PIN	SIGNAL	INPUT	OUTPUT	OE CTRLD BY
0	5V			
1	5V			
2	GND			
3	GND			
4	BFQ2	NOT USED	NOT USED	
5	BFQ1	NOT USED	NOT USED	
6	OUTIDENT		X	ALWAYS
7	INIDENT	X		
8	OUTGRANT		X	ALWAYS
9	INGRANT	X		
10	BD0	X	X	INPUT
11	IOXE	X		
12	BD1	X	X	INPUT
13	BD2	X	X	INPUT
14	BD3	X	X	INPUT
15	BINPUT		X	CONNECT
16	BD4	X	X	INPUT
17	BD5	X	X	INPUT
18	BD6	X	X	INPUT
19	MPX ADDRESS		X	CONNECT
20	BD7	X	X	INPUT
21	BD8	X	X	EINPUT
22	BD9	X	X	EINPUT
23	BCONNECT		X	CONNECT
24	BD10	X	X	EINPUT
25	BD11	X	X	EINPUT
26	BD12	X	X	EINPUT
27	INT 10			OC
28	BD13	X	X	EINPUT
29	BD14	X	X	EINPUT
30	BD15	X	X	EINPUT
31	INT 11	X	X	OC
32	BA0	X		
33	BA1	X		
34	BA2	X		
35	INT 12		X	OC
36	BA3	X		
37	BA4	X		
38	BA5	X		
39	INT 13		X	OC
40	BA6	X		
41	BA7	X		
42	BA8	X		
43	INT 15		X	OC
44	BA9	X		
45	BA10	X		
46	BA11		X	CONNECT
47	DMA REQUEST		X	OC
48	BA12		X	CONNECT
49	BA13		X	CONNECT
50	BA14		X	CONNECT
51	DMA DATA READY	X		
52	BA15		X	CONNECT
53	BA16		X	CONNECT
54	BA17		X	CONNECT
55	MASTER CLEAR	X		
96	GND			
97	GND			
98	5V			

Emulating the 1022 card

OC and ALWAYS, 8 signals driven by one 74LVC245.

OC implemented by a schottky diode

Maybe we could use only one interrupt signal and thus save 4 signals.

INPUT control 8 signals driven by one 74LVC245

EINPUT control 8 signals driven by one 74LVC245

CONNECT controls nOE for 10 signals driven by two 74LVC245

16 Databus signals level converted by two 74LVC245 OE controlled by BUS THREESTATE signal

16 signals, BA0-BA10, INIDENT, INGRANT, DMA DATA READY, MASTER CLEAR, IOXE always active input signals. nOE tied to GND. Two 74LVC245

Total 9 74LVC245 to achieve level conversion to 3.3V

Total 66 signals

~~Switches: 6 + 4 signals~~ ~~10 signals~~ CAR and BA Switches can be written by the STM32F407 to a register in the FPGA

uP interface. Multiplexed A/D. 16 signals.

Write strobe Output Enable

ALE Chip Select

20 signals

JTAG - 5 signals, CLK 1 signal, CONFIG_DONE and nCONFIG signal. Total 8 signals

FPGA has 101 signals - need to use a three state databus. Now we need 88 signals.

Need three signlas to control output three state drivers, INPUT, EINPUT and CONNECT.

