

Stamp9G45

Technical Reference

Stamp9G45: Technical Reference

Copyright © 2011 taskit GmbH

All rights to this documentation and to the product(s) described herein are reserved by taskit GmbH.

This document was written with care, but errors cannot be excluded. Neither the company named above nor the seller assumes legal liability for mistakes, resulting operational errors or the consequences thereof. Trademarks, company names and product names may be protected by law. This document may not be reproduced, edited, copied or distributed in part or in whole without written permission.

This document was generated on 2012-01-06T15:37:08+01:00.

Table of Contents

1. Introduction	1
2. Scope	2
3. Overview of Technical Characteristics	3
3.1. CPU	3
3.2. Memory	3
3.3. Interfaces and external signals	3
3.4. Miscellaneous	4
3.5. Power Supply	4
3.6. Dimensions	4
4. Hardware Description	5
4.1. Mechanics	5
4.2. AT91SAM9G45 Processor Core	5
4.3. Memory	5
4.3.1. NAND Flash	5
4.3.2. LPDDR-SDRAM	6
4.3.3. EEPROM	6
4.3.4. SRAM	6
4.4. Bus Matrix	7
4.5. Advanced Interrupt Controller (AIC)	7
4.6. Battery Backup	8
4.7. Reset Controller (RSTC)	8
4.8. Serial Number	8
4.9. Peripheral Input/Output Controller (PIO)	8
4.10. Clock Generation	9
4.10.1. Processor Clocks	9
4.10.2. Programmable Clocks	10
4.11. Power Management Controller (PMC)	10
4.11.1. Function	10
4.11.2. Power Management	11
4.12. Real-time Timer (RTT)	11
4.13. Timer Counter (TC)	11
4.14. Periodic Interval Timer (PIT)	12
4.15. Watchdog Timer	12
4.16. Real-time Clock (RTC)	12
4.17. True Random Number Generator (TRNG)	12
4.18. Peripheral DMA Controller (PDC)	12
4.19. Debug Unit (DBGU)	13
4.20. JTAG Unit	13
4.21. Two-wire Interface (TWI)	13
4.22. Multimedia Card Interface (MCI)	14
4.23. USB Host Port (UHP)	14
4.24. USB Device Port (UDP)	15
4.25. Ethernet MAC (EMAC)	15
4.26. Universal Synchronous Asynchronous Receiver and Transmitter (USART)	15
4.27. Synchronous Peripheral Interface (SPI)	17
4.28. Synchronous Serial Controller (SSC)	17

Stamp9G45

4.29. AC97 Controller (AC97C)	18
4.30. Image Sensor Interface (ISI)	18
4.31. LCD controller	18
4.31.1. LCDC Initialisation and LCD Power Sequencing	19
4.31.2. LCDC Frame Buffer	19
4.32. Touch Screen ADC Controller (TSADCC)	20
5. Design Considerations	21
5.1. Ethernet Controller (EMAC)	21
5.2. USB	21
5.2.1. USB Host Controller (UHP)	21
5.2.2. USB Device Controller (UDP)	22
5.3. Memory Bus	23
A. Peripheral Color Codes	25
B. Peripheral Identifiers	26
C. Address Map (Physical Address Space)	27
D. Stamp9G45 Pin Assignment	29
E. Stamp9G45 Electrical Characteristics	33
F. Stamp9G45 Clock Characteristics	34
G. Stamp9G45 Environmental Ratings	35
H. Stamp9G45 Dimensions	36
I. Starterkit Schematics	37

List of Figures

5.1. Buffered Memory Bus (PIOC) 1.8V - 3.3V	24
H.1. Stamp9G45 Dimensions	36
I.1. Starterkit FX8	37
I.2. Starterkit Buffer	38
I.3. Starterkit Memory	39
I.4. Starterkit Serial	40
I.5. Starterkit Ethernet	41
I.6. Starterkit USB, Power	42

List of Tables

4.1. Bus Matrix Masters	7
4.2. Bus Matrix Slaves	7
4.3. AT91SAM9G45 Clocks	10
4.4. AC97 I/O Lines	18
4.5. LCDC palette entry	19
4.6. LCDC 24 bit memory organization	19
B.1. Peripheral Identifiers	26
C.1. Physical Address Space	27
D.1. Pin Assignment IO Interface	29
D.2. Pin Assignment IO Interface	30
E.1. Electrical Characteristics	33
F.1. Clock Characteristics	34
G.1. Environmental Ratings	35

1. Introduction

The Stamp9G45 is intended to be used as a small size "intelligent" CPU module as well as a universal Linux CPU card. It can be used anywhere where restricted energy and space requirements play a role. The design of the Stamp9G45 is limited to the processors core needs like DDRAM and Flash, thus giving the customer a wide-ranged choice of configurations of the peripherals and environment. Featuring an integrated LCD/TFT and touch controller applications with graphical needs can be realized cost-efficient and individually.

The Stamp9G45 has all the necessary interfaces to support a huge variety of peripheral devices. Equipped with a 16-Bit parallel bus it gives fast access to a number of chips and additional devices.

The ARM architecture as a modern and widely supported processor architecture is currently the platform of choice for medium performance embedded devices. Almost all major processor manufacturers have ARM products in their portfolio.

The availability of the widespread operating system "Linux" for the ARM platform opens access to a broad range of software, including tools, drivers, and software libraries. Programs written for ARM can easily be employed on the PC platform for testing and debugging.

Examples of actual or potential applications are: protocol converters, measuring and test equipment, data-logging, as well as simple or more complex control and automation tasks.

2. Scope

This document describes the most important hardware features of the Stamp9G45. It includes all informations necessary to develop a customer specific hardware for the Stamp9G45. The Operating System Linux is described in a further document.

The manual comprises only a brief description of the AT91SAM9G45 processor, as this is already described in depth in the manual of the manufacturer Atmel. Descriptions of the ARM core ARM926EJ-S are available from Atmel and also at <http://www.arm.com>. It is much recommended to have a look at these documents for a thorough understanding of the processor and its integrated peripherals.

3. Overview of Technical Characteristics

3.1. CPU

Atmel AT91SAM9G45 Embedded Processor featuring an ARM926EJ-S™ ARM® Thumb® Core

- CPU Clock 400 MHz
- 32KB Instruction Cache
- 32KB Data Cache
- Memory Management Unit (MMU)
- 3.3V Supply Voltage, 1.8V Memory Bus Voltage, 1.0V Core Voltage

3.2. Memory

- 128 MB NAND flash memory (optional up to 1GB)
- 128 MB LPDDR-SDRAM (optional up to 512 MB)
- 64 KB SRAM
- 128 Bytes EEPROM
- Onboard Micro-SD Card Slot

3.3. Interfaces and external signals

- 2x 100-pin fine-pitch low-profile Connectors (Hirose FX8)
- Ethernet 10/100 Mbit MAC
- USB 2.0 High Speed Host
- USB 2.0 Full Speed OTG (USB-On-the-Go)
- USB 2.0 High Speed Device
- Four USARTs
- One UART
- One Synchronous Serial Controller (SSC, I²S)
- One Serial Peripheral Interface (SPI)
- Two Two Wire Interface (TWI, I²C)
- One High Speed MultiMedia Card Interface

- Three PWM
- 4-wire Touch Controller
- AC'97 Sound Controller
- LCD/TFT Controller (1280 x 860 pixels)
- JTAG debug port
- Digital Ports - up to 100 available
- Control Signals: IRQs, BMS, SHDN, WKUP
- Two Programmable Clocks
- Image Sensor Interface
- Analog-to-Digital Converter
- 16-Bit parallel CPU-Bus

Some of the various functions are realized by multiplexing connector pins; therefore not all functions may be used at the same time (see Appendix D, *Stamp9G45 Pin Assignment*)).

3.4. Miscellaneous

- Three 16-Bit Timer/Counter
- True Random Number Generator
- Real Time Timer (RTT), with battery backup support
- RTC
- Periodic Interval Timer (PIT)
- Watchdog Timer (WDT)
- Unique Hardware Serial Number

3.5. Power Supply

- 3.3V power supply
- 3V backup power supply, e.g. from a lithium battery

3.6. Dimensions

- Dimensions: 53.6x38x6 mm (WxDxH)

4. Hardware Description

4.1. Mechanics

The Stamp9G45 was designed as a flexible CPU-Module, which can be connected to base boards via 2x 100-pin fine pitch low profile Hirose ® FX8 connectors

The size of the Stamp9G45's PCB is only 53.6x38x6.0 mm fitting it in even the smallest design. While having implemented the sensible CPU, DDRAM and Flash design it still exports almost all possible CPU-Pins on it's connectors to allow a flexible design on base boards

The Stamp9G45 has an on-board Micro SD-Card slot, thus supporting even large memories needs in its compact design

4.2. AT91SAM9G45 Processor Core

The AT91SAM9G45 runs at 400 MHz with a memory bus frequency of 132 MHz.

Here are some of the most important features of the SAM9G45 ARM926EJ-S core:

- 32 Kbyte Data Cache, 32 Kbyte Instruction Cache, Write Buffer
- Two 32 Bit Data Bus
- ARM v4 and v5 Memory Management Unit (MMU)
- ARM v5 32-bit Instruction Set, ARM Thumb 16-bit Instruction Set supported
- DSP Instruction Extensions
- ARM Jazelle® Technology for Java® Acceleration
- EmbeddedICE™ Debug Communication Channel Support

Some of these features - like Jazelle - are currently not supported by the operating system of the product.

