Eidgenössische Technische Hochschule Zürich
Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Institut für Integrierte Systeme
Integrated Systems Laboratory

SHIVA: Correlator/Demodulator Chip for Direct-Sequence Spread-Spectrum RAKE-Receiver

Reto Zimmermann, Matthias Neeracher

Technical Report No. 94/9

May 1994

Design and Test Report

Design: Matthias Neeracher, Reto Zimmermann
Integrated Systems Laboratory, ETH Zürich

Project Partners: Hans Kaufmann, Stephen Lingwood
Ascom Tech AG, Hombrechtikon

Bruno Haller
Communication Technology Laboratory, ETH Zürich

Support: Norbert Felber, Hubert Kaeslin
Integrated Systems Laboratory, ETH Zürich

This work was supported by the Swiss Commission of Scientific Research (KWF).

1 Project Abstract

1.1 Direct-Sequence Spread-Spectrum Systems

Microcellular wireless communications systems using direct-sequence spread-spectrum (DSSS) techniques are becoming widely used e. g. for wireless local area networks (wireless LAN). In a DSSS system, transmission bandwidth is spread by multiplication of the data bits with a pseudonoise (PN) code sequence, which e. g. allows the use of code division multiple access (CDMA). Typical indoor application problems like frequency-selective fading effects and multipath propagation are handled by a RAKE-receiver that permits the simultaneous reception of multiple propagation paths by correlation with different synchronized versions of the spreading code and by multipath diversity combining.

1.2 SHIVA Operation

SHIVA integrates the code correlation and data demodulation operations of a spread-spectrum RAKE-receiver on one chip. Correlation of the received signal with the synchronized PN code is performed within six correlator arms, which are set to different signal delays caused by the multipath propagation. All arms can initially be used for code acquisition. In reception mode, two arms are dedicated for code tracking while the remaining four arms perform data correlation. The demodulation part performs phase estimation and correction and a weighted path recombination of the correlated data bits. For code frequency tracking, a frequency error detection signal is generated. The chip handles BPSK and QPSK modulated data with differential or direct encoding as well as BPSK and QPSK spreading codes.

SHIVA accepts PN code lengths of 15 – 1023 chips with chip rates up to $f_c = 16.4 MHz$ and using four samples per chip. This allows typical data transfer rates of $16\,kbit/s$ – $2.2\,Mbit/s$ with processing gains of 12 – $27\,dB$. The chip can be controlled and data be read out via a 16-bit processor interface.

1.3 SHIVA Application

In addition to the SHIVA correlator/demodulator chip, the spread-spectrum receiver requires a quadrature demodulator, two A/D converters and two chip matched filters at the input side. Moreover, a PN code generator provides the spreading codes and the entire system is controlled by a digital signal processor (DSP).

SHIVA is a prototype chip which is used for verification of the system architecture and the data processing algorithms. A final VLSI is planned for higher data rates and for integration of even more system components on one chip.

2 Architecture and Operation

Figure 1 shows the architecture of SHIVA. The three main building blocks consisting of the DSP interface, the correlator arms and the demodulator are described in the subsequent sections. The remaining smaller circuit parts are described in the following.

Two internal clocks are generated by division of the master clock (Clk) by two resp. four. The few circuit parts that process the incoming signal samples are clocked by the master clock ($65.5\,MHz$ = sample rate). The demodulator works at half this frequency ($32.7\,MHz$) in order to get 30 computation cycles per received symbol (smallest code length is 15 chips). The rest of the circuit including the processor interface and the correlator arms are clocked at $16.4\,MHz$ (chip rate).

The incoming signal samples (SI, SQ) are registered, formatted appropriately (according to Format) and then applied to the six correlator arms.

The two PN codes (PN1, PN2) are fed into 32 bit long shift registers to generate the desired code delay window of 32 chips. Then they are converted for BPSK or QPSK spreading accordingly and also applied to the correlators.

A counter, which is reset by CodeStart, provides the actual PN code chip number for the correlators to determine the start of their individually delayed code sequence.

Finally a very simple control logic block generates the triggering signals for sampling the correlator outputs and starting the demodulation process.

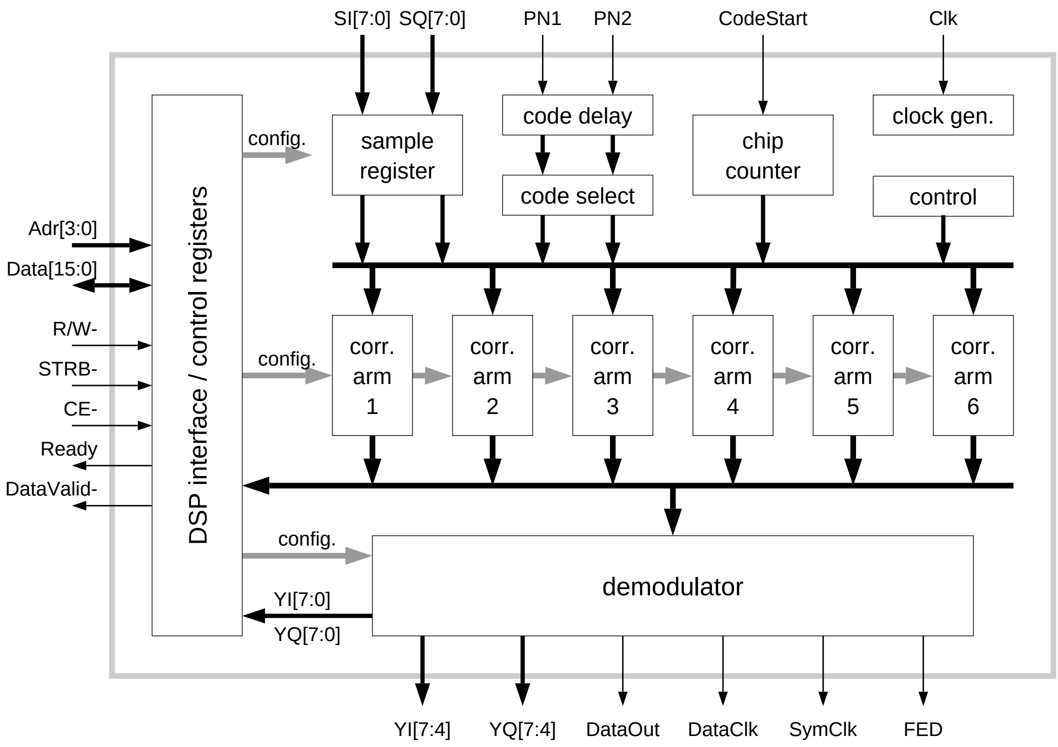

Figure 1: Block diagram of SHIVA.

2.1 Correlator Arms

Shiva contains six equivalent correlator arm circuits. All of them can be used in scan mode for code acquisition, but only four arms (2, 3, 5 and 6) have their outputs connected to the demodulator and thus can be used in data despreading and demodulation mode.

QPSK despreading of the received signal is performed by correlation with the synchronized complex spreading sequence (PN1, PN2), i. e. by multiplication with the conjugate complex spreading sequence and integration over the length of the code (see Fig. 2). For BPSK despreading, PN2 is set constantly to zero.

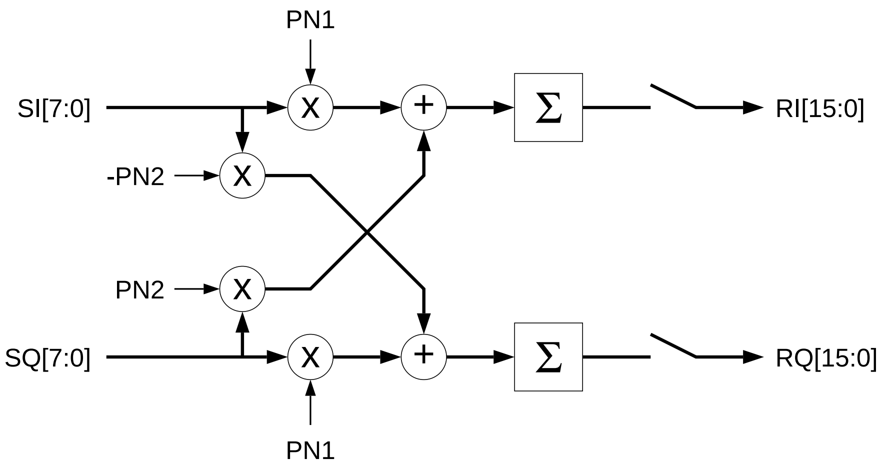

Figure 2: QPSK despreading.

Figure 3 depicts the block diagram of a correlator arm. The arm position register holds the code delay value for the corresponding arm. The input register loads one out of four sequentially arriving samples per chip according to the two LSBs of the arm position. The higher order bits of the arm position select from PN1 and PN2 one chip out of the 32 chip window generated by the chip delay shift register. The signal samples are then multiplied by these code bits (i. e. negated, zeroed or left unchanged) and integrated in the subsequent accumulator. When the chip number equals the delay position of the correlator arm, the integration result is read out and the accumulator is reset. A window of eight bits is then selected out of the 16 bit wide correlation result according to a scale value (programmed via the DSP interface). If an overflow is detected on the bits lying above the scaling window, the overflow flag is set and the 8 bit result is saturated to the maximum or minimum value respectively.

Because the correlation results of the different arms do not become valid at the same time caused by the different code delays, a second register is necessary in each arm (not shown in Fig. 3). These registers are enabled simultaneously in all arms at the moment when all arms have valid correlation results belonging to the same data symbol. At the next clock cycle, the demodulator can start its computations with all correlator outputs being valid for an entire symbol length.

