Evolvable Systems for Space Applications

Jason D. Lohn

Evolvable Systems Group
Computational Sciences Division
NASA Ames Research Center
Mountain View, CA USA

Stanford, CS426/BMI226
November 24, 2003

Outline

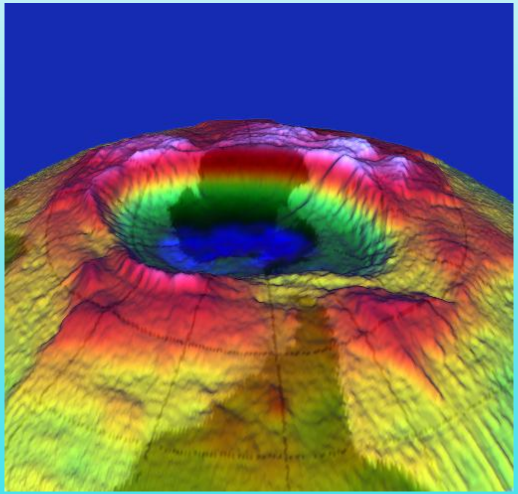
- NASA Interest in Evolvable Systems
- Example Applications
  - Self-repairing FPGAs
  - Circuit Design
  - Antenna Design
- Future & Conclusion

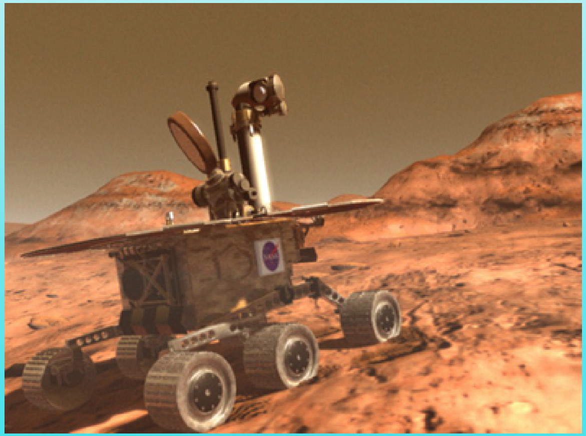
The NASA Mission

To understand and protect our home planet

To explore the Universe and search for life

To inspire the next generation of explorers



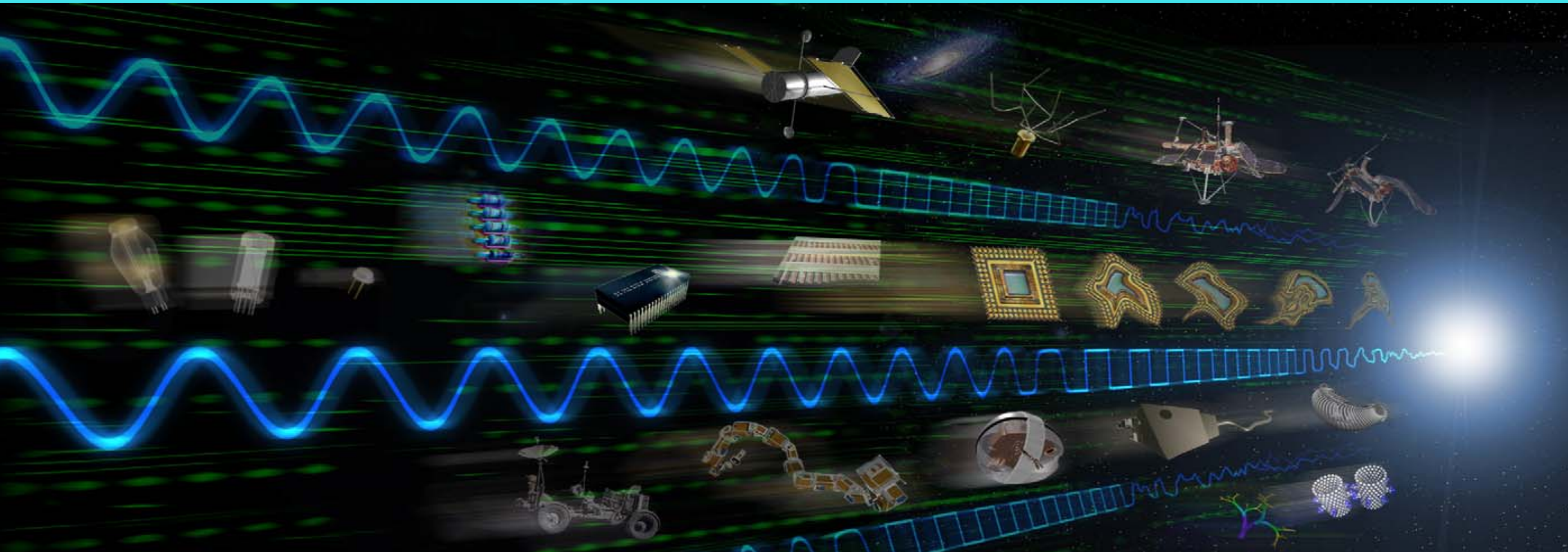


What Are Evolvable Systems?

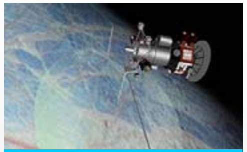
A system that changes over time to become better…

Changing its shape, its function, being automatically designed…

All under the guidance of a process derived from biological evolution.



Examples of Evolvable Systems



Automatic Fault Recovery

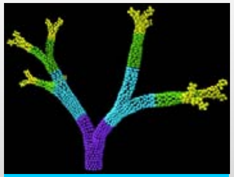
Electronics Survive Extreme Radiation, Temperature

Planetary Exploration

Evolved Controller Circuits Enable Mobile Drilling

Distributed Robotic Control

Coordination Among Modular Robotic Structures



Automated Design with New Devices

Defect Tolerant Nanosystems

Evolutionary and Adaptive Algorithms

NASA Interest in Evolvable Systems

Abundance of:

- Optimization problems
- Design problems
- Applications requiring automation, fault recovery, control

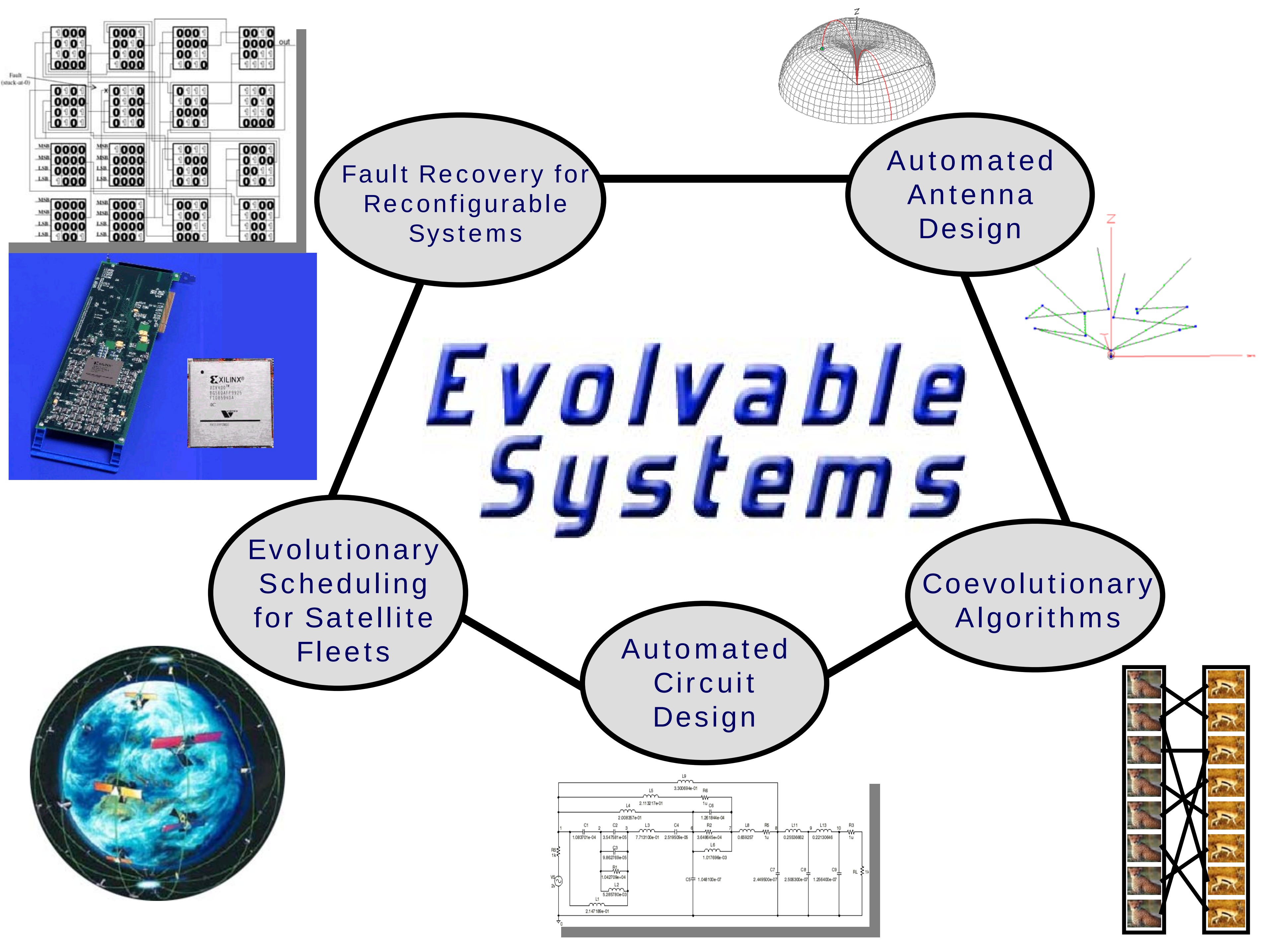
Agency-wide:

- Many projects using EAs
- Ames & JPL each have groups dedicated to evolvable systems

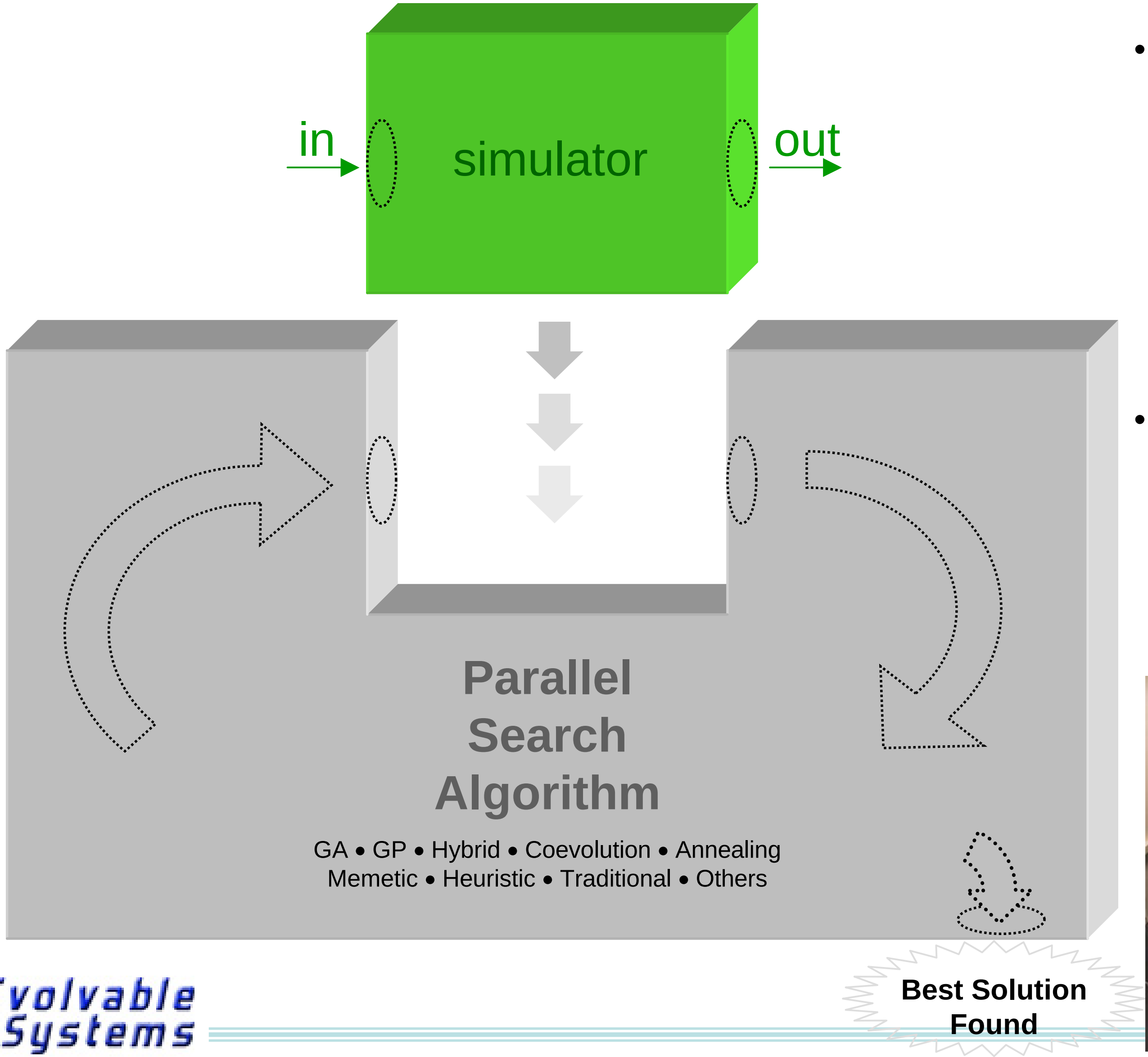
Human-Competitive Performance

- Evolutionary Algorithms are beginning to design complex engineering structures at the level of human experts

