Introduction to Berkeley UAV Project BEAR

David H. Shim, Tak-Kuen Koo, Frank Hoffmann,
Omid Shakernia, Jin Kim, Cedric Ma,
Bruno Sinopoli, Tullio Celano,
William Morrison, Shahid Rashid, Santosh Phillp
Shankar Sastry
University of California, Berkeley

September 28, 1999

Presentation Outline

1. Introduction of Berkeley UAV Fleet
2. Overview of Berkeley UAV Navigation System

Berkeley BEAR UAV Research

The Goal

Develop theoretical methodologies and experimental platform for

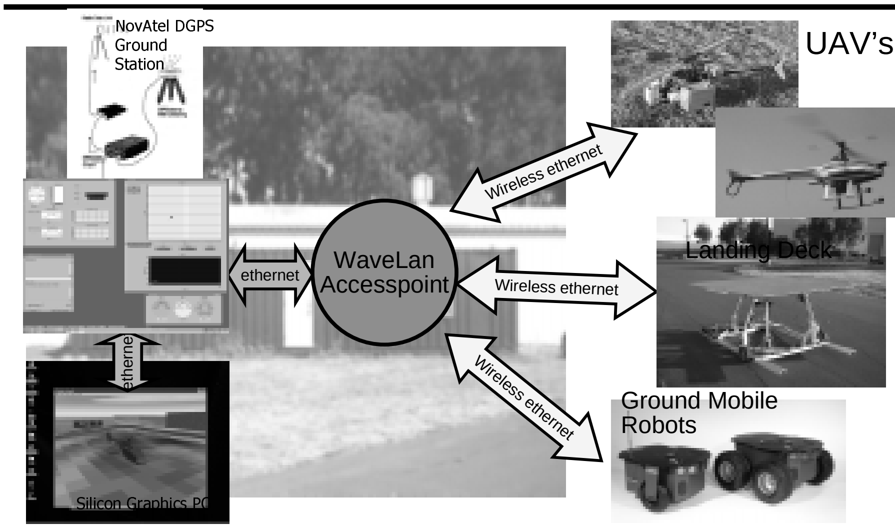
- vision-based landing
- multi-agent coordination
- landing on a pitching ship deck
- advanced guidance and control law

Berkeley BEAR UAV Research

3. Project History

August 1996: First helicopter assembled. Payload test (Ursa Minor 1)
October 1997: Ursa Minor 1 assembled based on Crossbow Accelerometer
April 1998: Second helicopter assembled: Ursa Minor 2
July 1998: Ursa Minor reconfigured with MotionPak based navigation
August 1998: Ursa Magnus joined
August 1998: Participated in UAVS competition at HAMMER, WA
September 1998: First instrumented flight using Ursa Minor 2
October 1998: Ursa Minor 3 joined
December 1999: research paper presented in IEEE CDC at Tampa, FL
Dec. 1998-Mar. 1999: mu-synthesis based attitude controller tested
March 1999: Boeing DQI-NP based navigation system implemented
June 1999: Yamaha R-50 purchased. Named to "Ursa Magnus"
August 1999: First instrumented flight using Ursa Magnus 2

