Outline

- Field Programmable Gate Arrays
  - Historical perspective
- Programming Technologies
- Architectures
  - PALs, PLDs, and CPLDs
  - FPGAs
    - Programmable logic
    - Interconnect network
    - I/O buffers
    - Specialized cores
- Programming Interfaces
History

- **Programmable Logic Arrays ~ 1970**
  - Implement any set of sum-of-products logic equations
  - Incorporated in VLSI devices

- **Programmable Logic Devices ~ 1980**
  - MMI Programmable Array Logic (PAL)
    - 16L8 – combinational logic only
    - 16R8 – sequential logic only
  - AMD 22V10 and Lattice 16V8
  - Complex PLDs – arrays of PLDs with routing network

- **Field Programmable Gate Arrays ~ 1985**
  - Xilinx Logic Cell Array (LCA)

- **CPLD & FPGA architectures became similar ~2000**
  - Incorporation of RAMs and other specialized cores
    - Programmable system-on-chip
Programming Technologies

- PLAs were mask programmable
- PALs used fuses for programming
- Early PLDs & CPLDs used floating gate technology
  - Erasable Programmable Read Only Memory (EPROM)
    - Ultra-violet erasable (UVEPROM)
    - Electrically erasable (EEPROM)
    - Flash memory came later and was used for CPLDs
- FPGAs used RAM for programming
- Later trends
  - Fuses were replaced with anti-fuses
    - Better reliability
  - Large CPLDs went to RAM-based programming
Programming Technologies

- **RAM**
  - Volatile – must configure after power-up
  - In-System Re-programmable (ISR)
  - Run-Time Reconfiguration (RTR)
    - dynamic reconfiguration while system is operating

- **Floating gate technologies**
  - Non-volatile but re-usable
    - UV EPROM, EEPROM, and flash memory
  - In-System Programmable (ISP)
    - EEPROM and flash memory
  - In-System Re-programmable (ISR)
    - Flash memory

- **Fuse/anti-fuse**
  - Non-volatile but not re-usable
  - One Time Programmable (OTP)
PALs

16L8 – combinational logic

- 10 to 16 inputs, each with true and complement signal
- 2 to 8 outputs, each with
  - 7 product terms can AND any of up to 16 inputs or their complements
  - Tri-state control product term for inverting output buffer
  - When output in tri-state, I/O pin can be used as input
    - High impedance output with no signal driven
PALs

16R8 – sequential logic

- 8 inputs, each with true & complement
- 8 outputs, each with
  - D flip-flop
  - With feedback for FSMs
  - 8 product terms that can AND any of:
    - 8 inputs or their complements
    - 8 feedbacks or their complements from D flip-flops

- One clock for all FFs
- One tri-state control for all outputs

C. Stroud 8/06
PLDs

22V10 replaced all PALs
- Combinational and/or sequential logic
  - Macrocell program bits C0, C1
- Up to 22 inputs w/complement
- Up to 10 outputs, each with
  - Macrocell
  - 8-16 product terms
  - Tri-state control product term
- Global
  - preset & clear PTs
  - clock

C. Stroud 8/06
PLDs

- **16V8**
  - Up to 16 inputs (bit & bitbar)
  - Up to 8 outputs, each with
    - 8 product terms (PTs), or
    - 7 with tri-state control (PT)
  - Macrocell similar to 22V10
    - More programming options
  - Ability to select adjacent pin
    - Allows embedded registers

---

[Diagram of 16V8 PLD]

**C. Stroud 8/06**

**FPGAs**
CPLDs

Cypress Semiconductor 374 CPLD Architecture
84-pin package w/~6 Vcc and 8 Gnd pins
36 inputs to AND-plane w/84 PTs and partially programmable OR-plane

Logic Block Diagram
CPLDs

- An array of PLDs
  - Global routing resources for connections
    - PLDs to other PLDs
    - PLDs to/from I/O pins
- Example: Cypress 39K
  - Each Logic Block (LB) similar to a 22V10
  - Each cluster of 8 LBs has two 8K RAMs & one 4K dual-port RAM/FIFO
    - Programmable Interconnect Modules (PIMs) provide interconnections
  - Array of up to 24 clusters with global routing
## Ranges of Resources

