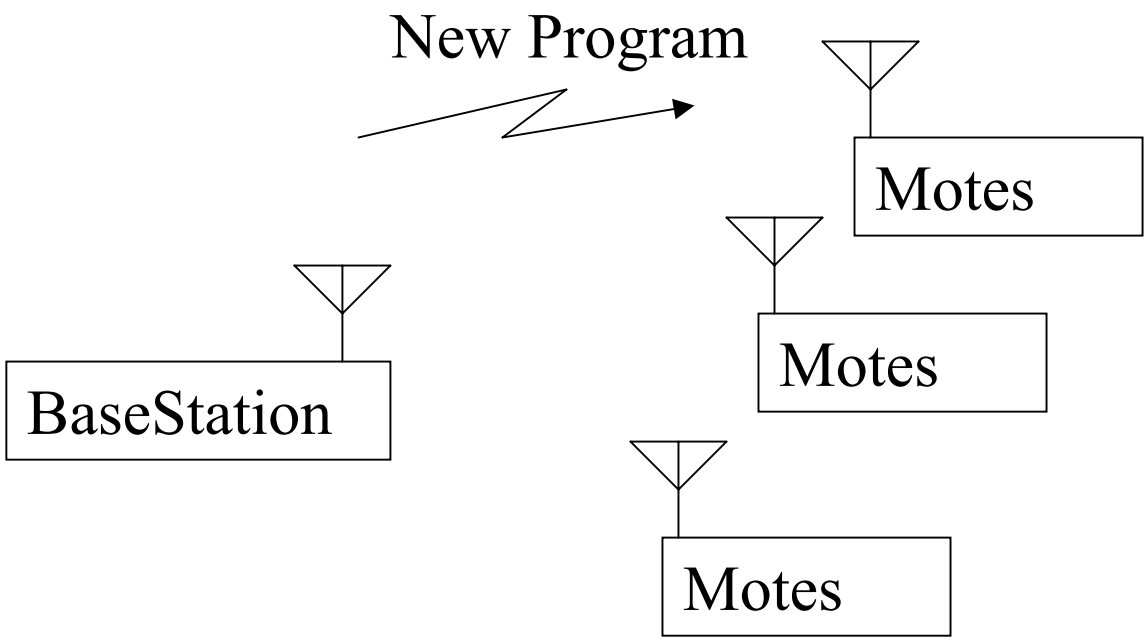
XNP

- Crossbow In-Network Programming



What Is XNP?

- Adds Wireless In-Network re-programming to any TOS Application
- Individual or Group Mote Updates
- Host Side GUI Control program
- Compatible with Single and Multi-hop Routing (forward & reverse path)

How XNP Works

- Two Phases
  - Download Application's SREC file from Host to local FLASH Memory
  - Re-program Mote
- Implemented as a private Active Message Service – all XNP radio messages are processed independently of Application

Step #1 XNP Download

- Host Broadcasts Start Download message
- Host broadcasts srec file (2 TOS packets per srec record/capsule) to all Motes.
- Active Motes store capsules in FLASH
- Host Queries motes for any “missing/lost” capsules
- Motes request “missing” capsule id
- Host transmits lost capsule
- Repeat until all Active Motes have complete image

Step #2 XNP Re-Program

- Host Broadcasts Re-Program / Re-Boot command with Program ID
- Active Motes verify Program ID matches downloaded ID
- Motes store their current Mote ID in non-volatile memory
- Motes re-program themselves and re-boot
- Application fetches Mote ID from non-volatile memory and restores ID.

