

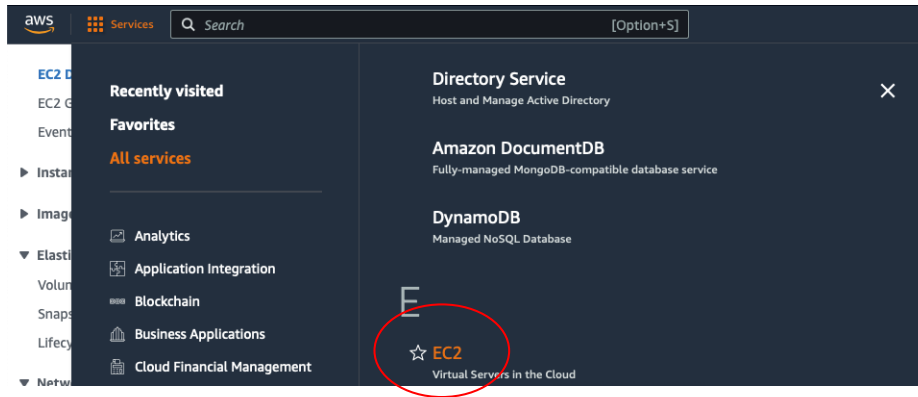
# CPU Run on AWS

Programming FPGAs for Economics:  
An Introduction to Electrical Engineering Economics

---

Bhagath Cheela, Alessandro Peri, André DeHon, Jesús Fernández-Villaverde

# Steps




1. **Log into your AWS account:**
2. Navigate to the Home Console
3. Select **EC2**
4. Launch Instance

## Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

**Launch instance**



**Migrate a server** 

# Steps: Name and tags

---

## Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

### Name and tags [Info](#)

Name

[Add additional tags](#)

# Select Amazon Linux Instance

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu®

Windows

Microsoft

Red Hat

Red Hat

SUSE LI

SUS

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-0eb5115914ccc4bc2 (64-bit (x86), uefi-preferred) / ami-0ca6dd60dbe07d5ad (64-bit (Arm), uefi)

Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible ▼

Description

Amazon Linux 2023 AMI 2023.3.20240304.0 x86\_64 HVM kernel-6.1

Architecture

64-bit (x86) ▼

Boot mode

uefi-preferred

AMI ID

ami-0eb5115914ccc4bc2

Verified provider

# Select CPU Instance

▼ Instance type [Info](#) | [Get advice](#)

Instance type

**m5n.large**

Family: m5n 2 vCPU 8 GiB Memory Current generation: true

On-Demand RHEL base pricing: 0.179 USD per Hour

On-Demand Linux base pricing: 0.119 USD per Hour

On-Demand Windows base pricing: 0.211 USD per Hour

☒ All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

Repeat the same for all other FPGA instances: **m5n.4xlarge**, **m5n.24xlarge**

# Key pair

## ▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

Proceed without a key pair (Not recommended)

Default value ▼



[Create new key pair](#)

For information on how to create a new key pair go [here](#)

# Launch m5n.large Instance

Repeat the same for all other CPU instances

▼ Summary

Number of instances

Info

1

Software Image (AMI)

FPGA Developer AMI

ami-02ab431c7b3297b00

Virtual server type (instance type)

z1d.2xlarge

Firewall (security group)

New security group

Storage (volumes)

2 volume(s) - 125 GiB











Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel

Launch Instance

Review commands

# EC2 Instances

Instances (1/2) <a href="#">Info</a>										<a href="#">Connect</a>
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>									Running ▼	
<input type="checkbox"/>	Name 	Instance ID	Instance state 	Instance type 	Status check	Alarm status	Availability Zone 	Public IPv4 DNS 		
<input type="checkbox"/>	3k-100-4	i-02362075ede6cd75e	✔ Running  	z1d.2xlarge	✔ 2/2 checks passed	<a href="#">View alarms</a> +	us-west-2a	ec2-54-188-219-255.us...		
<input checked="" type="checkbox"/>	cpu-run	i-00757b5113b27214b	✔ Running  	m5n.large	⌚ Initializing	<a href="#">View alarms</a> +	us-west-2a	ec2-35-88-29-46.us-we...		