4.3. Memory

The Stamp9G45 is equipped with two 32-Bit external bus interfaces, EBI0 and EBI1. Only a 16-Bit bus of EBI0 is exported on the interface connectors of the Stamp9G45. The memory bus voltage is 1.8 V and runs at 133 MHz. The memory bus voltage is different from normal operating voltage, which is 3.3 V. This has to be considered, when designing additional peripherals connected to the memory bus. Eventually buffer chips are necessary.

4.3.1. NAND Flash

The Stamp9G45 is equipped with a 128 MB NAND flash with 100000 erase and write cycles. It is organized in 128KB blocks. Customer specific adaptations are possible up to 1 GB on-board NAND flash. It is connected to chip select three (NCS3) of the microcontroller.

NAND flash has a different organisation of transistors than the common used NOR flash. While it allows a much higher density and thus increases the storage amount, it leads to some differences which need to be kept in mind.

Typically NAND flash is organized in pages and blocks, similar to hard disks. Pages are 512, 2048 or 4096 bytes in size, typical block sizes are 16, 128, 256 or 512 KB. Reading and programming are performed on page basis. Programming can only be done sequentially in one block.

Additionally NAND flash requires bad block management, either by the driver software or by a separate controller chip. Most NAND devices are shipped with bad blocks. These are identified and marked according to a specified bad block strategy. Further bad blocks may be detected during runtime. They are detected via an ECC. If a bad block is detected the data is written to a different, good block, and the bad block table is updated. So the overall memory capacity gradually shrinks as more and more blocks are marked bad.

This error detection is done by software like U-boot and Linux. Additionally NAND flash is subject to limited write and erase cycles. These are typically 100.000 cycles per block. So it is highly recommended to use wear levelling filesystems.

4.3.2. LPDDR-SDRAM

The Stamp9G45 is equipped with 128MB LPDDR-SDRAM (Low power DDR-SDRAM). Customer specific adaptations allow configurations up to 512MB. In 128MB and 256MB configurations, the LPDDR-SDRAM is connected to EBIO. The external Bus is not affected. In 512MB configuration 256MB of the LPDDR-SDRAM are connected to chip select one (NCS1) of the microcontroller's EBIO.

DDR-SDRAM allows random access to any of its memory area and is volatile memory. DDR-SDRAM (Double Data Rate) takes over data at the rising and falling edge of a clock pulse, thus achieving almost twice the bandwidth than a similar connected SDRAM. It has a synchronous interface, that means it waits for a clock signal before responding to control inputs and is therefore synchronized with the CPU bus. The clock is used to drive a final state machine in the chip, which allows to accept new instructions, before the previous one has finished executing.

4.3.3. EEPROM

The Stamp9G45 is equipped with a 128 bytes EEPROM, connected to the Dallas™ 1 wire bus.

EEPROM stands for Electrically Erasable Programmable Read-Only Memory and is non-volatile memory, which is used to store small amounts of data like calibration or configuration data. EEPROMS are byte-wise erasable, thus allowing true random access.

4.3.4. SRAM

The Stamp9G45's microcontroller is equipped with 64 KB internal SRAM. The internal SRAM can be accessed in one bus cycle and may be used for time critical sections of code or interrupt handlers.

4.4. Bus Matrix

The bus matrix of AT91SAM-controllers allows many master and slave devices to be connected independly of each other. Each master has a decoder and can be defined specially for each master. This allows concurrent access of masters to their slaves (provided the slave is available).

The bus matrix is thus the bridge between external devices connected to the EBI, the microcontroller's embedded peripherals and the CPU core.

Master 0	ARM926™ Instruction
Master 1	ARM926™ Data
Master 2	PDC
Master 3	USB HOST OHCI
Master 4	DMA
Master 5	DMA
Master 6	ISI Controller DMA
Master 7	LCD DMA
Master 8	Ethernet MAC DMA
Master 9	USB Device High Speed DMA
Master 10	USB Host High Speed EHCI DMA
Master 11	Reserved

Table 4.1. Bus Matrix Masters

Slave 0	Internal SRAM
Slave 1	Internal ROM
	USB OHCI
	USB EHCI
	UDP High Speed RAM
	LCD User Interface
	Reserved
Slave 2	DDR Port 0
Slave 3	DDR Port 1
Slave 4	DDR Port 2
Slave 5	DDR Port 3
Slave 6	External Bus Interface
Slave 7	Internal Peripherals

Table 4.2. Bus Matrix Slaves

4.5. Advanced Interrupt Controller (AIC)

The Advanced Interrupt Controller can handle up to 32 internal or external interrupt sources. The AIC integrates an 8-level priority controller. Interrupt sources can be programmed to be level sensitive or edge triggered. The polarity can be programmed for all external interrupt sources.

Moreover, all PIO lines can be used to generate a PIO interrupt. However, the PIO lines can only generate level change interrupts, that is, positive as well as negative edges will generate an interrupt. The PIO interrupt itself (PIO to AIC line) is usually programmed to be level-sensitive. Otherwise interrupts will be lost if multiple PIO lines source an interrupt simultaneously.

On the Stamp9G45 only GPIO interrupts are available. The list of peripheral identifiers, which are used to program the AIC can be found in Table B.1, “Peripheral Identifiers”

4.6. Battery Backup

The following parts of the AT91SAM9G45 Processor can be backed-up by a battery:

- Slow Clock Oscillator
- Real Time Timer
- Reset Controller
- Shutdown Controller
- RTC
- General Purpose Backup Registers

It is recommended to always use a backup power supply (normally a battery) in order to speed up the boot-up time and to avoid reset problems.

4.7. Reset Controller (RSTC)

The embedded microcontroller has an integrated Reset Controller which samples the backup and the core voltage. The presence of a backup voltage (VDDDBU) when the card is powered down speeds up the boot time of the microcontroller.

4.8. Serial Number

Every Stamp9G45 has a unique 48-bit hardware serial number chip which can be used by application software. The chip is a Dallas® one-wire-chip. A Linux driver is provided. Additionally it functions as the 128 Byte EEPROM.

4.9. Peripheral Input/Output Controller (PIO)

The Stamp9G45 has a maximum of 105 freely programmable digital I/O ports on its connectors. These pins are also used by other peripheral devices.

The Parallel Input/Output Controller(PIO) manages up to 32 programmable I/O ports. Each I/O port is associated with a bit number in the 32 bit register of the user interface. Each I/O port may be configured for general purpose I/O or assigned to a function of an integrated peripheral device. In doing so multiplexing with 2 integrated devices is possible. That means a pin may be used as GPIO, device A or device B. The PIO Controller also features a synchronous output providing up to 32 bits of data output in a single write operation.

The following characteristics are individually configurable for each PIO pin:

- PIO enable
- Peripheral A or B enable
- Output enable
- Output level
- Write Enable
- Level change interrupt
- Glitch filter: pulses that are lower than a half clock cycle are ignored
- Open-drain outputs
- Pull-up resistor

All configurations as well as the pin status can be read back by using the appropriate status register. Multiple pins of each PIO can also be written simultaneously by using the synchronous output register.

For interrupt handling, the PIO Controllers are considered as user peripherals. This means that the PIO Controller interrupt lines are connected among the interrupt sources 2 to 31. Refer to the PIO Controller peripheral identifier Table B.1, “Peripheral Identifiers” to identify the interrupt sources dedicated to the PIO Controllers. The PIO Controller interrupt can be generated only if the PIO Controller clock is enabled.

A number of the PIO signals might be used internally on the module. Care has to be taken when accessing the PIO registers in order not to change the settings of these internal signals, otherwise a system crash is likely to happen.

4.10. Clock Generation

4.10.1. Processor Clocks

The AT91SAM9G45 has no PLLB, but provides the 480 MHz USB Clock via a UPLL.

The embedded microcontroller generates its necessary clocks based on two crystal oscillators: One slow clock (SLCK) oscillator running at 32.768 KHz and one main clock oscillator running at 18.432 MHz. It has furthermore a internal slow clock oscillator, which can be battery backed-up.

From the main clock oscillator, the microcontroller generates two further clocks by using two PLLs. PLLA provides the processor clock (PCK) and the master clock (MCK). Typically PLLB provides the 48 MHz USB Clock and is normally used only for this purpose. The clocks of most peripherals derive from MCK. These include EBI, USART, SPI, TWI, SSC, PIT and TC.

Some peripherals like the programmable clocks and the timer counters (TC) can also run on SLCK. The real time timer (RTT) runs always on SLCK.

Hardware Description

Clock	Frequency	Source
PCK (Processor Clock)	800 MHz	PLLA
MCK (Master Clock)	133 MHz	PCK/3
USB Clock	480 MHz	UPLL
Slow Clock	32.768 KHz	Slow Clock Oscillator

Table 4.3. AT91SAM9G45 Clocks

4.10.2. Programmable Clocks

The programmable clocks can be individually programmed to derive their input from SLCK, PLLA, PLLB and Main Clock. Each PCK has a divider of 2, 4, 8, 16, 32 or 64.

The Stamp9G45 features two programmable clocks PCK0, PCK1.

