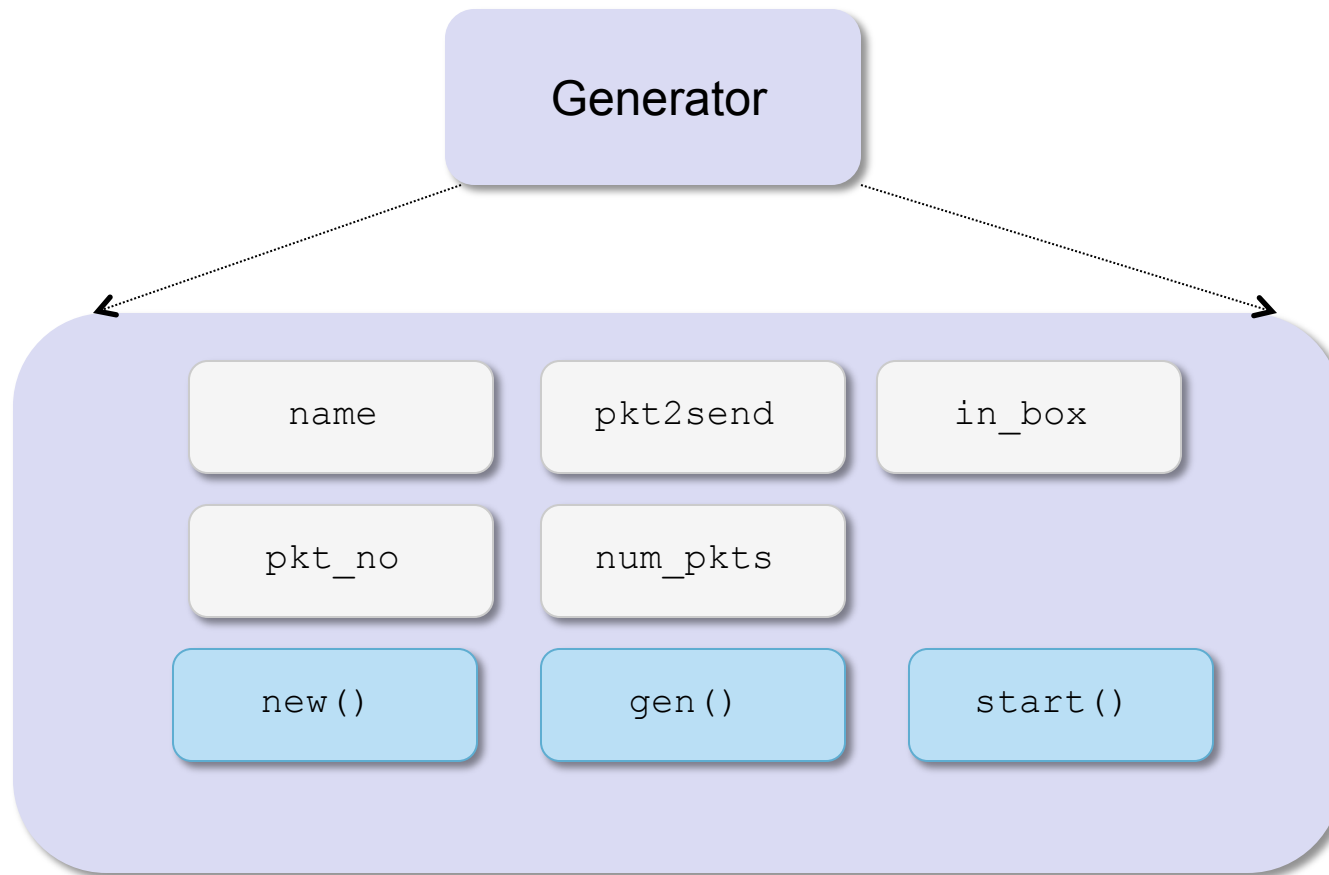# ASIC Verification
ECE745

Lab 3

Fall 2009
Dr. Meeta Yadav

Topics