- John Koza's list of human-competitive performance (many infringe on patents)


Evolvable Systems
Fault Recovery for Reconfigurable Systems
Automated Antenna Design
Coevolutionary Algorithms
Automated Circuit Design
Evolutionary Scheduling for Satellite Fleets

Evolutionary Design Engine



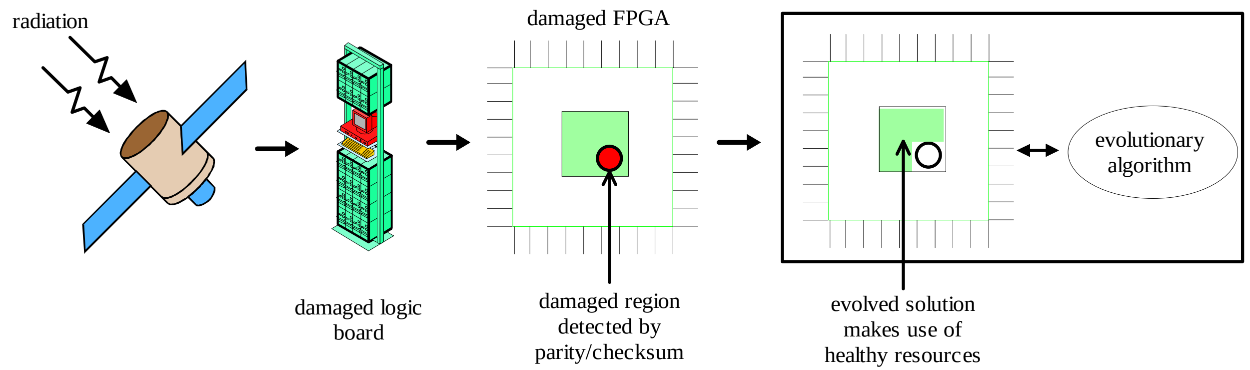
- Evolutionary Design Engine:
  - C/C++
  - Easy to "drop-in" new application areas
  - Parallel, any Unix
  - Integrated native code for NEC4 antenna simulator
  - Over 10,000 lines of code
- Beowulf cluster:
  - simulation farmed out to diskless nodes (currently 120 cpus)
  - low bandwidth communication

Fault Recovery For Reconfigurable Systems

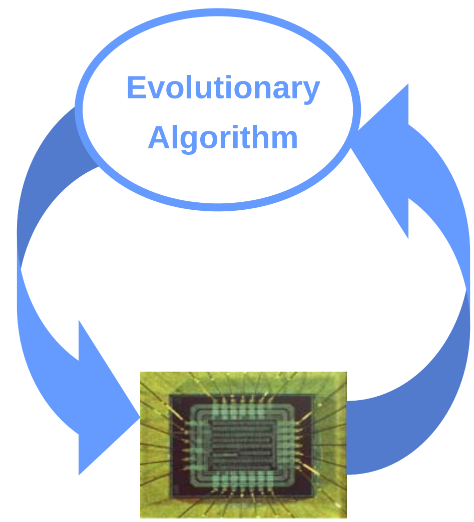
(1 Digital,1 Analog)

Fault Recovery

Dynamic Evolution for Fault Tolerance



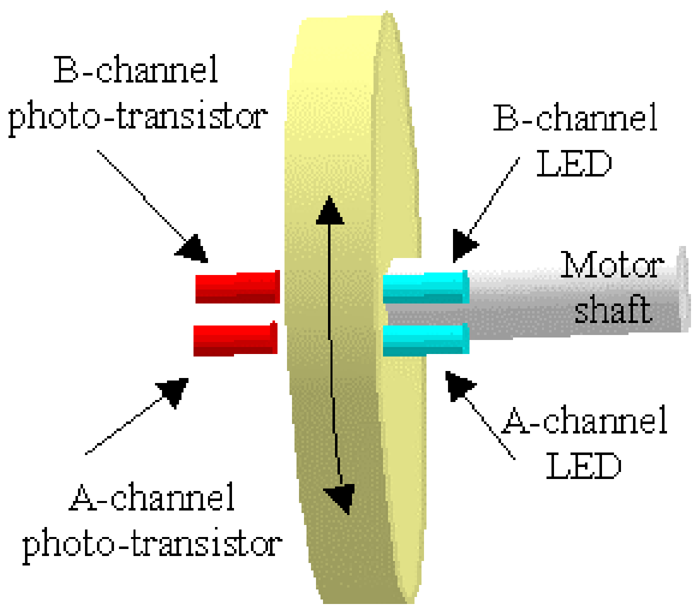
- Description:
  - Fault tolerance / self-repair in extreme environments
    - High temperature
    - High radiation
- Output: adaptive algorithms for autonomous self-repair of re-programmable logic chips

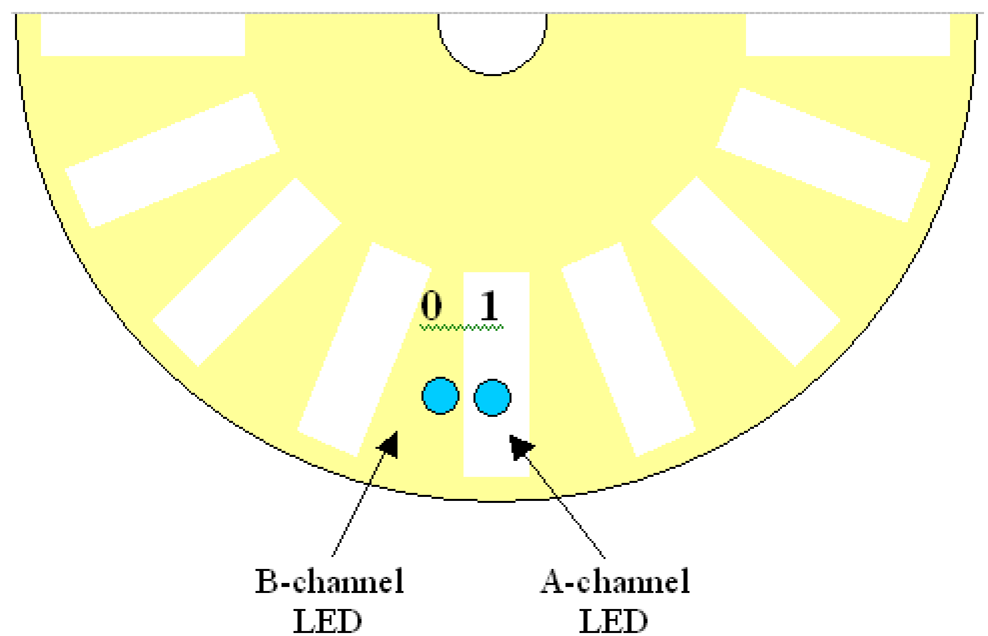


chip reprogrammed by algorithm

Quadrature Decoder

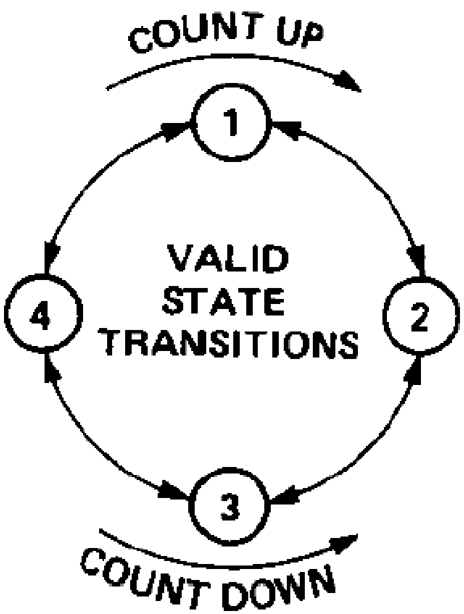
- Applications requiring determination of angular translation (or speed)
- Example: DC-motor to drive system for a mobile robot we may wish to move forward (or reverse) by a fixed distance
- Decoder determines rotation direction