In scan mode, the correlation results have to be processed by the controller DSP and thus must be readable via the processor interface. Because the real and the imaginary parts of a result are read out separately, it has to be guaranteed that they both belong to the same data symbol. This is realized with an additional register which holds the corresponding imaginary part (RQ) after the real part (RI) is read.

In scan mode, the arm position is automatically incremented by 1, 2, 4 or 8 chips (programmed via the processor interface) after a valid correlation result for the actual arm position has been read out. Each arm generates a valid and a lost flag for its correlation outputs. The valid flag is set whenever a correct value is written to the output register. The lost flag is set when a new value is written to the output register before the valid old value has been read. Both flags are reset when the output is read or when the arm position is changed.

All signals going to the processor interface are written out to a common bus by tristate buffers.

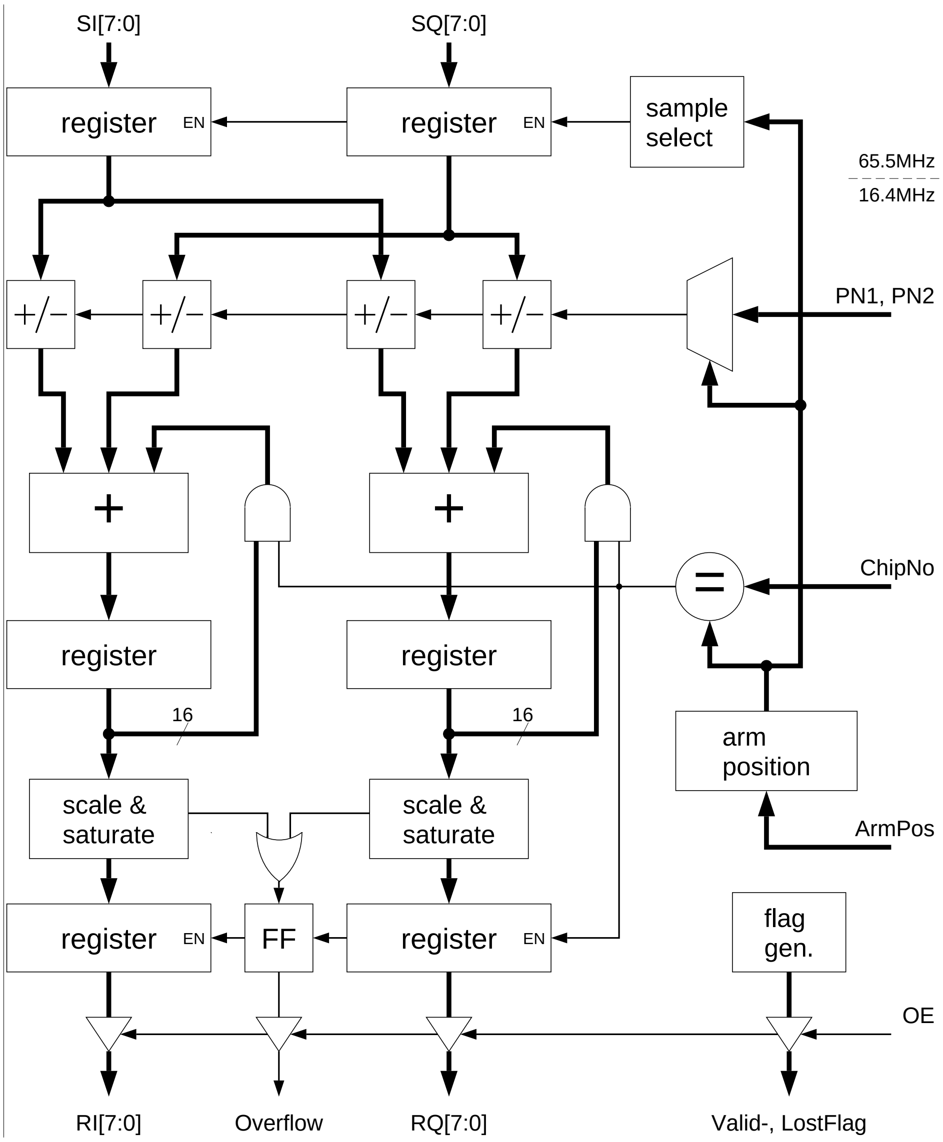

Figure 3: Block diagram of correlator arm.

Application Notes

- The valid and lost flags are reset when the real part of the correlator result (RI) is read.
- In scan mode, the arm position is auto-incremented after the real part of the correlator result (RI) is read.
- The imaginary part of the correlator result (RQ) is buffered when the real part (RI) is read.
- If the arm position remains unchanged (normal mode), after reading a correlation result, the next correlation output is valid as soon as the current correlation interval finishes (max. *code length / chip rate*). Thus correlation results can be read out at the data rate if the processor is fast enough.
- If the arm position is auto-incremented (scan mode), the first correct correlation result with the new delay position is available at the latest three correlation runs (max. $3 \times$ *code length / chip rate*) after the old result was read.
- If the arm position is changed by the processor, the first correct correlation result with the new delay position is available at the latest three correlation runs (max. $3 \times$ *code length / chip rate*) after the new arm position was written.
- For code lengths larger than 32, the arm position register is auto-incremented *modulo 32* (i. e. delay window length), so that the window is scanned repeatedly without resetting the register from outside.
- For code lengths smaller than 32 (i. e. 15 and 31), the arm position register is not incremented anymore when the code length is reached. For a new scanning round the register has to be reset by the processor.

2.2 Demodulator

Demodulation proceeds as shown in the block diagram in figure 4. For each symbol, the complex samples $\mathbf{r}_i$ produced by the four correlator arms dedicated to demodulation are combined by multiplying each with the complex conjugate of the joint phase and weighting estimate $\hat{\mathbf{X}}_i$ for the arm:

$$\mathbf{Y}[k] = \sum_{i=1}^{4} \mathbf{r}_i[k] \cdot \overline{\hat{\mathbf{X}}}_i[k]$$

From this soft decision, the hard decision $\hat{\mathrm{d}}$ can be calculated by taking the signs of the real and imaginary parts of $\mathbf{Y}$. Depending on whether QPSK or BPSK demodulation is used and on whether the internal differential decoding is in effect, one or two output bits are calculated according to the following table (where I and Q stand for the real and imaginary sign bit of $\hat{\mathrm{d}}$ and I$'$ and I$'$ stand for the values of I and Q for the previous symbol):

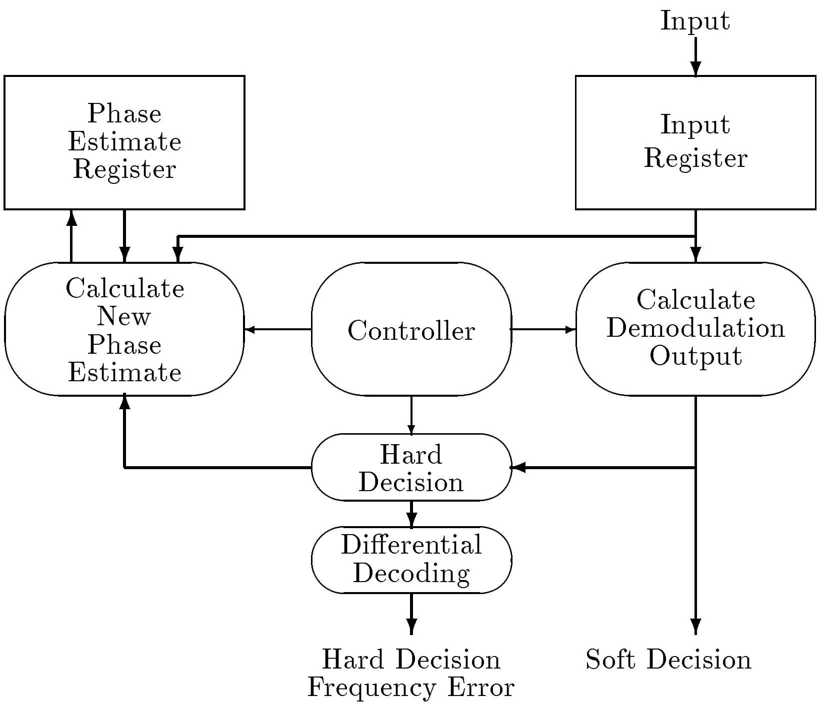

Figure 4: Demodulator Overview

BPSK/QPSK	Diff. Decoding	Bit 1		Bit 2	
QPSK	Yes	$\mathrm{I} \otimes \mathrm{I}'$	if $\mathrm{I} = \mathrm{Q}$	$\mathrm{Q} \otimes \mathrm{Q}'$	if $\mathrm{I} = \mathrm{Q}$
		$\mathrm{Q} \otimes \mathrm{Q}'$	otherwise	$\mathrm{I} \otimes \mathrm{I}'$	otherwise
	No	I		Q	
BPSK	Yes	$\mathrm{I} \otimes \mathrm{I}'$		—	
	No	I		—	

Furthermore, $\hat{\mathrm{d}}$ is used to update the $\hat{\mathbf{X}}_i$, employing an "exponential tail" IIR filter:

$$\hat{\mathbf{X}}_i[k] = \frac{N-1}{N}\hat{\mathbf{X}}_i[k-1] + \frac{1}{N}\mathbf{r}_i[k-1]\bar{\hat{\mathbf{d}}}[k-1]$$

where $N = 2^m, m \in \{0 \dots 7\}$. Finally, the sign of the frequency error is calculated and provided to the off-chip automatic frequency control.