- Generator and Driver Overview
- New features introduced in the lab
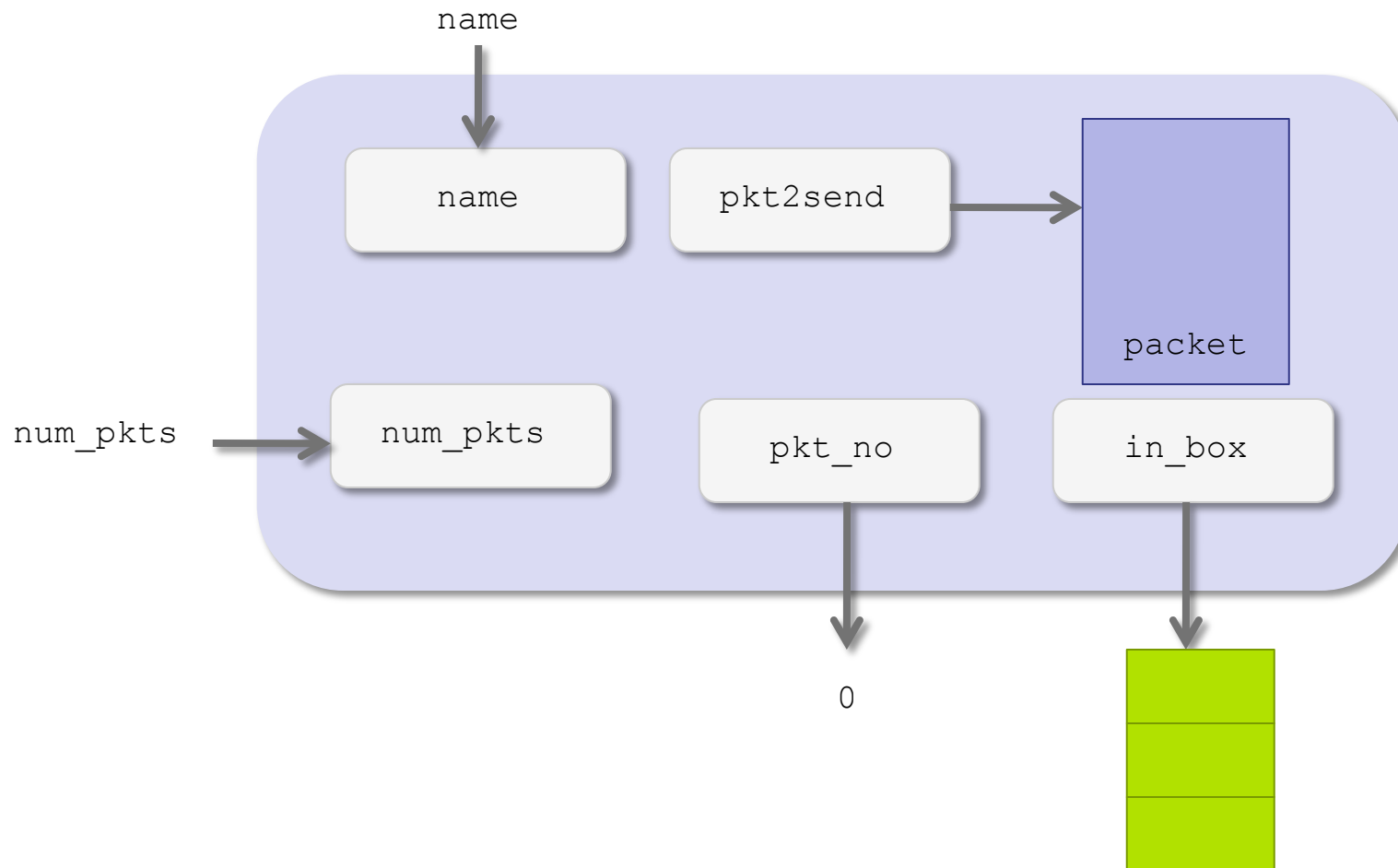