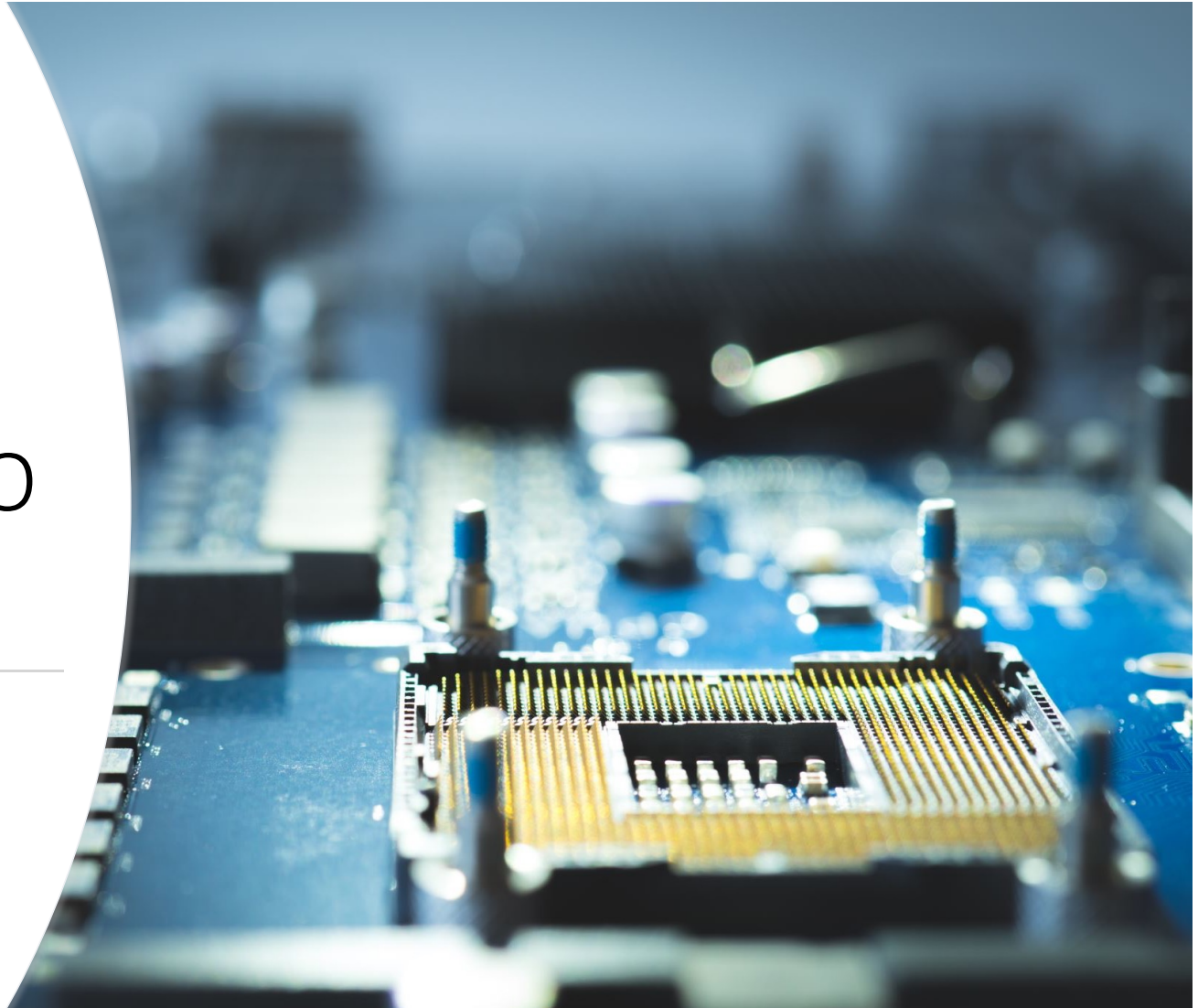
- In the top-left menu, select 'Instances'
- Copy the public IPv4 address in Visual Studio code





