Programmable SDR

CBIS 2007

Chris Wasser
Technical Director
Northrop Grumman

Definition

- SDR
  - Software-defined Radio
  - “wireless communication in which the transmitter modulation is generated or defined by a computer, and the receiver uses a computer to recover the signal”*

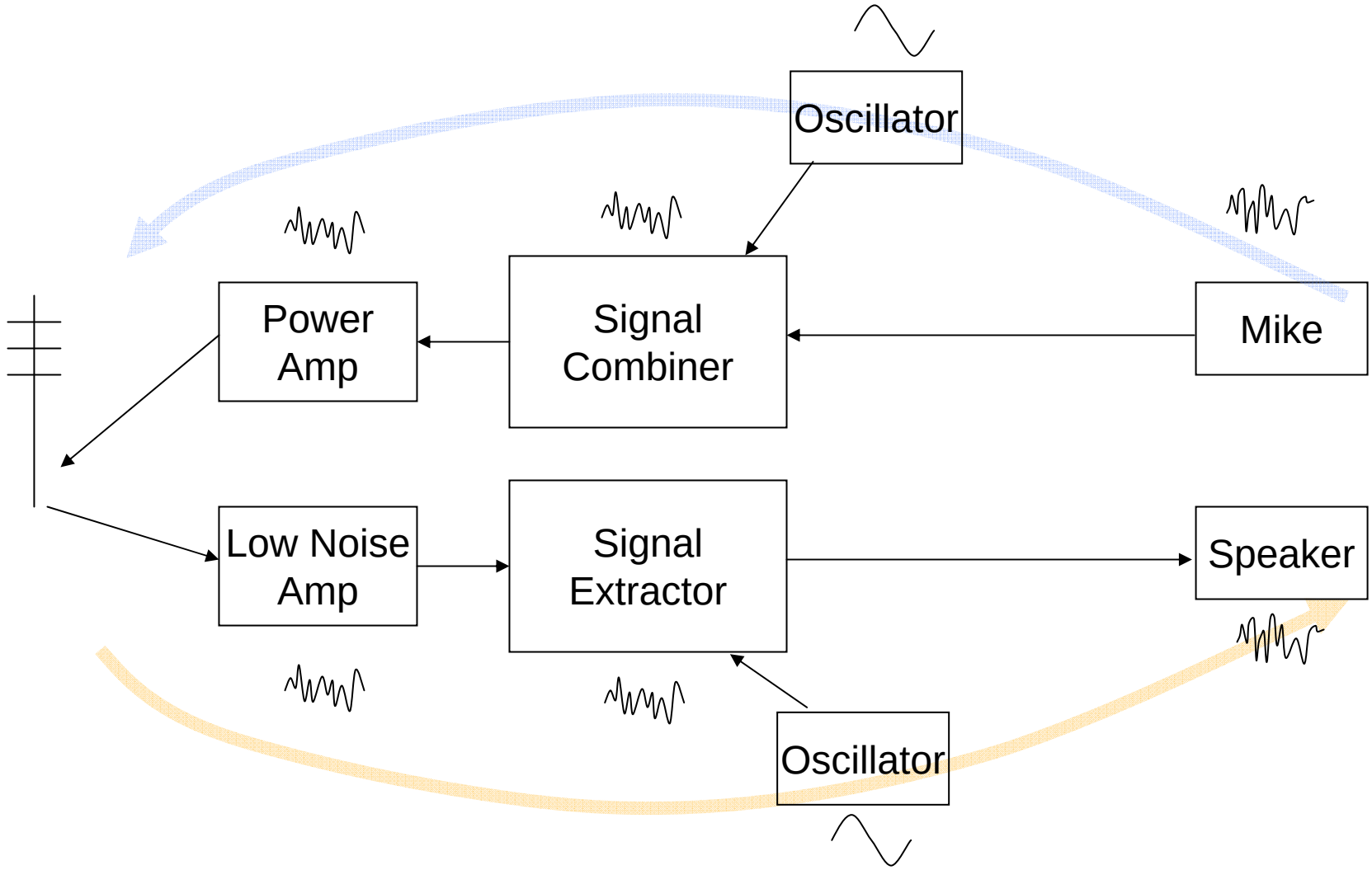
* http://searchnetworking.techtarget.com/sDefinition/0,,sid7_gci333184,00.html

Background

- Transmission of information (using Morse Code) via “wireless telegraphy” was developed by Guglielmo Marconi in 1895, although others had experimented with electricity and “spark-gap” transmission as much as 50 years prior.

- To this day, radios provide essential communication to military forces the world over.