How to Use XNP

- Add XNP Event Support functions
  - XNP signals a request to start XNP download
  - Application must release resources (External FLASH)
  - Application responds with GRANT or DENY
  - XNP signals Done when XNP download ends
- Add `Xnp.NPX_SET_IDS` call in INIT to restore Mote and Group Ids.
- Wire xnpc.nc into application
- Install Application and XNP Bootloader in Motes

Example Interface to XNP

- event result_t Xnp.NPX_DOWNLOAD_REQ(uint16_t wProgramID, uint16_t wEEStartP, uint16_t wEENofP)

```
{//Acknowledge NPX
call Xnp.NPX_DOWNLOAD_ACK(SUCCESS);
return SUCCESS;
}//event download_req
```

- event result_t Xnp.NPX_DOWNLOAD_DONE(uint16_t wProgramID, uint8_t bRet,uint16_t wEENofP){

```
return SUCCESS;
}//event download_done
```

- command result_t StdControl.init() {

```
call Xnp.NPX_SET_IDS(); //restore id s
… //standard init code
}
```

Example Component Wiring

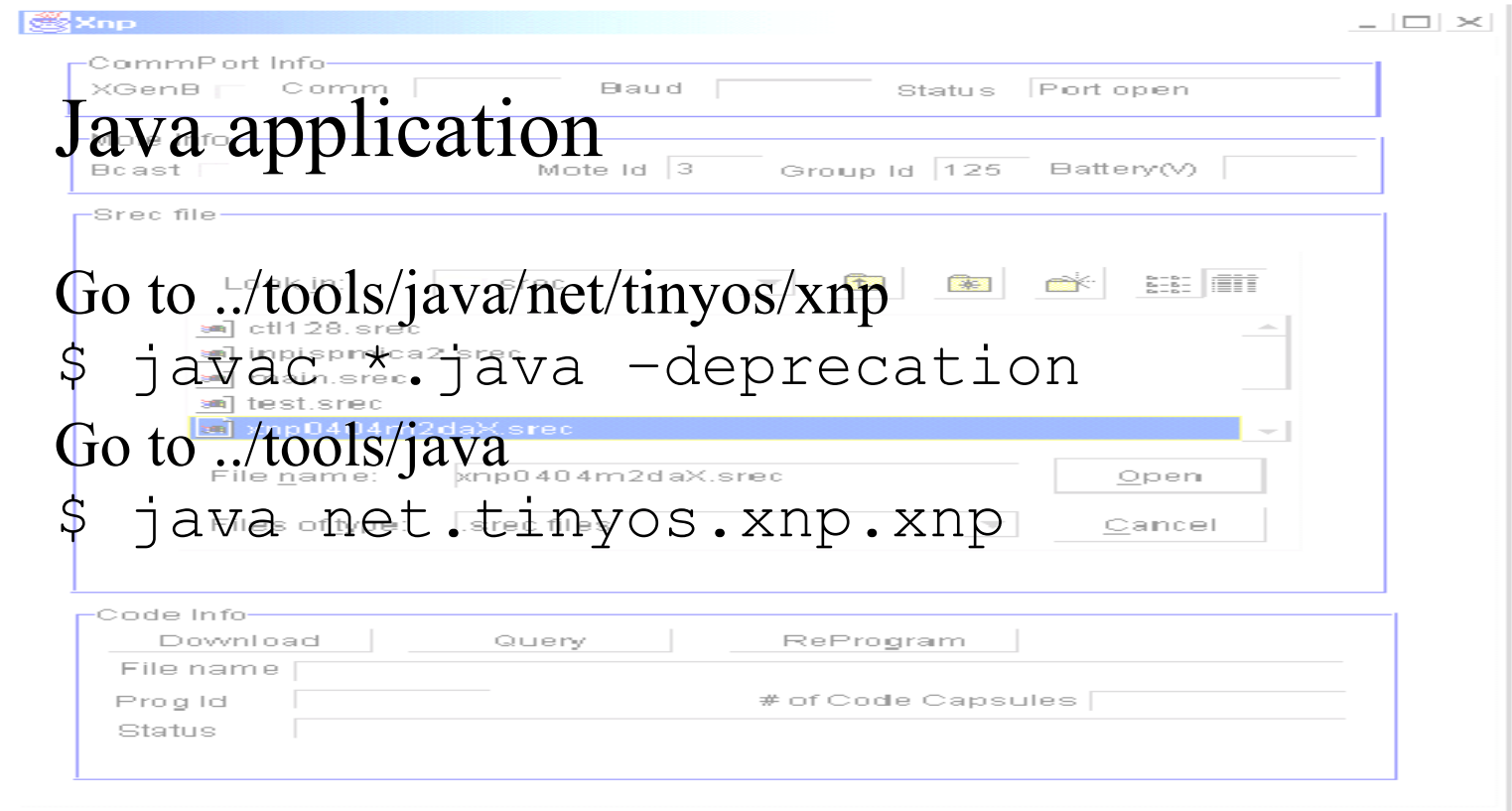
```
configuration XTestXnp {
  }
  implementation {
    components Main, GenericComm, ClockC, LedsC,
  XTestXnpM, XnpC;
    Main.StdControl -> XTestXnpM.StdControl;
    XTestXnpM.GenericCommCtl -> GenericComm;
    XTestXnpM.Clock -> ClockC;
    XTestXnpM.Leds -> LedsC;
    XTestXnpM.Xnp -> XnpC;
  }
```