Application Notes

- The choice of BSPK vs. QPSK mode and enabling differential decoding should be done at startup. Changing these parameters during operation is not recommended.
- The IIR filter parameter m may be changed at any time while SymClk is high.
- The demodulator is clocked at 32.7 MHz (1/2 of the master clock).

2.3 Processor Interface

The entire RAKE-receiver is controlled by a digital signal processor (TI TMS320 family). Its main task is to setup the system parameters used by Shiva and to control the correlator arm positions during code acquisition and scanning. Therefore all control parameters must be set and all required correlator and demodulator data must be read by the DSP via a dedicated asynchronous processor interface on Shiva. Fig. 5 shows the block diagram of the interface.

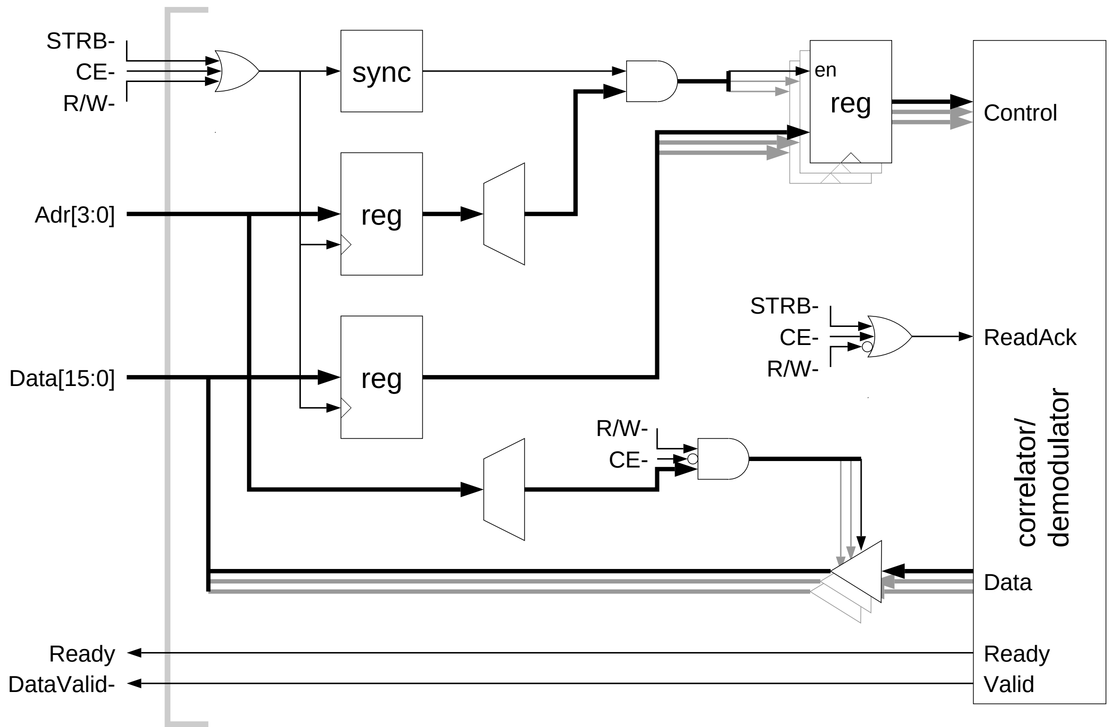

Figure 5: Block diagram of processor interface.

In write mode, the data and the address are loaded into the input registers on the rising edge of STRB. The strobe is then synchronized and enables the loading of the data into the control register specified by the address.

The read access is fully combinational except when the data to be read is currently changing on the chip. In this case the Ready-signal is tied low thus forcing the processor to insert wait cycles in its read access. As soon as the addressed data value becomes stable, Ready is tied high again and the processor can finish the read cycle. In read mode, the rising edge of STRB is interpreted on Shiva as acknowledge of the processor's read access and therefore can be used to reset the corresponding valid and lost flags and to buffer the imaginary part when reading the corresponding real part of complex data items (for data consistency reasons mentioned above).

DataValid- is low when all not masked correlator arms have valid data at their outputs. This signal can be used as interrupt signal for the processor to start processing of the new correlator outputs.

The 16 bit wide processor interface can also be used as 8 bit interface. The exact timings of the interface signals are specified in a later section.

Application Notes

- When reading the complex data items (RI, RQ) and (YI, YQ), the real part RI resp. YI has always to be read first. The corresponding imaginary part is then buffered and remains unchanged until the real part is read again. For (RI, RQ), this is done in each correlation arm individually.
- On read accesses to RI and YI, the handshake signal Ready is tied low two interface clock cycle before the value to be read changes, and it remains low for three cycles. Otherwise, the signal has high impedance (open-drain), so that it can be connected outside to other chips as well (wired-and). Therefore, the Ready signal must be tied high externally by a resistor (e. g. $500\,\Omega$).
- The valid and lost flags for (RI, RQ) and (YI, YQ) are reset asynchronously on the rising edge of STRB during the read operation of the real part.
- All write accesses are synchronized on chip, so that no data inconsistencies can result due to interface asynchronicities.
- The interface is clocked at 1/4 of the master clock (typ. 16.4 MHz).

Control registers (write)

Adr[3:0]	Data[15:0] Index	Range	Description
Correlator			
0 .. 5	[1:0]	0 .. 3	Arm position 0 .. 5 (sample number)
	[6:2]	0 .. 31	Arm position 0 .. 5 (chip number)
8	[2:0]	0 .. 4	Scaling factor for correlation results (0, 2, 4, 6, 8 bits shift left of 8-bit window)
9	[5:0]	0 .. 63	1 : scan mode, 0 : normal mode on arms 0 .. 5
	[7:6]	0 .. 3	Scan step size (1, 2, 4, 8 samples)
10	[5:0]	0 .. 63	Data valid mask for arms 0 .. 5 (1 : include, 0 : not include for generation of DataValid-)
	[7:6]	0, 1, 2	Code select: 0 : BPSK spreading by PN1 1 : QPSK spreading by (PN1, PN2) 2 : 2×BPSK (arm 0 .. 2 : PN1, arm 3 .. 5 : PN2)
Demodulator			
11	[2:0]	0 .. 7	Filter coefficient m
12	[0]	0, 1	Modulation type (0 : BPSK, 1 : QPSK)
	[1]	0, 1	Differential decoding (0 : off, 1 : on)

Data and status registers (read)

Adr[3:0]	Data[15:0] Index	Range	Description
0 .. 5	[7:0]	-128 .. 127	Correlator output real (RI) of arm 0 .. 5
	[15]	0, 1	0 : data valid on arm 0 .. 5
6	[7:0]	-128 .. 127	Demodulator output real (YI)
7	[5:0]		1 : correlator data valid on arms 0 .. 5
	[6]		1 : correlator data valid on all non-masked arms
	[7]		1 : overflow on any correlator arm detected
	[13:8]		1 : correlator data lost on arms 0 .. 5
	[14]		1 : demodulator data valid
	[15]		1 : demodulator data lost
8 .. 13	[7:0]	-128 .. 127	Correlator output imag. (RQ) of arm 0 .. 5
	[15]	0, 1	1 : overflow on correlator arm 0 .. 5 (RI or RQ) detected
14	[7:0]	-128 .. 127	Demodulator output imag. (YQ)
15	[5:0]		1 : correlator data lost on arms 0 .. 5
	[6]		1 : demodulator data valid
	[7]		1 : demodulator data lost
	[12:8]		Test signals

3 Testability

3.1 Test Structures

For Shiva, a high controllability and observability has been achieved using a partial scan with two separate scan paths which contain only 178 out of totally 1 304 flipflops. Fig. 6 gives an overview of the data path and the inserted scan registers.

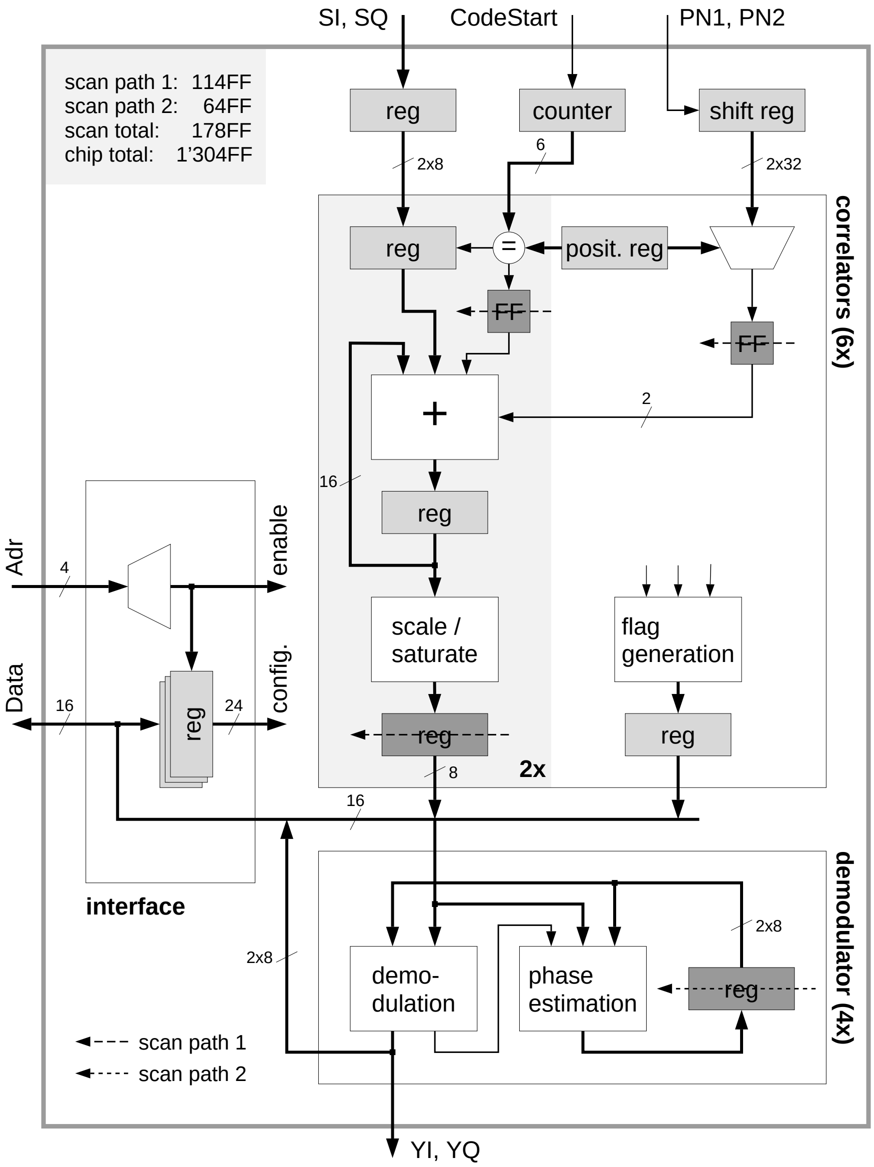

Figure 6: Partial scan of Shiva.