4.11. Power Management Controller (PMC)

4.11.1. Function

The PMC has a Peripheral Clock register which allows enabling or disabling of the clocks of all integrated peripherals individually using their "Peripheral Identifier" (see Table B.1, "Peripheral Identifiers"). The System Clock register allows enabling or disabling of each of the following clocks individually:

- Processor Clock
- ISI Clock
- USB Host Clock (common for both channels)
- USB Device Clock
- Programmable Clocks

The PMC status register provides "Clock Ready" or, respectively, "PLL Lock" status bits for each of these clocks. An interrupt is generated when any of these bits changes from 0 to 1. The PMC provides status flags for the

- Main Oscillator
- Master Clock
- PLLA
- PLLB
- Programmable Clocks

The Main Oscillator frequency can be measured by using the PMC Main Clock Frequency register. The SLCK is used as reference for the measurement.

4.11.2. Power Management

Using power management can dramatically reduce the power consumption of an Embedded Device. Via the PMC various clocks can be disabled or their speed reduced:

- stopping the PLLs (PLLA and / or PLLB)
- stopping the clocks of the various peripherals
- reducing the clock rates of peripherals, especially by changing MCK.

The PMC supports the following power-saving features: Idle mode and power-down mode. Please note, that not every operating system supports these modes.

- **Idle Mode.** In idle mode, the processor clock will be re-enabled by any interrupt. The peripherals, however, are only able to generate an interrupt if they still have a clock, so care has to be taken as to when a peripheral can be powered down.
- **Power-down Mode.** In many cases a system waits for a user action or some other rare event. In such a case, it is possible to change MCK to SLCK. Any external event which changes the state on peripheral pins (not the USB) can then be detected by the PIO controller or the AIC.

It should also be taken into account that when a PLL is stopped it will take some time to restart it. Changing the PLL frequencies or stopping them can therefore be done only at a moderate rate. If short reaction times are required, this is not a choice.

Additionally further measurements can reduce power consumption considerably:

- switching off the TFT supply voltage
- putting peripheral chips like Ethernet controller and / or PHY or serial driver devices in power down mode
- putting the SDRAM into self-refresh mode

4.12. Real-time Timer (RTT)

The Real-time Timer is a 32-bit counter combined with a 16-bit prescaler running at Slow Clock (SLCK = 32768 Hz). As the RTT keeps running if only the backup supply voltage is available, it is used as a Real-time clock.

The RTT can generate an interrupt every time the prescaler rolls over. Usually the RTT is configured to generate an interrupt every second, so the prescaler will be programmed with the value 7FFFh.

The RTT can also generate an alarm if a preprogrammed 32-bit value is reached by the counter.

4.13. Timer Counter (TC)

The Stamp9G45 features two blocks of timer counters with three counters each. Due to multiplexing four timer counters may be used with external signals.

The TC consists of three independent 16-bit Timer/Counter units. They may be cascaded to form a 32-bit or 48-bit timer/counter. The timers can run on the internal clock sources MCK/2, MCK/8, MCK/32, MCK/128, SLCK or the output of another timer channel. External clocks may be used as well as the counters can generate signals on timer events. They also can be used to generate PWM signals.

4.14. Periodic Interval Timer (PIT)

The PIT consists of a 20-bit counter running on MCK / 16. This counter can be preloaded with any value between 1 and 2^{20} . The counter increments until the preloaded value is reached. At this stage it rolls over and generates an interrupt. An additional 12-bit counter counts the interrupts of the 20 bit counter.

The PIT is intended for use as the operating system's scheduler interrupt.

4.15. Watchdog Timer

The watchdog timer is a 12-bit timer running at 256 Hz (Slow Clock / 128). The maximum watchdog timeout period is therefore equal to 16 seconds. If enabled, the watchdog timer asserts a hardware reset at the end of the timeout period. The application program must always reset the watchdog timer before the timeout is reached. If an application program has crashed for some reason, the watchdog timer will reset the system, thereby reproducing a well defined state once again.

The Watchdog Mode Register can be written only once. After a processor reset, the watchdog is already activated and running with the maximum timeout period. Once the watchdog has been reconfigured or deactivated by writing to the Watchdog Mode Register, only a processor reset can change its mode once again.

4.16. Real-time Clock (RTC)

The Real-time clock combines a complete time-of-day clock with alarm, a two-hundred-year Gregorian calendar and a programmable periodic interrupt. The time and calendar values are coded in BCD format.

4.17. True Random Number Generator (TRNG)

The True Random Generator (TRNG) passes the American NIST Special Publication 800-22 and the Diehard Random Tests Suites. It provides a 32-bit value every 84 clock cycles.

4.18. Peripheral DMA Controller (PDC)

The Peripheral DMA Controller (PDC) transfers data between on-chip serial peripherals and the on- and/or off-chip memories. The PDC contains unidirectional and bidirectional channels. The full-duplex peripherals feature unidirectional channels used in pairs (transmit only or receive only). The half-duplex peripherals feature one bidirectional channel. Typically full-duplex peripherals are USARTs, SPI or SSC. The MCI is a half duplex device.

The user interface of each PDC channel is integrated into the user interface of the peripheral it serves. The user interface of unidirectional channels (receive only or transmit only), contains two 32-bit memory pointers and two 16-bit counters, one set (pointer, counter) for current transfer and one set (pointer, counter) for next transfer. The bidirectional channel user interface contains four 32-bit memory pointers and four 16-bit counters. Each set (pointer, counter) is used by current transmit, next transmit, current receive and next receive.

Using the PDC removes processor overhead by reducing its intervention during the transfer. This significantly reduces the number of clock cycles required for a data transfer, which improves microcontroller performance. To launch a transfer, the peripheral triggers its associated PDC channels by using transmit and receive signals. When the programmed data is transferred, an end of transfer interrupt is generated by the peripheral itself. There are four kinds of interrupts generated by the PDC:

- End of Receive Buffer
- End of Transmit Buffer
- Receive Buffer Full
- Transmit Buffer Empty

The "End of Receive Buffer" / "End of Transmit Buffer" interrupts signify that the DMA counter has reached zero. The DMA pointer and counter register will be reloaded from the reload registers ("DMA new pointer register" and "DMA new counter register") provided that the "DMA new counter register" has a non-zero value. Otherwise a "Receive Buffer Full" or, respectively, a "Transmit Buffer Empty" interrupt is generated, and the DMA transfer terminates. Both reload registers are set to zero automatically after having been copied to the DMA pointer and counter registers.

4.19. Debug Unit (DBGU)

The Debug Unit is a simple UART which provides only RX/TX lines. It is used as a simple serial console for Firmware and Operating Systems.

4.20. JTAG Unit

The JTAG unit can be used for hardware diagnostics, hardware initialization, flash memory programming, and debug purposes. The JTAG unit supports two different modes, namely the "ICE Mode", and the "Boundary Scan" mode. It is normally jumpered for "ICE Mode".

JTAG interface devices are available for the unit. However, the use of them is not within the scope of this document.

4.21. Two-wire Interface (TWI)

The TWI is also known under the expression "I²C-Bus", which is a trademark of Philips and may therefore not be used by other manufacturers. However, interoperability is guaranteed. The TWI supports both master or slave mode.

The TWI uses only two lines, namely serial data (SDA) and serial clock (SCL). According to the standard, the TWI clock rate is limited to 400 kHz in fast mode and 100 kHz in normal mode, but configurable baud rate generator permits the output data rate to be adapted to a wide range of core clock frequencies.

Caveat: The TWI hardware unit has been known to be error prone in various microcontrollers, which has lead operating systems like Linux to use a bit-banging driver on the same pins instead.

4.22. Multimedia Card Interface (MCI)

The Stamp9G45 features a onboard Micro-SD-Card slot, which is connected to the MCI-B interface of the microcontroller. The MCI-A interface is provided for external additional use.

The MultiMedia Card Interface (MCI) supports the MultiMedia Card (MMC) Specification V3.11, the SDIO Specification V1.1 and the SD Memory Card Specification V1.0.

The MCI includes a command register, response registers, data registers, timeout counters and error detection logic that automatically handle the transmission of commands and, when required, the reception of the associated responses and data with a limited processor overhead. The MCI supports stream, block and multi-block data read and write, and is compatible with the Peripheral DMA Controller (PDC) channels, minimizing processor intervention for large buffer transfers.

The MCI operates at a rate of up to Master Clock divided by 2 and supports the interfacing of 2 slot(s). Each slot may be used to interface with a MultiMediaCard bus (up to 30 Cards) or with a SD Memory Card. Only one slot can be selected at a time (slots are multiplexed). A bit field in the SD Card Register performs this selection.

The SD Memory Card communication is based on a 9-pin interface (clock, command, four data and three power lines) and the MultiMedia Card on a 7-pin interface (clock, command, one data, three power lines and one reserved for future use). The SD Memory Card interface also supports MultiMedia Card operations. The main differences between SD and MultiMedia Cards are the initialization process and the bus topology.

4.23. USB Host Port (UHP)

In the current revision of the AT91SAM9G45 USB High speed is not working. It will work in the processor's next revision, which is expected in august 2011. The Stamp9G45 integrates two USB host ports supporting speeds up to 480 MBit/s. USB Host Port A is connected directly to the transceiver, USB Host Port B is multiplexed with the USB device port. Only one of them can be used at a time.

The controller is fully compliant with the Enhanced HCI(EHCI) specification. It supports both High-speed 480 Mbps and Full-speed 12 Mbps devices.

The USB Host Port (UHP) interfaces the USB with the host application. It handles Open HCI protocol (Open Host Controller Interface) as well as USB v2.0 Full-speed and Low-speed protocols.

The USB Host Port integrates a root hub and transceivers on downstream ports. It provides several high-speed half-duplex serial communication ports. Up to 127 USB devices (printer, camera, mouse, keyboard, disk, etc.) and an USB hub can be connected to the USB host in the USB "tiered star" topology.