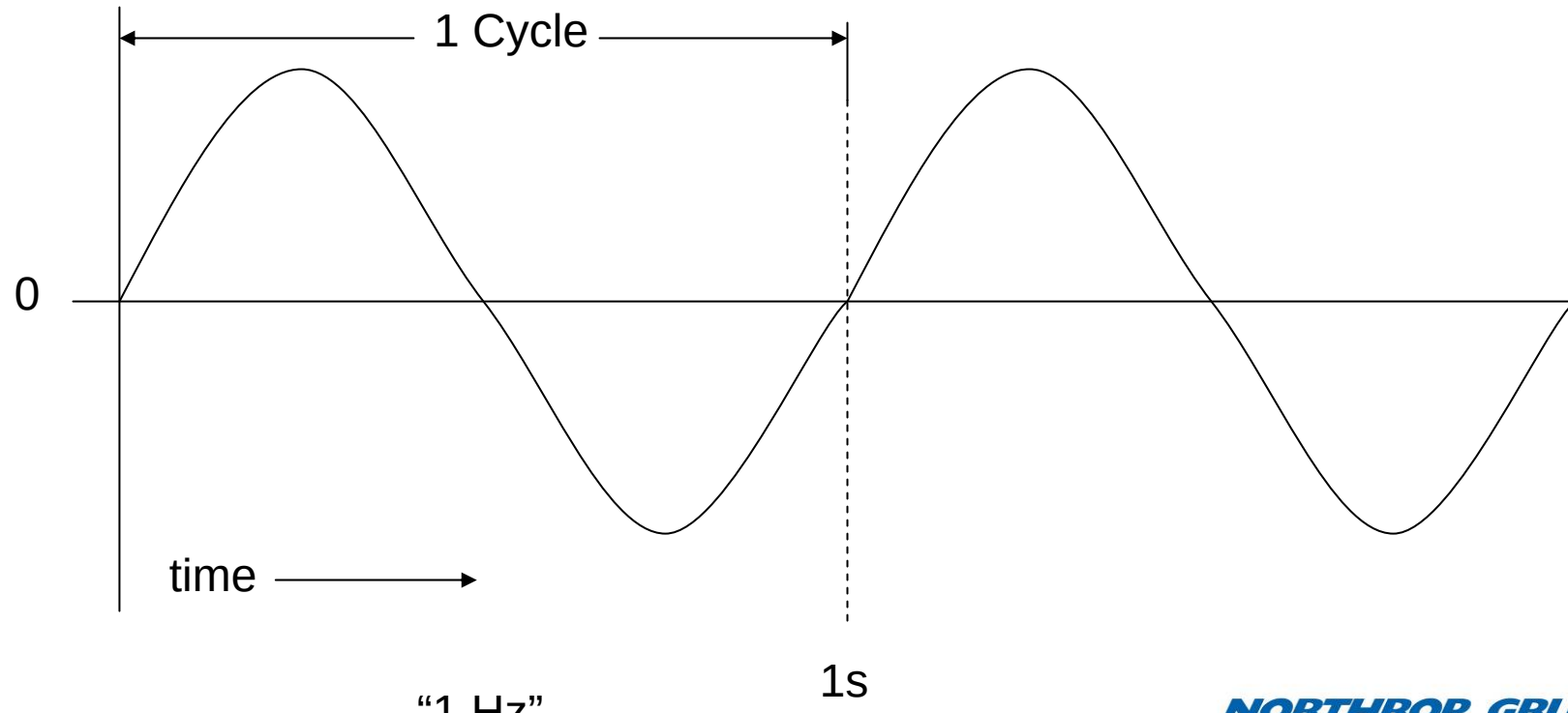
Radio Design



NORTHROP GRUMMAN

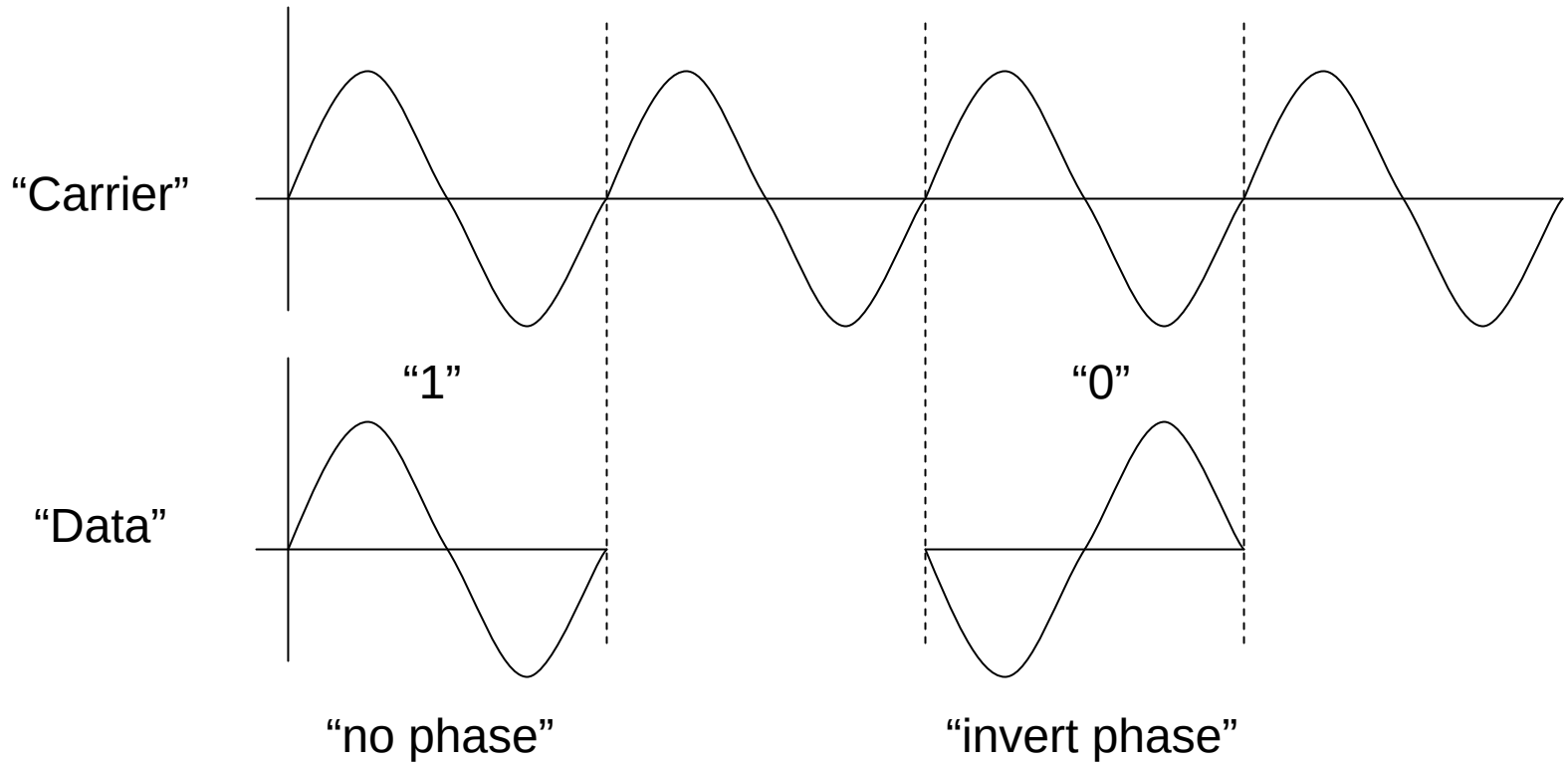
Radios