Scan path 1 (correlator arms)

Bit No.	FlipFlop	Pin
	Scan input	Scan1In (36)
Correlator arm 1		
1 .. 8	Bit 0 .. 7 of output register RQ (0 : LSB)	
9 .. 16	Bit 0 .. 7 of output register RI (0 : LSB)	
17	Output flipflop of delay multiplexer for PN1	
18	Output flipflop of delay multiplexer for PN2	
19	1. delay flipflop of flag generation enable signal	
Correlator arm 2		
20 .. 27	Bit 0 .. 7 of output register RQ	
28 .. 35	Bit 0 .. 7 of output register RI	
36	Output flipflop of delay multiplexer for PN1	
37	Output flipflop of delay multiplexer for PN2	
38	1. delay flipflop of flag generation enable signal	
Correlator arm 3		
39 .. 46	Bit 0 .. 7 of output register RQ	
47 .. 54	Bit 0 .. 7 of output register RI	
55	Output flipflop of delay multiplexer for PN1	
56	Output flipflop of delay multiplexer for PN2	
57	1. delay flipflop of flag generation enable signal	
Correlator arm 4		
58 .. 65	Bit 0 .. 7 of output register RQ	
66 .. 73	Bit 0 .. 7 of output register RI	
74	Output flipflop of delay multiplexer for PN1	
75	Output flipflop of delay multiplexer for PN2	
76	1. delay flipflop of flag generation enable signal	
Correlator arm 5		
77 .. 84	Bit 0 .. 7 of output register RQ	
85 .. 92	Bit 0 .. 7 of output register RI	
93	Output flipflop of delay multiplexer for PN1	
94	Output flipflop of delay multiplexer for PN2	
95	1. delay flipflop of flag generation enable signal	
Correlator arm 6		
96 .. 103	Bit 0 .. 7 of output register RQ	
104 .. 111	Bit 0 .. 7 of output register RI	
112	Output flipflop of delay multiplexer for PN1	
113	Output flipflop of delay multiplexer for PN2	
114	1. delay flipflop of flag generation enable signal	
	Scan output	Scan1Out (37)

- The scan path 1 improves the observability of the correlator arm circuits and makes the demodulator inputs (i. e. correlator outputs) controllable.
- It is clocked at 1/4 of the master clock (typ. 16.4 MHz).
- While the scan path 1 is enabled, no other parts of the circuit are disabled.

Scan path 2 (demodulator)

1. dimension		
Bit No.	FlipFlop	Pin
	Scan input	Scan2In (40)
1 .. 8	Bit 0 .. 7 of feedback register 2 for imag. part (0 : LSB)	
9 .. 16	Bit 0 .. 7 of feedback register 2 for real part (0 : LSB)	
	Scan output	Scan2Out (41)
2. dimension		
Reg. No.	Register	
1	Scan input : feedback register 2 (real & imag. part)	
2	Feedback register 3	
3	Feedback register 4	
4	Feedback register 1	
1	Scan output : feedback register 2	

- The scan path 2 makes the demodulator feedback registers observable and controllable.
- Only feedback register 2 is scannable (Scan2Mode = 0). In shift mode (Scan2Mode = 1), all four feedback registers are connected in a ring (i. e. circular shift register), so that every register can be read or written by shifting the value into resp. out of register 2.
- It is clocked at half the master clock frequency (typ. 32.7 MHz).
- While the scan path 2 is enabled, running demodulations are not halted, whereas the starting of new demodulation runs is disabled.

Additional test points

Additional signals of one correlator arm are made observable via the processor interface. They can be watched on the high order data bits of status register 15.

Adr[3:0]	Bit	Signal
15	Data[8]	Enable signal for sample input register of correlator arm 1
15	Data[9]	Result bit of comparison of arm position and chip number
15	Data[10]	Count flag for enabling auto-increment of arm position
15	Data[11]	Output bit of delay multiplexer for PN2
15	Data[12]	Output bit of delay multiplexer for PN1

3.2 Fault Simulation

For fault simulation and test vector generation, a special non-functional simulation has been written in order to achieve a high fault coverage with an acceptable number of test vectors.

Fault simulation results:

Number of test vectors		5 251
Stuck-at faults	total / not detected	11 888 / 162
	coverage	98.6 %
Stuck-at & open faults	total / not detected	38 434 / 1 234
	coverage	96.8 %

4 Application of SHIVA

Figure 7 shows the system overview of a direct-sequence spread-spectrum RAKE-receiver using SHIVA (see also [1]. Down-conversion of the RF signal and filtering has to be performed outside of SHIVA as well as PN code generation. Additionally a DSP is used for controlling the operation of SHIVA via the processor interface.

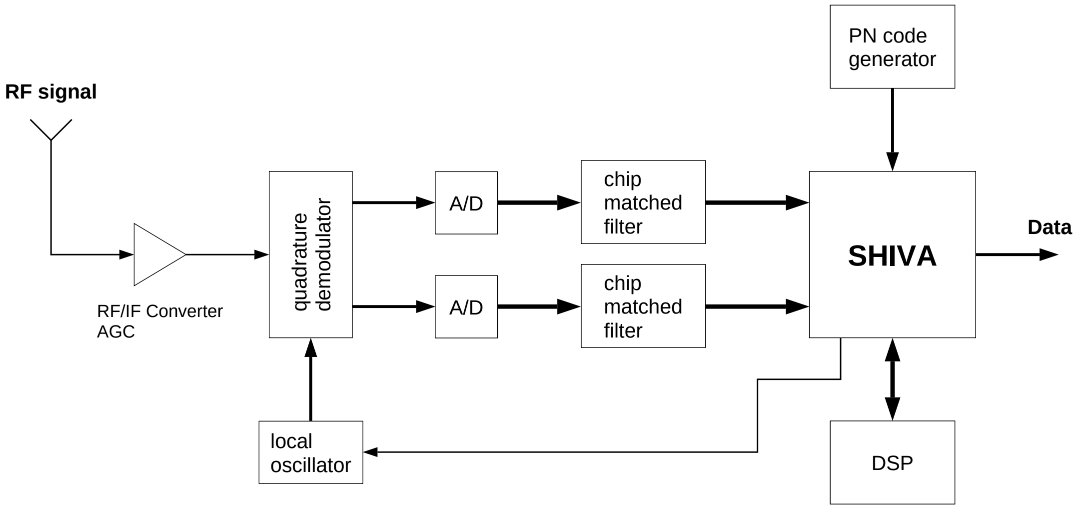

Figure 7: Application of SHIVA.

Typical system parameters of the RAKE-receiver:

Data modulation	BPSK / QPSK
Sample rate	65.5 MHz
Chip rate	16.4 MHz
Spreading factor	15 – 1023 (12 – 30 dB gain)
Data rate	16 $kbit/s$ – 2.2 $Mbit/s$
Spreading	BPSK / QPSK
RF signal frequency	2.44 GHz

5 SHIVA Floorplan and Layout

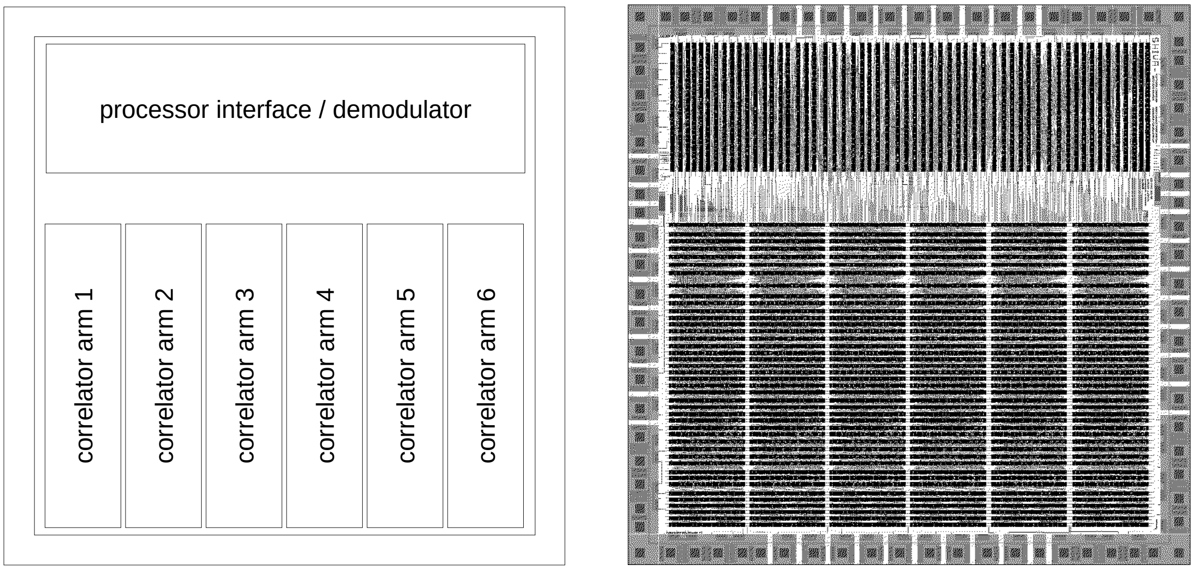

Figure 8: Floorplan and layout of SHIVA.