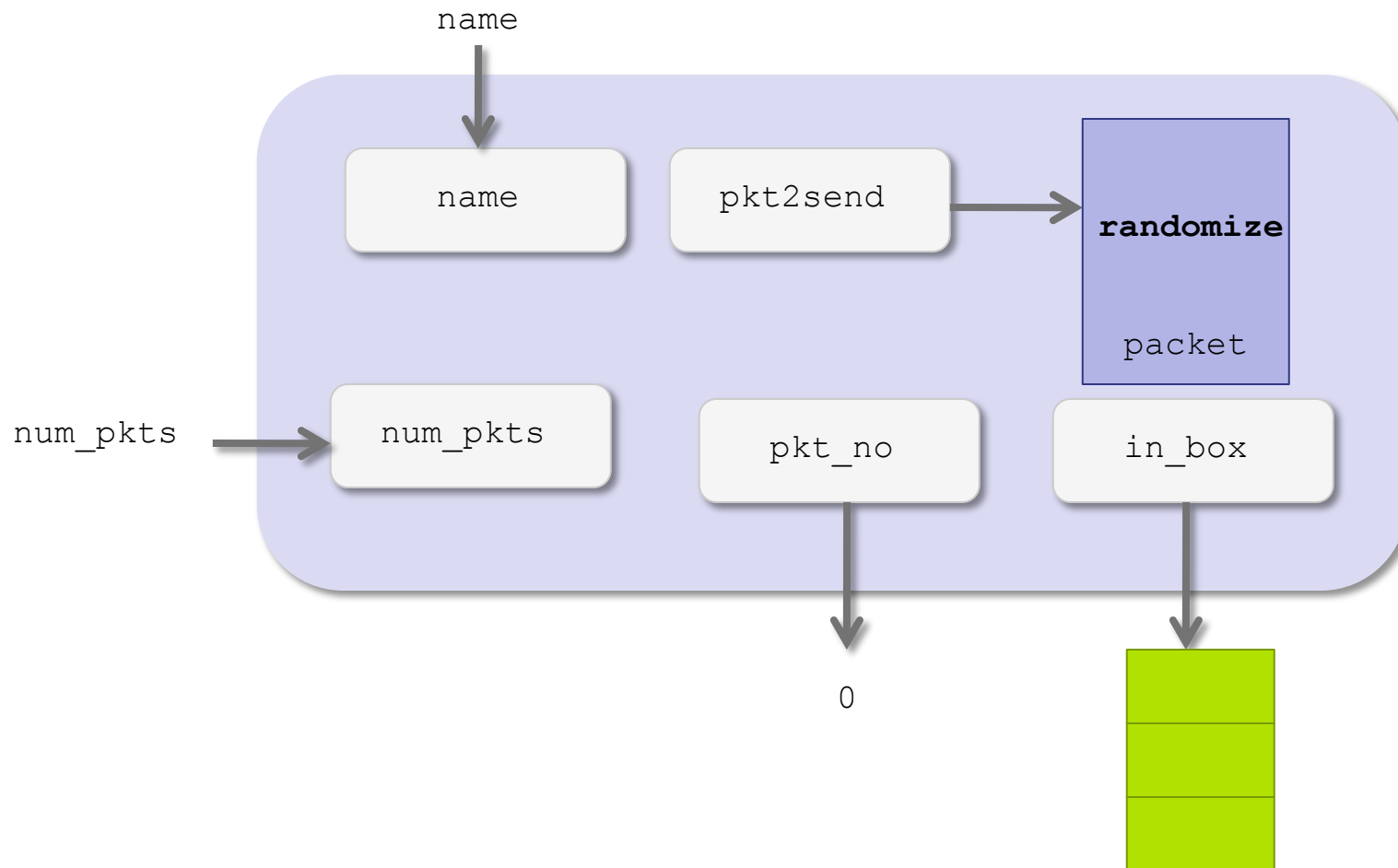# Generator
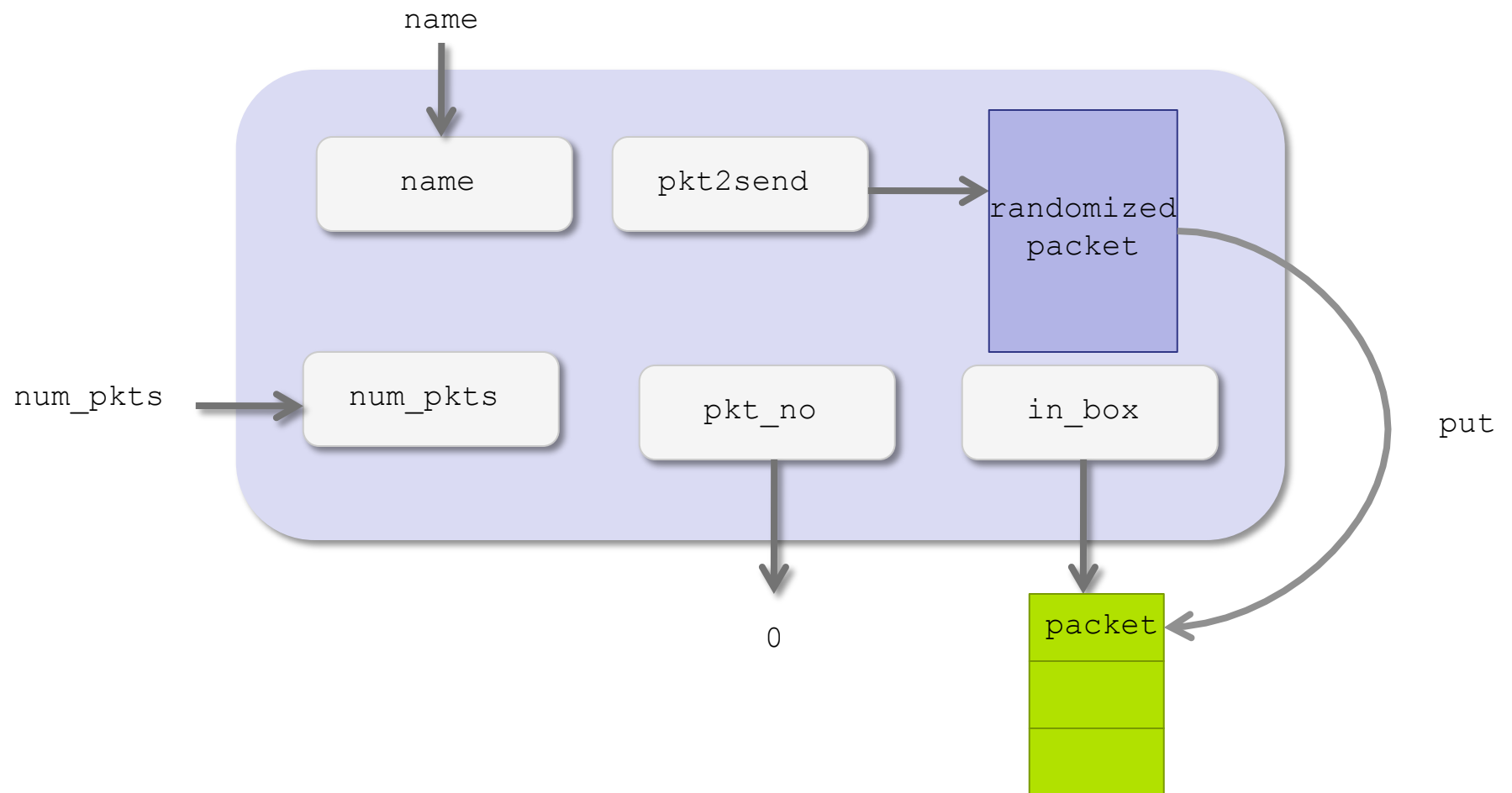
- Generator