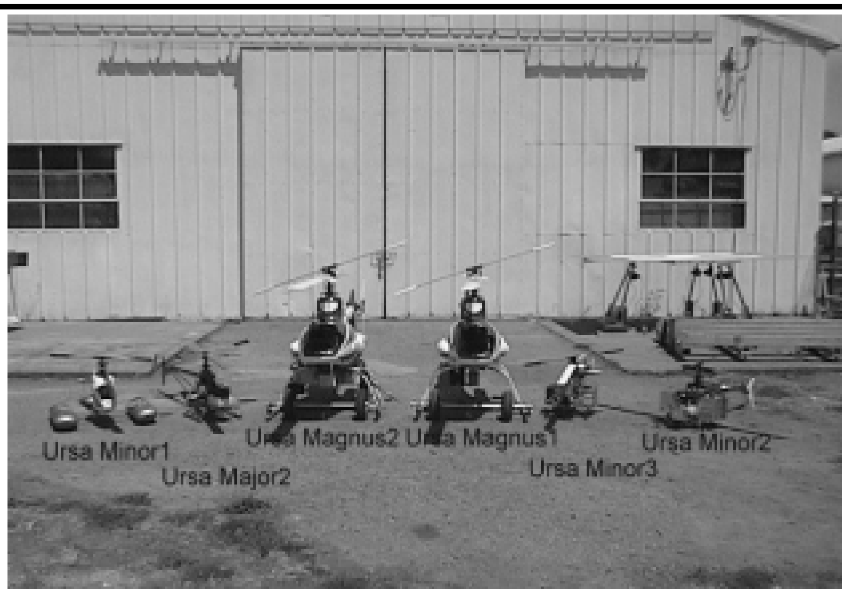
Berkeley UAV Research Platform



Ground Monitoring System

WaveLAN: T. John Koo
Pioneer mobile rotot: Omid Shakernia, Frank Hoffman
Pitching deck landing pad: Tulio

Berkeley BEAR Team Fleet Line-up



Berkeley BEAR Team Fleet Line-up

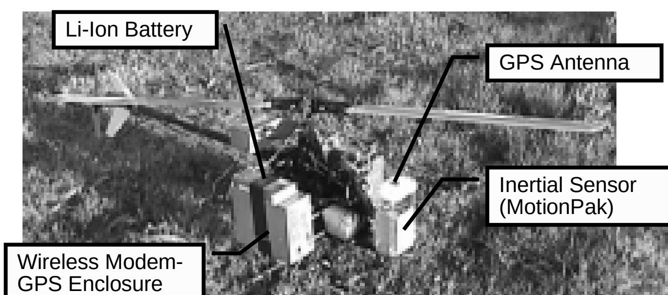
1. Ursa-Minor Series (1,2,3)
   Based on Kyosho Concept 60
   Modified OS FX-91(14cc) 2cycle engine (Ursa Minor 2,3)
   Payload: 5kg

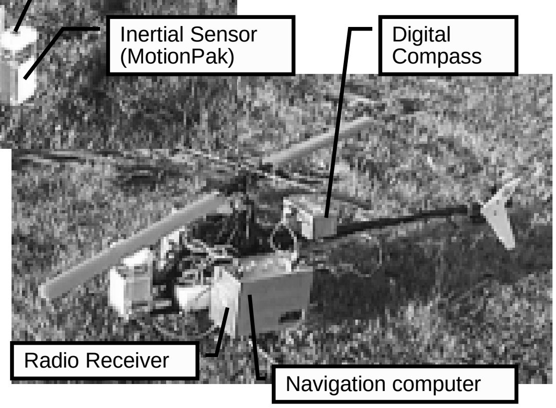
Ursa Minor 1: retired to Trainer
Ursa Minor 2: Engineering Plastic Composite body
MotionPak based navigation system
Ampro Pentium 233MMX Littleboard

Ursa Minor 3: Graphite body (stiffness)
Boeing DQI-NP based navigation
Cyrix MediaGX 233MHz PC104 board

DQI-NP: Digital Quartz Inertial Measurement Unit with Navigation Processor

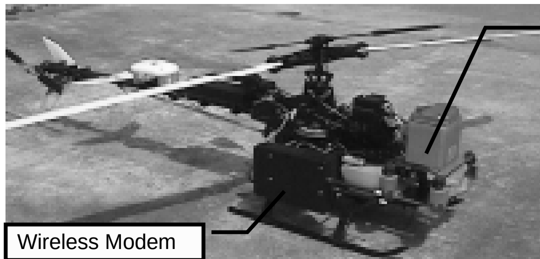
Ursa Minor 2

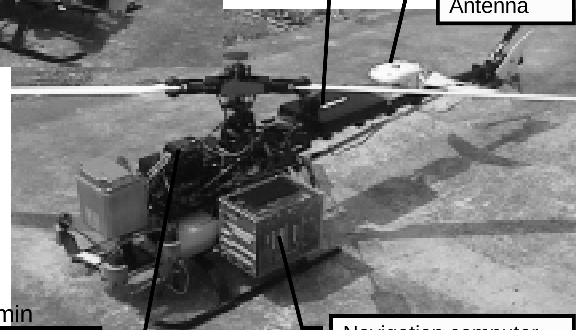




Length: 1.4m
Width: 0.39m
Height: 0.47m
Weight: 9.3 kg
Engine Output: 2.8 bhp
Rotor Diameter: 1.5m
Flight time: 15 min
System operation time: 30 min