- How do they work?
  - Electro-magnetic (RF) energy creation
    - Sine wave

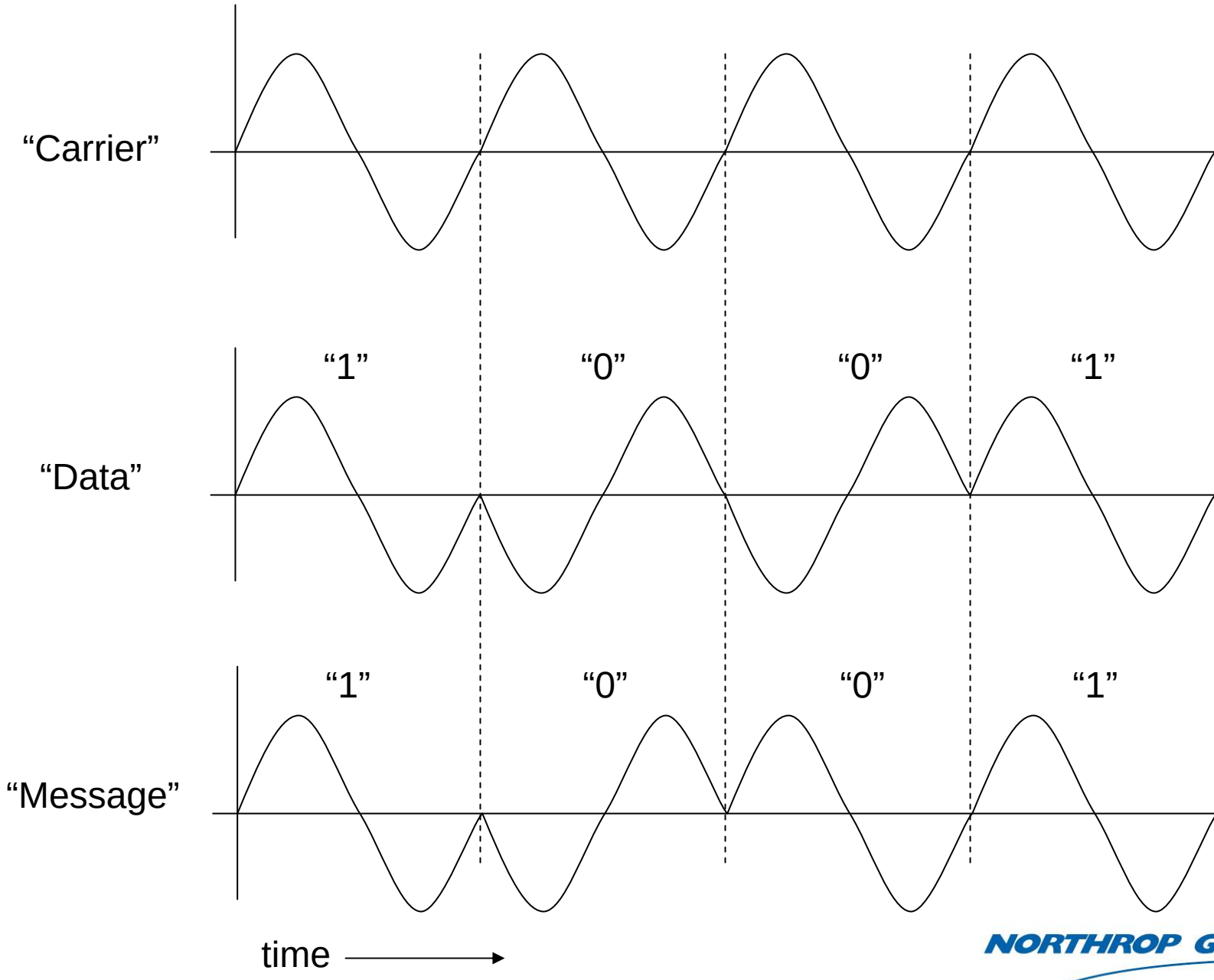


Radios (2)