Generator
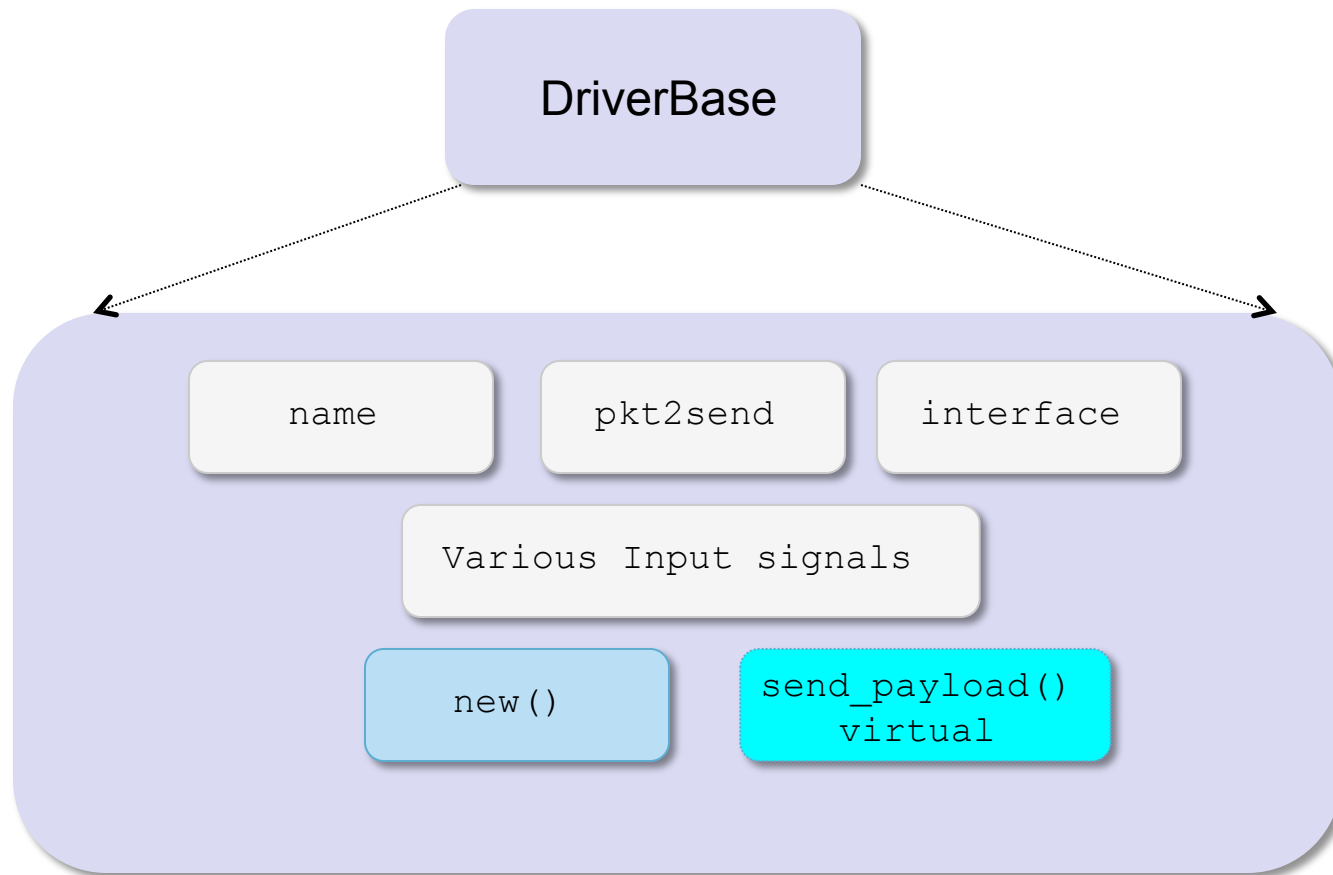
- Generator: new( )