~~Need one signal to control the input databus threestate.~~

Grand total 97 signals. Remains 4 unused signals in the FPGA

STM32F407 INTERFACE

74LVC OE SIGNALS

INPUT

CONNECT

THREE-STATE INPUT AND INPUT-SIGNAL OUTPUT CONROL

THREE-STATE INPUT AND EINPUT-SIGNAL OUTPUT CONROL

OC

OUTPUT

DATABUS IN ALWAYS ENABLED

99	5V								
1A	VREFB1N0	IO			DIFFIO_RX_L1n	DIFFOUT_L1n	Low_Speed	6	BA0
1A	VREFB1N0	IO			DIFFIO_RX_L1p	DIFFOUT_L1p	Low_Speed	7	BA1
1A	VREFB1N0	IO			DIFFIO_RX_L3n	DIFFOUT_L3n	Low_Speed	8	BA2
1A	VREFB1N0	IO			DIFFIO_RX_L3p	DIFFOUT_L3p	Low_Speed	10	BA3
1A	VREFB1N0	IO			DIFFIO_RX_L5n	DIFFOUT_L5n	Low_Speed	11	BA4
1A	VREFB1N0	IO			DIFFIO_RX_L5p	DIFFOUT_L5p	Low_Speed	12	BA5
1A	VREFB1N0	IO			DIFFIO_RX_L7n	DIFFOUT_L7n	Low_Speed	13	BA6
1A	VREFB1N0	IO			DIFFIO_RX_L7p	DIFFOUT_L7p	Low_Speed	14	BA7
1B	VREFB1N0	IO		JTAGEN				15	JTAG
1B	VREFB1N0	IO		TMS	DIFFIO_RX_L11	DIFFOUT_L11n	Low_Speed	16	JTAG
1B	VREFB1N0	IO	VREFB1N0					17	BA8
1B	VREFB1N0	IO		TCK	DIFFIO_RX_L11	DIFFOUT_L11p	Low_Speed	18	JTAG
1B	VREFB1N0	IO		TDI	DIFFIO_RX_L12	DIFFOUT_L12n	Low_Speed	19	JTAG
1B	VREFB1N0	IO		TDO	DIFFIO_RX_L12	DIFFOUT_L12p	Low_Speed	20	JTAG
1B	VREFB1N0	IO			DIFFIO_RX_L14	DIFFOUT_L14n	Low_Speed	21	BA9
1B	VREFB1N0	IO			DIFFIO_RX_L14	DIFFOUT_L14p	Low_Speed	22	BA10
1B	VREFB1N0	IO			DIFFIO_RX_L16	DIFFOUT_L16n	Low_Speed	24	BA11
1B	VREFB1N0	IO			DIFFIO_RX_L16	DIFFOUT_L16p	Low_Speed	25	BA12
2	VREFB2N0	IO	CLK0n		DIFFIO_RX_L18	DIFFOUT_L18n	High_Speed	26	BA13
2	VREFB2N0	IO	CLK0p		DIFFIO_RX_L18	DIFFOUT_L18p	High_Speed	27	CLK
2	VREFB2N0	IO	CLK1n		DIFFIO_RX_L20	DIFFOUT_L20n	High_Speed	28	BA14
2	VREFB2N0	IO	CLK1p		DIFFIO_RX_L20	DIFFOUT_L20p	High_Speed	29	BA15
2	VREFB2N0	IO	VREFB2N0					30	BA16
2	VREFB2N0	IO	PLL_L_CLKOUTn		DIFFIO_RX_L27	DIFFOUT_L27n	High_Speed	32	BA17
2	VREFB2N0	IO	PLL_L_CLKOUTp		DIFFIO_RX_L27	DIFFOUT_L27p	High_Speed	33	BD0 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B1n	High_Speed	38	BD1 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B1p	High_Speed	39	BD2 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B3n	High_Speed	41	BD3 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B3p	High_Speed	43	BD4 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B5n	High_Speed	44	BD5 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B5p	High_Speed	45	BD6 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B7n	High_Speed	46	BD7 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B7p	High_Speed	47	BD8 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B9n	High_Speed	50	BD9 OUT
3	VREFB3N0	IO	VREFB3N0					48	BD10 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B9p	High_Speed	52	BD11 OUT
3	VREFB3N0	IO						54	BD12 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B12n	High_Speed	55	BD13 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B12p	High_Speed	56	BD14 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B14n	High_Speed	57	BD15 OUT
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B14p	High_Speed	58	BD0 IN
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B16n	High_Speed	59	BD1 IN
3	VREFB3N0	IO			DIFFIO_TX_RX_	DIFFOUT_B16p	High_Speed	60	BD2 IN
4	VREFB4N0	IO	VREFB4N0					61	BD3 IN
4	VREFB4N0	IO						62	BD4 IN
4	VREFB4N0	IO			DIFFIO_TX_RX_	DIFFOUT_B23n	High_Speed	64	BD5 IN
4	VREFB4N0	IO			DIFFIO_TX_RX_	DIFFOUT_B23p	High_Speed	65	BD6 IN
4	VREFB4N0	IO						66	BD7 IN
4	VREFB4N0	IO			DIFFIO_TX_RX_	DIFFOUT_B27n	High_Speed	69	BD8 IN
4	VREFB4N0	IO			DIFFIO_TX_RX_	DIFFOUT_B27p	High_Speed	70	BD9 IN
5	VREFB5N0	IO			DIFFIO_RX_R1p	DIFFOUT_R1p	High_Speed	75	BD10 IN
5	VREFB5N0	IO			DIFFIO_RX_R2p	DIFFOUT_R2p	High_Speed	74	BD11 IN
5	VREFB5N0	IO			DIFFIO_RX_R1n	DIFFOUT_R1n	High_Speed	77	BD12 IN
5	VREFB5N0	IO			DIFFIO_RX_R2n	DIFFOUT_R2n	High_Speed	76	BD13 IN
5	VREFB5N0	IO			DIFFIO_RX_R7p	DIFFOUT_R7p	High_Speed	79	BD14 IN
5	VREFB5N0	IO						78	BD15 IN
5	VREFB5N0	IO			DIFFIO_RX_R7n	DIFFOUT_R7n	High_Speed	81	OUTIDENT
5	VREFB5N0	IO	VREFB5N0					80	INIDENT