6 Data Sheet

6.1 Technical Data

SHIVA

Process Technology		
VLSI Technology	1.5μm double-metal CMOS n-well	cmn16
Logic Structures		
Controllers:	FSM 1 (demodulator)	Moore
	used states / inputs / outputs	30 / 1 / 15
Test structures:	partial scan path 1 (correlators)	114 FFs
	partial scan path 2 (demodulator, 2-D)	64 FFs (4 $\times$ 16 FFs)
Pads:	inputs – TTL compatible	35
	outputs – TTL compatible	17
	bidirectionals – TTL compatible	16
	power: core / pad-frame	4 / 4
	ground: core / pad-frame	4 / 4
Physical Data		
Standard-cell blocks:	block 1 (interface, demodulator)	$6.5\,mm \times 1.7\,mm$
	cells / transistors	1 646 / 25 621
	blocks 2 – 7 (6 correlator arms)	$1.0\,mm \times 4.2\,mm$
	cells / transistors	661 / 11 640
Total:	cells / transistors	5 612 / 95 461
Chip geometry:	area of core	$42.5\,mm^2$
	area of chip	$7.6 \times 7.6 = 57.4\,mm^2$
Package:	ceramic leadless chip carrier, 84 pins	LCC 84
Timing Data		
Working frequency:	system clock (maximum, worst case)	$65.5\,MHz$
	internal clocks	$32.7\,MHz$ / $16.4\,MHz$
	data transfer rate	$16\,kbit/s - 2.2\,Mbit/s$

6.2 Package

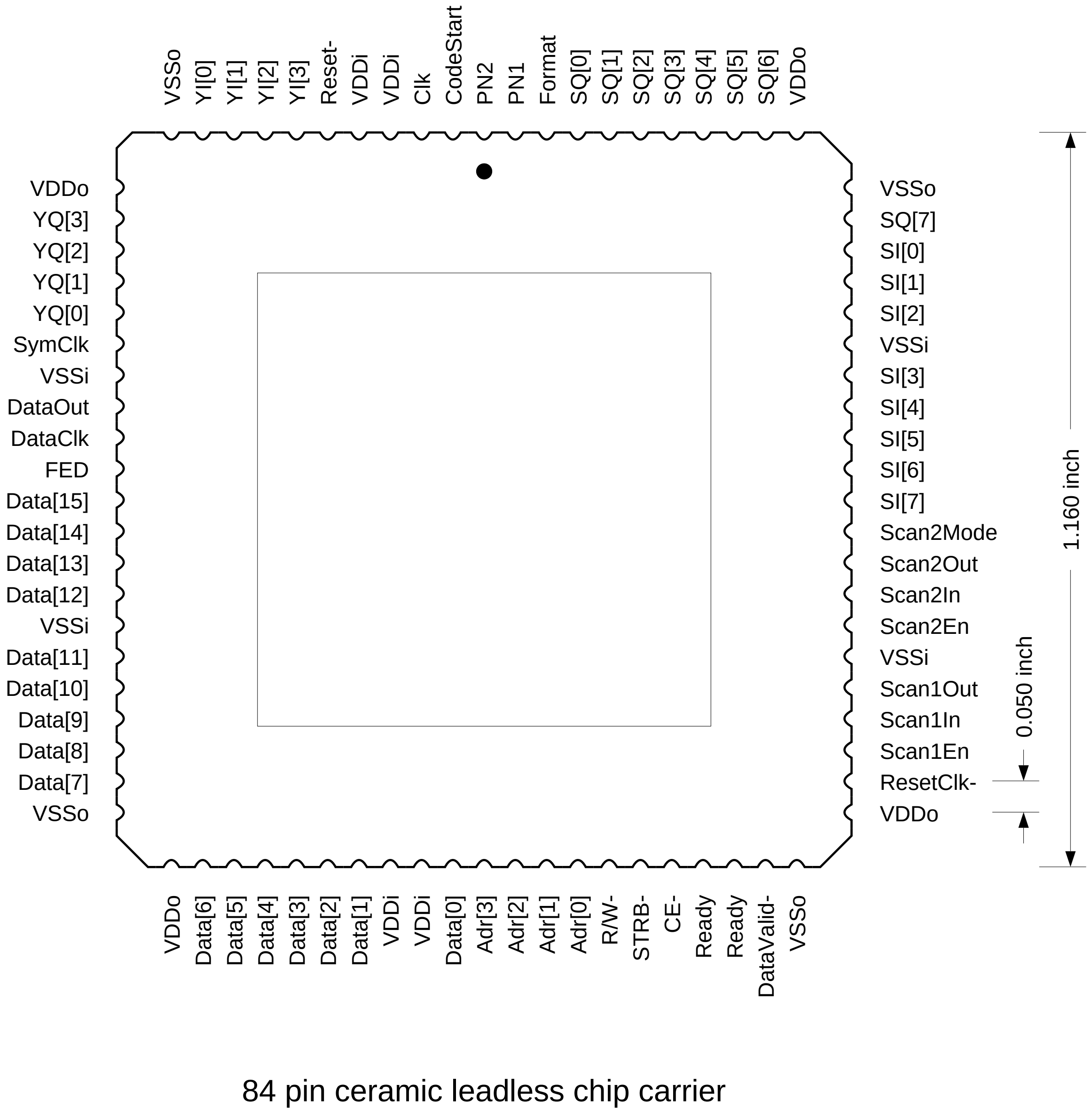

Figure 9: Shiva pin layout.

6.3 Pin Description

Signal	Pin	Type	Description
Processor interface			
Adr[3]	43	I	4 bit address bus (LSB : Adr[0])
Adr[2]	44	I	
Adr[1]	45	I	
Adr[0]	46	I	
Data[15]	22	I/O	16 bit bidirectional data bus (LSB : Data[0])
Data[14]	23	I/O	
Data[13]	24	I/O	
Data[12]	25	I/O	
Data[11]	27	I/O	
Data[10]	28	I/O	
Data[9]	29	I/O	
Data[8]	30	I/O	
Data[7]	31	I/O	
Data[6]	34	I/O	
Data[5]	35	I/O	
Data[4]	36	I/O	
Data[3]	37	I/O	
Data[2]	38	I/O	
Data[1]	39	I/O	
Data[0]	42	I/O	
R/W-	47	I	Read/write enable (0 : write control registers, 1 : read data or status registers), controls bidirectional data bus
STRB-	48	I	Data strobe on rising edge (*write access* : data are strobed into internal register, *read access* : signals chip that data is being read)
CE-	49	I	Chip enable, active low (1 : no registers are written, data bus is high impedance)
Ready	50 51	O O O	Interface ready signal for full handshake (open-drain) (0 (tied low) : processor inserts wait cycles during read access, 1 (high impedance) : processor can complete read access)
DataValid-	52	O	Data valid signal for processor interrupt, active low (0 : output data of all non-masked correlator arms are valid)

Signal	Pin	Type	Description
Inputs			
SI[7]	64	I	Signal input real (I) (@ 65.5 MHz, LSB : SI[0])
SI[6]	65	I	
SI[5]	66	I	
SI[4]	67	I	
SI[3]	68	I	
SI[2]	70	I	
SI[1]	71	I	
SI[0]	72	I	
SQ[7]	73	I	Signal input imag. (Q) (@ 65.5 MHz, LSB : SQ[0])
SQ[6]	76	I	
SQ[5]	77	I	
SQ[4]	78	I	
SQ[3]	79	I	
SQ[2]	80	I	
SQ[1]	81	I	
SQ[0]	82	I	
Format	83	I	Input format of SI, SQ (0 : 2's complement, 1 : offset-128)
PN1	84	I	PN codes 1 & 2 for BPSK/QPSK spreading (@ 16.4 MHz)
PN2	1	I	
CodeStart	2	I	Code start signal (1 : first bit of code sequence on PN1, PN2)
Clk	3	I	Master clock, 65.5 MHz
Reset-	6	I	Global asynchronous reset
Outputs			
YI[3]	7	O	4 MSBs of demodulator output real, valid at rising edge of SymClk (LSB : YI[0])
YI[2]	8	O	
YI[1]	9	O	
YI[0]	10	O	
YQ[3]	13	O	4 MSBs of demodulator output imag., valid at rising edge of SymClk (LSB : YQ[0])
YQ[2]	14	O	
YQ[1]	15	O	
YQ[0]	16	O	
SymClk	17	O	Symbol clock
DataOut	19	O	Detected data bit(s), valid at rising edge of DataClk ($BPSK$: 1 bit, $QPSK$: 2 bits sequentially)
DataClk	20	O	Data clock
FED	21	O	Frequency error detector (sign bit), valid at rising edge of SymClk