Ursa Minor3

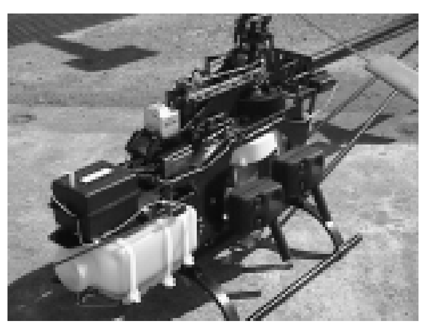




Length: 1.4m
Width: 0.39m
Height: 0.47m
Weight: 9.4 kg
Engine Output: 2.8 bhp
Rotor Diameter: 1.5m
Flight time: 15 min
System operation time: 30 min

Berkeley BEAR Fleet Line-up

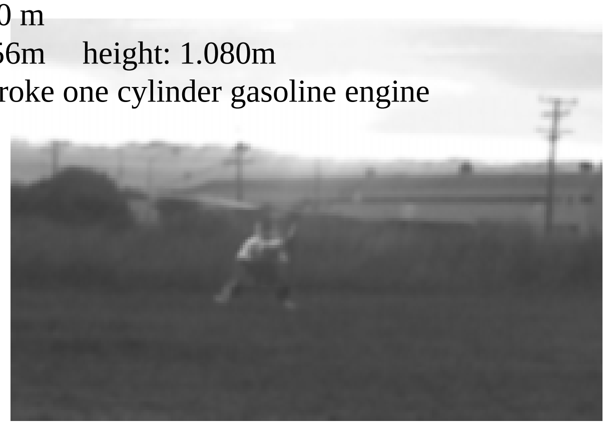
2. Ursa-Major Series
   Based on Bergen Industrial Twin helicopter
   Payload: >10kg



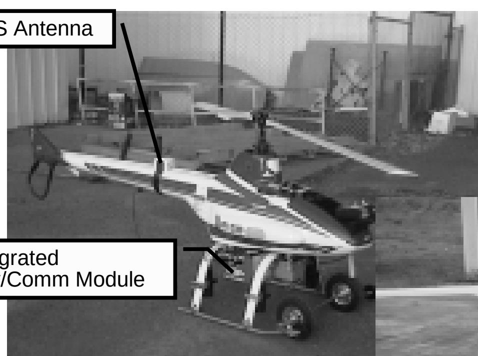
Berkeley BEAR Fleet Line-up

3. Ursa-Magnus Series
   Based on Yamaha R-50 industrial helicopter

   Rotor diameter: 3.070 m
   Fuselage length: 2.656m height: 1.080m
   Water-cooled, two stroke one cylinder gasoline engine
   Dry weight: 44kg
   Payload: >20 kg

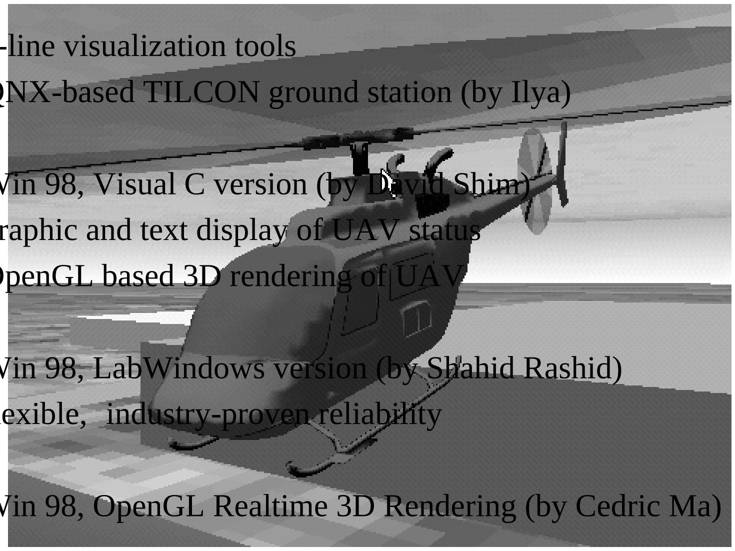


Onboard Hardware: Ursa Magnus2



Length: 3.6m
Height: 1.08m
Dry Weight: 44 kg Payload: 20kg
Engine Output: 20 hp
Rotor Diameter: 3.070m
Flight time: 30 min
System operation time: 60 min

Existing Tools- Visualization



- On-line visualization tools

1. QNX-based TILCON ground station (by Ilya)

2. Win 98, Visual C version (by David Shim)
   graphic and text display of UAV status
   OpenGL based 3D rendering of UAV