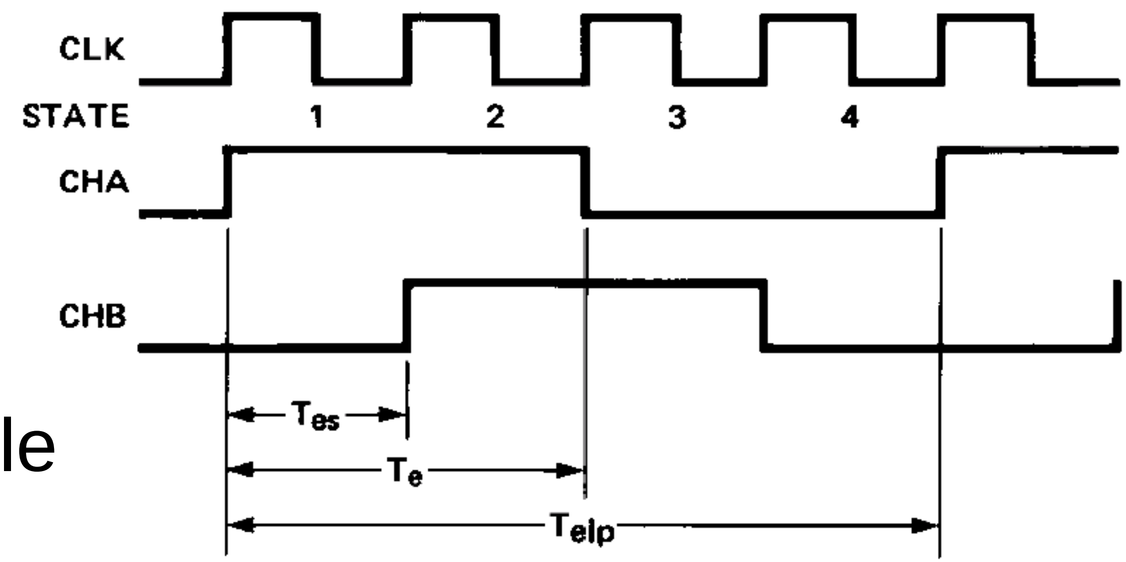


Quadrature Decoder

- Finite state machine



- Input and State traces

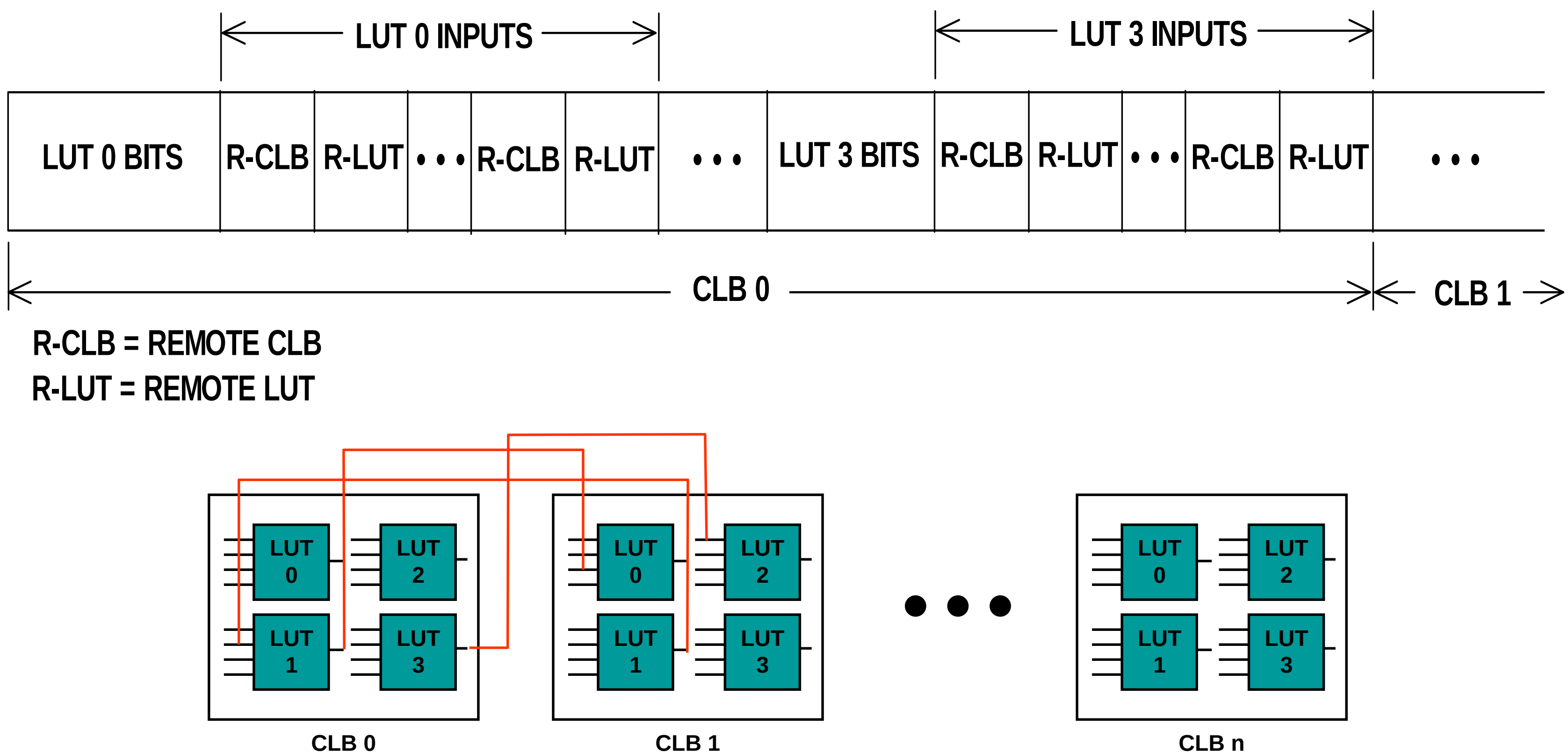


- State transition table

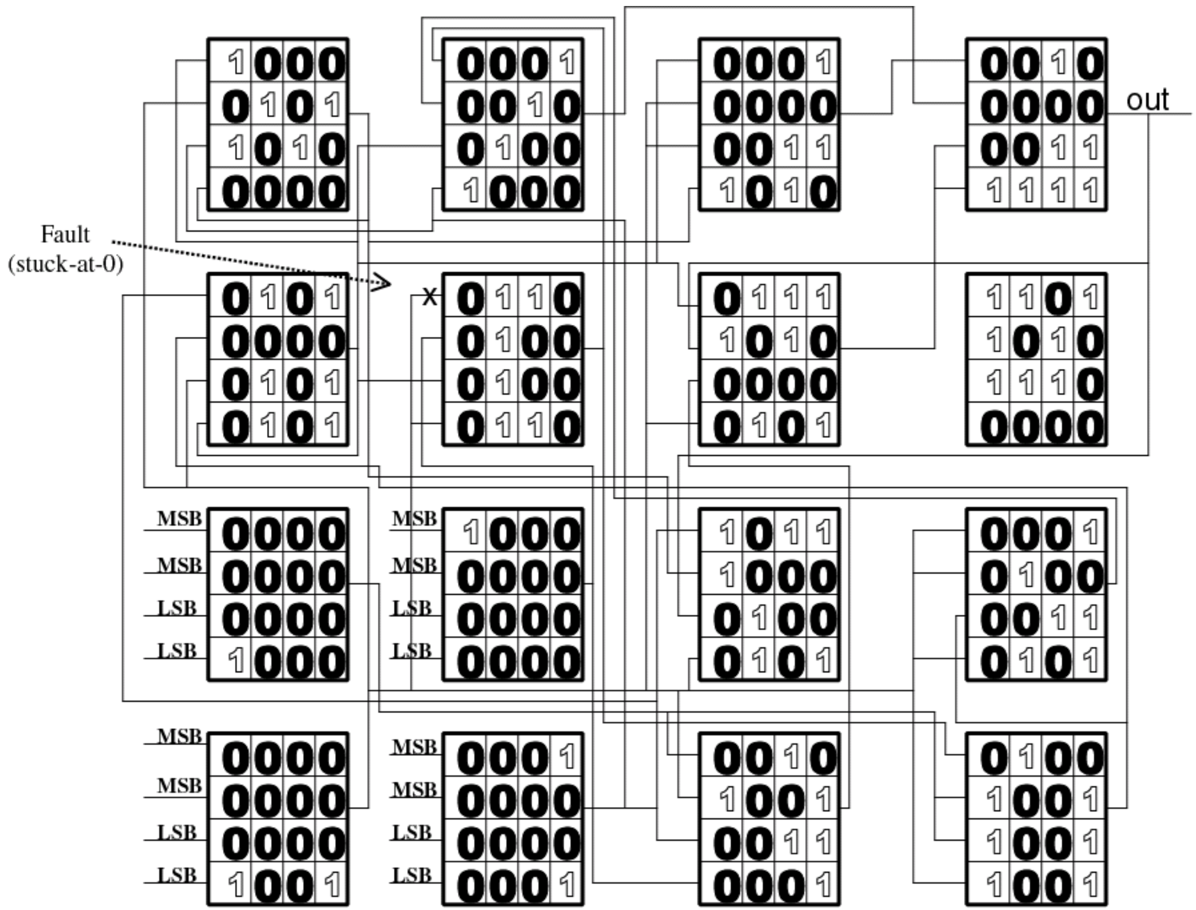
CHA	CHB	STATE
1	0	1
1	1	2
0	1	3
0	0	4

Genetic Representation

- Logic bits in the LUTs
- Routing bits specify how to connect LUT outputs to LUT inputs



Results



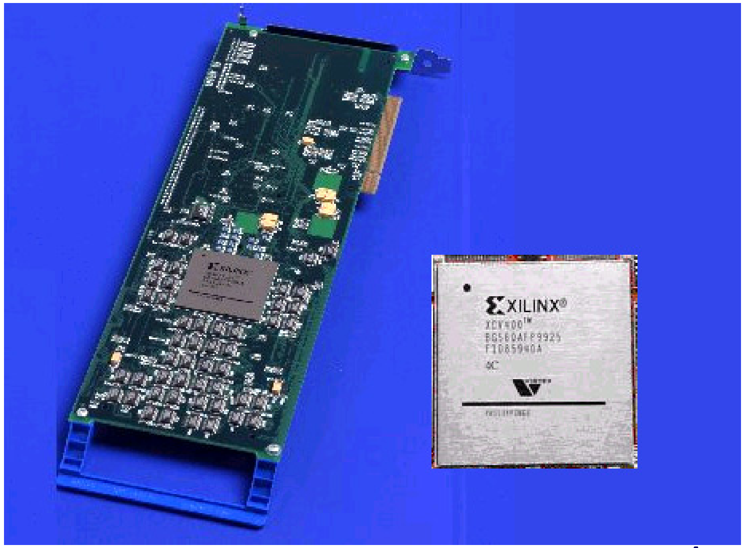
Evolved Quad Decoder Configuration

Hardware Evolution

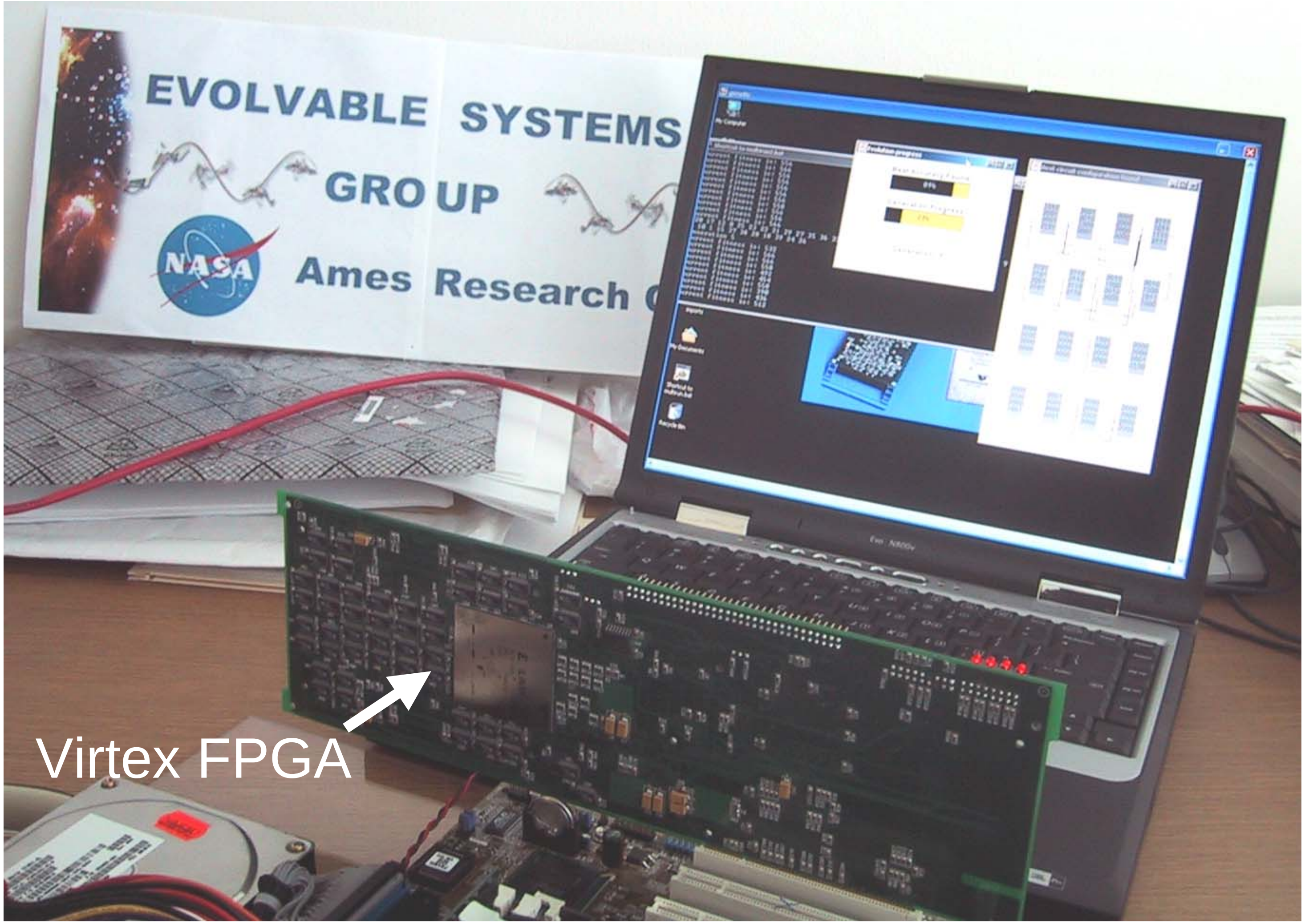
Hardware Evolution

- 20 random latchup faults are simulated
- Quad decoder is repaired automatically in minutes
- FPGA chip is reprogrammed about 5 times per second – we will soon go much faster

Celoxica RC1000 board
Xilinx Virtex 1000e

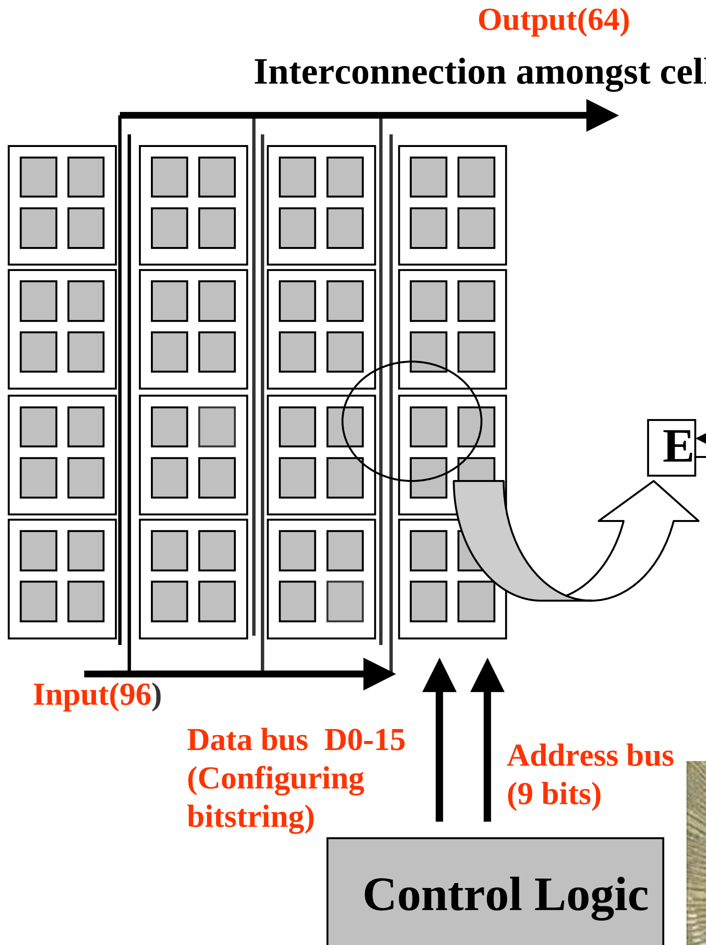


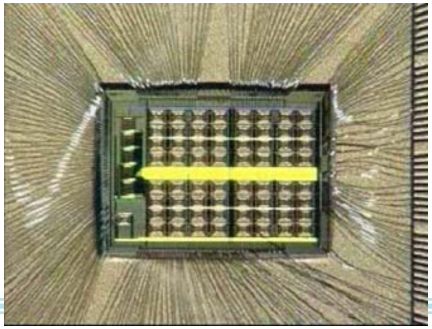
Hardware Setup



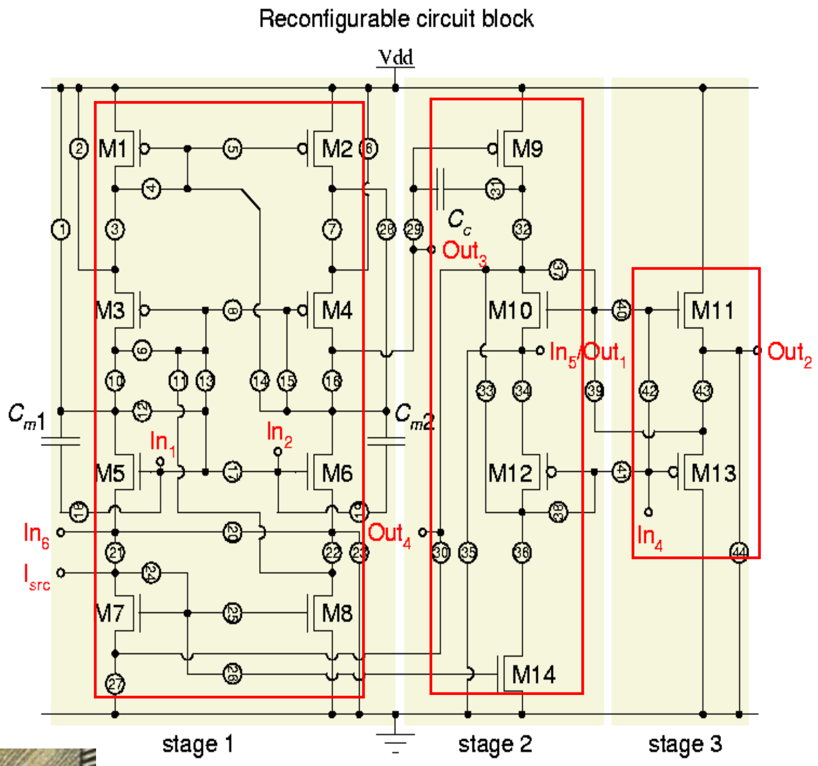
Programmable Transistor Array Cell – FPTA2