- Manipulate the RF to carry information

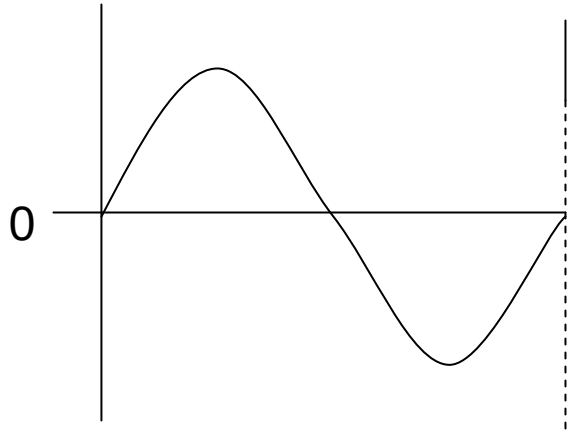


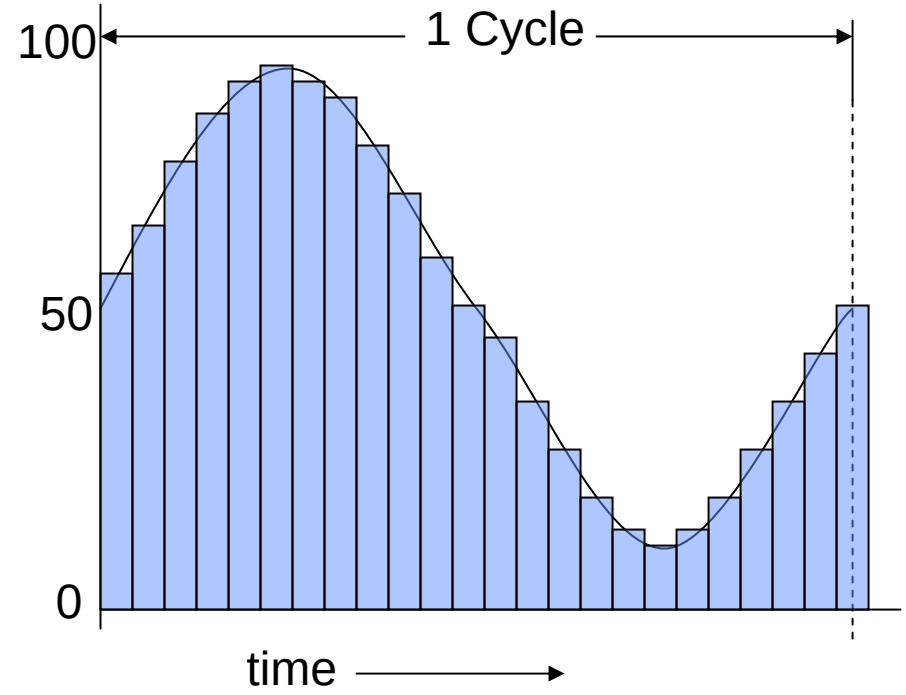
Radios (3)