3. Win 98, LabWindows version (by Shahid Rashid)
   flexible, industry-proven reliability

3. Win 98, OpenGL Realtime 3D Rendering (by Cedric Ma)

SEC September, 1999

Test Flight Video

Ursa Magnus 2: “frequency sweeping” flight for system id

SEC September, 1999

Test Flight Video

Ursa Minor-1: landing on the ship-motion deck

Navigation Software

- **Real-Time OS: QNX**

- **Advantages**

- Systematic way to write Real-time programs
- UNIX-like system architecture
- Very compact kernel
- Advanced Inter-Process Communication(IPC) support

- **Drawbacks**

- Latency issues
- Limited number of supporting hardware
- Limited number of available application software

Navigation Software- Features of QNX

- **Multiprocessing (priority)**

- **IPC**
  Messages, proxies, shared memories

- **Real-time feature**
  supports PC hardware interrupts via *Proxy*

- **Scheduling**: process priorities, states and algorithms
  Priorities: 0(lowest)-31(highest)
  States: *READY*, *BLOCKED*(*send-*, *receive-*, *reply-blocked*)
  Algorithm:*FIFO*, *Round-robin*, *Adaptive*

Navigation Software- Network support

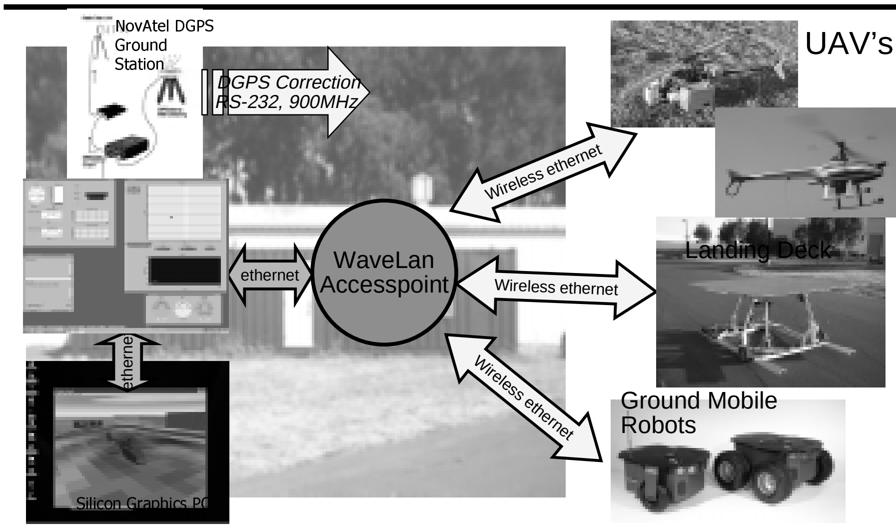
Network support of QNX

- QNX-only network: seamless virtual machine distributed over network (supports TCP/IP, peer-to-peer RS-232)
  supports spawning processes and IPC over network

- Heterogeneous network: QNX and non-QNX computers (TCP/IP)
  Flight info download and command upload using TCP/IP

- CORBA on QNX: Orbix by IONA (www.iona.com)