<table>
<thead>
<tr>
<th>FPGA Resource</th>
<th>Small FPGA</th>
<th>Large FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Logic</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLBs per FPGA</td>
<td>256</td>
<td>25,920</td>
</tr>
<tr>
<td>LUTs and flip-flops per PLB</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td><strong>Routing</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wire segments per PLB</td>
<td>45</td>
<td>406</td>
</tr>
<tr>
<td>PIPs per PLB</td>
<td>139</td>
<td>3,462</td>
</tr>
<tr>
<td><strong>Specialized Cores</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bits per memory core</td>
<td>128</td>
<td>36,864</td>
</tr>
<tr>
<td>Memory cores per FPGA</td>
<td>16</td>
<td>576</td>
</tr>
<tr>
<td>DSP cores</td>
<td>0</td>
<td>512</td>
</tr>
<tr>
<td><strong>Other</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input/output cells</td>
<td>62</td>
<td>1,200</td>
</tr>
<tr>
<td>Configuration memory bits</td>
<td>42,104</td>
<td>79,704,832</td>
</tr>
</tbody>
</table>

C. Stroud 8/06

FPGAs
Basic PLB Architecture

- Look-up Table (LUT) implements truth table
- Memory elements:
  - Flip-flop/latch
  - Some FPGAs - LUTs can also implement small RAMs
- Carry & control logic implements fast adders/subtractors
A Simple PLB

- Two 3-input LUTs
  - Can implement any 4-input combinational logic function
- 1 flip-flop
  - Programmable:
    - Active levels
    - Clock edge
    - Set/reset
- 22 configuration memory bits
  - 8 per LUT
    - C0-7
    - S0-7
  - 6 controls
    - CB0-7

C. Stroud 8/06
Combinational Logic Functions

- Gates are combined to create complex circuits
- Multiplexer example
  - If $S = 0$, $Z = A$
  - If $S = 1$, $Z = B$
- Very common digital circuit
- Heavily used in FPGAs
  - $S$ input controlled by configuration memory bit
  - We’ll see it again

![Multiplexer example diagram]

<table>
<thead>
<tr>
<th>S</th>
<th>A</th>
<th>B</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Look-up Tables

- Recall multiplexer example
- Configuration memory holds outputs for truth table
- Internal signals connect to control signals of multiplexers to select value of truth table for any given input value
Look-up Table Based RAMs

- Normal LUT mode performs read operations
- Address decoder with write enable generates clock signals to latches for write operations
- Small RAMs but can be combined for larger RAMs
Interconnect Network

- Wire segments of varying length
  - $xN = N$ PLBs in length
    - 1, 2, 4, and 6 are most common
  - $xH$ = half the array in length
  - $xL$ = length of full array

- Programmable Interconnect Points (PIPs)
  - Also known as Configurable Interconnect Points (CIPs)
  - Transmission gate connects to 2 wire segments
  - Controlled by configuration memory bit
    - 0 = wires disconnected
    - 1 = wires connected
PIPs

- Break-point PIP
  - Connect or isolate 2 wire segments

- Cross-point PIP
  - Turn corners

- Multiplexer PIP
  - Directional and buffered
  - Select 1-of-\(N\) inputs for output
    - Decoded MUX PIP – \(N\) config bits select from \(2^N\) inputs
    - Non-decoded MUX PIP – 1 config bit per input

- Compound cross-point PIP
  - Collection of 6 break-point PIPs
    - Can route to two isolated signal nets
Spartan 3 Routing Resources

PLB consists of 4 slices

over 2,400 PIPs
mostly MUX PIPs

x6 wire segments

x2 wire segments

xH & xL wire segments

over 450 total wire segments in PLB

C. Stroud 8/06
FPGAs

- Recent trend - incorporate specialized cores
  - RAMs – single-port, dual-port, FIFOs
    - 128 bits to 36K bits per RAM
    - 4 to 575 per FPGA
  - DSPs – 18x18-bit multiplier, 48-bit accumulator, etc.
    - up to 512 per FPGA
  - Microprocessors and/or microcontrollers
    - up to 2 per FPGA
      - Hard core processor
    - Support soft core processors
      - Synthesized from HDL into programmable resources
FPGA Architectures

- **4000/Spartan**
  - $N \times N$ array of unit cells
  - Unit cell = CLB + routing
    - Special routing along center axes
  - I/O cells around perimeter

- **Virtex/Spartan-2**
  - $M \times N$ array of unit cells
  - Added block 4K RAMs at edges

- **Virtex-2/Spartan-3**
  - Block 18K RAMs in array
  - Added 18x18 multipliers with each RAM
  - Added PowerPCs in Virtex-2 Pro

- **Virtex-4/Virtex-5**
  - Added 48-bit DSP cores w/multipliers
  - I/O cells along columns for BGA
Specialized Cores