Generator

- Generator: gen( )

name

| name | pkt2send | randomize |
| | | packet |

num_pkts → num_pkts   pkt_no   in_box

0

Generator

- Generator: start( )

DriverBase

- DriverBase

DriverBase

- ## DriverBase: new ( )

DriverBase   Execute Interface

name	pkt2send	interface

Various Input signals

Drive Signals into the DUT
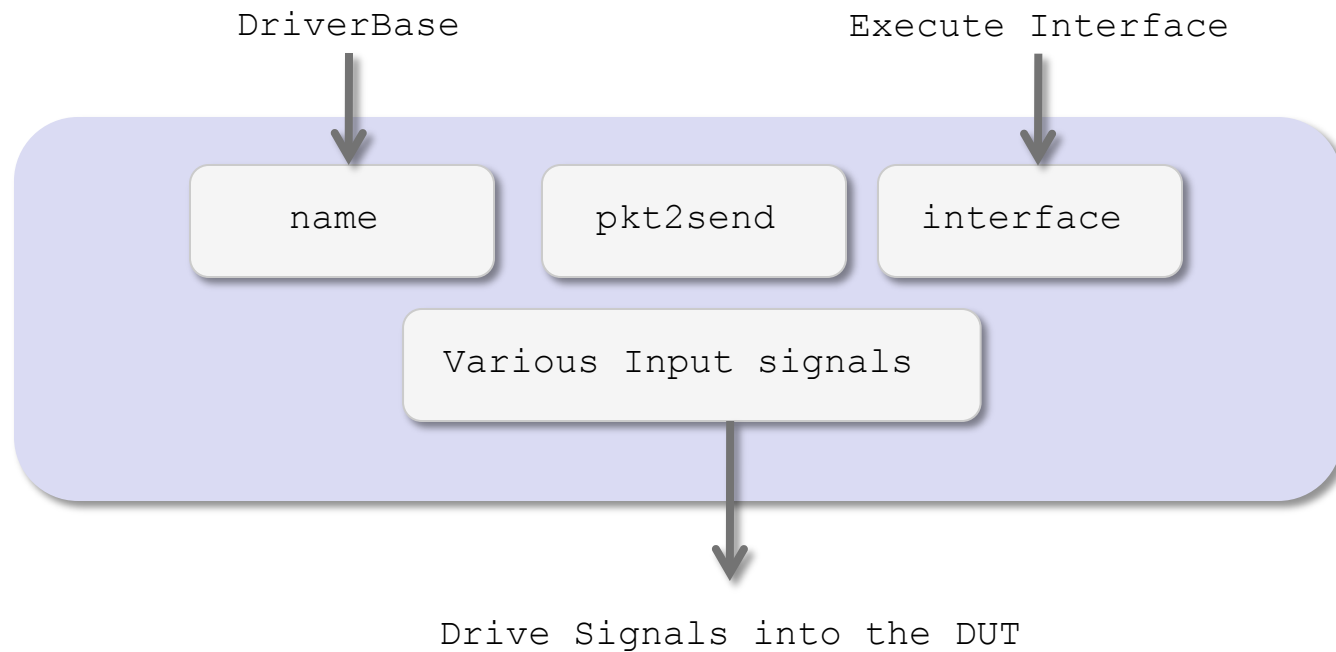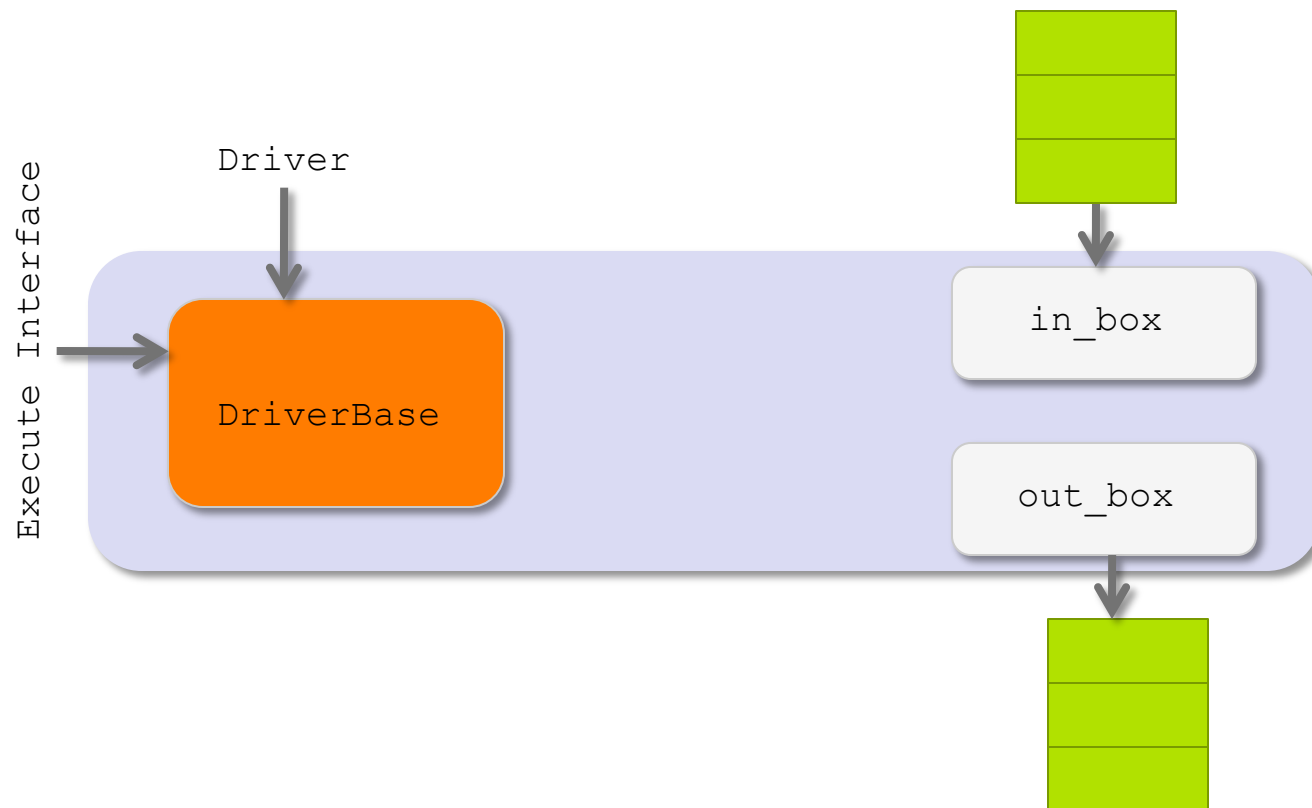
Driver