Another Way to Create RF

- create “sine wave” from scratch
  - approximate the shape using digital values
  - using a “Digital to Analog Converter” (DAC)

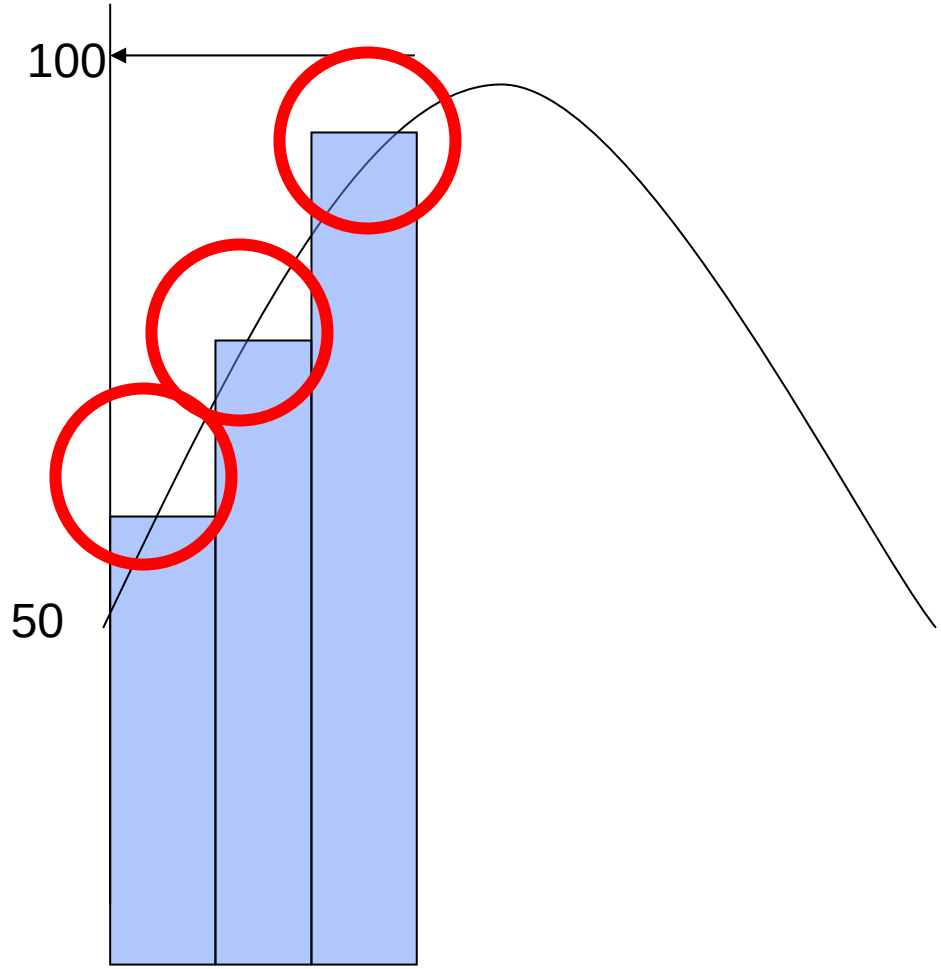




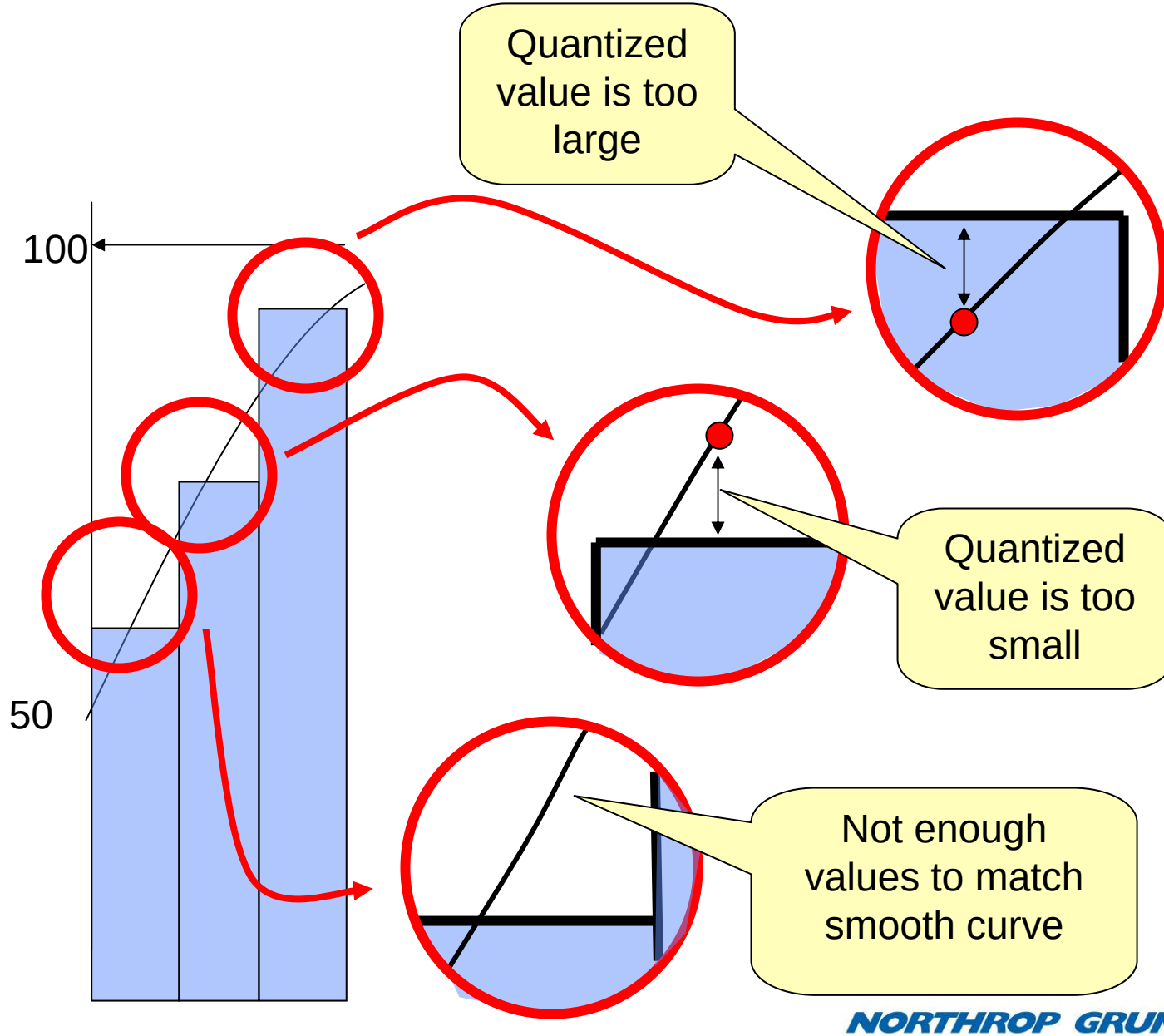
NORTHROP GRUMMAN

Some problems with the approximation

- Since the number values are “quantized”
  - rigid set of whole values (0, 1, 2, 3…)



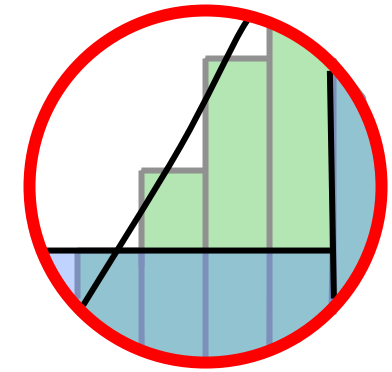
Some problems with the approximation (2)



NORTHROP GRUMMAN

What to do?

- Better quantization
  - "higher resolution"
    - More expensive



- More values
  - remember Calculus?
  - produce the values faster
    - Nyquist theorem* identifies the lower limit
    - 2B, or 2 x (bandwidth)

* http://en.wikipedia.org/wiki/Nyquist-Shannon_sampling_theorem

A Short Aside…

- We have to say it… JTRS
  - Joint Tactical Radio System
- Driving force behind SDR design and implementation
  - separate the protocol (waveform) from the hardware
- Design standard architecture
  - Software Communication Architecture (CA)
  - allows waveforms to be converted into RF signals
- Design of standard “waveforms”