4.24. USB Device Port (UDP)

In the current revision of the AT91SAM9G45 USB High speed is not working. It will work in the processor's next revision, which is expected in august 2011. The Stamp9G45 integrates one USB device port supporting speeds up to 480 MBit/s. It is multiplexed with the USB Host Port B. Only one of them can be used at a time.

The controller is fully compliant with the Enhanced HCI(EHCI) specification. It supports both High-speed 480 Mbps and Full-speed 12 Mbps devices.

The USB Device Port (UDP) is compliant with the Universal Serial Bus (USB) V2.0 full-speed device specification. The USB device port enables the product to act as a device to other host controllers.

The USB device port can also be implemented to power on the board. One I/O line may be used by the application to check that VBUS is still available from the host. Self-powered devices may use this entry to be notified that the host has been powered off. In this case, the pullup on DP must be disabled in order to prevent feeding current to the host. The application should disconnect the transceiver, then remove the pullup.

4.25. Ethernet MAC (EMAC)

The EMAC module implements a 10/100 MBit/s Ethernet MAC compatible with the IEEE 802.3 standard using an address checker, statistics and control registers, receive and transmit blocks, and a DMA interface.

The address checker recognizes four specific 48-bit addresses and contains a 64-bit hash register for matching multicast and unicast addresses. It can recognize the broadcast address of all ones, copy all frames, and act on an external address match signal.

An individual 48-bit MAC address (ETHERNET hardware address) is allocated to each product. This number is stored in flash memory. It is recommended not to change the MAC address in order to comply with IEEE Ethernet standards.

To completely implement ethernet an additional physical layer interface is needed (PHY). A sample implementation is found on the Starterkit Board.

4.26. Universal Synchronous Asynchronous Receiver and Transmitter (USART)

The Stamp9G45 has up to four independent USARTs, not including the debug unit.

The Universal Synchronous Asynchronous Receiver Transceiver (USART) provides one full duplex universal synchronous asynchronous serial link. Data frame format is widely

programmable (data length, parity, number of stop bits) to support a maximum of standards. The receiver implements parity error, framing error and overrun error detection. The receiver time-out enables handling variable-length frames and the transmitter timeguard facilitates communications with slow remote devices. Multidrop communications are also supported through address bit handling in reception and transmission.

The USART supports the connection to the Peripheral DMA Controller, which enables data transfers to the transmitter and from the receiver. The PDC provides chained buffer management without any intervention of the processor.

Six different modes are implemented within the USARTs:

- Normal (standard RS232 mode)
- RS485
- Hardware Handshaking
- ISO7816 Protocol: T=0 or T=1
- IrDA

RS485. In RS485 operating mode the RTS pin is automatically driven high during transmit operations. If RTS is connected to the "enable" line of the RS485 driver, the driver will thus be enabled only during transmit operations.

Hardware Handshaking. The hardware handshaking feature enables an out-of-band flow control by automatic management of the pins RTS and CTS. The receive DMA channel must be active for this mode. The RTS signal is driven high if the receiver is disabled or if the DMA indicates a buffer full condition. As the RTS signal is connected to the CTS line of the connected device, its transmitter is thus prevented from sending any more characters.

ISO7816. The USARTs have an ISO7816-compatible mode which permits interfacing with smart cards and Security Access Modules (SAM). Both T=0 and T=1 protocols of the ISO7816 specification are supported.

IrDA. The USART features an infrared (IrDA) mode supplying half-duplex point-to-point wireless communication. It includes the modulator and demodulator which allows a glueless connection to the infrared transceivers. The modulator and demodulator are compliant with the IrDA specification version 1.1 and support data transfer speeds ranging from 2.4 kb/s to 115.2 kb/s.

Signals of the Serial Interfaces. All UARTs/USARTs have one receiver and one transmitter data line (full duplex). Not all USARTs are implemented with full modem control lines. Furthermore the available lines depend largely on the used multiplexing. Most modem control lines can be implemented with standard digital ports.

Hardware Interrupts. There are several interrupt sources for each USART:

- Receive: RX Ready, (DMA) Buffer Full, End of Receive Buffer

- Transmit: TX Ready, (DMA) Buffer Empty, End of Transmit Buffer, Shift Register Empty
- Errors: overrun, parity, framing, and timeout errors
- Handshake: the status of CTS has changed
- Break: the receiver has detected a break condition on RXD
- NACK: non acknowledge (ISO7816 mode only)
- Iteration: the maximum number of repetitions has been reached (ISO7816 mode only)

Please refer to the chapter about the DMA unit (PDC) for a description of the "Buffer Full" and "End of Receive / Transmit Buffer" events.

4.27. Synchronous Peripheral Interface (SPI)

The Stamp9G45 features two SPI ports, with four respectively one chipselect available.

The Serial Peripheral Interface (SPI) circuit is a synchronous serial data link that provides communication with external devices in Master or Slave Mode. It also enables communication between processors if an external processor is connected to the system.

The Serial Peripheral Interface is essentially a shift register that serially transmits data bits to other SPIs. During a data transfer, one SPI system acts as the "master" which controls the data flow, while the other devices act as "slaves" which have data shifted into and out by the master.

A slave device is selected when the master asserts its NSS signal. If multiple slave devices exist, the master generates a separate slave select signal for each slave (NPCS). The SPI system consists of two data lines and two control lines:

- Master Out Slave In (MOSI): This data line supplies the output data from the master shifted into the input(s) of the slave(s).
- Master In Slave Out (MISO): This data line supplies the output data from a slave to the input of the master. There may be no more than one slave transmitting data during any particular transfer.
- Serial Clock (SPCK): This control line is driven by the master and regulates the flow of the data bits. The master may transmit data at a variety of baud rates; the SPCK line cycles once for each bit that is transmitted. The SPI baudrate is Master Clock (MCK) divided by a value between 1 and 255
- Slave Select (NSS): This control line allows slaves to be turned on and off by hardware.

Each SPI Controller has a dedicated receive and transmit DMA channel.

4.28. Synchronous Serial Controller (SSC)

The Stamp9G45 has one SSC interface available, depending on the multiplexing of the pins.

The SSC supports many serial synchronous communication protocols generally used in audio and telecom applications such as I2S, Short Frame Sync, Long Frame Sync, etc.

The SSC has separated receive and transmit channels. Each channel has a data, a clock and a frame synchronization signal (RD, RK, RF, resp. TD, TK, TF). Both a receive and a transmit DMA channel are assigned to each SSC.

4.29. AC97 Controller (AC97C)

AC97 Component Specification 2.2 compliant AC97 digital controller. It supports mono or stereo up to 20 bit sample length and features a sampling rate up to 48 KHz.

Pin	Description	Type
AC97CK	12.288-MHz bit-rate clock	Input
AC97RX	SDATA_IN	Input
AC97FS	48-KHz frame indicator	Output
AC97TX	SDATA_OUT	Output

Table 4.4. AC97 I/O Lines

4.30. Image Sensor Interface (ISI)

The Image Sensor Interface (ISI) supports direct connection to the ITU-R BT. 601/656 8-bit mode compliant sensors and up to 12-bit grayscale sensors. It receives the image data stream from the image sensor on the 12-bit data bus. This module receives up to 12 bits for data, the horizontal and vertical synchronizations and the pixel clock. The reduced pin count alternative for synchronization is supported for sensors that embed SAV (start of active video) and EAV (end of active video) delimiters in the data stream.

The Image Sensor Interface interrupt line is generally connected to the Advanced Interrupt Controller and can trigger an interrupt at the beginning of each frame and at the end of a DMA frame transfer. If the SAV/EAV synchronization is used, an interrupt can be triggered on each delimiter event.

For 8-bit color sensors, the data stream received can be in several possible formats: YCbCr 4:2:2, RGB 8:8:8, RGB 5:6:5 and may be processed before the storage in memory. The data stream may be sent on both preview path and codec path if the bit CODEC_ON in the ISI_CR1 is one. To optimize the bandwidth, the codec path should be enabled only when a capture is required.

In grayscale mode, the input data stream is stored in memory without any processing. The 12-bit data, which represent the grayscale level for the pixel, is stored in memory one or two pixels per word, depending on the GS_MODE bit in the ISI_CR2 register. The codec datapath is not available when grayscale image is selected.

4.31. LCD controller

The LCD controller supports single and double scan monochrome and color passive STN LCD modules and single scan active TFT LCD modules with a resolution of up to 2048x2048 with a color depth of up to 24 bits per pixel.

The LCD controller relies on a relatively simple frame buffer concept, which means that all graphics and character functions have to be implemented in software: character sets and graphic primitives are not integrated in the controller.

4.31.1. LCDC Initialisation and LCD Power Sequencing

LCD cells (pixels) should not be subjected to DC power for prolonged periods of time, as chemical decomposition might take place. The LCD controller therefore provides for a strict AC control of the LCD pixels. To do so, the LCD controller has to be initialized appropriately. Switching on the LCD supply voltage therefore has to take place after the LCDC initialization or shortly before.

Accordingly, the LCDC should not be powered down without deactivating the LCD supply voltage. The same is true if the LCDC is stopped indirectly by stopping the respective clock source, namely the PLLA.

The LCD backlight supply is not involved in these considerations. It may be switched on or off at any time independently of the state of the LCDC.

4.31.2. LCDC Frame Buffer

The LCDC Frame Buffer typically resides in the external RAM.