Installation of XNP Bootloader

- Install Application (w/ XNP services)
- Install Bootloader in upper Memory
- Makerules has this capability built-in
  - `$ make install.<moteid> inp <platform>`
- *Bootfiles*
  - Inpispm2.srec Mica2
  - Inpispm2d.srec Mica2Dot
- Makerules error – fix this line:

```
inp: FORCE      $(PROGRAMER) $(PROGRAMMER_FLAGS_DAPA) --upload if=$(BOOTLOADER)
```

XNP Host User Interface



Java application

Go to ../tools/java/net/tinyos/xnp

```
$ javac *.java -deprecation
```

Go to ../tools/java

```
$ java net.tinyos.xnp.xnp
```

Demonstration Example #1 Install XNPBlink on Mote

- $ cd apps/xnpblink
- $ make install.12 mica2
  - With mica2 and MIB attached to LPT/Programmer port
- $ uisp –dprog=dapa –dlpt=3 - -upload if=inpispm2.srec
  - Or use the "—upload command line" from step 2 with if=inpispm2.srec (inpispm2d for dots)

Demonstration Example #2 “New Program”

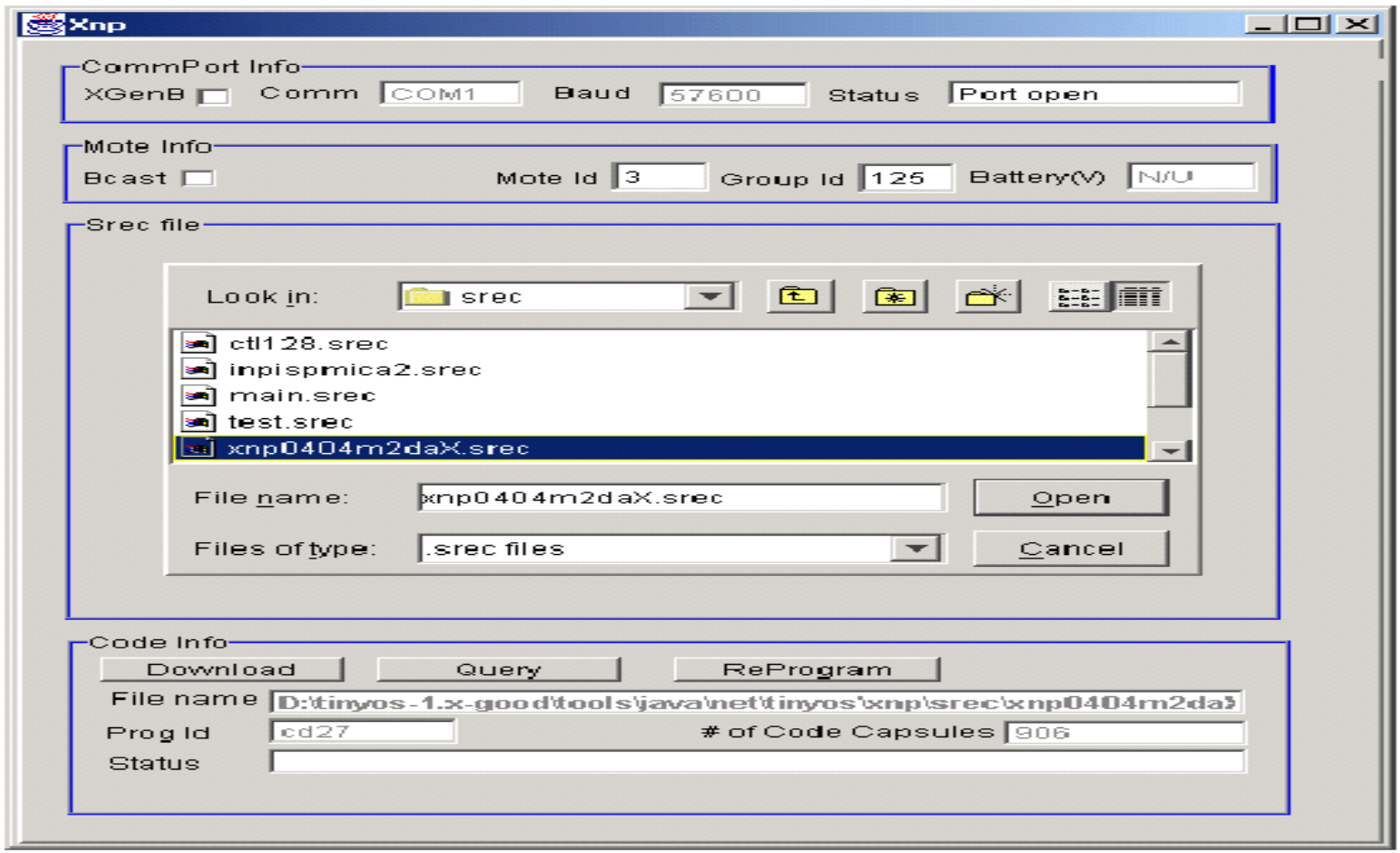
- Build the “new” program to download
  - Cd to XNPCount
  - $make mica2
  - This creates a main.srec file under ../build/mica2