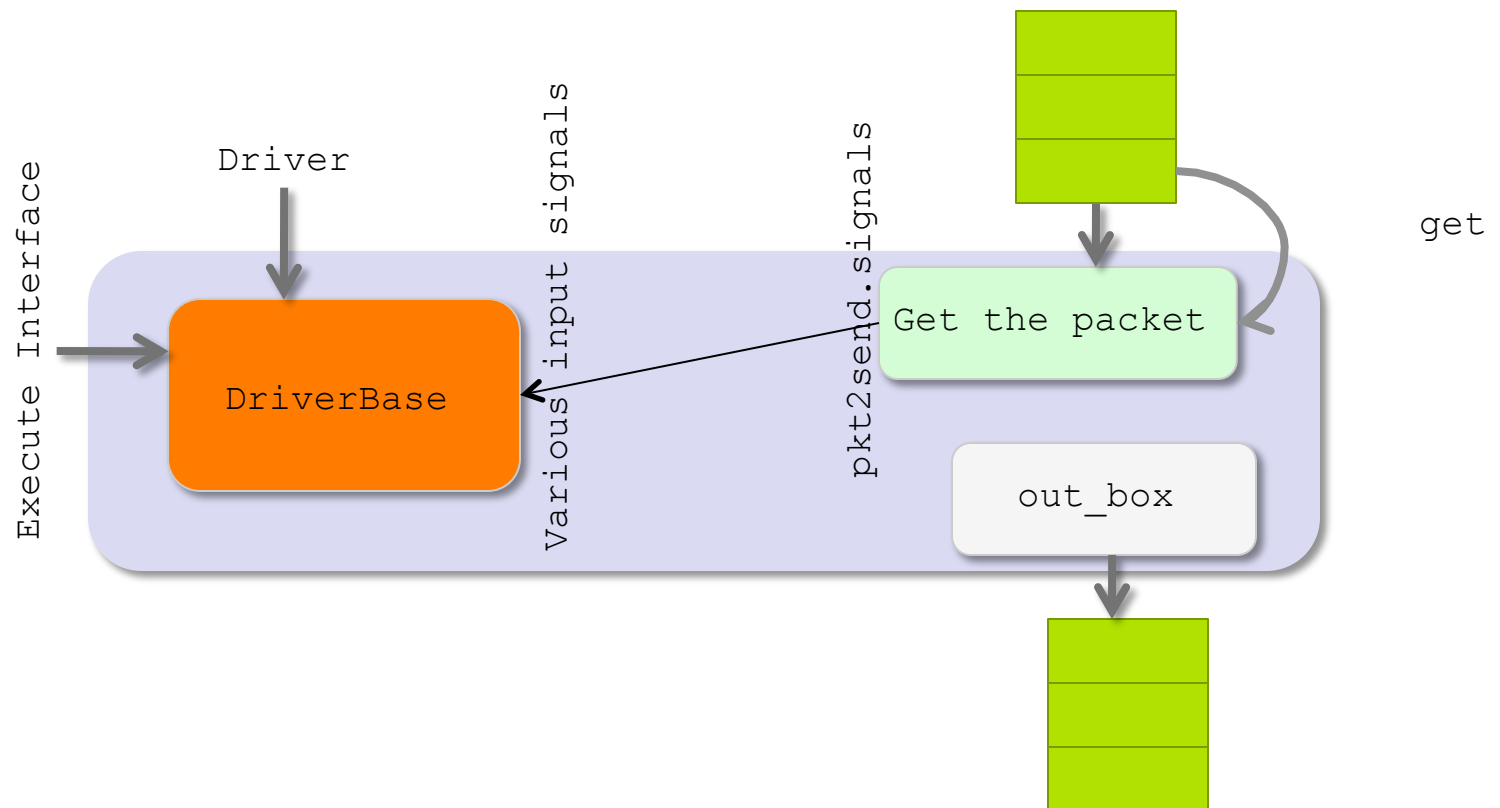- ## Driver extends