The LCDC video memory is organized as a frame buffer in a straight forward way. It supports color depths of 1, 2, 4, 8, 16, or 24 bit per pixel. The video data is stored in a packed form with no unused bits in the video memory.

The color resolutions of 1, 2, 4, and 8 bpp (bits per pixel) use a palette table which is made up of 16-bit entries. The value of each pixel in the frame buffer serves as an index into the palette table. The value of the respective palette table entry is output to the display by the LCDC, see Table 4.5, “LCDC palette entry”.

Bit[14..10]	Bit[9..5]	Bit[4..0]
Blue[7..3]	Green[7..3]	Red[7..3]

Table 4.5. LCDC palette entry

The bits 2..0 of each color channel are not used in the palettized configuration — they are set to 0.

The same scheme as above is used in the 16-bit color resolution configuration, although in this case the frame buffer entry is output directly to the display instead of indexing a palette table.

In the 24-bit color resolution configuration, each frame buffer entry consists of one byte for each color, see Table 4.6, “LCDC 24 bit memory organization”.

Bit[23..16]	Bit[15..8]	Bit[7..0]
Blue[7..0]	Green[7..0]	Red[7..0]

Table 4.6. LCDC 24 bit memory organization

If the LCD Module has lower color resolution (fewer bits per color component), only the most significant bits of each component are used.

The Linux frame buffer driver offers a function which returns the information about the frame buffer structure including the assignment of each frame buffer bit to a color channel bit. It is recommended that graphics software uses this function in order to achieve a correct color representation.

4.32. Touch Screen ADC Controller (TSADCC)

The Stamp9G45 has additional to touch panel support three ADC channels available.

The Touch Screen ADC Controller is a 10-bit Analog-to-Digital Converter supporting resistive touch screen panels. It can be used as Touch Screen Controller, ADC or both supporting eight lines maximum. It integrates a 8-to-1 analog multiplexer for analog to digital conversions of up to 8 analog lines, four power switches that measure both axis positions and a pen-interrupt and pen-loss switch.

The conversions extend from 0V to TSADVREF, an external voltage reference for better accuracy. It supports 8-bit or 10-bit resolution mode. Every channel can be enabled and disabled separately. It supports 10-bit 384 Ksamples/sec.

5. Design Considerations

5.1. Ethernet Controller (EMAC)

The emac needs an additional PHY design. The emac supports both, MII and RMI interface.

Please take care of the specific layout requirements of the Ethernet port when designing a base board. The two signals of the transmitter pair (ETX+ and ETX-) should be routed in parallel (constant distance, e.g. 0.5mm) with no vias on their way to the RJ45-jack. The same is true for the receiver pair (ERX+ and ERX-). No other signals should be crossing or get next to these lines. If a ground plane is used on the base board, it should be omitted in the vicinity of the Ethernet signals.

A 1nF / 2kV capacitor should be connected between board ground and chassis ground (which is usually connected to the shield of the RJ45-jack).

5.2. USB

On Stamp9G45 the Pull-down resistors on the lines are already implemented on Stamp9G45 itself.

5.2.1. USB Host Controller (UHP)

External Parts. A few external parts are required for the proper operation of the UHP:

- Pull-down resistors on each line of approximately 15 k Ω . These should be installed even if the UDP is not to be used at all in order to keep the signals from floating.
- Series resistors of 27 Ω (5%) on each line.
- Small capacitors (e.g. 15pF) to ground on each line (optional).
- ESD protection devices are recommended for applications which are subject to external contact. The restrictions with regard to capacitive loading have to be applied when selecting a protection device.
- A circuit to generate the 5V VBUS supply voltage.

V_{BUS} considerations for USB Host. A USB host port has to provide a supply voltage V_{BUS} of 5V +- 5% which has to be able to source a maximum of 500mA, or 100mA in case of battery operation. Please refer to the appropriate rules in the USB specification. A low ESR capacitor of at least 120 μ F has to be provided on V_{BUS} in order to avoid excessive voltage drops during current spikes.

V_{BUS} has to have an over-current protection. The over-current drawn temporarily on V_{BUS} must not exceed 5A. Polymeric PTCs or solid state switches are recommended by the specification. Suitable PPTCs are "MultiFuse" (Bourns), "PolyFuse" (Wickmann/Littelfuse), "PolySwitch" (Raychem/Tyco).

It is required that the over-current condition can be detected by software, so that V_{BUS} can be switched off or be reduced in power in such a case.

Layout considerations. The external resistors should be placed in the vicinity of the module's connector. The two traces of any of the differential pairs (USB-Host A+ and USB-Host A- , as well as USB-Host B+ and USB-Host B-) should not encircle large areas on the base board, in order to reduce signal distortion and noise. They are preferably routed closely in parallel to the USB connector.

5.2.2. USB Device Controller (UDP)

External Parts. A few external parts are required for the proper operation of the UDP:

- Pull-down resistors on each line of approximately 330 k Ω . These should be installed even if the UDP is not to be used at all in order to keep the signals from floating.
- Series resistors of 27 Ω (5%) on each line.
- A voltage divider on the 5V USB supply voltage V_{BUS} converting this voltage to 3.3V (1.8V), e.g. 27 k Ω / 47 k Ω , for the V_{BUS} monitoring input (USB_CNX).
- ESD protection devices are recommended for applications which are subject to external contact. The restrictions with regard to capacitive loading have to be applied when selecting a protection device.

The USB specification (revision 2.0 of April 27, 2000, chapter 7.1.1.1) demands a cable with a characteristic impedance of 90 $\Omega \pm 15\%$. Matching this impedance would require a driver output impedance of 45 Ω on both the negative and the positive driver. Nevertheless, the controller demands external resistors of 27 Ω . These would correspond to intrinsic resistances of the drivers of 18 Ω .

The USB specification demands a switchable pull-up resistor of 1.5 k Ω on USB-Device+ which identifies the UDP as a full speed device to the attached host controller. On this module, this resistor is integrated on the chip. It can be switched on or off using the "USB Pad Pull-up Control Register", which is part of the "Bus Matrix User Interface" (not the "USB Device Port User Interface", as one might expect). This pull-up resistor is required to be switchable in order not to source current to an attached but powered down host. This would otherwise constitute an irregular condition on the host. The software has to take care of this fact.

The capacitors are intended to improve the signal quality (edge rate control) in dependency of the specific design. They are not mandatory. The total capacitance to ground of each USB pin, the PCB trace to the series resistor, and the capacitor must not exceed 75pF.

Operation with V_{BUS} as a Supply. Special care has to be taken if the module is powered by the V_{BUS} supply. Please refer to the appropriate rules in the USB specification with regard to inrush current limiting and power switching. As the module draws more than 100mA in normal mode, it is a "high-power" device according to the specification (<100mA = "low-power", 100..500mA = "high-power"). It therefore requires staged

switching which means that at power-up it should draw not more than 100mA on VBUS. The capacitive load of a USB device on VBUS should be not higher than 10 μ F.

Layout considerations. The external resistors should be placed in the vicinity of the module's connector. The traces of the differential pair (USB-Device+ and USB-Device-) should not encircle large areas on the base board, in order to reduce signal distortion and noise. They are preferably routed closely in parallel to the USB connector.

5.3. Memory Bus

On the Stamp9G45 the memory bus is driven with 1.8V. This affects the voltages of PIOC-controller pins, they are 1.8V as well. Not affected are the ADC-Channels, which have their own ADV_{REF} . The V_{MEM} pins on the module are pin one and two of the bus interface. If pins of PIOC or the memory bus are in use on the customer's design it is highly recommended to implement buffers on both memory bus and PIOC pins.

The memory bus is used inside of the module. It can be either 1.8V or 3.3V. The V_{mem} pin of the module is powered by the module itself. Do not power this pin externally to maintain inter-product dependencies. A difference between V_{mem} and VCC may also affect the behaviour of one PIO-controller of the respective module.

To connect 3.3V chips to the memory bus or to maintain compatibility between various products it is recommended to implement buffer chips on the memory bus externally, like shown in Figure 5.1, "Buffered Memory Bus (PIOC) 1.8V - 3.3V"

To connect 5V chips the same schematics can be used with suitable buffer chips.