- TSMC 0.18micron 1.8v; Area 5mmX 7mm; 64 reconfigurable cells; 96 analog in, 64 analog out
- 256 pins; 16bits data bus/9bits address bus; 5000 programming bits
- Each cell OpAmp level



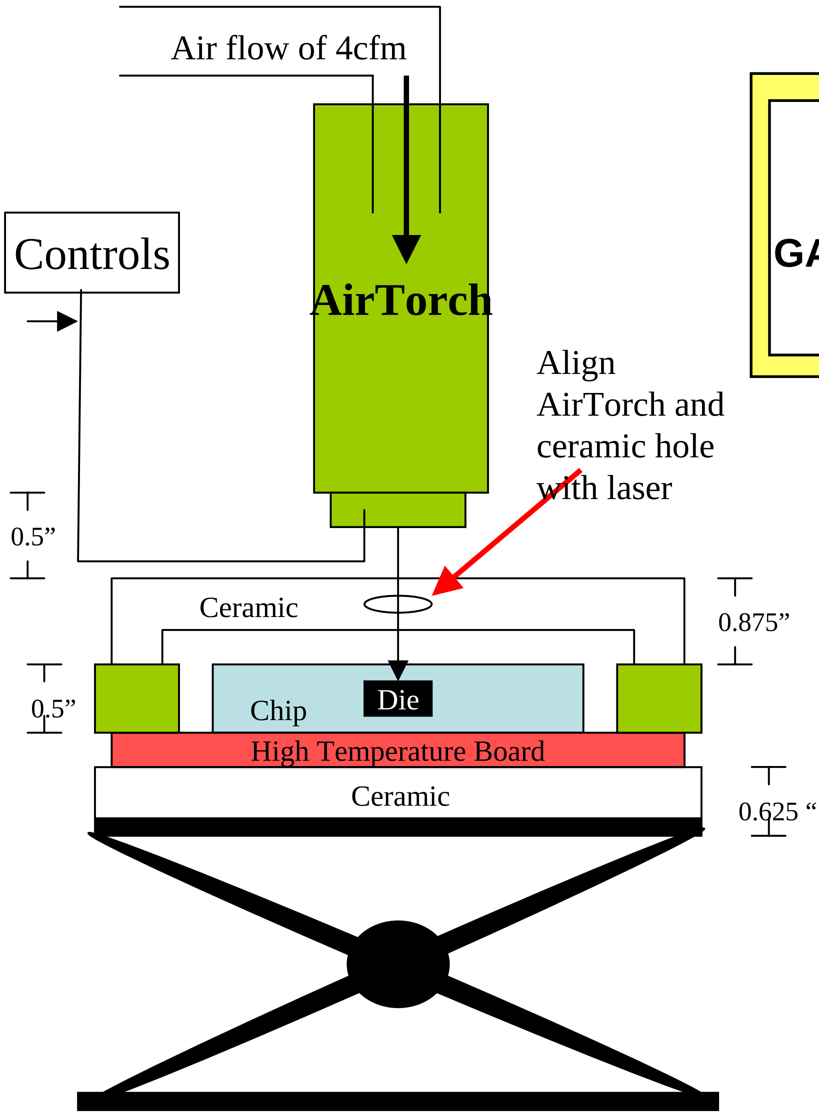


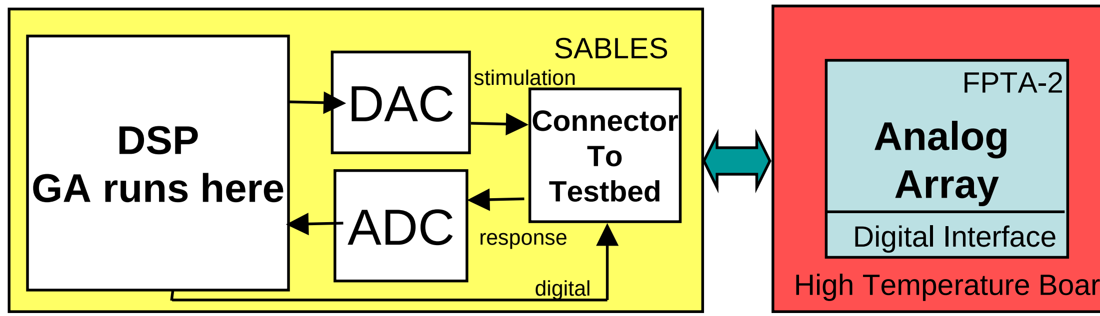
Chip Architecture

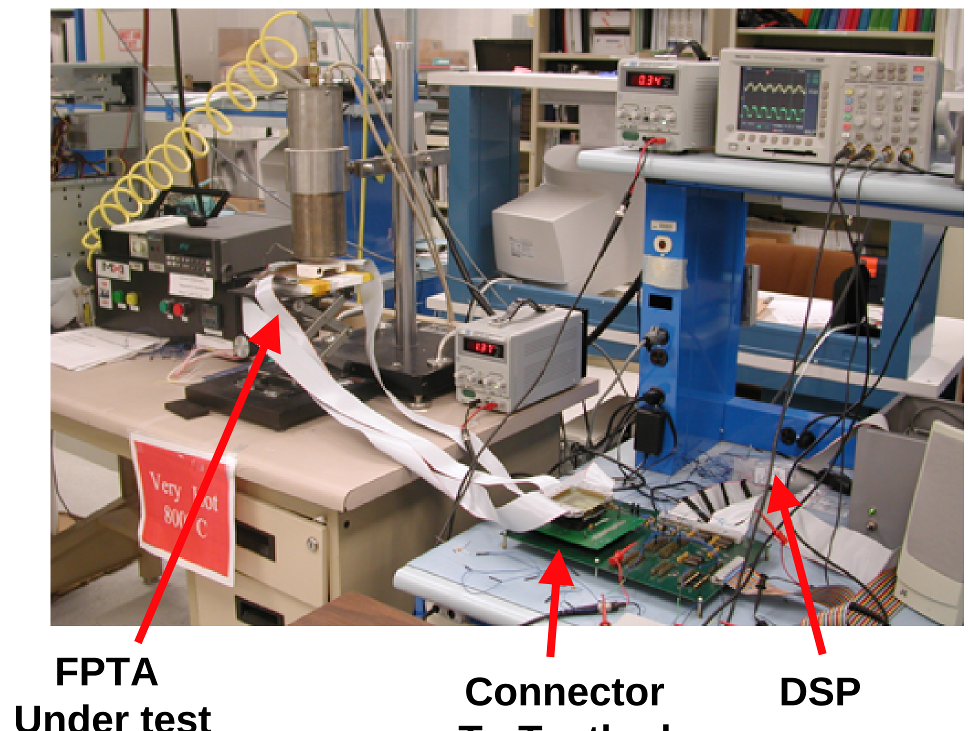


Cell Schematic

Extreme Environment integrating FPTA-2 and SABLES

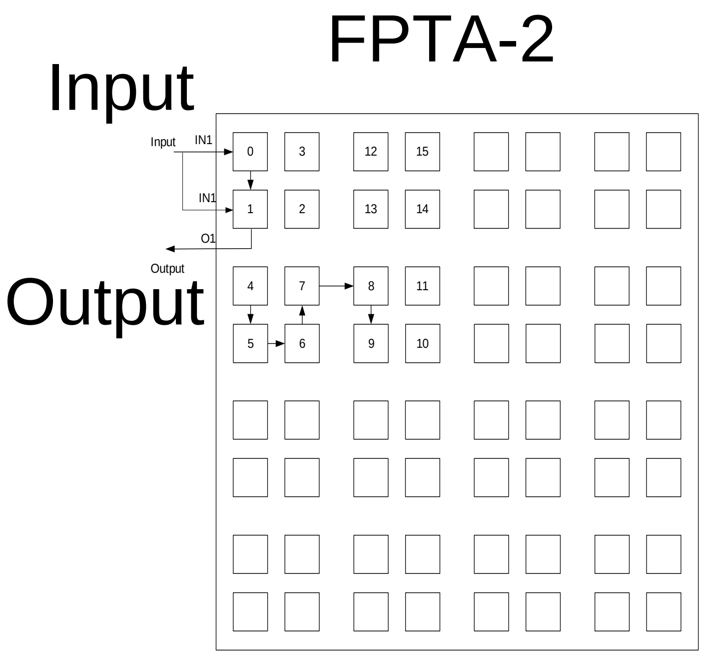




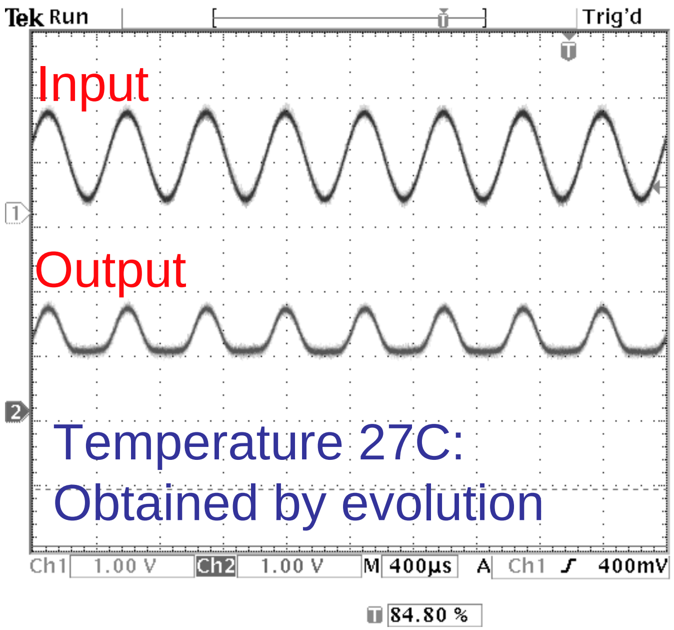


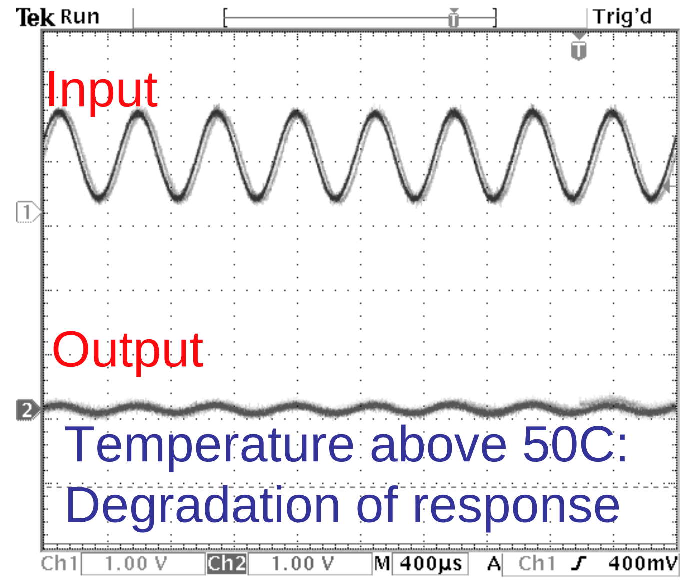
Recovery of Half-wave rectifier at 280C

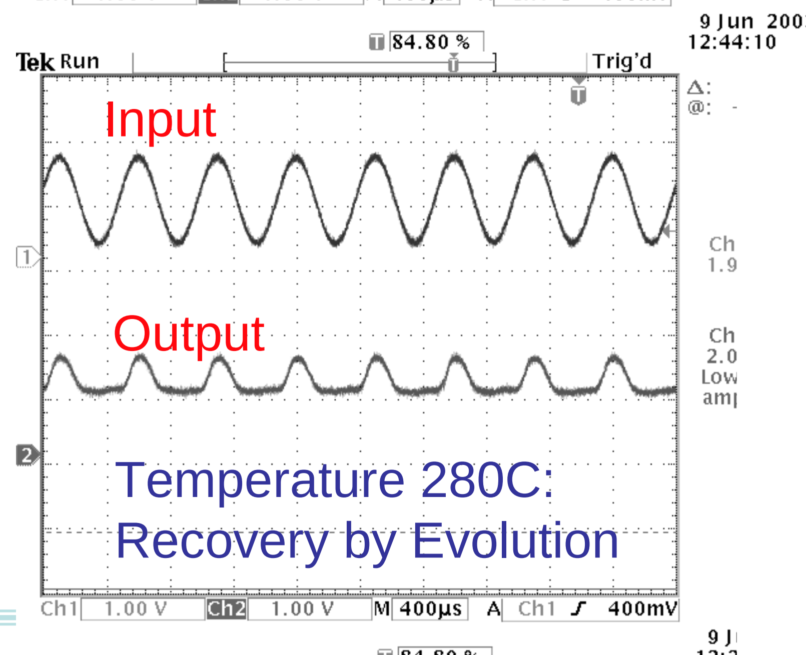
- Reconfigurable Cell (FPTA-2)
  - 2 cells: 0,1
  - Input: 1 Analog Input (0.9 Volt amplitude – 2 kHz)
- Evolution Mechanism:
  - Goal: Synthesize Half-way rectifier at 280ºC.
  - Constrained evolution to Cell0 and Cell1
  - Fitness Function: Absolute error to the target;
- Genetic Algorithm
  - 400 individuals
  - Chromosome: 160 bits
  - less than 50 generations – 1 min