Demonstration Example #3 Basestation Setup

- Install Xgenericbase onto a mote. Attach to serial port

  *Connect “BaseStation” to serial port.
- $ cd tinyos-1.x/tools/java/net/tinyos/xnp
- $ javac *.java –deprecation
- $ cd tinyos-1.x/tools/java
- $ java net.tinyos.xnp.xnp &
- Opens up XNP Gui

XNP GUI


Xnp
CommPort Info
XGenB
Comm
COM1
Baud
57600
Status
Port open
Mote Info
Bcast
Mote Id
3
Group Id
125
Battery(V)
N/U
Srec file
Look in:
srec
ctl128.srec
inpispmica2.srec
main.srec
test.srec
xnp0404m2daX.srec
File name:
xnp0404m2daX.srec
Open
Files of type:
.srec files
Cancel
Code Info
Download
Query
ReProgram
File name
D:\tinyos-1.x-good\tools\java\net\tinyos\xnp\srec\xnp0404m2da
Prog Id
cd27
# of Code Capsules
906
Status

XNP GUI

- Select XGenB
- Set GROUPID to your ID
- Select BCAST
- SREC File Selection
  - Navigate and select apps/xnpcount/build/mica2/main.srec
- Verify XNPBlink mote is turned ON (blink)
- Press Download
  - XNPBlink leds should change to fast blinking
  - Takes about 5-10 minutes
  - Watch Query / Missing packets update
- Press PROGRAM
  - BLINK changes to COUNT