NORTHROP GRUMMAN

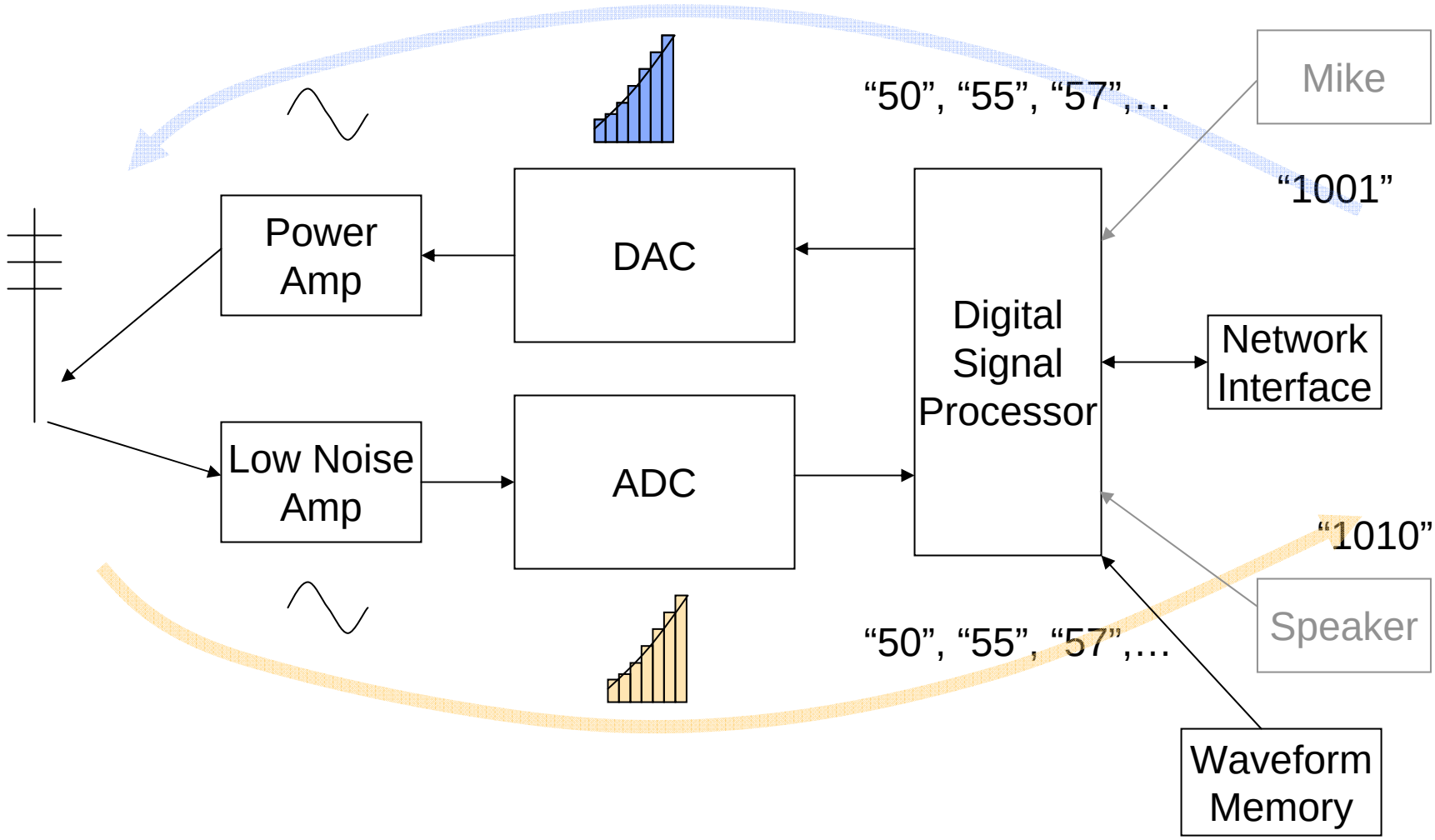
Waveforms

- ISO OSI Layers 1-3
  - physical RF signals
  - data encoding schemes
    - low-level channel sharing
    - security, error detection/correction
  - networking
    - high-level channel sharing
    - Internet Protocol (IP)
    - Quality of Service (QoS)