DriverBase
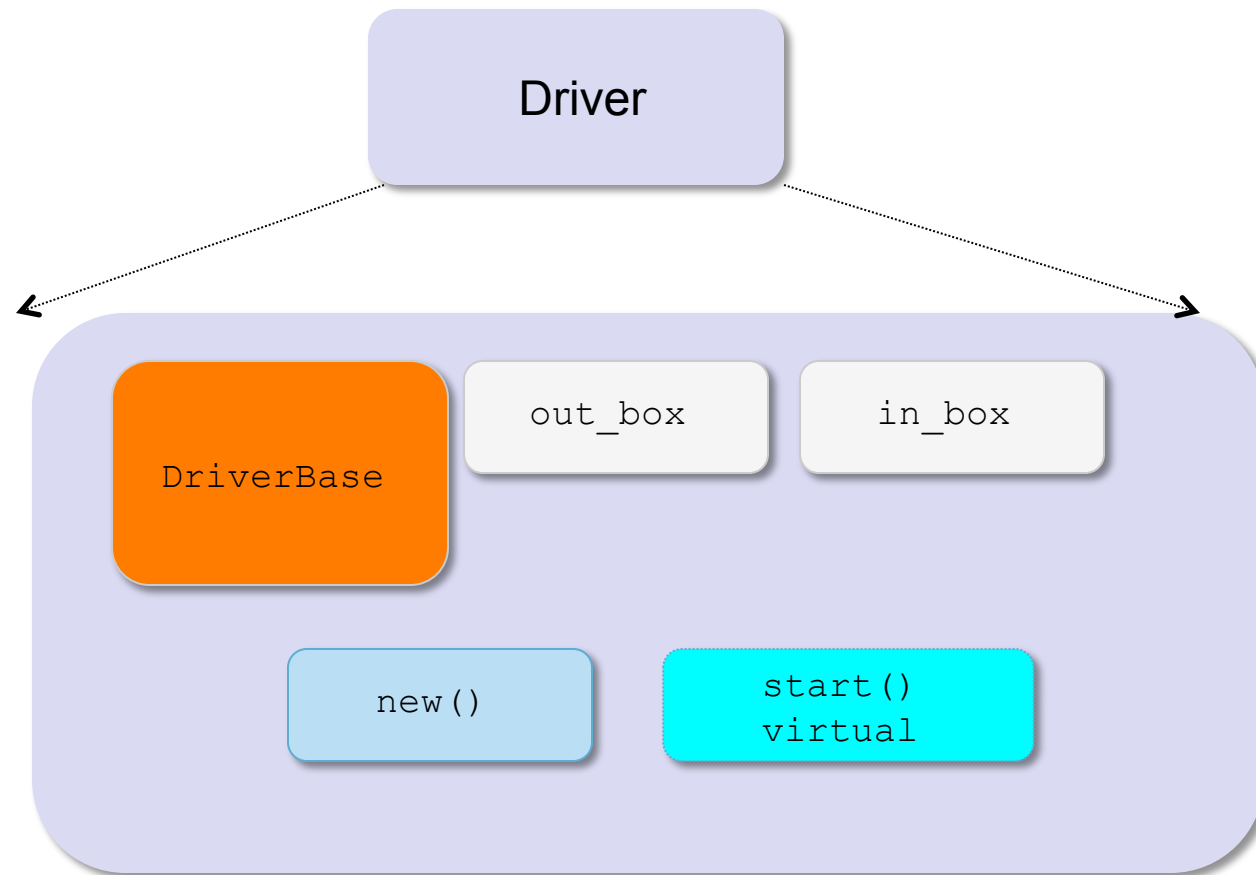
Driver