Recovery of Halfwave Rectifier at 280C







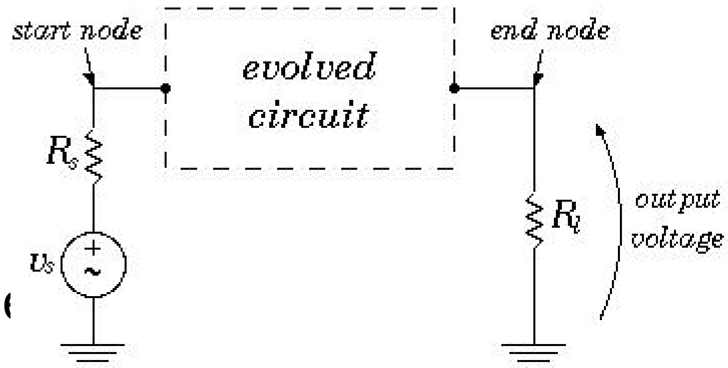
Recovery by Evolution of Halfway rectifier:

- 2 Cells of FPTA2 used by evolution
- Input Freq.: 2 kHz
- Input Amplitude: 0.9 Volt

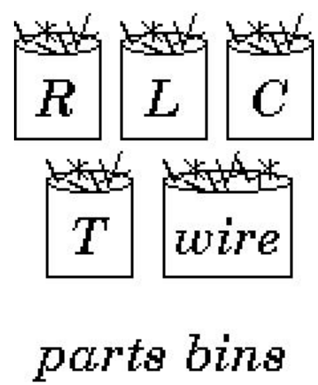
Automated Circuit Design

Circuit Construction

- Template circuit



- Set of selectable



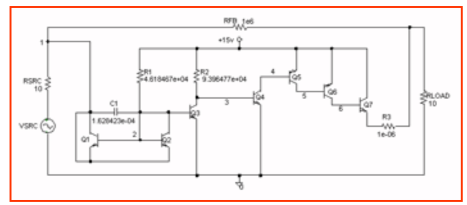
Circuit Representation

- a small circuit construction language is used
- instructions involve placing components end-to-end (laying a trail)
- examples:
  - place_resistor(144.92, new_node)
  - place_transistor(ground, new_node, previous_node)
  - place_capacitor(183e-6, ground)

From Representation to Fitness Calculation

genetic representation

circuit design



SPICE circuit simulator

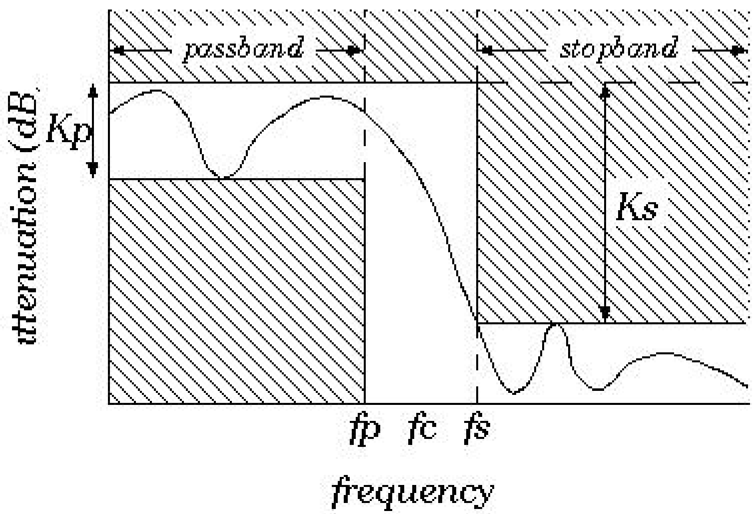
circuit-constructing program

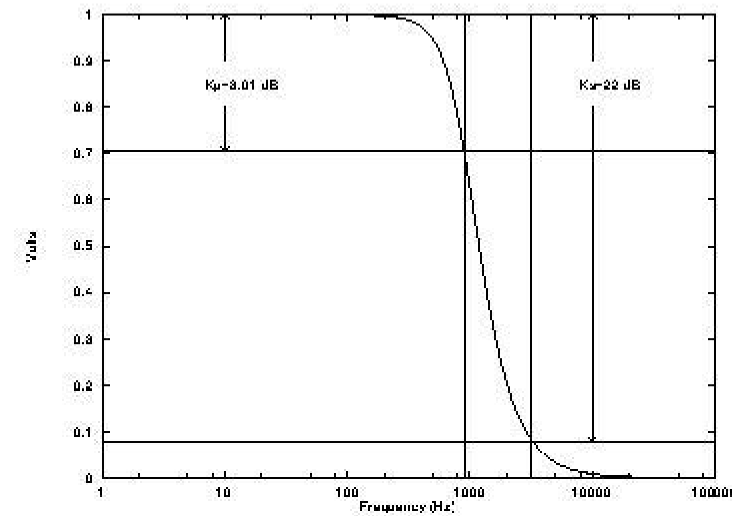
SPICE netlist

Fitness Calculation

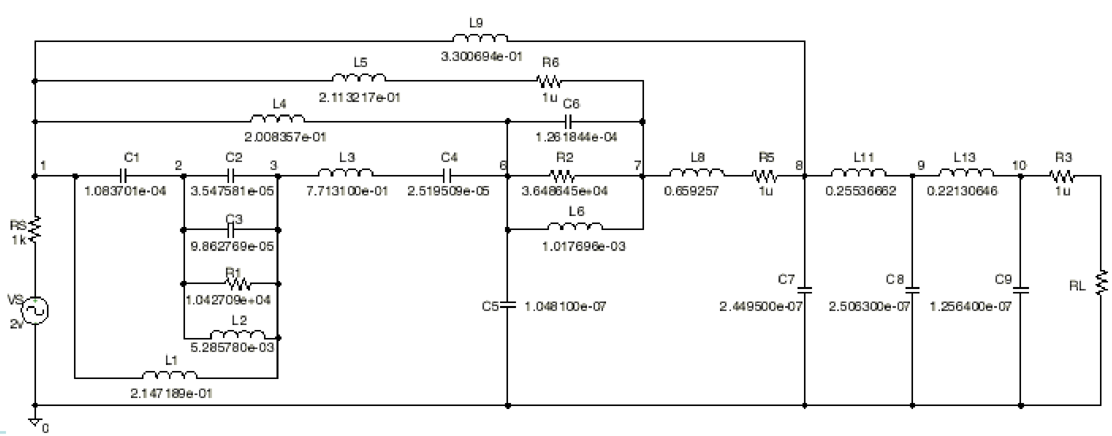
Circuit Design Application

- Design and optimization

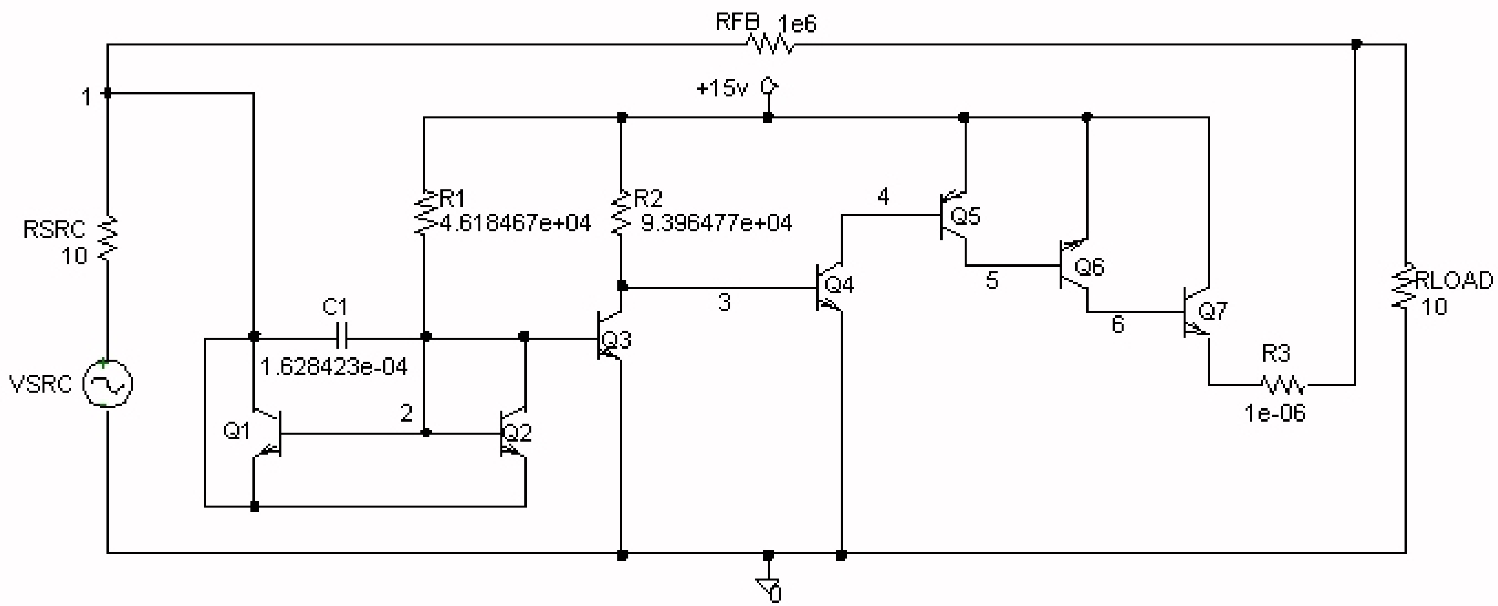




evolved circuit



Evolved 85 dB Amplifier



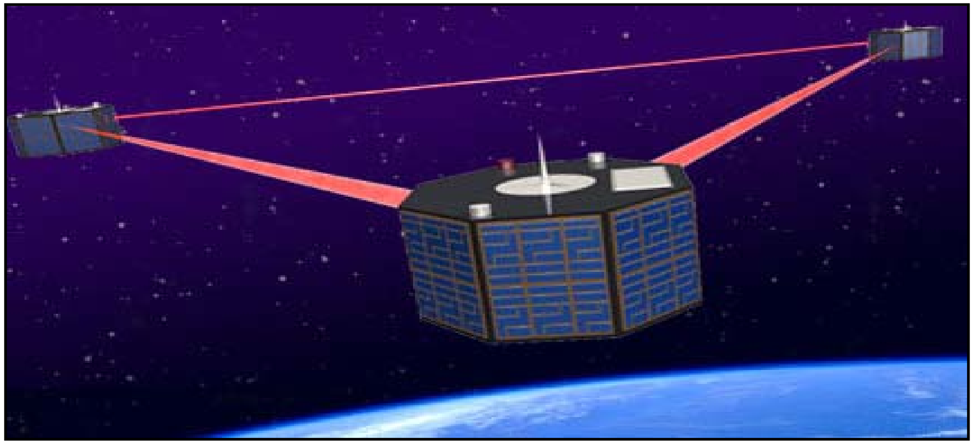
<u>**Evolved Circuit-Constructing Program**</u>

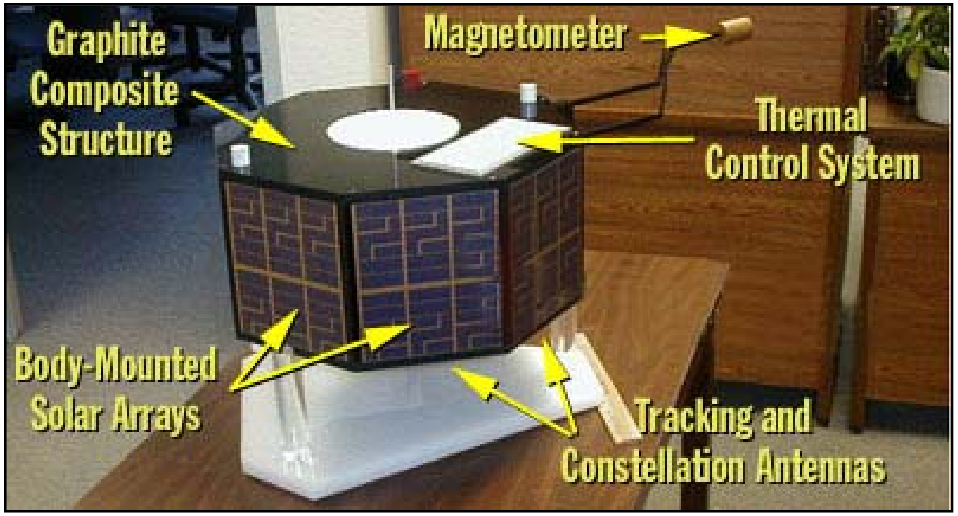
```
transistor(N, ACTIVE_NODE, NEW_NODE, INPUT_NODE);
transistor(N, BASE, ACTIVE_NODE, PREVIOUS_NODE);
resistor_cast_to_ps(4.618467e+04);
capacitor_cast_to_input(1.628423e-04);
transistor(N, NEW_NODE, ACTIVE_NODE, GROUND_NODE);
resistor_cast_to_ps(9.396477e+04);
transistor(N, NEW_NODE, ACTIVE_NODE, GROUND_NODE);
transistor(P, NEW_NODE, ACTIVE_NODE, PS_NODE);
transistor(N, NEW_NODE, ACTIVE_NODE, PS_NODE);
transistor(N, PS_NODE, ACTIVE_NODE, NEW_NODE);
resistor_move_to_output(1e-06);
```