C. Stroud 8/06

Virtex and Spartan II   Virtex II and Spartan 3

4K-bit RAMs           18K-bit RAMs and 18×18-bit multipliers
Programmable RAMs

- 18 Kbit dual-port RAM
- Each port independently configurable as
  - 512 words x 36 bits
    - 32 data bits + 4 parity bits
  - 1K words x 18 bits
    - 16 data bits + 2 parity bits
  - 2K words x 9 bits
    - 8 data bits + 1 parity bit
  - 4K words x 4 bits (no parity)
  - 8K words x 2 bits (no parity)
  - 16K words x 1 bit (no parity)

- Each port has independently programmable
  - clock edge
  - active levels for write enable, RAM enable, reset
FPGA Configuration Memorys

- PLB addressable
  - Good for partial reconfiguration
  - X-Y coordinates of PLB location to be written
    - Requires tag to identify which resources will be configured

- Frame addressable
  - Vertical or horizontal frame
  - Access to all PLBs in frame
    - Only portion of logic and routing resources accessible in a given frame
    - Many frames to configure PLBs
      - Major address for column, minor address for frame
Frames vs. Column Type

Day #1 FPGA Verification Course

Number of Frames

- **Virtex1/Spartan2**
- **Virtex2pro**
- **Spartan3**
- **Virtex4**

CLB, IOB/TERM, IOI/DSP, RAM routing, RAM content, center
Virtex-4 Architectures

V4LX

V4SX

V4FX

PowerPC location

Day #1 FPGA Verification Course
Configuration Memory

- **Frame order**
  - CLBs, IOBs, DSPs, & center column form main portion
  - BRAMs come after

- **Frames span 16 rows (V5=20)**
  - 2.5 words per row (V5=2)
  - All columns have INT switch box routing
    - 3,312 PIPs \( \approx \) first 18.5 frames

- **Total frames/column**
  - CLBs = 22 frames
  - DSPs = 21 columns
  - Center column = 33 frames
  - IOBs = 30 frames
    - Left & right cols in LX & SX
  - BRAMs & GTs = 20 frames

- 2 frames at end of row

\[
N = \# \text{ columns}
\]
\[
X = (\# \text{ rows}/16)-1
\]
Virtex-5 Architectures

- Similar architecture, frame structure and order
  - I/O cells not along outside column on right side
  - “Center” column (Xs) not in center of array
    - More columns to right side of “center” column
  - Similar top/bottom and config row format
  - 41 words (32-bit) per frame
    - Hamming bits in middle word of frame

<table>
<thead>
<tr>
<th>part</th>
<th>#rows</th>
<th>Legend</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>80</td>
<td>#=#CLBcols</td>
</tr>
<tr>
<td>50</td>
<td>120</td>
<td>D=DSPs</td>
</tr>
<tr>
<td>85</td>
<td>120</td>
<td>R=RAMs</td>
</tr>
<tr>
<td>110</td>
<td>160</td>
<td>O=I/O cells</td>
</tr>
<tr>
<td>220</td>
<td>160</td>
<td>X=IO&amp;DCM</td>
</tr>
<tr>
<td>330</td>
<td>240</td>
<td>T/C=T only</td>
</tr>
</tbody>
</table>

C. Stroud 9/07

FPGAs
Virtex-5 FX30T

- 5,120 slices
  - 4 FFs & 6-input LUTs
- 68 DPRAMs/FIFOs
  - 36Kbits
- 64 DSPs
  - 24x18 mult & 48-bit ALU
- 1 PowerPC 440
- 1 PCI Express
- 4 Ethernet MACs
- 8 Gigabit Xceivers
FPGA Configuration Memorys

- **PLB addressable**
  - Good for partial reconfiguration
  - X-Y coordinates of PLB location to be written
    - Requires tag to identify which resources will be configured

- **Frame addressable**
  - Vertical or horizontal frame
  - Access to all PLBs in frame
    - Only portion of logic and routing resources accessible in a given frame
    - Many frames to configure PLBs
      - Major address for column, minor address for frame

Hybrid i.e.: Virtex-4 Virtex-5 Virtex-6
Configuration Interfaces

- **Master** – FPGA retrieves its own configuration from ROM after power-up
  - Serial or Parallel options

- **Slave** – FPGA configured by external source (i.e., a µP)
  - Serial or Parallel options
  - Used for dynamic reconfiguration
  - Can also read configuration memory contents

- **Boundary Scan Interface**
  - 4-wire IEEE standard serial interface for testing
  - Write and read access to configuration memory
    - Not available in all FPGAs
    - Used for dynamic partial reconfiguration
  - Interfaces to FPGA core
    - Not available in all FPGAs
    - Connections between Boundary Scan Interface and internal routing network and PLBs (Xilinx provides 2-4 of these ports)

