegebene Pin-Belegung, so dass einzelne Module untereinander ausgetauscht werden können. Damit kann der Sensorknoten schnell und kostengünstig an neue Anforderungen und veränderte Umgebungsbedingungen angepasst werden.

Abbildung 10: IHPStack mit vier Modulen (Power, Controller, Transceiver und Amplifier)

Im Rahmen von SMART wurde für den IHPStack ein IHP430x Microcontroller-Modul erstellt. Dieses Modul kann als vollwertige Alternative zu den bereits vorhandenen Microcontroller-Modulen eingesetzt werden. Die Schnittstelle zum RASIP bleibt hierbei jedoch ungenutzt, da das Modul zu einem Zeitpunkt entstand, als die Entwicklung des RASIP bereits abgebrochen wurde.

Neben dem Microcontroller-Modul wurde ein Modul speziell für die Demonstration des Anti-Tamper-ICs erstellt. Dieses kann in den IHPStack integriert und über ein MCU-Modul gesteuert werden. Obwohl die letztendliche Integration der Anti-Tamper-Komponenten in den RASIP ausblieb, konnte hiermit die korrekte Funktionsweise validiert und präsentiert werden.

Das Konzept und die Implementierung des IHPStack wurde im Rahmen des Projektes auf der Konferenz Sensornets 2012, „*1st International Conference on Sensor Networks*“ in Rom, Italien präsentiert.

UPM HiReCookie

Der UPM HiReCookie ist ebenfalls ein modularer Sensorknoten mit einem einheitlichen Connector Interface. Im Gegensatz zum IHPStack verwendet dieser einen FPGA als Rechenkern. Der Einsatz von Low Power Microcontroller im UPM HiReCookie ist nicht vorgesehen. Aufgrund der Rechenkapazität des verwendeten FPGAs konnte auf dem HiReCookie die Videokompression umgesetzt werden. Hierzu wurde der Sensorknoten im Rahmen des Projektes um ein Kameramodul erweitert. Für die Datenübertragung standen ein ZigBee- und ein WLAN-Modul zur Verfügung.

Um der SMART-Middleware eine einheitliche Hardware-Plattform zu bieten, wurden in den FPGA des HiReCookie der IHP430x integriert. Damit war die Nutzung der Software ohne aufwendige Portierungen auf beiden Varianten der SMART-Sensorknoten möglich. Da der HiReCookie auf einem Xilinx Spartan 6 LX150 basiert, musste der IHP430x auf diesen FPGA portiert werden. Um die Komplexität und damit den Ressourcenbedarf des Systems, siehe Abbildung 11, gering zu halten, wurde eine speziell angepasste Variante des Designs verwendet.