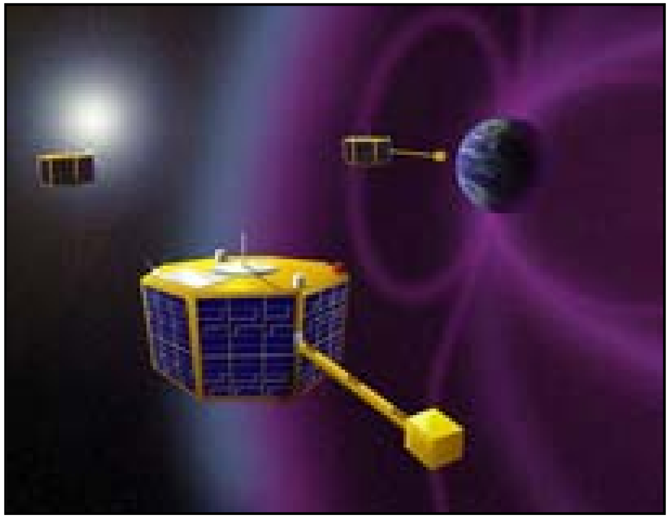
Automated Antenna Design and Optimization

ST5 Mission

- New Millennium Program mission
- Three nanosats
- Measure effect of solar activity on the Earth's magnetosphere







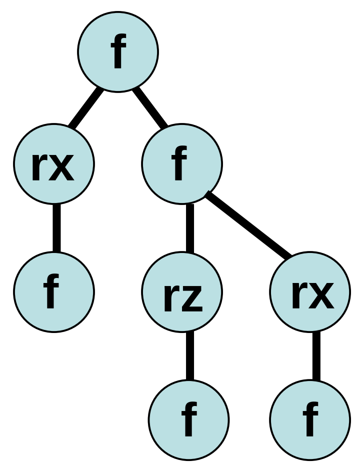
ST5 Antenna Requirements

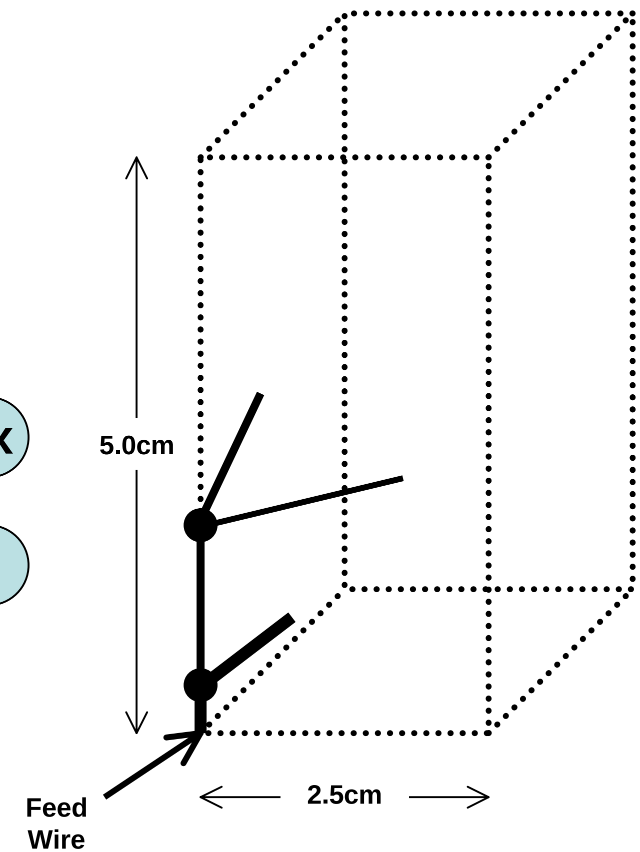
- Transmit Frequency: 8470 MHz
- Receive Frequency: 7209.125 MHz
- Antenna RF Input: 1.5W = 1.76 dBW = 31.76 dBm
- VSWR: < 1.2 : 1 at the antenna input port at Transmit Freq, < 1.5 : 1 at the antenna input port at Receive Freq
- Antenna Gain Pattern: Shall be 0 dBic or greater for angles 40 <= theta <=80; 0 <= phi <= 360
- Antenna pattern gain (this shall be obtained with the antenna mounted on the ST5 mock-up) shall be 0.0 dBic (relative to anisotropic circularly polarized reference) for angles 40 <= theta <=80; 0 <= phi <= 360, where theta and phi are the standard spherical coordinate angles as defined in the IEEE Standard Test Procedures for Antennas, with theta=0 to direction perpendicular to the spacecraft top deck. The antenna gain shall be measured in reference to a right hand circular polarized sense (TBR).
- Desired: 0 dBic for theta = 40, 2 dBic for theta = 80, 4 dBic for theta = 90, for 0 <= phi <= 360
- Antenna Input Impedance: 50 ohms at the antenna input port
- Magnetic dipole moment: < 60 mA-cm^2
- Grounding: Cable shields of all coaxial inputs and outputs shall be returned to RF ground at the transponder system chasis. The cases of all comm units will be electrically isolated from the mounting surface to prohibit current flow to the spacecraft baseplate.
- Antenna Size: diameter: < 15.24 cm (6 inches), height: < 15.24 cm (6 inches)
- Antenna Mass: < 165 g

ST5 Quadrifilar Helical Antenna

Antenna Genotype

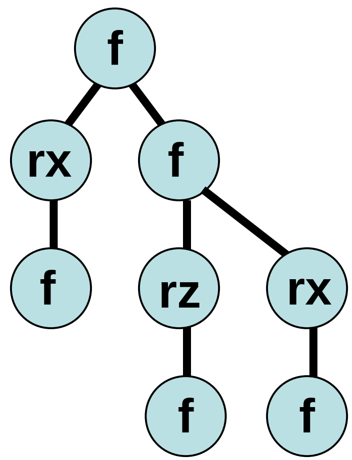
- Genotype specifies design of 1 arm in 3-space
- Genotype is tree-structured computer program that builds a wire form
- Commands:
  - forward(length radius)
  - rotate_x(angle)
  - rotate_y(angle)
  - rotate_z(angle)
- Branching in genotype → branching in wire form





Antenna Genotype

- Forward commands can have 0,1,2, or 3 children
- Rotate commands have exactly 1 child (always non-terminal)



EA Parameters

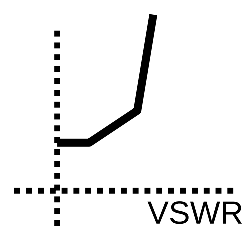
- Generational
- Rank-based selection
- Elitism of 2
- Population=200 (seeded w/individuals from previous run)
- Crossover: 1pt, 50%
- Mutation: 50%:
  - Mutate a constant,
  - Mutate a function (terminals remain terminals),

  or
  - Add/delete node (for adds: terminals remain terminals)
- Crossover, mutation mutually exclusive

Fitness Function

- Antenna designs are evaluated by NEC4 running on Linux Beowulf supercomputer
- 3 copies w/added noise are evaluated for each design
- Fitness function (to be minimized):

  F = VSWR_Score * Gain_Score * Penalty_Score
- VSWRs from both freqs are scaled and multiplied



- Gain is sampled at 5 degree increments between theta=40 and theta=90

  f = 0 if gain > 0.5 dB

  f = 0.5 – gain if gain < 0.5 dB
- Penalty: proportional to # gain samples less than 0.01 dB

Fitness Function: VSWR

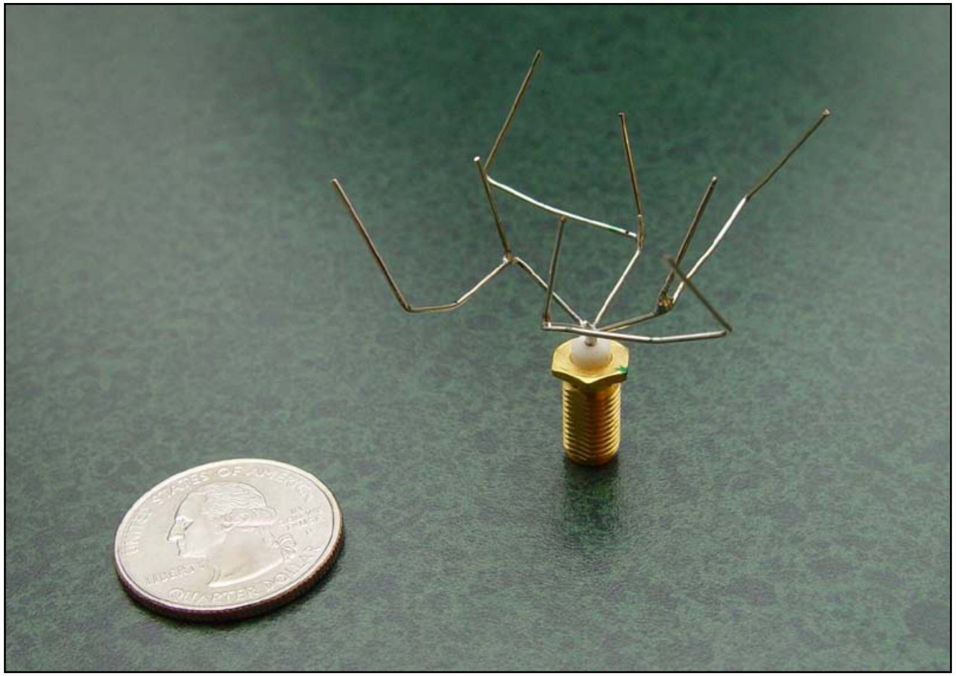
• voltage standing wave ratio: quantifies reflected-wave interference
• the ratio between the highest voltage and the lowest voltage in the signal envelope along a transmission line

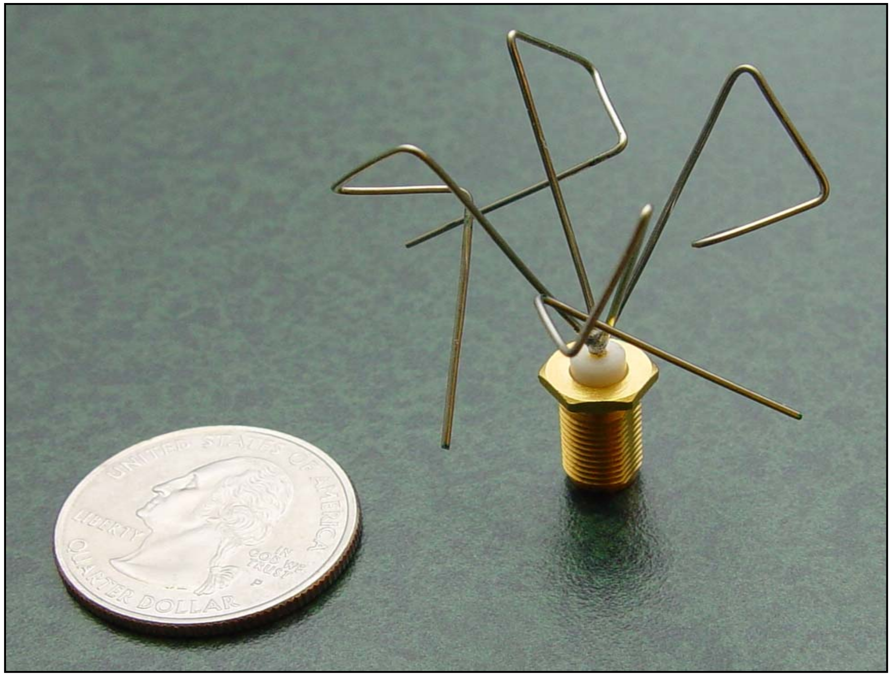
$$
\begin{aligned}
v_r &= \text{VSWR at receive frequency} \\
v_r' &= \begin{cases} v_r + 2.0(v_r - 1.25) & \text{if } v_r > 1.25 \\ v_r & \text{if } 1.25 > v_r > 1.1 \\ 1.1 & \text{if } v_r < 1.1 \end{cases} \\
v_t &= \text{VSWR at transmit frequency} \\
v_t' &= \begin{cases} v_t + 2.0(v_t - 1.15) & \text{if } v_t > 1.15 \\ v_t & \text{if } 1.15 > v_t > 1.1 \\ 1.1 & \text{if } v_t < 1.1 \end{cases} \\
vswr &= v_r' v_t'
\end{aligned}
$$

Fitness Function: Gain

$$
\begin{aligned}
gain_{ij} &= \text{gain at } \theta = 5^\circ i,\ \phi = 5^\circ j \\
gain(i,j) &= \begin{cases} 0 & \text{if } gain_{ij} > 0.5 \\ 0.5 - gain_{ij} & \text{if } gain_{ij} < 0.5 \end{cases} \\
gain &= 1 + 0.1 \sum_{i=8}^{i<19} \sum_{j=0}^{j=72} gain(i,j)
\end{aligned}
$$