Signal	Pin	Type	Description
Test			
ResetClk-	55	I	Clock divider reset (for test)
Scan1En	56	I	Correlator scan path enable (1 : scan mode, 0 : normal mode)
Scan1In	57	I	Correlator scan path input
Scan1Out	58	O	Correlator scan path output
Scan2En	60	I	Demodulator scan path enable (1 : scan mode, 0 : normal mode)
Scan2In	61	I	Demodulator scan path input
Scan2Out	62	O	Demodulator scan path output
Scan2Mode	63	I	Demodulator scan path mode (0 : scan mode, 1 : shift mode)
Power			
VDDo	12	P	Power ($5\,V$), pad-frame
	33	P	
	54	P	
	75	P	
VDDi	4	P	Power ($5\,V$), core
	5	P	
	40	P	
	41	P	
VSSo	11	P	Ground ($0\,V$), pad-frame
	32	P	
	53	P	
	74	P	
VSSi	18	P	Ground ($0\,V$), core
	26	P	
	59	P	
	69	P	

Pin	Signal	Type	Pad	Pin	Signal	Type	Pad
1	PN2	I	TTL	43	Adr[3]	I	TTL
2	CodeStart	I	TTL	44	Adr[2]	I	TTL
3	Clk	I	TTL	45	Adr[1]	I	TTL
4	VDDi	P		46	Adr[0]	I	TTL
5	VDDi	P		47	R/W-	I	TTL
6	Reset-	I	TTL	48	STRB-	I	TTL
7	YI[3]	O	TTL, $4mA$	49	CE-	I	TTL
8	YI[2]	O	TTL, $4mA$	50	Ready	O	TTL, $16mA$, open-drain
9	YI[1]	O	TTL, $4mA$	51	Ready	O	TTL, $16mA$, open-drain
10	YI[0]	O	TTL, $4mA$	52	DataValid-	O	TTL, $4mA$
11	VSSo	P		53	VSSo	P	
12	VDDo	P		54	VDDo	P	
13	YQ[3]	O	TTL, $4mA$	55	ResetClk-	I	TTL, pull-up
14	YQ[2]	O	TTL, $4mA$	56	Scan1En	I	TTL, pull-down
15	YQ[1]	O	TTL, $4mA$	57	Scan1In	I	TTL, pull-down
16	YQ[0]	O	TTL, $4mA$	58	Scan1Out	O	TTL, $4mA$
17	SymClk	O	TTL, $4mA$	59	VSSi	P	
18	VSSi	P		60	Scan2En	I	TTL, pull-down
19	DataOut	O	TTL, $4mA$	61	Scan2In	I	TTL, pull-down
20	DataClk	O	TTL, $4mA$	62	Scan2Out	O	TTL, $4mA$
21	FED	O	TTL, $4mA$	63	Scan2Mode	I	TTL, pull-down
22	Data[15]	I/O	TTL, $8mA$	64	SI[7]	I	TTL
23	Data[14]	I/O	TTL, $8mA$	65	SI[6]	I	TTL
24	Data[13]	I/O	TTL, $8mA$	66	SI[5]	I	TTL
25	Data[12]	I/O	TTL, $8mA$	67	SI[4]	I	TTL
26	VSSi	P		68	SI[3]	I	TTL
27	Data[11]	I/O	TTL, $8mA$	69	VSSi	P	
28	Data[10]	I/O	TTL, $8mA$	70	SI[2]	I	TTL
29	Data[9]	I/O	TTL, $8mA$	71	SI[1]	I	TTL
30	Data[8]	I/O	TTL, $8mA$	72	SI[0]	I	TTL
31	Data[7]	I/O	TTL, $8mA$	73	SQ[7]	I	TTL
32	VSSo	P		74	VSSo	P	
33	VDDo	P		75	VDDo	P	
34	Data[6]	I/O	TTL, $8mA$	76	SQ[6]	I	TTL
35	Data[5]	I/O	TTL, $8mA$	77	SQ[5]	I	TTL
36	Data[4]	I/O	TTL, $8mA$	78	SQ[4]	I	TTL
37	Data[3]	I/O	TTL, $8mA$	79	SQ[3]	I	TTL
38	Data[2]	I/O	TTL, $8mA$	80	SQ[2]	I	TTL
39	Data[1]	I/O	TTL, $8mA$	81	SQ[1]	I	TTL
40	VDDi	P		82	SQ[0]	I	TTL
41	VDDi	P		83	Format	I	TTL
42	Data[0]	I/O	TTL, $8mA$	84	PN1	I	TTL

7 Timing Specifications

Parameter			slow	typ	fast	meas	Unit
Clock Timing							
f_{cp}	Clk frequency	max	66	125	200	>100	MHz
t_{cp}	Clk period	min	15	8	5	<10	ns
t_{cpl}	Clk low time	min	7	4	3	<4	ns
t_{cph}	Clk high time	min	7	4	3	<4	ns
Input Timings							
t_{su}(S)	SI, SQ setup time	min	-14	-7	-4	-1.6	ns
t_h(S)	SI, SQ hold time	min	17	9	6	8.0	ns
t_{su}(PN)	PN code inputs setup time	min	-10	-5	-3	0.4	ns
t_h(PN)	PN code inputs hold time	min	17	9	6	8.2	ns
t_{su}(Sin)	Scan path inputs setup time	min	6	4	2	3.0	ns
t_h(Sin)	Scan path inputs hold time	min	16	9	6	4.9	ns
Output Timings							
t_{con}(Y)	YI, YQ contamination delay time	min	33	17	10	17.5	ns
t_d(Y)	YI, YQ propagation delay time	max	40	21	13	22.4	ns
t_{con}(Dout)	Demodulator outputs cont. delay time	min	33	17	10	15.9	ns
t_d(Dout)	Demodulator outputs prop. delay time	max	38	20	13	22.5	ns
t_{con}(Iout)	Interface outputs cont. delay time	min	39	20	12	24.0	ns
t_d(Iout)	Interface outputs prop. delay time	max	54	28	18	25.0	ns
t_{con}(Sout)	Scan path outputs cont. delay time	min	33	17	10	17.7	ns
t_d(Sout)	Scan path outputs prop. delay time	max	40	21	13	23.4	ns
Processor Interface Read Timings							
t_{su}(RS)	CE-, R/W-, Adr su. t. before STRB- low	min	12	7	4	-1.0	ns
t_h(RS)	CE-, R/W-, Adr ho. t. after STRB- high	min	3	2	1	4.8	ns
t_w(RS)	STRB- low pulse duration	min	7	4	3	<4	ns
t_{en}(RD)	Data being driven after CE-, R/W-	min	14	7	4	–	ns
t_d(RD)	Data valid after CE-, R/W-, Adr valid	max	37	20	12	18.6	ns
t_{dis}(RD)	Data tri-state after CE-, R/W-	max	16	9	6	–	ns
t_d(RR)	Ready valid after CE-, R/W-, Adr	max	33	18	11	16.3 *	ns
Processor Interface Write Timings							
t_{su}(WS)	CE-, R/W- setup time before STRB- low	min	12	7	4	-1.0	ns
t_h(WS)	CE-, R/W- hold time after STRB- high	min	3	2	1	4.8	ns
t_w(WS)	STRB- low pulse duration	min	7	4	3	<4	ns
t_{su}(WD)	Adr, Data setup time before STRB- high	min	-9	-4	-2	-0.5	ns
t_h(WD)	Adr, Data hold time after STRB- high	min	12	7	4	7.7	ns

* = fall time; rise time = $40\,ns$ ($R = 500\,\Omega$ pull-up, $I_{OL} = 9.8\,mA$), $24.8\,ns$ ($R = 200\,\Omega$, $I_{OL} = 23\,mA$)

Timing measurement conditions:

	VDD	Temp.	Process	Pin load
slow	$4.5\,V$	$85°\,C$	worst case	$35\,pF$
typ	$5.0\,V$	$25°\,C$	typical	$35\,pF$
fast	$5.5\,V$	$25°\,C$	best case	$35\,pF$
meas	$5.0\,V$	$25°\,C$	measured	$35\,pF$

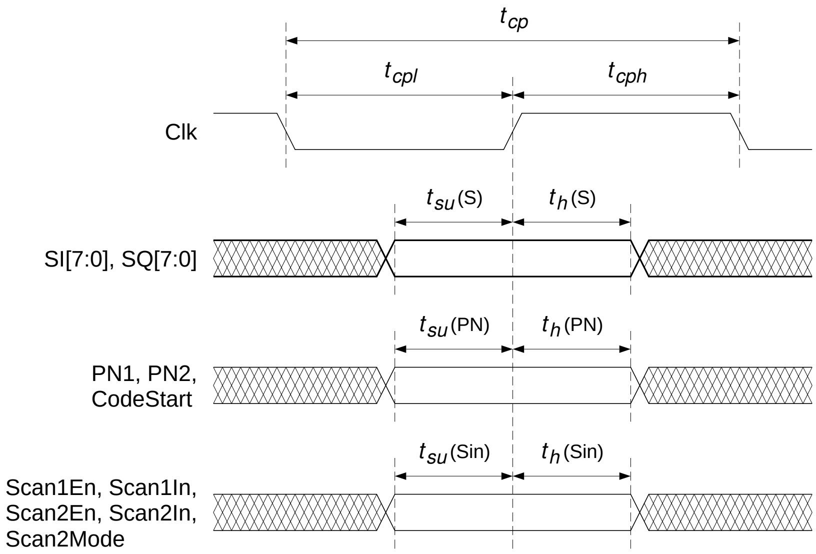

Figure 10: Timing diagram: Inputs.