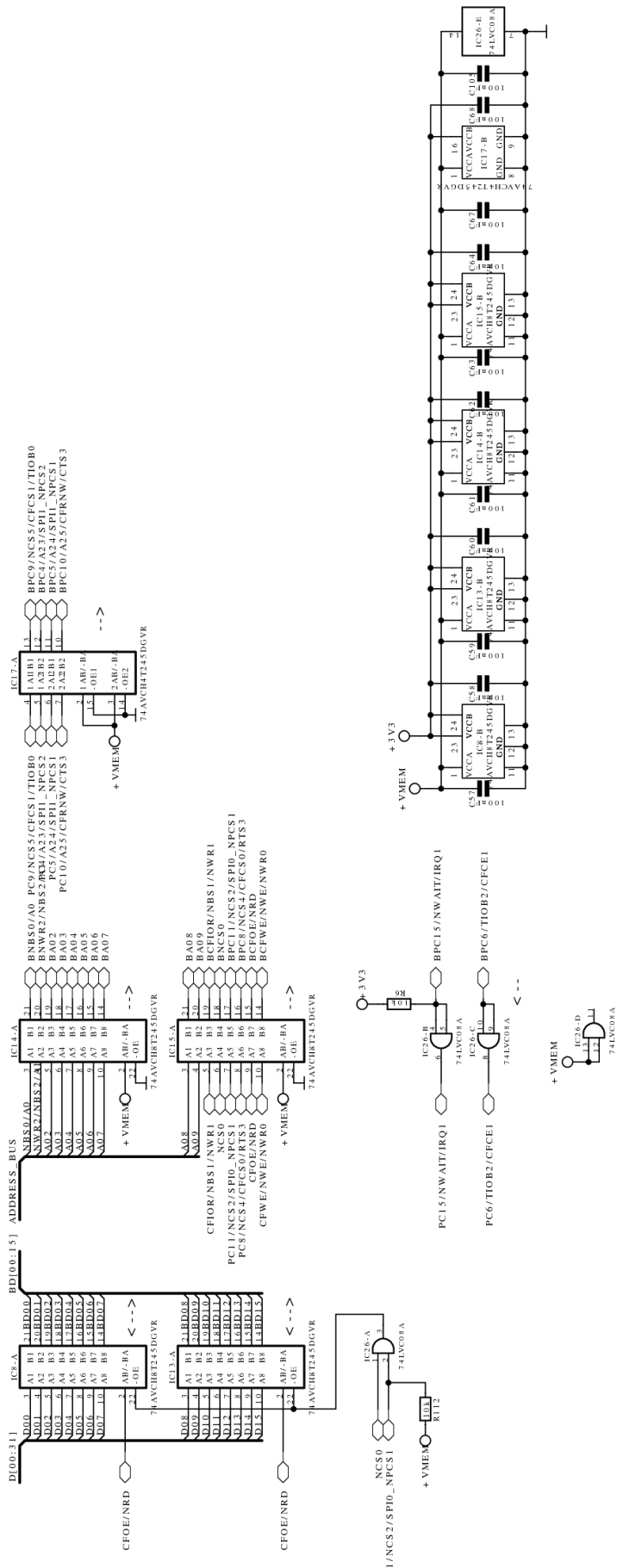


Figure 5.1. Buffered Memory Bus (PIOC) 1.8V - 3.3V

Appendix A. Peripheral Color Codes

This table matches the color used to identify various peripherals in tables.

Power Supply/Ground
USART
Debug UART
TWI (I ² C-Bus)
SD-Card/MMC
SPI
USB Host
USB Device
Reserved
Synchronous Serial Controller (SSC)
JTAG
Control
Ethernet
Genral Purpose I/O Port
Programmable Clock Output
Analog-to-digital Converter
Timer Counter
Image Sensor Interface
LCD/TFT Controller Interface
Embedded Trace Macrocell
Static Memory Controller
Compact Flash Interface
Pulse Width Modulator
Touch Controller
AC97 Sound Interface

Appendix B. Peripheral Identifiers

ID	Mnemonic	Peripheral Name	External Interrupt
0	AIC	Advanced Interrupt Controller	FIQ
1	SYSC	System Controller Interrupt	
2	PIOA	Parallel I/O Controller A	
3	PIOB	Parallel I/O Controller B	
4	PIOC	Parallel I/O Controller C	
5	PIOD/PIOE	Parallel I/O Controller D/E	
6	TRNG	True Random Number Generator	
7	US0	USART 0	
8	US1	USART 1	
9	US2	USART 2	
10	US3	USART 3	
11	MCI0	High Speed Multi Media Card Interface 0	
12	TWI0	Two-Wire Interface 0	
13	TWI1	Two-Wire Interface 1	
14	SPI0	Serial Peripheral Interface	
15	SPI1	Serial Peripheral Interface	
16	SSC0	Synchronous Serial Controller 0	
17	SSC1	Synchronous Serial Controller 1	
18	TC0..TC5	Timer Counter 0,1,2,3,4,5	
19	PWM	Pulse Width Modulation Controller	
20	TSADCC	Touch Screen ADC Controller	
21	DMA	DMA Controller	
22	UHPHS	USB Host High Speed	
23	LCDC	LCD Controller	
24	AC97C	AC97 Controller	
25	EMAC	Ethernet MAC	
26	ISI	Image Sensor Interface	
27	UDPHS	USB Device High Speed	
28	Reserved		
29	MCI1	High Speed Multi Media Card Interface 1	
30	Reserved		
31	AIC	Advanced Interrupt Controller	IRQ

Table B.1. Peripheral Identifiers

Appendix C. Address Map (Physical Address Space)

After the execution of the remap command the 4 GB physical address space is separated as shown in the following table. Accessing these addresses directly is only possible if the MMU (memory management unit) is deactivated. As soon as the MMU is activated the visible address space is changed completely. If absolute memory addresses should be accessed within an application, the corresponding address space has first to be mapped to the virtual address space using `mmap` or `ioremap` under Linux.

Address (Hex)	Mnemonic	Function
00 0000	Boot Memory	NCS0 or internal ROM or internal SRAM (depending on BMS and REMAP)
10 0000	ITCM	Internal SRAM A 32 kByte, mapped for instructions
20 0000	DTCM	Internal SRAM B 32 kByte, mapped for data
30 0000	SRAM	Internal SRAM 64 kByte
40 0000	ROM	Internal ROM
50 0000	LCDC	LCD/TFT controller
60 0000	UDPHS	USB Device Port (DMA)
70 0000	USB OHCI	USB OHCI controller
80 0000	USB EHCI	USB EHCI controller
1000 0000	EBI_1 NCS0	Chip Select 0
2000 0000	EBI_1 NCS1	Chip Select 1: DDRAM
3000 0000	EBI_1 NCS2	Chip Select 2
4000 0000	EBI_1 NCS3	Chip Select 3: NAND
5000 0000	EBI_1 NCS4	Chip Select 4: CF_1
6000 0000	EBI_1 NCS5	Chip Select 5: CF_2
7000 0000	EBI_0 NCS0	Chip Select 0: DDRAM
FFF7 8000	UDPHS	USB Device Port
FFF7 C000	TC0, TC1, TC2	3 Timer Counter, 16-Bit
FFF8 0000	MCI_0	Multimedia Card / SD-Card Interface #0
FFF8 4000	TWI_0	Two Wire Interface #0(I ² C)
FFF8 8000	TWI_1	Two Wire Interface #1(I ² C)
FFF8 C000	USART0	Synchronous or Asynchronous Serial Port #0
FFF9 0000	USART1	Synchronous or Asynchronous Serial Port #1
FFF9 4000	USART2	Synchronous or Asynchronous Serial Port #2
FFF9 8000	USART3	Synchronous or Asynchronous Serial Port #3
FFF9 C000	SSC0	Serial Synchronous Controller #0(I ² S)
FFFA 0000	SSC1	Serial Synchronous Controller #1(I ² S)
FFFA 4000	SPI0	Serial Peripheral Interface #0
FFFA 8000	SPI1	Serial Peripheral Interface #1
FFFA C000	AC97	AC97 Interface
FFFB 0000	TSADC	Touch Controller Interface

Address Map (Physical Address Space)

Address (Hex)	Mnemonic	Function
FFFB 4000	ISI	Image Sensor Interface
FFFB 8000	PWM	Pulse Width Modulator
FFFB C000	EMAC	Ethernet Controller
FFFC C000	TRNG	True Random Number Generator
FFFD 0000	MCI_1	Multimedia Card / SD-Card Interface #1
FFFD 4000	TC3, TC4, TC5	3 Timer Counter, 16-Bit
FFFF E200	ECC	Error Correction Controller
FFFF E400	DDRS DR	DDRAM Controller #1
FFFF E600	DDRS DR	DDRAM Controller #0
FFFF E800	SMC	Static Memory Controller
FFFF EA00	MATRIX	Bus Matrix User Interface
FFFF EC00	DMAC	Bus Matrix User Interface
FFFF EE00	DBGU	Debug Unit, including UART
FFFF F000	AIC	Advanced Interrupt Controller
FFFF F200	PIOA	32 Bit Parallel I/O Controller A
FFFF F400	PIOB	32 Bit Parallel I/O Controller B
FFFF F600	PIOC	32 Bit Parallel I/O Controller C
FFFF F800	PIOD	32 Bit Parallel I/O Controller D
FFFF FA00	PIOE	32 Bit Parallel I/O Controller E
FFFF FC00	PMC	Power Management Controller
FFFF FD00	RSTC	Reset Controller, Battery Powered
FFFF FD10	SHDWC	Shutdown Controller, Battery Powered
FFFF FD20	RTT	Real-time Timer 32 Bit, Battery Powered
FFFF FD30	PIT	Periodic Interval Timer 32 Bit
FFFF FD40	WDT	Watchdog Timer
FFFF FD50	SCKCR	Serial Clock Register
FFFF FD60	GPBR	4 General Purpose Backup Registers