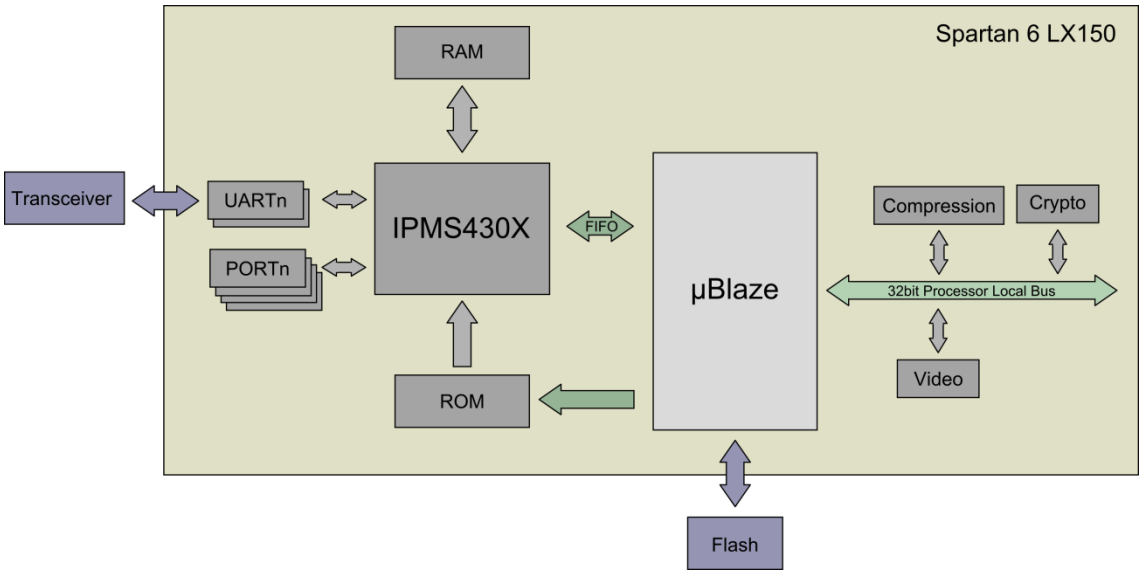


Abbildung 11: Integration des IHP430x in den HiReCookie FPGA

Für das Rekonfigurieren der Hardware-Einheiten im FPGA wird ein µBlaze Controller eingesetzt. Dieser bildet auch die Schnittstelle zum Processor Local Bus (PLB), welcher für den Datenverkehr zu den rekonfigurierbaren Einheiten verwendet wird. Der µBlaze ist ein 32-bit-Processor-IP von Xilinx und kann in deren FPGAs frei verwendet werden. Eine Verwendung in anderen FPGAs oder eine Umsetzung als Silicium Device ist jedoch ausgeschlossen, so dass er nicht als Prozessorkern für alle SMART-Varianten eingesetzt werden kann. Im HiReCookie dient er als Hardware-Controller und übernimmt auch die Konfiguration des IHP430x-Kerns. Hierzu sind beide Controller über einen gemeinsamen Speicher und über ein FIFO-Interface miteinander verbunden. Der gemeinsame Speicher wird als Textspeicher (ROM) in den IHP430x eingeblendet. Beim Systemstart lädt der µBlaze zunächst den Programmcode des IHP430x aus dem externen Flash-Speicher in den gemeinsamen Speicher und löst anschließend dessen Reset. Damit kann beim IHP430x-Kern auf eine Instanziierung des Flash-Controllers verzichtet werden.

Der IHP430x-Kern ist für die Ausführung der SMART-Middleware verantwortlich. Diese enthält unter anderem den Protokoll-Stack für die drahtlose Datenübertragung über das ZigBee-Transceiver-Modul. Das Transceiver-Modul ist per UART an den IHP430x-Kern angeschlossen. Für eine Übertragung der Daten zwischen der SMART-Middleware und den rekonfigurierbaren Hardwaremodulen wurde ein bidirektionales FIFO-nterface umgesetzt. Hierüber kann der IHP430x Daten an den µBlaze übertragen, der diese anschließend an die jeweilige Hardware-Einheit weiterleitet. Die Ergebnisse der Hardware-Einheit bzw. Sensor- und Videodaten können vom µBlaze ebenso über das FIFO-Interface an den IHP430x-Kern übertragen werden.

2.3. Abschlusspräsentation

Als Abschluss des Projektes wurde am 19. April 2013 in Schimatari, Griechenland eine Präsentation der Ergebnisse durchgeführt. An der Präsentation nahmen alle Projektpartner, außer Lippert und TUBS, und Programm Officer der EU teil.

Das IHP präsentierte in Kooperation mit anderen Projektpartnern die FPGA-Variante des SMART-Sensorknotens. Hierbei wurde gezeigt, dass der IHP430x erfolgreich in den FPGA integriert werden

konnte und mit den anderen Hardware-Einheiten verbunden ist. Zudem wurden die drahtlose Übertragung und die Verschlüsselung von Daten präsentiert.

Außerdem wurden die Ergebnisse der RASIP-Variante in Form des IHP430x präsentiert. Der IHP430x war als Modul des IHPStack verfügbar. Es wurde die Einbindung in der GNU Toolchain, die Nutzung des internen Flash-Speichers und die Verschlüsselung von Daten mittels der integrierten AES-Hardware-Einheit präsentiert.

Darüber hinaus wurde die Portierung des TinyOS Betriebssystems kurz vorgestellt, welche als Basis für die SMART-Middleware dient.

3. Zahlenmäßiger Nachweis

Dem IHP sind im Rahmen des SMART-Projektes die folgenden Kostenpositionen gefördert worden.

Position	Beschreibung
0812	Beschäftigungsentgelt E12 - E15
0817	Beschäftigungsentgelt E1 - E11
0835	Vergabe von Aufträgen
0843	Sonstige allgemeine Verwaltungsausgaben
0846	Dienstreisen

Tabelle 3: Kostenpositionen des IHP

Durch die Beschäftigungsentgelte konnte über die Projektlaufzeit ein wissenschaftlicher Mitarbeiter Vollzeit beschäftigt werden. Darüber hinaus wurde ein technischer Mitarbeiter für die Messung der gefertigten ICs zeitweise beschäftigt.

Für die Fertigung der ICs und der Sensorknoten-Module wurden Aufträge an Dritte erteilt. Hierbei handelte es sich im Speziellen um das Verpacken der ICs und die Fertigung und Bestückung von Leiterplatten. Darüber hinaus wurden über die Position 0835 externe Kosten, die bei der Fertigung der ICs entstanden sind, bezahlt. Zu diesen zählen unter anderem Maskenkosten und anteilig Chemikalienkosten.