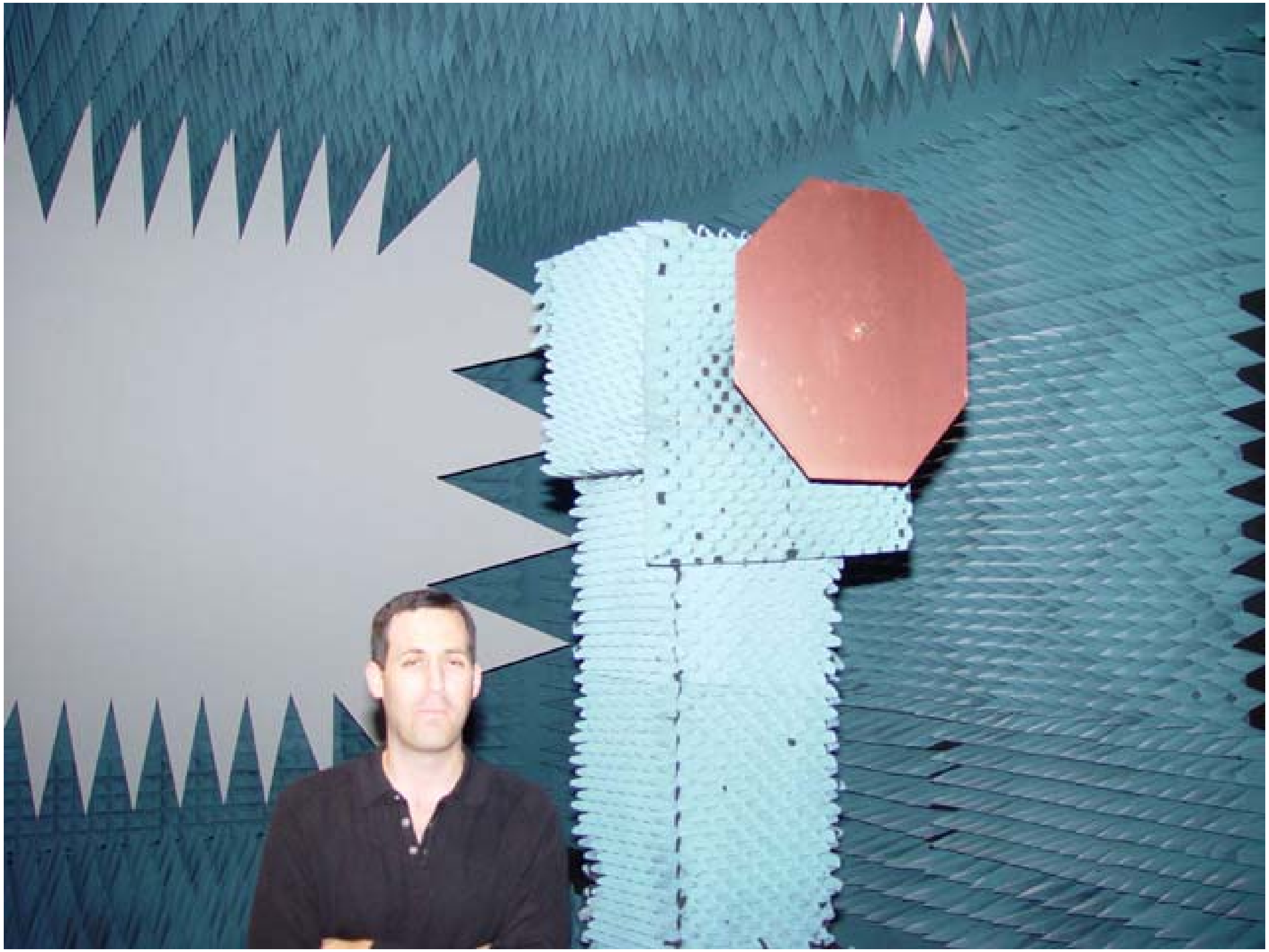
Evolved Antennas



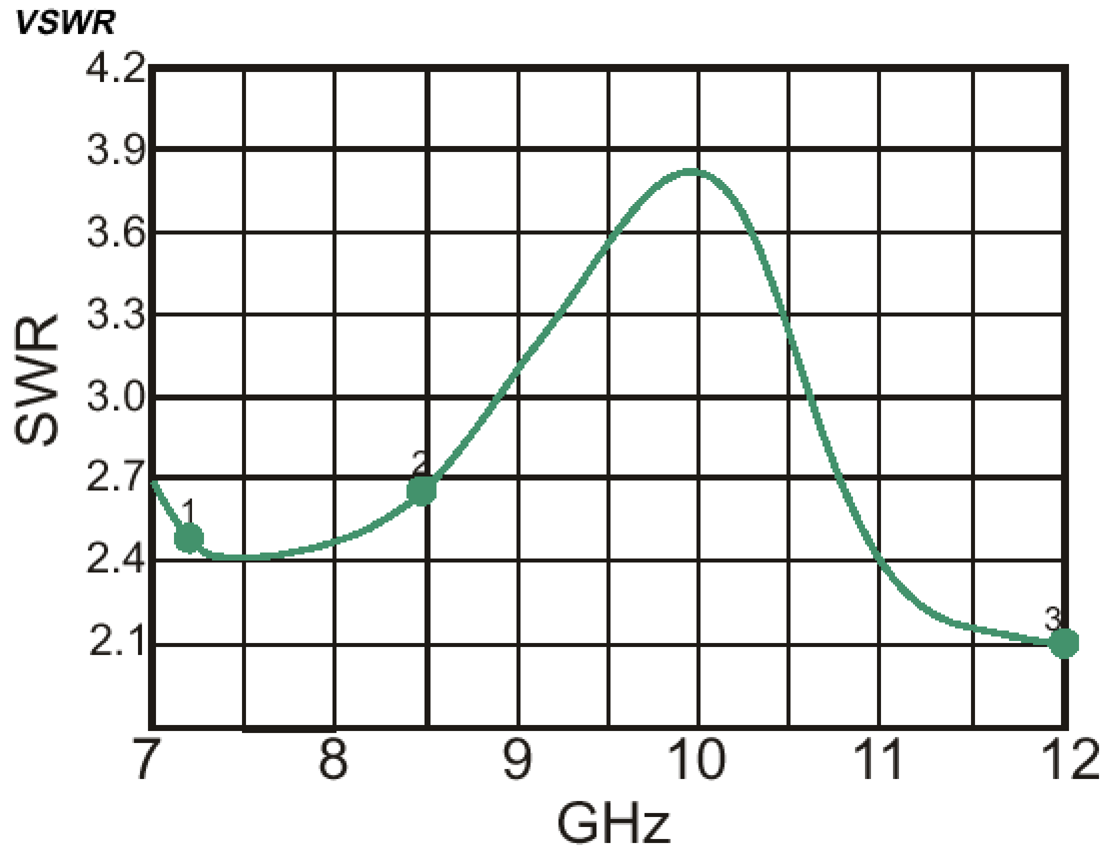


ST5-3-10 **ST5-4W-03**

Antenna for ST5 Mission

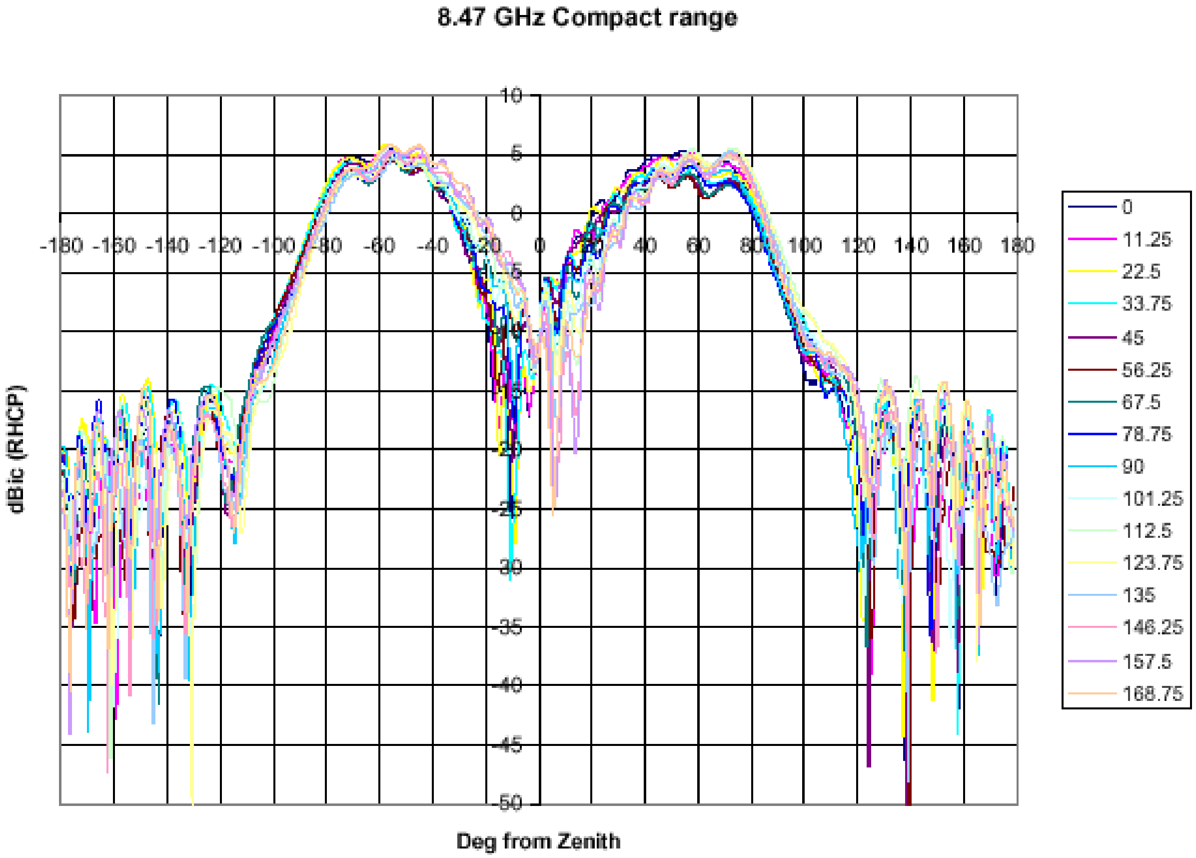


VSWR for Antenna ST5-3-10



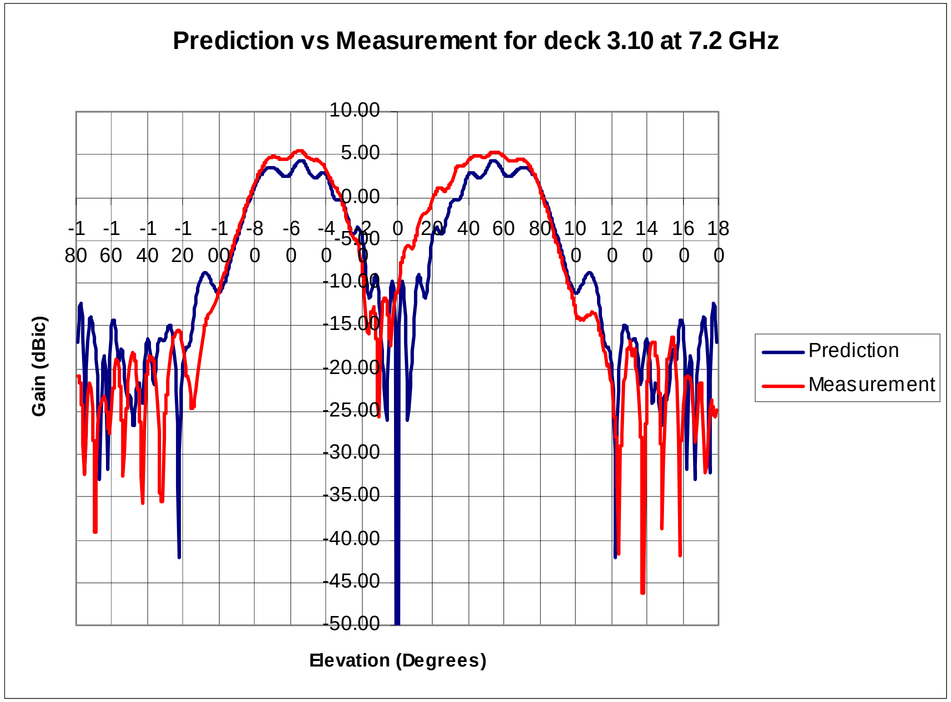
Data measured at GSFC on prototype ST5-3-10, Build LIR-2003-02-08