5	VREFB5N0	IO			DIFFIO_RX_R10	DIFFOUT_R10p	High_Speed	85	OUTGRANT
5	VREFB5N0	IO			DIFFIO_RX_R11	DIFFOUT_R11p	High_Speed	84	INGRANT
5	VREFB5N0	IO			DIFFIO_RX_R10	DIFFOUT_R10n	High_Speed	87	IOXE
5	VREFB5N0	IO			DIFFIO_RX_R11	DIFFOUT_R11n	High_Speed	86	BINPUT
6	VREFB6N0	IO	CLK2p		DIFFIO_RX_R14	DIFFOUT_R14p	High_Speed	88	MPX ADDRESS
6	VREFB6N0	IO	CLK2n		DIFFIO_RX_R14	DIFFOUT_R14n	High_Speed	89	BCONNECT
6	VREFB6N0	IO	CLK3p		DIFFIO_RX_R16	DIFFOUT_R16p	High_Speed	90	MASTER CLEAR
6	VREFB6N0	IO	CLK3n		DIFFIO_RX_R16	DIFFOUT_R16n	High_Speed	91	DMA REQUEST
6	VREFB6N0	IO			DIFFIO_RX_R18	DIFFOUT_R18p	High_Speed	92	DMA DATA READY
6	VREFB6N0	IO			DIFFIO_RX_R18	DIFFOUT_R18n	High_Speed	93	AD0
6	VREFB6N0	IO	DPCLK3		DIFFIO_RX_R26	DIFFOUT_R26p	High_Speed	96	AD1
6	VREFB6N0	IO	VREFB6N0					97	AD2
6	VREFB6N0	IO	DPCLK2		DIFFIO_RX_R26	DIFFOUT_R26n	High_Speed	98	AD3
6	VREFB6N0	IO			DIFFIO_RX_R27	DIFFOUT_R27p	High_Speed	99	AD4
6	VREFB6N0	IO			DIFFIO_RX_R28	DIFFOUT_R28p	High_Speed	100	AD5
6	VREFB6N0	IO			DIFFIO_RX_R27	DIFFOUT_R27n	High_Speed	101	AD6
6	VREFB6N0	IO			DIFFIO_RX_R28	DIFFOUT_R28n	High_Speed	102	AD7
6	VREFB6N0	IO			DIFFIO_RX_R33	DIFFOUT_R33p	High_Speed	105	AD8
6	VREFB6N0	IO			DIFFIO_RX_R33	DIFFOUT_R33n	High_Speed	106	AD9
7	VREFB7N0	IO			DIFFIO_RX_T1p	DIFFOUT_T1p	High_Speed	110	AD10
7	VREFB7N0	IO			DIFFIO_RX_T1n	DIFFOUT_T1n	High_Speed	111	AD11
7	VREFB7N0	IO	VREFB7N0					112	AD12
7	VREFB7N0	IO						113	AD13
7	VREFB7N0	IO						114	AD14
7	VREFB7N0	IO			DIFFIO_RX_T10	DIFFOUT_T10p	High_Speed	118	AD15
7	VREFB7N0	IO			DIFFIO_RX_T10	DIFFOUT_T10n	High_Speed	119	nADS
8	VREFB8N0	IO			DIFFIO_RX_T16	DIFFOUT_T16p	Low_Speed	120	nOE
8	VREFB8N0	IO		DEV_CLRn	DIFFIO_RX_T16	DIFFOUT_T16n	Low_Speed	121	nWE
8	VREFB8N0	IO		DEV_OE				122	nCE
8	VREFB8N0	IO	VREFB8N0					123	INPUT
8	VREFB8N0	IO		CONFIG_SEL				126	EINPUT
8	VREFB8N0	IO			DIFFIO_RX_T19	DIFFOUT_T19p	Low_Speed	124	CONNECT
8	VREFB8N0	Input_only		nCONFIG				129	nCONFIG
8	VREFB8N0	IO			DIFFIO_RX_T19	DIFFOUT_T19n	Low_Speed	127	INT11
8	VREFB8N0	IO			DIFFIO_RX_T20	DIFFOUT_T20p	Low_Speed	130	INT12
8	VREFB8N0	IO			DIFFIO_RX_T20	DIFFOUT_T20n	Low_Speed	131	INT13
8	VREFB8N0	IO			DIFFIO_RX_T22	DIFFOUT_T22p	Low_Speed	132	INT14
8	VREFB8N0	IO		CRC_ERROR	DIFFIO_RX_T22	DIFFOUT_T22n	Low_Speed	134	INT15
8	VREFB8N0	IO						135	
8	VREFB8N0	IO		nSTATUS	DIFFIO_RX_T24	DIFFOUT_T24p	Low_Speed	136	
8	VREFB8N0	IO		CONF_DONE	DIFFIO_RX_T24	DIFFOUT_T24n	Low_Speed	138	CONF_DONE
8	VREFB8N0	IO			DIFFIO_RX_T26	DIFFOUT_T26p	Low_Speed	140	
8	VREFB8N0	IO			DIFFIO_RX_T26	DIFFOUT_T26n	Low_Speed	141	
		GND						3	
		GND						4	
		GND						95	
		GND						83	
		GND						68	
		GND						63	
		GND						53	
		GND						42	
		GND						142	
		GND						137	
		GND						133	
		GND						125	
		GND						116	
		GND						104	
		VCCIO1A						9	
		VCCIO1B						23	
		VCCIO2						31	