Berkeley UAV Network Configuration

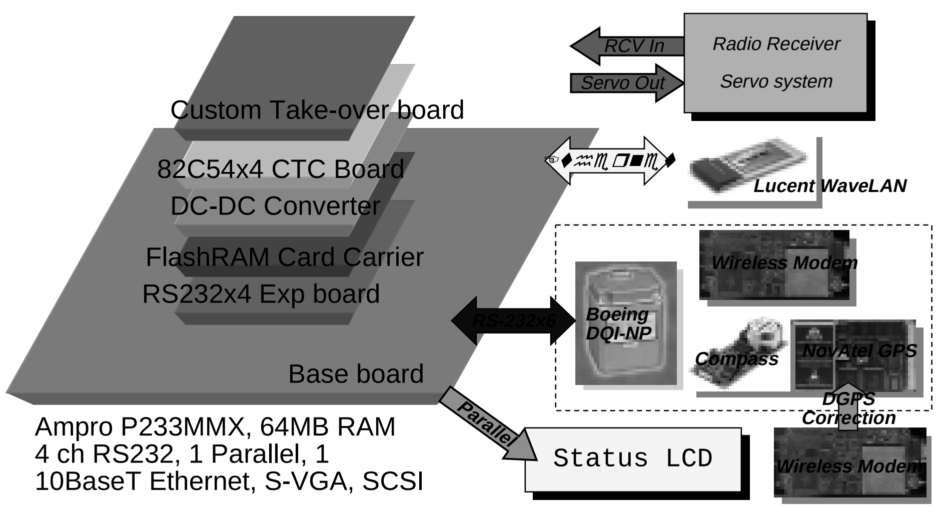


Ground Monitoring System

WaveLAN: T. John Koo
Pioneer mobile rotot: Omid Shakernia, Frank Hoffman
Pitching deck landing pad: Tullio

Navigation Hardware

1. Navigation Computer Configuration



Ampro P233MMX, 64MB RAM
4 ch RS232, 1 Parallel, 1
10BaseT Ethernet, S-VGA, SCSI

Navigation Hardware

- Navigation computer

- Navigation Sensors
  Boeing DQI-NP, NovAtel GPS(Millen RT-2), Digital Compass,
  Ultrasonic height meter (x4)

- Miscellaneous sensors: contact switch(x4), engine encoder

- Communications
  900MHz Wireless modem(x2), Lucent WaveLan(2.4GHz)

- Vision System
  Stereo camera on Pan-Tilt-Verge Platform, Video transmitter
  On-board vision processing computer (Ampro P233MMX)

SEC September, 1999

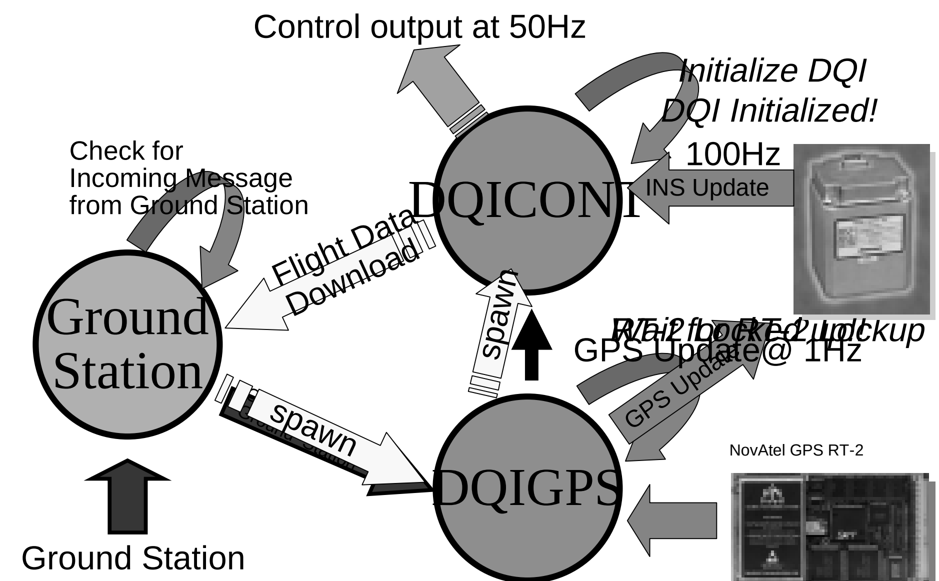
Navigation Software: DQI-NP-Based

Tasks of Onboard Navigation Computer Software

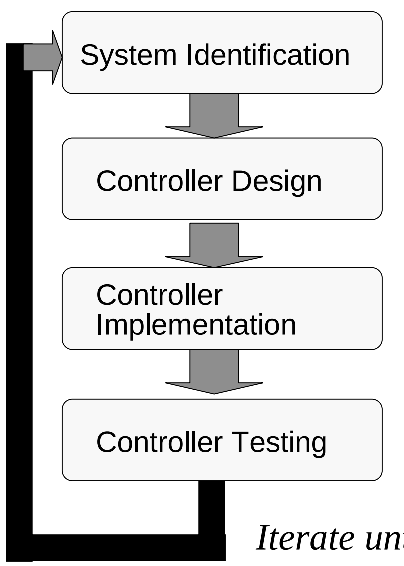
1. Initialize navigation sensors(GPS, IMU)
2. Acquire sensor information from GPS, IMU, Ultrasonic height meters
3. Update the Boeing DQI-NP using GPS measurements
4. Calculate the stabilizing and tracking control output
5. Download system status and navigation information
6. Process uploaded command from ground station