- **Other configuration interfaces in some FPGAs**

C. Stroud 9/07  FPGAs  35
# Xilinx Configuration Interface Pins

<table>
<thead>
<tr>
<th>Name</th>
<th>Direction</th>
<th>Driver Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dedicated Pins</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCLK</td>
<td>Input/Output</td>
<td>Active</td>
<td>Configuration clock. Output in Master mode.</td>
</tr>
<tr>
<td>PROGRAM</td>
<td>Input</td>
<td>Active</td>
<td>Asynchronous reset to configuration logic.</td>
</tr>
<tr>
<td>DONE</td>
<td>Input/Output</td>
<td>Active/ Open-Drain</td>
<td>Configuration status and start-up control.</td>
</tr>
<tr>
<td>M2, M1, M0</td>
<td>Input</td>
<td>Open-Drain</td>
<td>Configuration mode selection.</td>
</tr>
<tr>
<td>TMS</td>
<td>Input</td>
<td>Boundary</td>
<td>Boundary-scan tap controller.</td>
</tr>
<tr>
<td>TCK</td>
<td>Input</td>
<td>Boundary</td>
<td>Boundary-scan clock.</td>
</tr>
<tr>
<td>TDI</td>
<td>Input</td>
<td>Boundary</td>
<td>Boundary-scan data input.</td>
</tr>
<tr>
<td>TDO</td>
<td>Output</td>
<td>Active</td>
<td>Boundary-scan data output.</td>
</tr>
<tr>
<td><strong>Dual Function Pins</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIN (D0)</td>
<td>Input/Output</td>
<td>Active Bidirectional</td>
<td>Serial configuration data input.</td>
</tr>
<tr>
<td>D[0:7]</td>
<td>Input/Output</td>
<td>Active Bidirectional</td>
<td>Slave Parallel configuration data input, readback data output.</td>
</tr>
<tr>
<td>CS</td>
<td>Input</td>
<td>Chip Select</td>
<td>Chip Select (Slave Parallel only).</td>
</tr>
<tr>
<td>WRITE</td>
<td>Input</td>
<td>Active</td>
<td>Active Low write select, read select (Slave Parallel only).</td>
</tr>
<tr>
<td>BUSY/DOUT</td>
<td>Output</td>
<td>Open-Drain/ Active</td>
<td>Busy/Ready status for Slave Parallel (open-drain). Serial configuration data output for serial daisy-chains (active).</td>
</tr>
<tr>
<td>INIT</td>
<td>Input/Output</td>
<td>Open-Drain</td>
<td>Delay configuration, indicate configuration clearing or error.</td>
</tr>
</tbody>
</table>
Spartan 3
Master Modes

Figure 1-1: Spartan-3 Generation Self-Loading (Master) Configuration Modes
Master mode

- Configuration sequence during power-up of device
  - Typically from
    - Serial EPROM
    - Master Serial
    - Parallel EPROM
    - Master Parallel
      - 8-bit
      - 32-bit
Spartan 3 Slave Configuration

(a) Slave Serial mode

Processor, Microcontroller
SERIAL_DATA
CLOCK

Spartan-3 Generation FPGA
DIN
CCLK

Processor, Microcontroller
DATA[7:0]
SELECT
READ/WRITE
READY/BUSY
CLOCK

Spartan-3 Generation FPGA
D[7:0]
CSI_B
RDWR_B
BUSY
CCLK

(b) JTAG mode

JTAG Tester, Processor, Microcontroller
DATA_OUT
MODE_SELECT
CLOCK
DATA_IN

Spartan-3 Generation FPGA
TDI
TMS
TCK
TDO

(c) Slave Parallel mode (SelectMAP)