		VCCIO3						49
		VCCIO3						40
		VCCIO4						67
		VCCIO5						82
		VCCIO6						94
		VCCIO6						103
		VCCIO7						117
		VCCIO8						139
		VCCIO8						128
		VCCA1						35
		VCCA2						34
		VCCA3						5
		VCCA3						107
		VCCA4						143
		VCCA5						71
		VCCA6						2
		VCC_ONE						73
		VCC_ONE						72
		VCC_ONE						51
		VCC_ONE						37
		VCC_ONE						36
		VCC_ONE						144
		VCC_ONE						115
		VCC_ONE						109
		VCC_ONE						108
		VCC_ONE						1

Altera MAX 10	10M04SCE144C8G	4k LE	9.72 Euro	1LE is one LUT and one bit register
	10M02SCE144C8G	2k LE	6.27 Euro	
	10M08SCE144C8G	8k LE	14.84 Euro	
PSU 3.3V	TPS54229_PDSO-8	TPS54229DDAR	1,35 Euro	
CONF_DONE LED driver	DMN65D8L	DMN65D8LQ-7	0,17 euro	
50 MHz oscillator	ASDMB-50MHz	ASDMB-50.000MH	1,05 euro	**Registers needed**
8 MHz kristall				
USB	USB kontakt.			Block Address Register I BAR 16 bits
JTAG - FPGA				Block Address Register II 16 bits Max 823 cylinders. But maybe all bits are tested by test programs?
JTAG - STM32				Status Register 16 bits
Serieport TX/RX				Control Word 16 bits
Serial SPI FLASH for configuration storage		AT25DN256-SSHF	0,255 Euro	Read Seek condition 15 bits
				ECC count register 14 bits A sector is 8192 bits + 56 bit ECC
STM32F407VET6			9.67 Euro	ECC pattern register 12 bits
				ECC control 3 bits
				Word Count 14 bits Max 18 sectors = 9 k data
				Summa 122 bits
				Smallest FPGA is more than enough.
				ECC Disk
				CWR bit 15=0 CWR bit 15 = 1
				IOX 1540 READ CORE ADDRESS READ CORE ADDRESS In the 1096 module. Not needed.
				IOX 1541 LOAD CORE ADDRESS LOAD CORE ADDRESS In the 1096 module. Not needed.
				IOX 1542 READ SEEK CONDITION READ ECC COUNT Is ECC really necessary?
				IOX 1543 LOAD BLOCK ADDR I LOAD BLOCK ADDR II
				IOX 1544 READ STATUS REGISTER READ ECC PATTERN
				IOX 1545 LOAD CONTROL WORD LOAD ECC CONTROL
				IOX 1546 READ BLOCK ADDRESS I READ BLOCK ADDRESS II For test mode - just read back register content
				IOX 1547 LOAD WORD COUNT LOAD ECC CONTROL

Big Disk It doesn't make sense to have the trouble with the ECC disk?