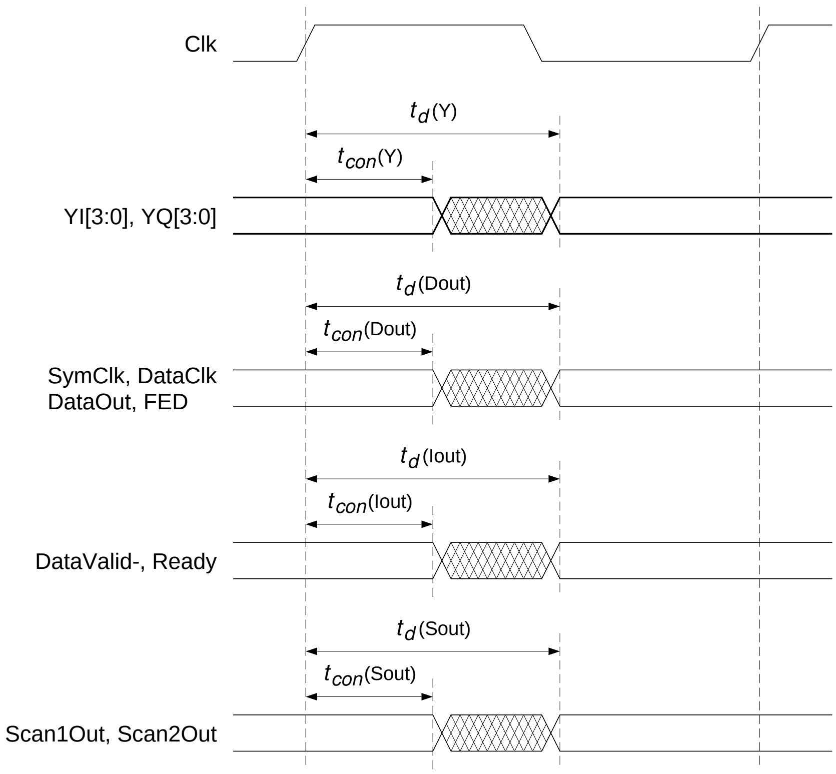

Figure 11: Timing diagram: Outputs.

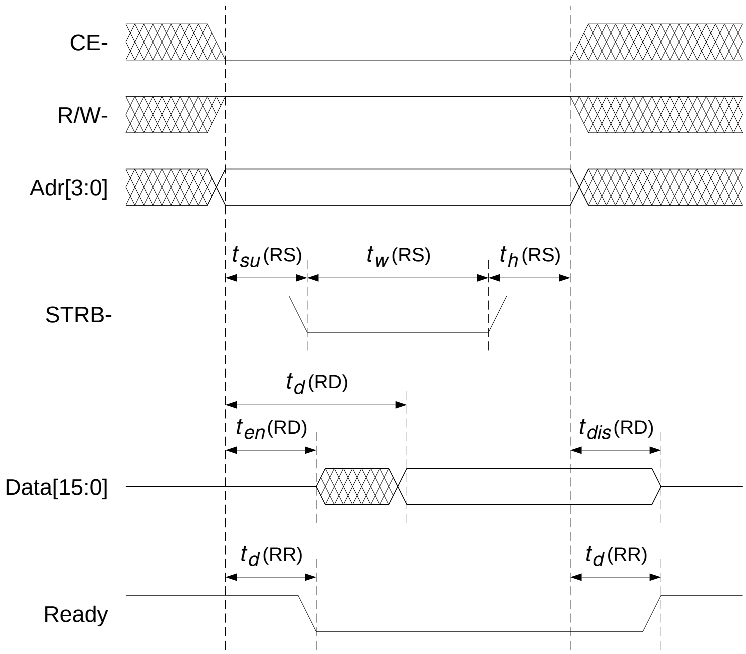

Figure 12: Timing diagram: Read access from processor interface.

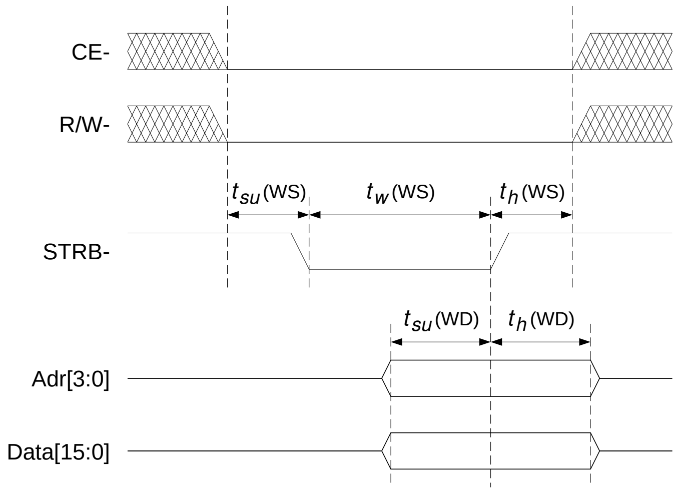

Figure 13: Timing diagram: Write access from processor interface.

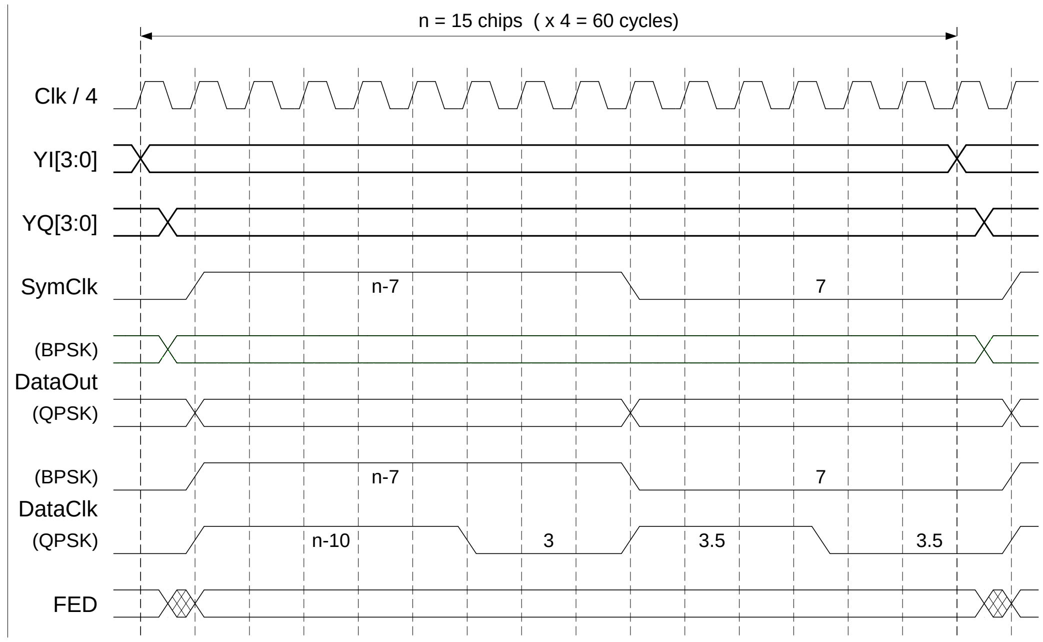

Figure 14: Logic timing diagram: Demodulator outputs and output clocks.

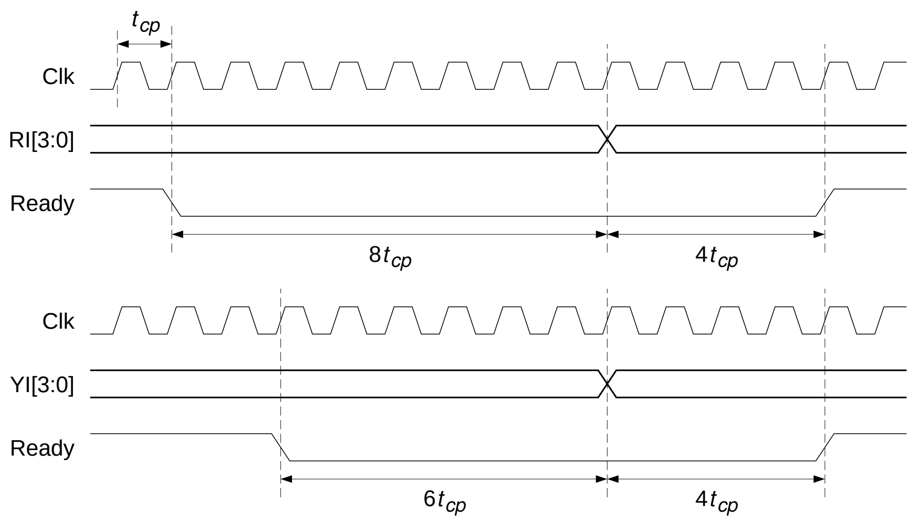

Figure 15: Logic timing diagram: Ready handshake signal timing (read access on RI resp. YI).

8 Electrical Characteristics

Recommended operating conditions:

Parameter		min	typ	max	Unit
VDD	Supply voltage	4.5	5	5.5	V
V_{IL}	Low-level input voltage	-0.5		0.8	V
V_{IH}	High-level input voltage	2.0		VDD	V
V_{OL}	Low-level output voltage,				
	$I_{OL} = 4\,mA$ (16 mA for Ready)			0.4	V
V_{OH}	High-level output voltage,				
	$I_{OH} = 4\,mA$ (16 mA for Ready)	2.4			V
T_J	Junction temperature	0		100	$^\circ C$

Electrical characteristics at $25^\circ\,C$:

Parameter		Test Conditions	min	typ	max	Unit
IDD	static current	$VDD = 5\,V,\ f_{cp} = 0\,MHz$		0.3		mA
IDD	dynamic current	$VDD = 5\,V,\ f_{cp} = 60\,MHz$		142		mA

9 Test Report

Totally 30 chips (two series of 15 chips each) have been fabricated and packaged by TIM3[1]/CMP. The packaged prototypes have been tested and their timing and power parameters measured on the IMS XL60 ASIC tester. Tests and measurements have been performed with a functional test vector set (~16 000 vectors) and a special test vector set with a high fault coverage (5 251 vectors, 98.6 % stuck fault coverage).