Figure 1-2: Spartan-3 Generation Downloaded (Slave) Configuration Modes

C. Stroud 8/06

FPGAs 39
Spartan 3 Daisy Chains

Table 1-4: Number of Bits in an Uncompressed FPGA Bitstream Image

<table>
<thead>
<tr>
<th>Spartan-3 Generation FPGA Family</th>
<th>FPGA Part Number</th>
<th>Number of Configuration Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartan-3A/3AN FPGA</td>
<td>XCS850A</td>
<td>437,912</td>
</tr>
<tr>
<td></td>
<td>XCS6200A</td>
<td>1,096,128</td>
</tr>
<tr>
<td></td>
<td>XCS8400A</td>
<td>1,856,250</td>
</tr>
<tr>
<td></td>
<td>XCS6700A</td>
<td>2,782,410</td>
</tr>
<tr>
<td></td>
<td>XCS8100A</td>
<td>3,755,256</td>
</tr>
<tr>
<td>Spartan 3LE FPGA</td>
<td>XCS8100E</td>
<td>391,344</td>
</tr>
<tr>
<td></td>
<td>XCS8250E</td>
<td>1,353,728</td>
</tr>
<tr>
<td></td>
<td>XCS8500E</td>
<td>2,270,208</td>
</tr>
<tr>
<td></td>
<td>XCS81200E</td>
<td>3,911,184</td>
</tr>
<tr>
<td></td>
<td>XCS81000E</td>
<td>3,909,596</td>
</tr>
<tr>
<td>Spartan-3 FPGA</td>
<td>XCS850</td>
<td>439,264</td>
</tr>
<tr>
<td></td>
<td>XCS3200</td>
<td>1,047,416</td>
</tr>
<tr>
<td></td>
<td>XCS8400</td>
<td>1,699,156</td>
</tr>
<tr>
<td></td>
<td>XCS82000</td>
<td>3,223,488</td>
</tr>
<tr>
<td></td>
<td>XCS81500</td>
<td>5,214,784</td>
</tr>
<tr>
<td></td>
<td>XCS62000</td>
<td>7,673,624</td>
</tr>
<tr>
<td></td>
<td>XCS84000</td>
<td>11,316,284</td>
</tr>
<tr>
<td></td>
<td>XCS85000</td>
<td>13,271,096</td>
</tr>
</tbody>
</table>

Figure 1-3: Spartan-3 Generation Configuration Daisy-Chain Options

C. Stroud 8/06

FPGAs

40
Configuration Techniques

- **Full configuration & readback**
  - Simple configuration interface
  - Internal automatic calculation of frame address
  - Long download time for large FPGAs

- **Partial reconfiguration & readback**
  - Only change portions of configuration memory with respect to reference design
    - Reduces download time for reconfiguration
  - Requires more complicated interface
    - Command Register (CMR)
    - Frame Length Register (FLR)
    - Frame Address Register (FAR)
    - Frame Data Register
      - Input (FDRI) – for download
      - Output (FDRO) – for readback (*note separate access*)
Configuration Techniques

- Compressed configuration
  - Requires multiple frame write capability
    - Write identical frames of config data to multiple frame addresses
  - Extension of partial reconfiguration interface capabilities
    - Frame address is much smaller than frame of configuration data
  - Reduces download time for initial configuration depending on
    - Regularity of system function design
    - % utilization of array
      - Unused portions written with default configuration data
Full Configuration Example

- Dummy Word 0xFFFFF
- Synchronize Word 0xAA995566
- CMD Write 0x30008001
  - Reset CRC 0x00000007
- FLR Write 0x30016001
  - FLR = 0x00000024
  - Frame length = 37 words
    - 1,184 bits ÷ 32 bits/word
- COR Write 0x30012001
  - COR Write 0x00003FE5
- IDCODE Write 0x3001C001
  - Device ID = 0x0140D093 (3S50)
- MASK Write 0x3000C001
  - MASK = 0x00000000
- CMD Write 0x30008001
  - Switch CCLK 0x00000009
- FAR Write 0x30002001
  - FAR = 0x00000000 (full config)
- CMD Write 0x30008001
  - Write CFG 0x00000001
- FDRI Write 0x30004000
  - # words to write 0x50003555

Xilinx ASCII Bitstream
Created by Bitstream I.32
Design name: s3mod7.ncd
Architecture: spartan3
Part: 3s50tq144
Date: Tue Sep 04 15:50:09 2007
Bits: 439264

`start of actual configuration data`
Partial Reconfiguration Example

- Dummy Word 0xFFFFFFFF
- Synchronization Word 0xAA995566
- CMD Write 0x30008001
  - Reset CRC 0x00000007
- IDCODE Write 0x3001C001
  - Device ID = 0x0140D093 (3S50)
- COR Write 0x30012001
  - COR Write Packet Data 0x00003FE5
- CMD Write 0x30008001
  - Shutdown 0x0000000B
- CRC Write 0x30000001
  - CRC = 0x00002CE9
- CMD Write 0x30008001
  - AGhigh 0x00000008
- CMD Write 0x30008001
  - WCFG 0x00000001
- FAR Write 0x30002001
  - FAR = 0x00080000 (partial config)
- Part Reconfig Reg Write 0x3001E001
  - Null 0x00000000
- FDRI Write 0x300042E4
  - #words to write 0x000002E4

... 4 NOOPs 0x20000000
01100000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
01100000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
... 16 NOOPs 0x20000000
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00000000000000000000000000000001
00110000000000000000000000000001
00110000000000000000000000000001
... 26656 bits

start of actual configuration data