IOX 1540	READ CORE ADDRESS		
IOX 1541	LOAD CORE ADDRESS		
IOX 1542	READ SEEK CONDITION	12 bits	
IOX 1543	LOAD BLOCK ADDRESS	16 bits	
IOX 1544	READ STATUS REGISTER	15 bits	But how many of the error bits are needed?
IOX 1545	LOAD CONTROL WORD	16 bits	
IOX 1546	READ BLOCK ADDRESS		Read back block address in test mode
IOX 1547	LOAD WORD COUNT REGISTER	13 bits	Max 8k words transfer
		Sum 72 bits	

Programmed IO

8 bit IO address base register

8 bit IO BASE is matched against BA3-BA10 when IOXE is LOW. Then CONNECT is set LOW. If BA0 is LOW then also INPUT is set LOW and enable MD0-MD15 on the databus. BA0-BA2 selects the register to be gated on MD0-MD15 OUT signals.
INPUT signal gates the CLK pulse to store data into selected register.

Possible need to synchronize the BD0/BD15 data inputs with a 16 bit buffer register clocked by main clock.

DMA

From Device to Memory
STM32F407 writes WC counter with the amount of data to transfer. A direction register set to high indicate the transfer goes from device to memory. As soon as the WC is written the write buffer empty status is set.
STM32F407 writes to buffer register. This will clear the write buffer empty status. This also set the DMA REQUEST flip-flop and a DMA request signal is generated IF WC register is non-zero

IF INGRANT active AND DMA REQUEST flip-flp is INACTIVE then enable OUTGRAN signal
IF INGRANT active AND DMA REQUEST flip-flp is ACTIVE then set CONNECT signal LOW.
Enable BA11-BA14 on address bus. Gate CAR register on BA11-BA14.
Enable MD0-MD15 on databus. GATE data buffer onto MD0-MD15

From Memory to Device

Interrupt

Why is the 6 bit code in the IDENT instruction so weird?
BA0-BA5 Give a coded value for the external interrupt level where the program is seeking to identify an interrupt.

Instruction	Code	5 LSB bits
IDENT PL10	143604	0 0 0 1 0 0
IDENT PL11	143611	0 0 1 0 0 1
IDENT PL12	143622	0 1 0 0 1 0
IDENT PL13	143643	1 0 0 0 1 1

Programmable interrupt request level will require a 4 bit interrupt level register
Writing this register will start the interrupt sequence.
A 8 bit register providning the IDENT code to be sent back is required.

When INIDENT is LOW AND BA2-BA5 is matching INTERRUPT REUQEST LEVEL an INPUT and CONNECT signal is generated.
This also gates the 8 bit IDENT register on the BD0-BD7 OUT signals and enables the BD0-BD7 bus driver.

The four MSB here seems to map to the interrupt level

When INIDENT is received LOW and BA2-BA5 is NOT matching OUTIDENT is LOW.

But what is the reason for the 2 LSB bits?

I/O manual say "Count-code"

Could it be two ways of telling exactly the same thing? The two lower bits is a binary code for interrupt level.

The 1022 module only makes use of a single bit, BA3 corresponding to the INT 11.

Configuring the FPGA

The FPGA shall be possible to be configured through the JTAG port or from the STM32 microprocessor.

The STM32 reads a bitstream file from the SD card FATFS file system and use this as configuration information.

FPGA JTAG port of board is parallell to the with the STM32.

JTAG ports

Two JTAG ports. One for FPGA and one for STM32.

FPGA config from STM32

https://www.intel.com/content/www/us/en/programmable/support/support-resources/support-centers/devices/cfg-index/cfg-jtag.html

Jam STAPL player	Reads .jam or .jbc files.	
JRunner	Reads .cdf and .rbf files.	https://github.com/RichardPlunkett/jrunner-beaglebone