Das IHP verfolgt einen vertikalen Forschungsansatz und betreibt im eigenen Haus eine Chipfabrikation zu Forschungszwecken, deren hohe Kosten anteilig von den anderen Abteilungen entsprechend dem Anteil an der Nutzung dieser Chipherstellung mitgetragen werden. Die entstandenen Kosten wurden im Rahmen des SMART-Projektes durch die Position 0843 gedeckt.

Die Reisemittel wurden für die Reisen zu den Projekttreffen, zum Besuch der Artemis-Veranstaltungen und zur Teilnahme an Fachkonferenzen genutzt.

4. Notwendigkeit und Angemessenheit der geleisteten Arbeiten

Der Einsatz von verteilten drahtlosen Sensornetzen ist eine der Triebfedern für die Entwicklungen in den Bereichen Industrie 4.0, Home Automation and Ambient Assisted Living sowie Medical Life Care. Aktuelle Marktprognosen zeigen, dass der Bedarf in den nächsten Jahren überproportional steigen wird. Hiermit verbunden ist aber auch der Bedarf nach Sicherheitslösungen für die Klasse von Geräten. Insbesondere die eingeschränkten Ressourcen und die Besonderheiten bei den Ad-Hoc-

Netzwerken führen dazu, dass bestehende Sicherheitslösungen, wie sie im Bereich der Bürogeräte/PCs seit langen zu finden sind, nur eingeschränkt wiederverwendet werden können.

Die zentrale Ausrichtung des SMART-Projektes zur Entwicklung eines rekonfigurierbaren Sensorknotens kann eine mögliche Lösung für aktuelle und zukünftige Probleme darstellen. Mit dem schnellen Austauschen von essentiellen Hardwarekomponenten zur Laufzeit können Ressourcenbeschränkungen wirksam umgangen werden. So zeigen die Ergebnisse des Projektes, dass komplexe Algorithmen wie ECC oder auch H.264 auch auf dieser Klasse von Geräten eingesetzt werden können, ohne dass andere Einschränkungen dafür entgegengenommen werden müssen.

Obwohl die Integration der Komponenten während der Projektlaufzeit nicht vollständig abgeschlossen wurde, konnten die Funktionsfähigkeit und das erfolgreiche Zusammenarbeiten der Komponenten nachgewiesen werden. Hierzu hat auch insbesondere die Idee des konfigurierbaren Sensorknotens, wie der HiRe Cookie und der IHPStack, beigetragen. In den Bereichen Forschung und Rapid Prototyping besitzen modulare Sensorknoten entscheidende Vorteile, da sie schnell und kosteneffizient an veränderte Bedingungen angepasst werden können.

5. Nutzen und Verwendbarkeit der Ergebnisse

5.1. Verwendung in weiteren Forschungsprojekten

Das IHP und dessen Abteilung System Design ist stetig aktiv an verschiedenen Forschungsprojekten beteiligt. Ein Schwerpunkt liegt hierbei beim Entwurf und der Entwicklung von Sensorknoten im sicherheitskritischen Umfeld. Die gewonnenen Erkenntnisse aus dem SMART-Projekt können direkt und indirekt wiederverwendet werden. Zusätzlich ermöglichen Gemeinsamkeiten zwischen den Projekten die Nutzung von Synergien. So wird insbesondere der IHP430x auch in anderen Projekten verwendet.

5.1.1. Matrix - Middleware für die Realisierung internetbasierter telemedizinischer Dienste

In dem Projekt Matrix wurden der Entwurf und die Fertigung des IHP430x maßgeblich beeinflusst. So wurden im Rahmen des Projektes erste Versionen des Mikrocontrollers gefertigt und erste Arbeiten zur Portierung von Softwarekomponenten vorgenommen. Entwicklung, die im Rahmen von SMART entstanden sind, konnten in diesen Mikrocontrollern getestet werden, ohne dass zusätzliche Kosten entstanden sind.

5.1.2. Tampres - Tamper Resistant Sensor Node

Mit dem Projekt TAMPRES werden die Arbeiten zu den Anti-Tamper-Einheiten des SMART-Sensorknotens vorgeführt. Darüber hinaus werden Analysen zur Angreifbarkeit von digitalen Schaltungen durchgeführt. Hierbei werden auch die in SMART verwendeten Cryptomodule wiederverwendet.

5.1.3. Aeternitas - Energy Efficient Wakeup System for Wireless Sensor Nodes

Das Projekt Aeternitas beschäftigt sich mit der Entwicklung eines sicheren WakeUp-Transceivers für Ultra-Low-Power-Sensorknoten. Mit dem SMART-Projekt konnten hierzu erste Erkenntnisse durch den Einsatz der Clock-Gating-Komponenten gewonnen werden. Zusätzlich ist der Einsatz der Cryptomodule vorgesehen.