The tests have shown that only six out of 30 tested chips are fully functional (yield = 20 %). Six other chips have exactly localizable faults with small effect that still allow their usage as prototype circuits with only slightly reduced functionality. All other chips have more severe defects which result in failure of important parts of the circuit. Most of the defective circuits show a considerable increase in static current, too. Six defects on the metal layers have been localized visually under the microscope.

The six correct chips work up to 100 MHz (limit of the IMS tester). They consume a static current of about 0.3 mA and a dynamic current of 142 mA @ 60 MHz. The measured timings are given in the table in the timing specifications section. The following table summarizes the maximum measured static currents ($IDD_{stat.}$) and a fault characterization for all tested chips.

[1] Techniques de l'Informatique, des Mathématiques, de la Microélectronique et de la Microscopie quantitative, Grenoble Cedex, France

Chip No.	$IDD_{stat.}$ (mA)	Fault characterization
0	0.35	stuck-0 on multiplexer input I1 (S[2] = 1) in PN1 delay register in correlator arm 3 → false results on correlator 3 (demod. results are o.k.)
1	0.32	no fault
2	0.32	same fault as chip 0 → false results on correlator 3
3	0.32	no fault
4	3.80	circular area (Ø ≈ 100 μm, visible) with no metal-2 in interface/demodulator standard cell area → all correlator arms same false results
5	1.42	open-1 fault (visible) on metal-2 reset wire between correlator arms 2 and 3 → correlator arms 0..2 not resettable (except by power-up)
6	2.17	fault in demodulator → demodulator outputs false
7	5.19	fault ? → arm 1 most results false; arms 0, 2..5 most results o.k., same false results
8	12.48	fault in correlator arm 5 → RQ false if not saturated
9	16.10	fault ? → arms 0, 1, 4, 5 stuck-1 on RI[1], RQ[1]; arms 2, 3 more false result bits
10	5.50	fault in interface → all correlator arms same false results
11	11.90	fault in demodulator → all demodulator results false
12	0.24	fault ? → arms 0..3 same false results; arms 4, 5 same false results, but different from other arms
13	2.50	fault in demodulator → most demodulator results false
14	5.80	fault ? → all correlator arms same randomly false and correct result
15	0.32	no fault
16	0.32	no fault
17	84.10	big metal-1 short in pad ring (visible) → everything fails
18	1.99	fault in correlator arm 5 → false results on correlator 5
19	6.51	fault in correlator arm 1 → false results on correlator 1
20	0.96	fault in demodulator → demodulator results false
21	0.32	same fault as chip 0 → false results on correlator 3
22	20.00	circular area with no metal-2 in big routing channel (visible) → everything fails
23	0.51	stuck-0 on output RI[2] of correlator 3 → false results on correlator 3
24	0.32	no fault
25	187.00	circular area with no metal-2 in correlator 2 (visible) → everything fails (many stuck-at on outputs)
26	0.91	fault in correlator 4 → false results on correlator 4
27	4.09	fault ? → all correlator results false
28	23.40	open-0 on metal-2 output wire DataClk (visible) → everything else o.k.
29	0.32	no fault

All chips with faults in the correlator arms or the interface yield false results on the correlator and demodulator outputs, except for the chips 0, 2, 21 and 23 where the results of the faulty correlator arm 3 are not fed into the demodulator. The chips with faults in the demodulators, however, give correct results at the correlator outputs.

The following table summarizes the operation limitations for each chip:

Chip No.	Characterization
1, 3, 15, 16, 24, 29	fully functional
5	circuit reset correctly only after power-up
28	output DataClk not usable
0, 2, 21, 23	results of correlator 3 not usable (scan arm)
6, 11, 13, 20	demodulator results not usable
19	results of correlator 1 and demodulator not usable
26	results of correlator 4 and demodulator not usable
8, 18	results of correlator 5 and demodulator not usable
4, 7, 9, 10, 12, 14, 17, 22, 25, 27	entire circuit not usable

With more testing effort, the individual faults could be characterized and localized even more exactly. The implemented test structures are sufficient for efficient chip testing and moderate fault localization. However, additional test structures would be helpful for more exact and efficient fault localization, but would have had a considerable impact on chip area and circuit speed.

The distribution and diversity of the characterized faults lead to the conclusion that the bad yield can not be caused by design errors (e. g. due to timing problems). Additionally, the high measured static currents of most chips imply a high defect rate in a bad process run. The only open question is the fact that three chips have exactly the same stuck-at fault, especially when considering that no masks were used within this process.

Figure 16 shows the approximate fault locations estimated from the chips' behavior.

Further tests will be performed on the new HP83000-Tester for exact speed measurements.

10 Acknowledgements

We would like to thank Hans Kaufmann, Stephen Lingwood and Bruno Haller for providing us with all the required system specifications needed for carrying out a successful ASIC design.

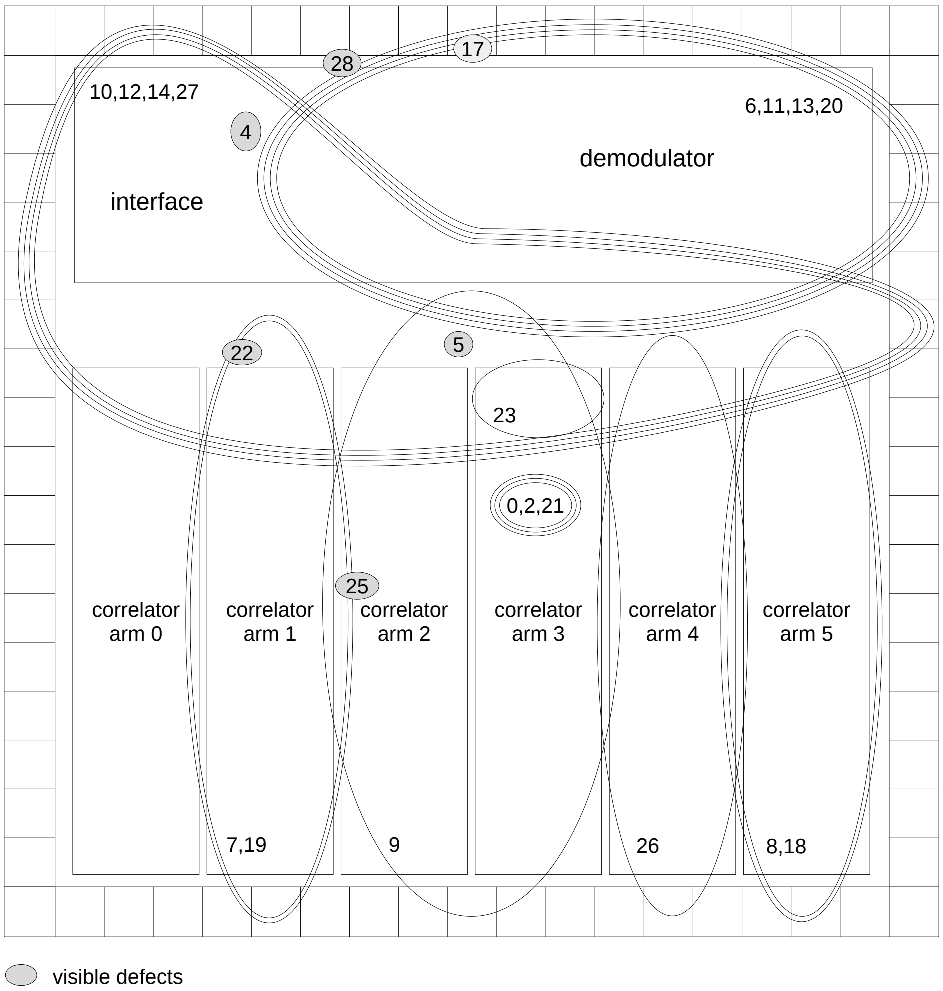

Figure 16: Estimated fault locations

References

[1] S. D. Lingwood, H. Kaufmann and B. Haller: *ASIC implementation of a direct-sequence spread-spectrum RAKE-receiver.* IEEE Vehicular Technology Conference, VTC'94, Stockholm, June 1994.

[2] D. Beeler and H. Kaufmann: *Time integrating correlator (TIC) for real-time processing of spread-spectrum signals.* IEEE 1990 Custom Integrated Circuits Conference, CICC'90, Boston, May 1990.

[3] H. Kaufmann: *Time integrating correlator (TIC).* Internal report, Ascom Tech AG, System Design, Hombrechtikon, 1993.

[4] *DIGSST: Wireless Data Modem for Local Area Communication.* Ascom, January 1992.

[5] U. Fawer and H. Kaufmann: *Digital Direct-Sequence Spread-Spectrum RAKE-Receiver – Algorithms and Implementation.* Communication Technology Laboratory, ETH Zürich, March 1992.

[6] M. Neeracher: *Quetzalcoatl: The Design of a Fast Digital Filter.* Integrated Systems Laboratory, ETH Zürich, July 1992.

[7] R. Dixon: *Spread Spectrum Systems.* John Wiley & Sons, 1984.