Table C.1. Physical Address Space

Appendix D. Stamp9G45 Pin Assignment

Pin	GPIO	Periph. A	Periph. B	Add. Function	Add. Function	Periph. B	Periph. A	GPIO	Pin
1	VMEM				VMEM				2
3	EBI1 A0/NBS0				EBI1 A1/EBI1_NBS2/NWR2				4
5	EBI1 A2				EBI1 A3				6
7	EBI1 A4				EBI1 A5				8
9	EBI1 A6				EBI1 A7				10
11	EBI1 A8				EBI1 A9				12
13	EBI1 A10				EBI1 A11				14
15	EBI1 A12				EBI1 A13				16
17	EBI1 A14				EBI1 A15				18
19	EBI1 A16/BA0				EBI1 A17/BA1				20
21	EBI1 A18						EBI1 A19	PC2	22
23	PC3	EBI1 A20					EBI1 A21/ NANDCLE	PC4	24
25	PC5	EBI1 A22/ NANDALE					EBI1 A23	PC6	26
27	PC7	EBI1 A24					EBI1 A25/ CFRNW	PC12	28
29	GND				GND				30
31	PD8	AC97FS	TIOB5			TCLK5	AC97CK	PD9	32
33	PD7	AC97TX	TIOA5				AC97RX	PD6	34
35	EBI1 NCS0				EBI1 NCS1/SDCS				36
37	PC13	EBI1 NCS2					EBI1 NCS3/ NANDCS	PC14	38
39	PC10	EBI1 NCS4/ CFCS0	TCLK2			CTS2	EBI1 NCS5/ CFCS1	PC11	40
41	PC0	DQM2					DQM3	PC1	42
43	EBI1 NRD/CFOE				EBI1 NWR0/NWE/CFWE				44
45	EBI1 NWR1/NBS1/CFIOR				EBI1 NWR3/NBS3/CFIOW				46
47	PC8	CFCE1				RTS2	CFCE2	PC9	48
49	GND				GND				50
51	VCC				VCC				52
53	EBI1 D0				EBI1 D1				54
55	EBI1 D2				EBI1 D3				56
57	EBI1 D4				EBI1 D5				58
59	EBI1 D6				EBI1 D7				60
61	EBI1 D8				EBI1 D9				62
63	EBI1 D10				EBI1 D11				64

Pin	GPIO	Periph. A	Periph. B	Add. Function	Add. Function	Periph. B	Periph. A	GPIO	Pin
65	EBI1 D12				EBI1 D13				66
67	EBI1 D14				EBI1 D15				68
69	GND				GND				70
71	PB20	ISI_D0					ISI_D1	PB21	72
73	PB22	ISI_D2					ISI_D3	PB23	74
75	PB24	ISI_D4					ISI_D5	PB25	76
77	PB26	ISI_D6					ISI_D7	PB27	78
79	PB28	ISI_PCK					ISI HSYNC	PB30	80
81	PB29	ISI VSYNC				PCK1	ISI MCK	PB31	82
83	PB8	TXD3	ISI_D8			ISI_D9	RXD3	PB9	84
85	PB10	TWD1	ISI_D10			ISI_D11	TWCK1	PB11	86
87	SHDN				WKUP				88
89	NRST				VBATT				90
91	RTCK						NWAIT	PC15	92
93	NTRST				JTAGSEL				94
95	TDI				TMS				96
97	TDO				TCK				98
99	GND				GND				100

Table D.1. Pin Assignment IO Interface

Pin	GPIO	Periph. A	Periph. B	Add. Function	Add. Function	Periph. B	Periph. A	GPIO	Pin
1	VCC				VCC				2
3	PA17	ETXCK					ERXDV	PA15	4
5	PD22	TIOA2		TSAD2	TSAD3		TCLK0	PD23	6
7	BMS				TSADVREF				8
9	PE3	LCD VSYNC					LCD HSYNC	PE4	10
11	PE5	LCD DOTCK					LCDDEN	PE6	12
13	PE2	LCDDC				LCDD2	LCDD0	PE7	14
15	PE8	LCDD1	LCDD3			LCDD4	LCDD2	PE9	16
17	PE10	LCDD3	LCDD5			LCDD6	LCDD4	PE11	18
19	PE12	LCDD5	LCDD7			LCDD10	LCDD6	PE13	20
21	PE14	LCDD7	LCDD11			LCDD12	LCDD8	PE15	22
23	PE16	LCDD9	LCDD13			LCDD14	LCDD10	PE17	24
25	GND				GND				26
27	PE18	LCDD11	LCDD15			LCDD18	LCDD12	PE19	28
29	PE20	LCDD13	LCDD19			LCDD20	LCDD14	PE21	30
31	PE22	LCDD15	LCDD21			LCDD22	LCDD16	PE23	32
33	PE24	LCDD17	LCDD23				LCDD18	PE25	34

Stamp9G45 Pin Assignment

Pin	GPIO	Periph. A	Periph. B	Add. Function	Add. Function	Periph. B	Periph. A	GPIO	Pin
35	PE26	LCDD19					LCDD20	PE27	36
37	PE28	LCDD21					LCDD22	PE29	38
39	PE30	LCDD23					SPI0 SPCK	PB2	40
41	PB0	SPI0 MISO					SPI0 MOSI	PB1	42
43	PB3	SPI0 NPCS0			GPAD4	PWM0	SPI0 NPCS1	PD24	44
45	PD25	SPI0 NPCS2	PWM1	GPAD5		SPI1 NPCS1	TSADTRG	PD28	46
47	PA20	TWD0					TWCK0	PA21	48
49	GND				GND				50
51	VCC				VCC				52
53	PB19	TXD0	SPI0 NPCS2			SPI0 NPCS1	RXD0	PB18	54
55	PB17	SPI1 NPCS0	RTS0			CTS0	SPI1 MOSI	PB15	56
57	PA6	MCI0 DA4	ETX2			ETX3	MCI0 DA5	PA7	58
59	PA8	MCI0 DA6	ERX2			ERX3	MCI0 DA7	PA9	60
61	PB4	TXD1					RXD1	PB5	62
63	PD16	RTS1					CTS1	PD17	64
65	PB6	TXD2					RXD2	PB7	66
67	PD20	TIOA0		TSAD0	TSAD1		TIOA1	PD21	68
69	PB13	DTXD					DRXD	PB12	70
71	PD29	TCLK1	SCK1		GPAD6	PWM2	PCK0	PD26	72
73	PD14	TF1					RF1	PD15	74
75	PD12	TK1	PCK0				RK1	PD13	76
77	PD10	TD1					RD1	PD11	78
79	PA1	MCI0 CDA	TIOA3			TCLK3	MCI0 CK	PA0	80
81	PA2	MCI0 DA0	TIOB3			TCLK4	MCI0 DA1	PA3	82
83	PA4	MCI0 DA2	TIOA4			TIOB4	MCI0 DA3	PA5	84
85	HDMA				HDPA				86
87	HDMB				HDPB				88
89	DDM				DDP				90
91	GND				GND				92
93	PA10	ETX0					ETX1	PA11	94
95	PA12	ERX0					ERX1	PA13	96
97	PA16	ERXER					ETXEN	PA14	98

Stamp9G45 Pin Assignment

Pin	GPIO	Periph. A	Periph. B	Add. Function	Add. Function	Periph. B	Periph. A	GPIO	Pin
99	PA18	EMDC					EMDIO	PA19	100

Table D.2. Pin Assignment IO Interface

Appendix E. Stamp9G45 Electrical Characteristics

Ambient temperature 25°C, unless otherwise indicated

Symbol	Description	Parameter	Min.	Typ.	Max	Unit
V_{CC}	Operating Voltage		3.0	3.3	3.6	V
V_{MEM}	Memory Bus Voltage		1.65	1.8	1.95	V
V_{RES}	Reset Treshhold			2.9		V
T_{RES}	Duration of Reset Pulse		150		280	ms
V_{IH}	High-Level Input Voltage	3.3V	2.0		$V_{CC} + 0.3$	V
		(PIOC4 - PIOC31) 1.8V	1.26		2.1	V
V_{IL}	Low-Level Input Voltage	3.3V	-0.3		0.8	V
		(PIOC4 - PIOC31) 1.8V	-0.3		0.54	V
P	Normal Operation			345		mW
	Full Load	max.		457		mW
	Stand-By			266		mW
	Power-Down			125		mW
V_{BATT}	Battery Voltage		2.0	3.0	V_{CC}	V
I_{BATT}	Battery Current	Ambient temp. = 25°C		5		μ A
		Ambient temp. = 70°C			17	μ A
		Ambient temp. = 85°C			22	μ A

Table E.1. Electrical Characteristics

Appendix F. Stamp9G45 Clock Characteristics

Symbol	Description	Dependency	Tolerance	Typical Value	Unit
MAINCK	Main Oscillator frequency			12.000	MHz
SLCK	Slow Clock			32.768	KHz
PLLACK	PLLA Clock	MAINCK		800.000	MHz
PCK	Processor Clock	PLLACK		400.000	MHz
MCK	Master Clock	PCK		133.000	MHz
DDCK	DDRAM Clock	MCK		266.000	MHz
BCK	Baudrate Clock	MCK	1.5%	8.25(max)	MHz
UPLLCK	USB Clock	MAINCK	0.25%	480.000	MHz

Table F.1. Clock Characteristics

Appendix G. Stamp9G45 Environmental Ratings

Symbol	Description	Parameter	Operating		Storage		Unit
			Min.	Max.	Min.	Max.	
T _A	Ambient temperature		-30	85	-45	85	°C
	Relative Humidity	no condensation		90		90	%RH
	Absolute Humidity		<= Humidity@T _A = 60°C, 90%RH				
	Corrosive Gas		not admissible				