- Driver: new( )

Driver

- ## Driver: start( )

New Things in Lab

- ## Parameterized Mailboxes
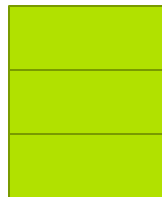 - To tie a mailbox to transfer only one kind of data

```
typedef mailbox #(Packet) in_box_type;
In_box_type in_box;
```

in_box

Can only transfer handles of
Packet class

New Things in Lab

- ## extern
  - Allows you to move method definitions outside the body of the class declarations

```
class Generator
…
extern function new (…)
…
endclass    in_box

function Generator::new(…)
```

New Things in Lab

- ## new ( ) function in the extended class
  - If the base class constructor has any arguments the constructor in the extended class must have a constructor and must call the base constructor on its first line

```
class Driver extends DriverBase
…

function Driver::new (string...)
   super.new(name, Execute);
```

New Things in Lab

- ## Virtual Interface
    - Virtual interfaces provide a mechanism for separating abstract models and test programs from the actual signals that make up the design.
    - A virtual interface allows the same subprogram to operate on different portions of a design, and to dynamically control the set of signals associated with the subprogram.
    - Instead of referring to the actual set of signals directly, users are able to manipulate a set of virtual signals.
    - Changes to the underlying design do not require the code using virtual interfaces to be re-written.
    - By abstracting the connectivity and functionality of a set of blocks, virtual interfaces promote code-reuse
    - Can be passed as arguments to task and functions

```
virtual Execute_io.TB Execute
```

New Things in Lab

- Virtual Interface
    - Virtual interfaces is a handle or a pointer to the physical interface
    - Must be initialized

Thank You