5.1.4. Tele-Diagnostic - THz sensors and tools for bioanalysis and wireless biosensor networks

Das Ziel des Tele-Diagnostic-Projektes ist die Kombination von in-vitro-Untersuchungen und in-vitro-Diagnostik für die Bioanalyse mittels drahtloser Sensornetze zur Datenübertragung. Mit dem IHP430x verfügt das IHP bereits über einen leistungsfähigen Mikrocontroller, der im Rahmen des Projektes um die notwendigen Peripherien zur in-vitro-Diagnostik erweitert wird. Der IHP430x V4 aus dem SMART-Projekt kann hierbei als Entwicklungs- und Erprobungsplattform verwendet werden.

5.1.5. UNIKOPS - Universell konfigurierbare Sicherheitslösung für Cyber-Physikalische heterogene Systeme

Die Entwicklung von Cyber Physical Systems hat dazu geführt, dass der Bedarf nach Sicherheitslösungen im Bereich der Sensornetze stark an Bedeutung gewonnen hat. Hierbei ist sowohl der physische Schutz als auch die Sicherung der Datenhaltung sowie deren Übertragung von entscheidender Bedeutung. Mit dem SMART-Projekt wurden auf beiden Gebieten Forschungsergebnisse erzielt, die direkt bzw. indirekt wiederverwendet werden können.

5.2. Nutzung in Forschung und Lehre

Das IHP verfügt über enge Kooperationen mit Hochschulen in Berlin und Brandenburg. So werden unter anderem zwei Lehrstühle der BTU Cottbus durch Mitarbeiter des IHPs geleitet. Die Forschungsinhalte des Lehrstuhls „Sicherheit in pervasiven Systemen“ stehen in direktem Zusammenhang mit den Forschungszielen des Projektes SMART. Die Entwicklung von sicheren Sensorknoten ist auch hier zentraler Schwerpunkt der Forschung. So können Erkenntnisse aus dem Projekt SMART direkt in der Lehre wiederverwendet werden.

Darüber hinaus werden für Studierende der Hochschulen Berlin/Brandenburg Praktika und Abschlussarbeiten am IHP angeboten. Die zu bearbeitenden Themen sind hierbei oftmals auf den Bereich der sicheren eingebetteten Systeme ausgelegt. Mit dem IHPStack und dem IHP430x kann den Studierenden hierfür Hardware zur Verfügung gestellt werden, die sie selbständig weiterentwickeln und erproben können.

6. Fortschritte bei anderen Stellen

Die Erforschung und Entwicklung von sicheren drahtlosen Sensornetzen hat während der Laufzeit des Projektes stetig zugenommen. So haben insbesondere die Entwicklungen bei den Cyber Physical Systems einen hohen Forschungs- und Entwicklungsbedarf aufgezeigt. Die von SMART adressierten Schwerpunkte Sensorik, Videoverarbeitung und Energieeffizienz sind hierbei von zentraler Bedeutung. Dementsprechend vielfältig und umfangreich sind die neuen Erkenntnisse auf diesen Gebieten.

Demgegenüber stellen die Schlüsselfunktionen Rekonfigurierbarkeit und H.264-Videoverarbeitung immer noch ein Alleinstellungsmerkmal dar. Obwohl andere FPGA-basierte Ansätze für Sensorknoten existieren, sind die Fähigkeiten des HiRe Cookie im Bereich der partiellen Rekonfigurierbarkeit und Energieeffizienz bei dieser Art von Sensorknoten einzigartig. Eine vergleichbare Lösung zum Komprimieren von Videodaten in das Format H.264 in einem Low-Power-FPGA konnte bisher nicht ausgemacht werden.

7. Erfolgte und geplante Veröffentlichungen

[1] TandemStack - A Flexible and Customizable Sensor Node Platform for Low Power Applications

Oliver Stecklina, Dieter Genschow, Christian Goltz 24.-26. Februar 2012: Sensornets 2012 1st International Conference on Sensor Networks (Rom, Italien)

[2] Hybrid Simulation Environment for Rapid MSP430 System Design Test and Validation using MSPsim and SystemC

Oliver Stecklina, Frank Vater, Thomas Basmer, Erik Bergmann, Hannes Menzel 13.-15. April 2011: DDECS 2011 14th Symposium on Design and Diagnostics of Electronic Circuits and Systems (Cottbus, Deutschland)

[3] Adaptable Security in Wireless Sensor Networks by Using Reconfigurable ECC Hardware Coprocessors

Jorge Portilla, José Andrés Otero, Eduardo de la Torre, Teresa Riesgo, Oliver Stecklina, Steffen Peter, Peter Langendörfer, 29. September 2010: International Journal of Distributed Sensor Networks, Vol. 2010