Table G.1. Environmental Ratings

Appendix H. Stamp9G45 Dimensions

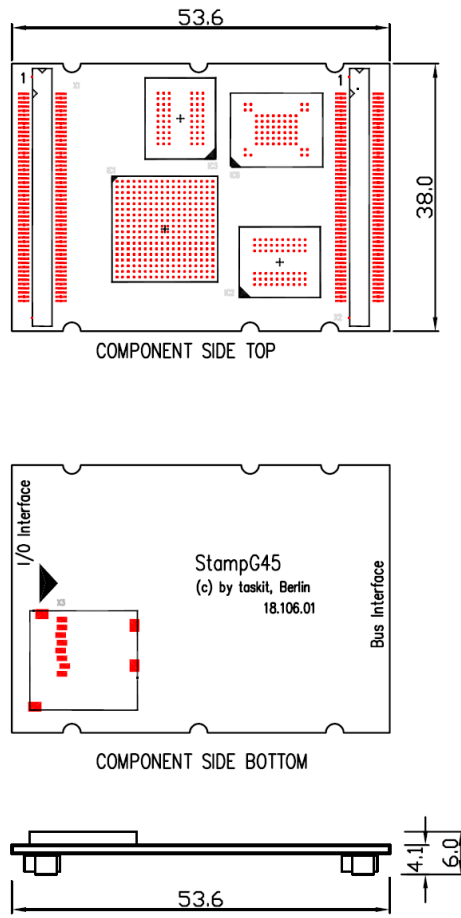
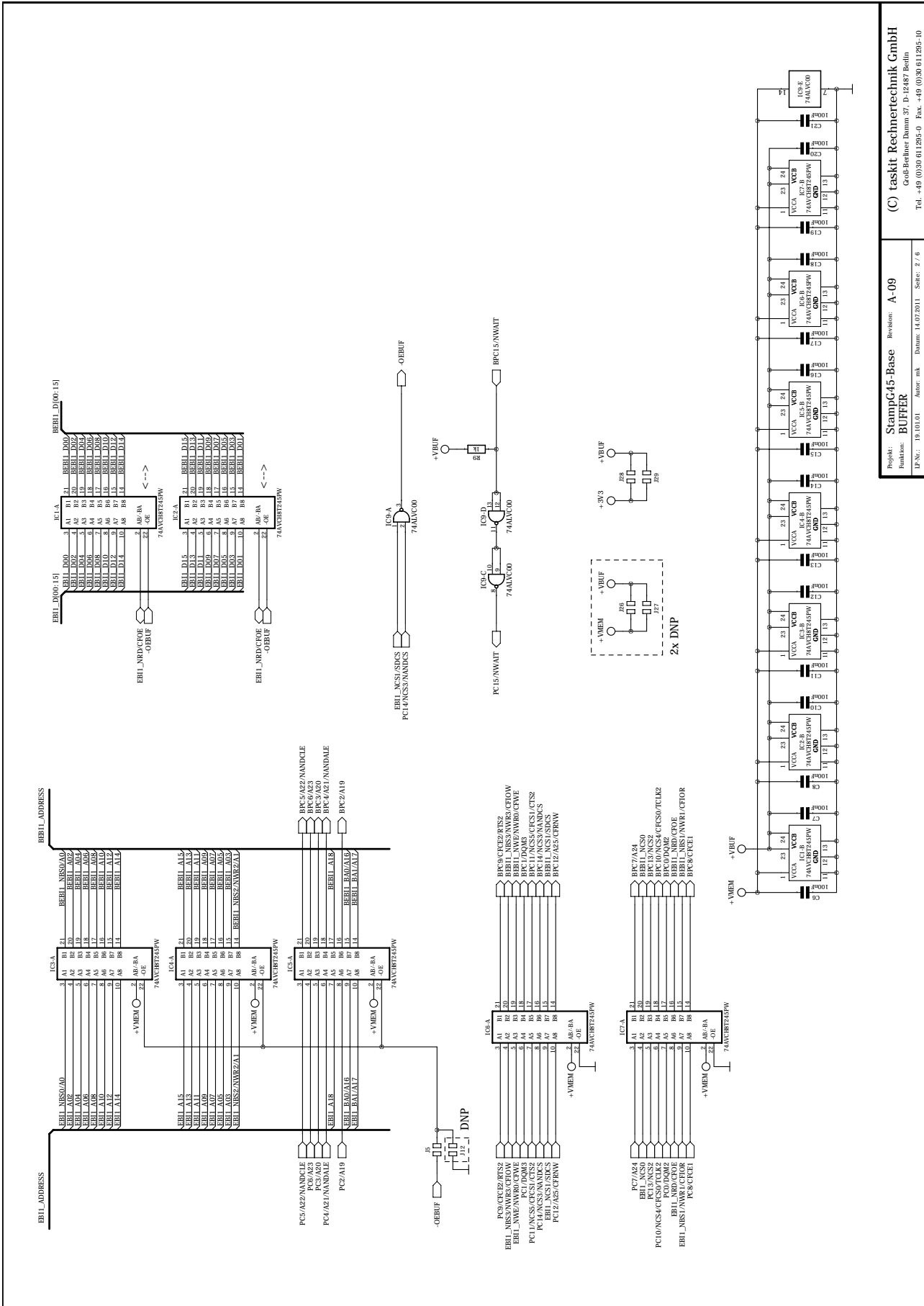


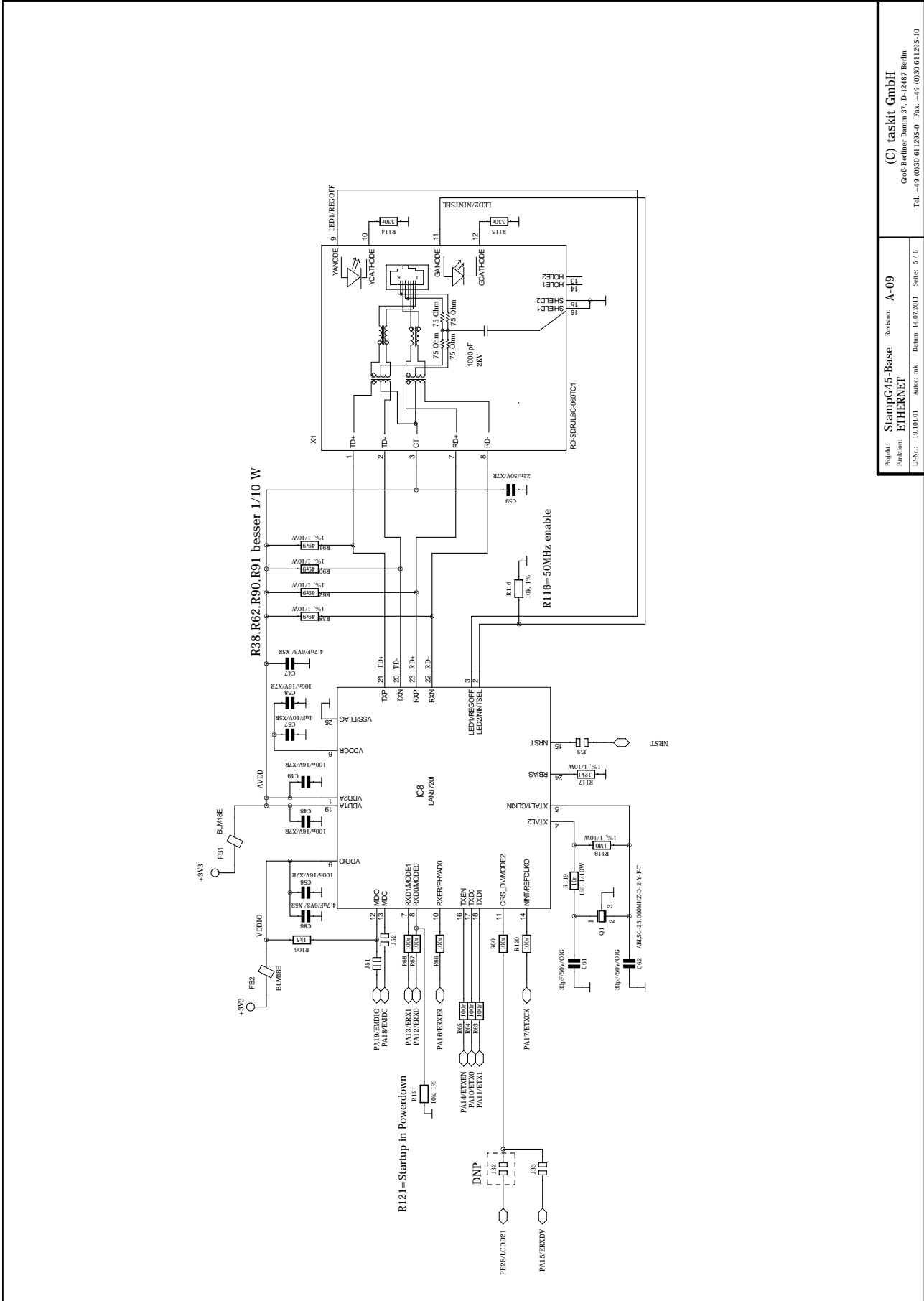
Figure H.1. Stamp9G45 Dimensions



Projekt: StampG45-Base Revision: A-09
 Funktion: BUFFER
 IP-Nr.: 19.101.01 Autor: mk Datum: 14.07.2011 Seite: 2 / 6

(C) taskit Rechnertechnik GmbH
 Groß-Böhrler Damm 37, D-12487 Berlin
 Tel. +49 (0)30 61 1295-0 Fax. +49 (0)30 61 1295-10

Figure I.2. Starterkit Buffer



Project: StampG45-Base Revision: A-09
 Funktion: ETHERNET
 IP-Nr.: 10.101.01 Autor: mk Datum: 14.07.2011 Seite: 5/6
 (C) taskit GmbH
 Groß-Beuthener Damm 37, D-12487 Berlin
 Tel.: +49 (0)30 61 1295-0 Fax: +49 (0)30 61 1295-10

Figure I.5. Starterkit Ethernet