SEC September, 1999

Navigation Software: DQI-NP-Based



System Integration Procedure



- Parameter acquisition (step through)
- System estimation/validation with input-output samples

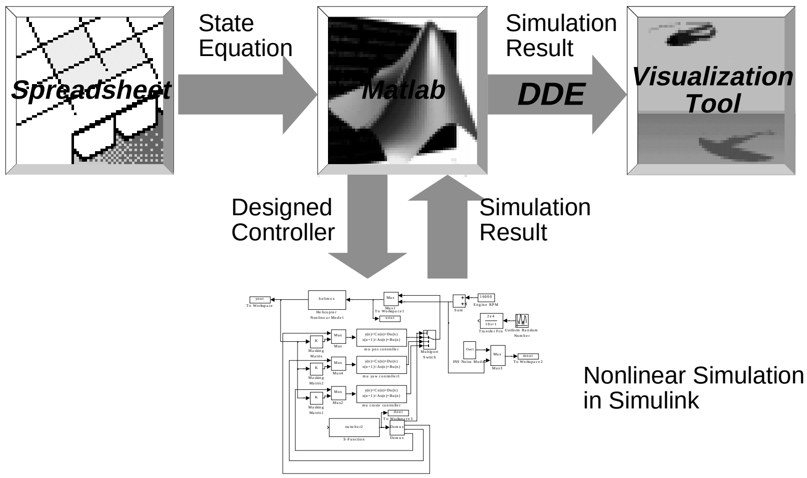
- Different design methods and benchmarks
- ***Robustness issue***

- Code generation (Manual or Computer-aided)

- To evaluate the controller performance
- Serious difficulties and risks involved

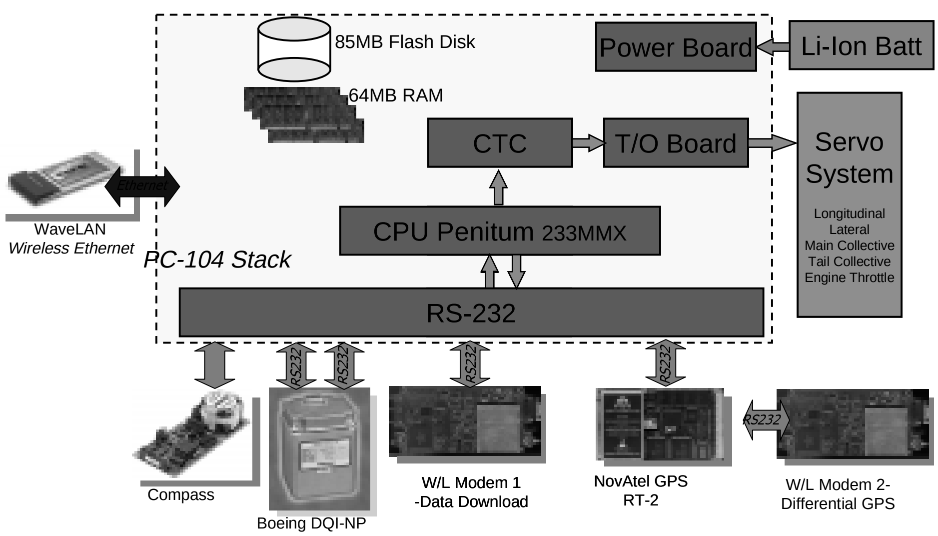
Iterate until satisfied

Integrated Development Environment



Navigation Hardware

1. Ursa Magnus 2: Boeing DQI-NP based system



UAV Design Philosophy

- **Functionality**

  install the necessary sensor systems and processors to perform the given task: autonomous take-off/landing, hover, way-point navigation
- **Light & Compact Design**

  Weight reduction for better flight performance

  Use compact parts and enclosures to minimize the inertia and maximize the rotor efficiency
- **Modularity**
- **Safety and Reliability**
- Minimization of interference among components

Complications in System Integration

- Base system: a model helicopter with unknown parameters
- Integration of diverse mechanical and electronic components, most of which were not intended to be used in harsh environment
- Interference among system components

  interference among W/L modem, GPS, Video Tx, radio controller...
- Exhausting process of Trial and Error
- Special issues in operation:

  - Not-friendly operation environment

    heat, vibration, shock, dirt, oily spray, etc...

  - RELIABLILTY, SAFETY