NORTHROP GRUMMAN

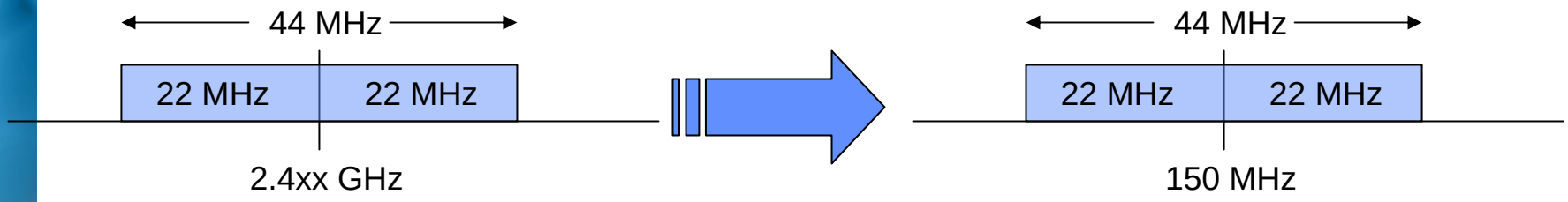
SDR Block Design

- hypothetical



Performance Example

- 802.11b
  - 2.4 GHz, 44MHz channel
    - in other words: data takes up only 44MHz



  - Nyquist rate = 2B = 2 * 44 MHz = 88 MHz sample rate
  - This leaves a LOT of CPU performance to spare!

Current Generation Hardware

- Xilinx Virtex-II Pro and Virtex 4 FX FPGAs have embedded 405 PowerPC processors
  - 400MHz,
  - FPGA implements signal processing instructions
- Intel Core2 Extreme
  - 2.93 GHz
  - High performance Math core
    - special signal processing instructions
- Texas Instruments TMS320C6x series
  - 1 GHz
  - special signal processing functions

NORTHROP GRUMMAN

CBRNE Design Example

- Current JCID design
  - CPU: Intel Xscale PXA255, 400MHz
  - Memory:
    - RAM: 256 MB
    - Flash: 128 MB
  - Embedded software: C++, ~16MB
    - Windows CE 5.0
  - Manages up to 4 sensors via serial (RS-232) or Ethernet
    - < 10% of the CPU (nominal)
  - Wireless (802.11b) mesh networking
    - separate subsystem

NORTHROP GRUMMAN

CBRNE Design Example (2)

- CPU:
  - Intel Xscale PXA255, 400MHz -> many choices
- Memory:
  - Most SDRs will have at least 256Mb for operation, plus non-volatile (flash) storage for waveforms
- Windows CE 5.0
  - possibly need to port S/W to another OS
    - likely Linux running “on top” of the SDR’s OS
- Serial (RS-232) or Ethernet
  - many JTRS SDRs will have Ethernet to support net-centric data comms
  - Serial is another matter… but chips are cheap
- Wireless mesh networking
  - part of the “waveform” design

NORTHROP GRUMMAN

Considerations

- Design issues include:
  - Different, possible “non typical” CPUs
    - PowerPC, embedded ARM, etc.
  - Real-time Operating Systems (RTOS)
    - as opposed to Windows or Linux
      - but there are options…
  - Power and heat
    - fast, powerful CPUs turn lots of electrons into lots of heat

NORTHROP GRUMMAN

CPUs

- Most current Sensor management software is either C, C++ or Java
  - PowerPC
    - GNU “gcc” compiler available
    - Java Virtual Machine (JVM) available
  - ARM
    - GNU “gcc” compiler available
    - Java Virtual Machine (JVM) available
  - Note on the JVM:
    - be wary of how much resources these take up
    - specialty versions
      - but watch for missing features

NORTHROP GRUMMAN

Operating Systems

- Most current Sensor management software is either Windows, WindowsCE or Linux
  - Windows / WindowsCE
    - NOT available for PowerPC
    - but WindowsCE is available for ARM
  - Linux
    - Both PowerPC and ARM available
      - as well as for many others…
  - Watch out for resource needs for OS over RTOS

NORTHROP GRUMMAN

Some Other Gotchas

- There are some “gotchas”
  - embedded devices tend to have “quirks”
    - limited memory
    - no support for virtual memory
    - special instructions to access Signal Processing features
  - Current SDR designs
    - have limited external connections
      - limited (or no) Ethernet
      - limited (or no) Serial port support
    - power hungry

NORTHROP GRUMMAN

Network Interaction

- Keep in mind that the application software is competing with the SDR software
  - SDR
    - low latency, high CPU demand, bursty
  - Application
    - moderate latency, moderate CPU demand, more regular
  - Watch out for applications that are:
    - Low latency
      - sensor management…
    - high CPU demand
      - M&S is probably a bad idea…

NORTHROP GRUMMAN

Conclusion

- SDRs will provide a ubiquitous computer platform across the military
- “Excess” capacity can be exploited
- Issues:
  - unique computing environment
    - RTOS, latency, specialty CPUs, odd form factors
- SDRs provide an excellent opportunity for embedding Force Protection capabilities with the forces we are protecting.

NORTHROP GRUMMAN

NORTHROP GRUMMAN

DEFINING THE FUTURE

Questions?