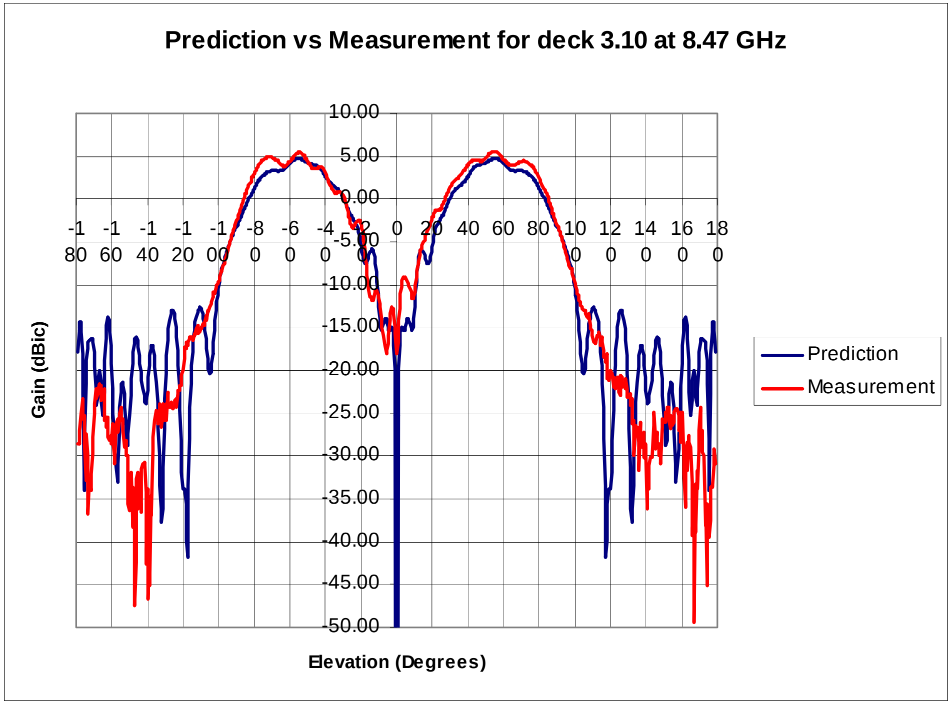
Gain for Antenna ST5-3-10



Data measured at GSFC on prototype ST5-3-10, Build LIR-2003-02-08

Prediction vs Measurement for Antenna ST5-3-10

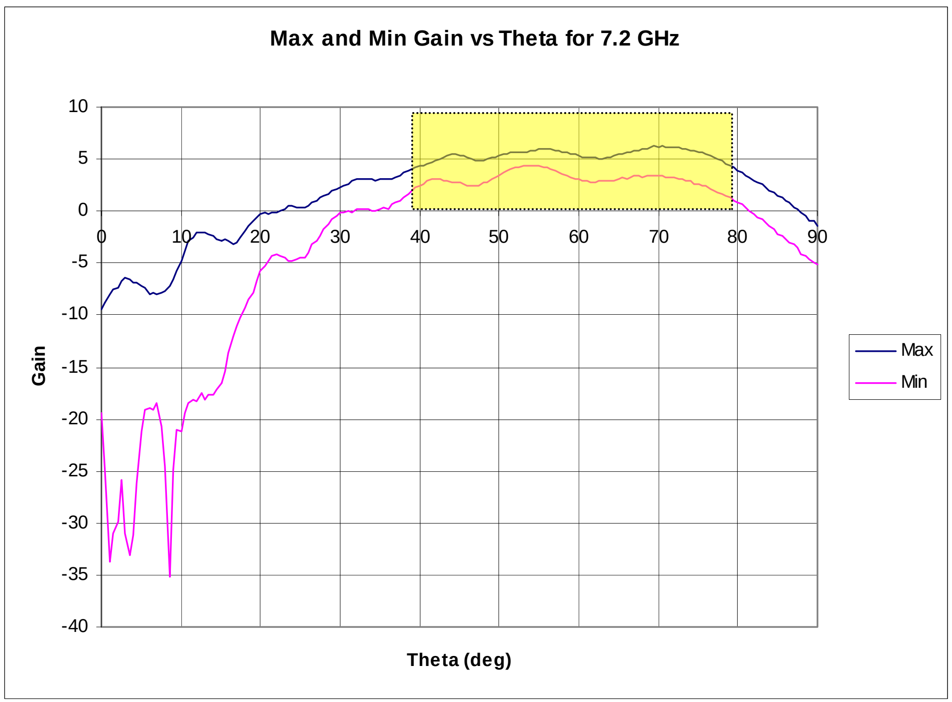




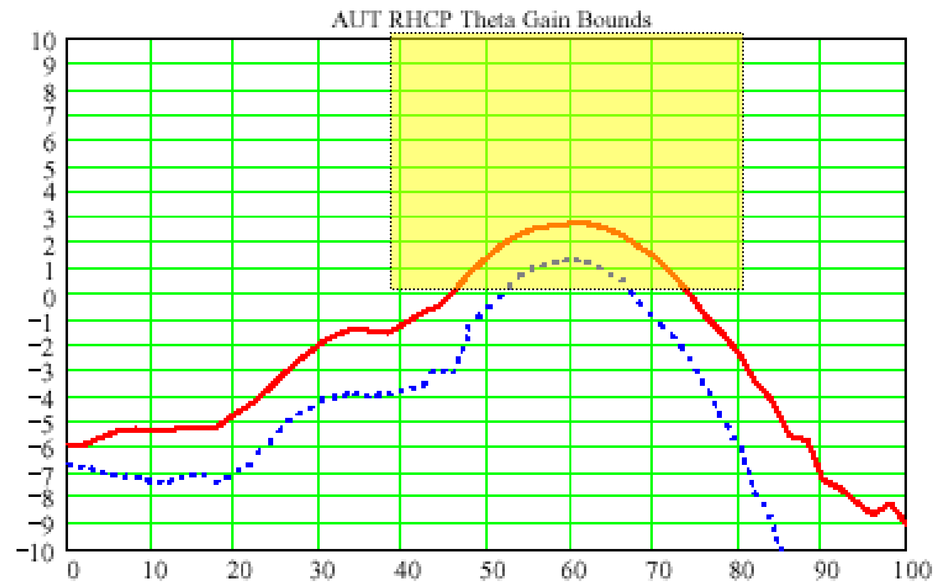
Phi = 0 degrees
NEC4 Simulator
Data measured at GSFC on prototype ST5-3-10, Build LIR-2003-02-08

Comparison: Pattern Analysis for Antennas @ 7.2 GHz

Evolved Antenna



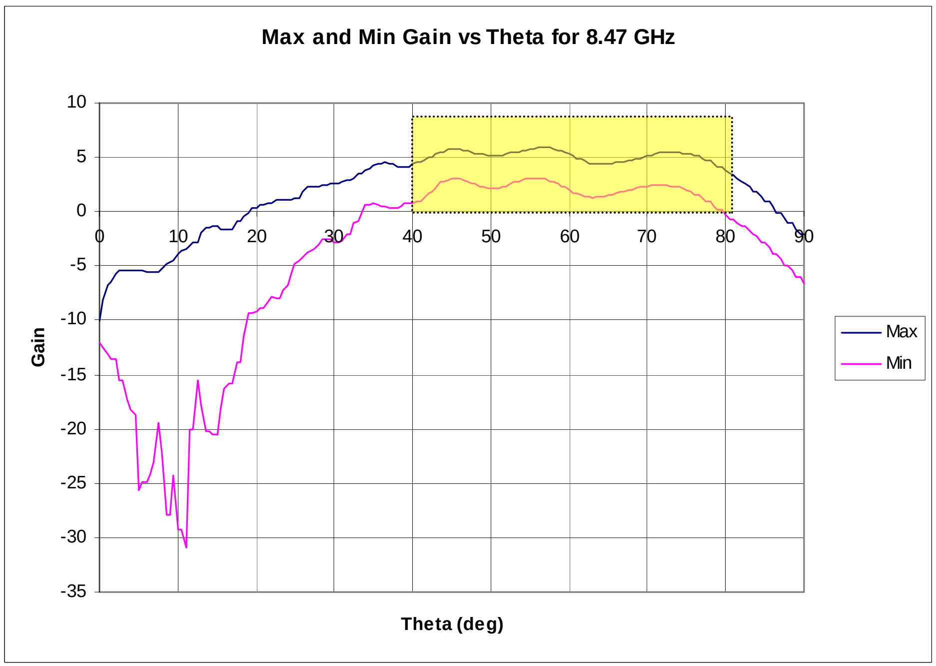
Conventional Antenna



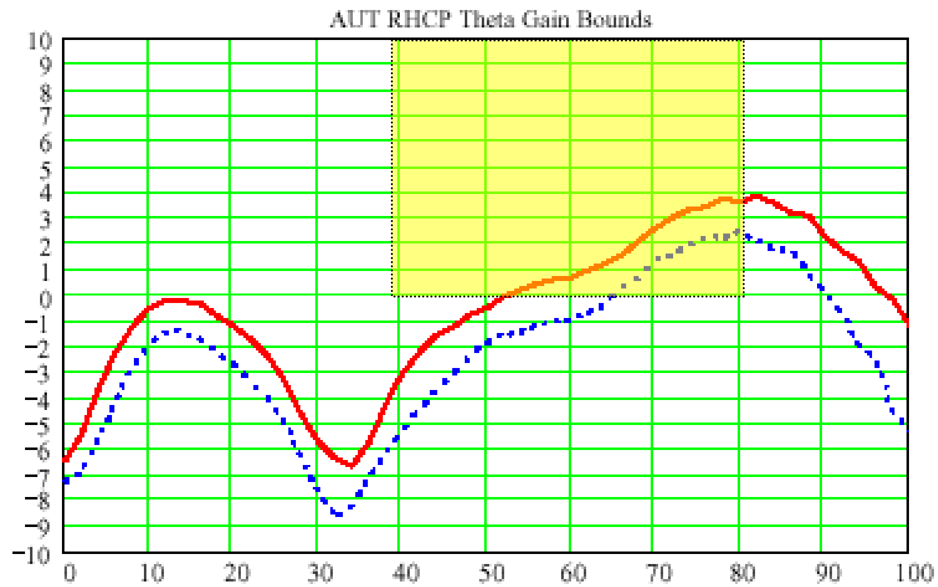
Shaded Yellow Box Denotes Area In-Spec, According to Original Mission Requirements

Comparison: Pattern Analysis for Antennas @ 8.47 GHz

Evolved Antenna

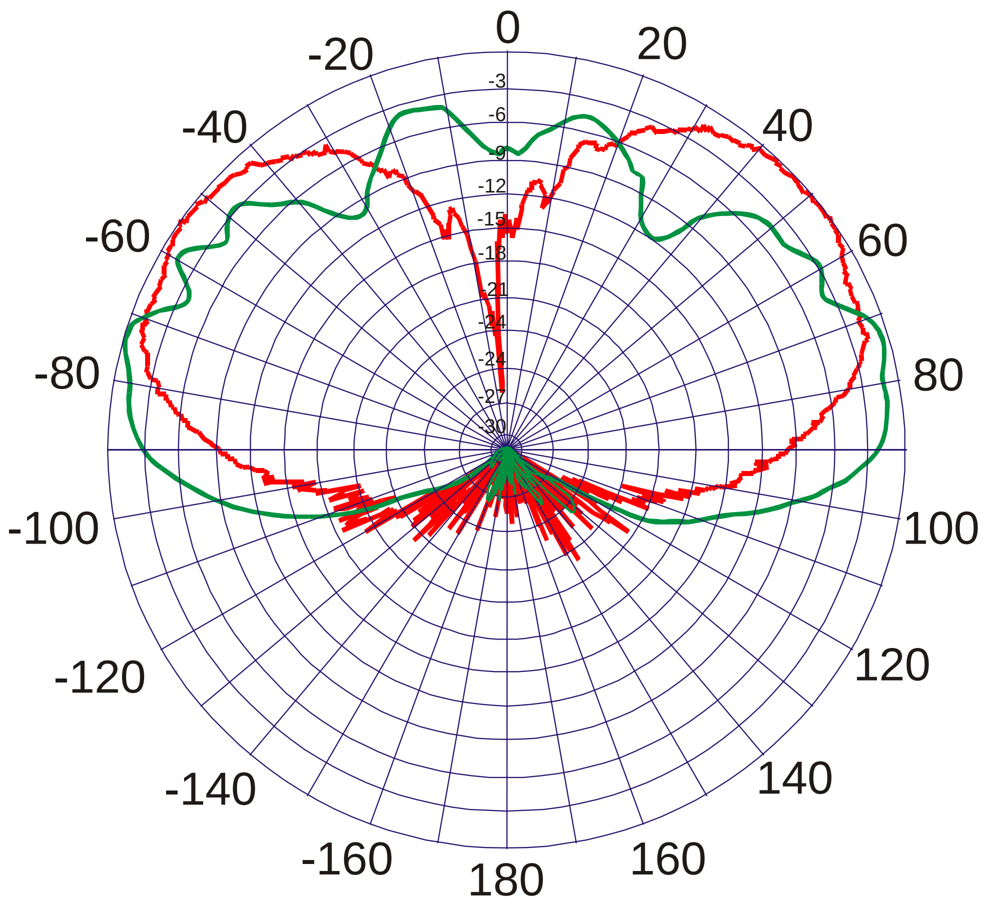


Conventional Antenna



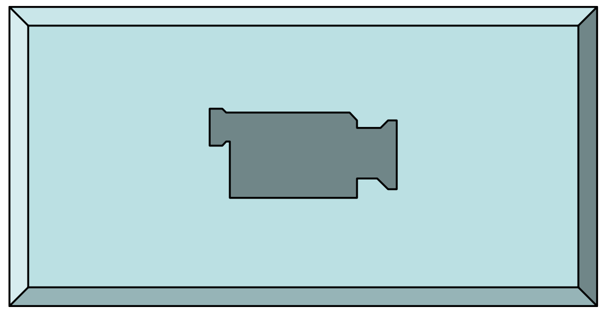
Shaded Yellow Box Denotes Area In-Spec, According to Original Mission Requirements

Directionality Comparison



Evolved Antenna (red) vs QHA (green)
Data is normalized to 0 dBic (RHCP) for each antenna
-3 dB for each division

Evolved Antenna Animation



- genetic representation
- genotype → phenotype mapping
- position on satellite

Benefits

- **Potential Lower Power**: evolved antenna achieves high gain (2-4dB) across a wider range of elevation angles
  - allows a broader range of angles over which maximum data throughput can be achieved
  - may require less power from the solar array and batteries
- **More Uniform Coverage**: Very uniform pattern with small ripples in the elevations of greatest interest (40 – 80 degrees)
  - allows for reliable performance as elevation angle relative to the ground changes
- **Inexpensive Design Cycle**: 3 person-months to run algorithms and fabricate the first prototype as compared to 5 person-months for contractor team
- **On Schedule to be First Evolved Hardware in Space when mission launches in 2004**

Future & Conclusion

Future

- Automated design across multiple levels: macro, micro, nano
- New antenna design problems
- Evolved nano-devices and architectures
- FPGA Self-Repair (real-time/subsecond repair)
- MEMS design & manufacturing problems
- Ornithopters (macro/micro)
- Modular robotic control

Conclusion

- Evolutionary design/optimization:
  - Fast design cycles save time/money
  - Fast design cycles allow iterative “what-if”
  - May discover designs that challenge expert designers
- Which NASA applications area a good fit for EAs?
- Evolved antenna for ST5:
  - Meets mission requirements
  - Unusual organic topology
  - Currently undergoing space qualification