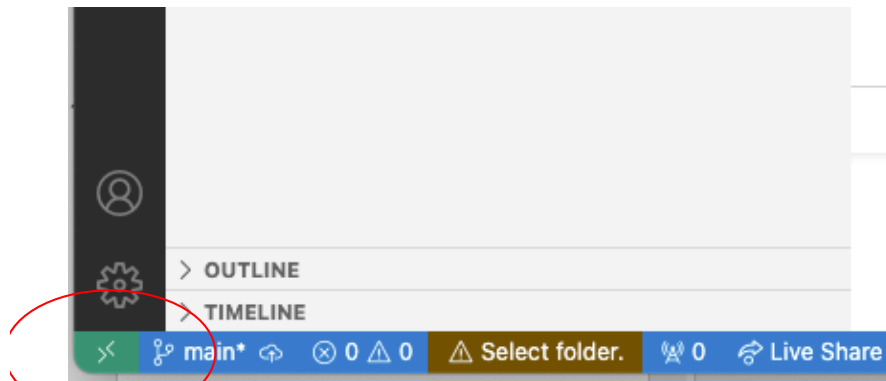
# VISUAL STUDIO CODE

---



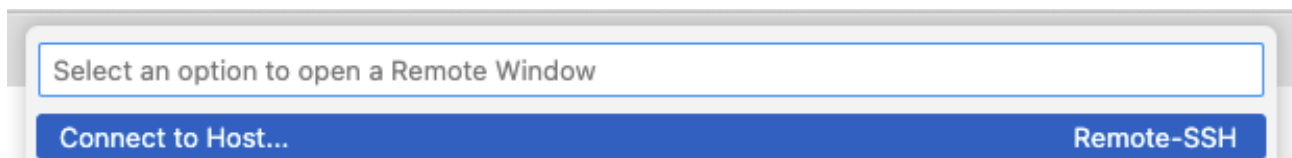
## Open a Remote Window

- On the bottom-left corner of Visual Studio Code click on the green button 'Open a Remote Window'
- Click on Connect to Host
- Click on Configure SSH Hosts.
  - Give a name to the Host (e.g. aws-ec-cpu)
  - Copy the public IP address (red circle) and set the path to your key
  - Set the user to **ec2-user**
- Connect to aws-ec2.



CLICK the GREEN BUTTON HERE

```
# ----- CONNECT TO AMAZON AWS
Host aws-ec-cpu
HostName ec2-34-219-191-114.us-west-2.compute.amazonaws.com
User ec2-user
IdentityFile ~/[YOUR KEY PATH].pem
# -----
```



# Setup the Instance

A solid orange horizontal bar is positioned below the title text.

# Install utilities and Copy Repo

- Initiate a terminal session on the AWS instance and run the subsequent script to install the utilities

```
sudo yum install git -y  
sudo yum install make -y  
sudo yum install tmux -y  
sudo yum install wget -y
```

- Clone our GitHub repository into a directory of your preference (e.g., /home/ec2-user):

```
git clone https://github.com/AleP83/FPGA-Econ.git
```

# AWS Configure

```
[centos@ip-10-0-1-68 ~]$ aws configure  
AWS Access Key ID [None]: █
```

1. Go to your aws account and set (one time thing):
  - AWS Access Key ID
  - AWS Secret Access Key
2. Go to the terminal in visual studio and type aws configure

```
aws configure  
AWS Access Key ID [*****xxxx]: <Your AWS Access Key ID>  
AWS Secret Access Key [*****xxxx]: <Your AWS Secret Access Key>  
Default region name: [us-west-1]: us-east-1  
Default output format [None]: json
```

3. Set:
  - AWS Access Key ID:
  - AWS Secret Access Key
  - Default Region name: us-west-2  
*Note: this depends in which region you launched your instance.*
  - Default output format [json]: json

# Modify the Makefile

- Set the AWS S3 Bucket Name. Specify the S3 bucket name by replacing S3-NAME-GOES-HERE

```
S3_EXE_BUCKET_NAME := S3-NAME-GOES-HERE
```

*Remark: The S3 bucket name must be globally unique within AWS. If an error occurs during bucket creation, it may be due to the name being already in use by another user.*

- Select the AWS region of the S3 bucket (default is us-west-2):

```
AWS_REGION := us-west-2
```

m5n.large

# Run all binaries in **m5n.large**

- Log into an **m5n.large** instance, previously set up following the instructions above
- **Initiate** `tmux` **terminal session**. To ensure your terminal session remains active throughout the potentially lengthy execution, initiate a terminal multiplexer session

```
tmux
```

- Execute all exercises in a single **m5n.large** instance

```
make cpu_results M5N=1x USE_AWS_S3_EXE=yes
```

## **Faster Replication Strategy.**

- Set up three separate **m5n.large** instances as described above.
- Run the following scripts from the terminal of each instance:

```
make cpu_results M5N=1xBATCH1 USE_AWS_S3_EXE=yes
```

```
make cpu_results M5N=1xBATCH2 USE_AWS_S3_EXE=yes
```

```
make cpu_results M5N=1xBATCH3 USE_AWS_S3_EXE=yes
```



m5n.4xlarge



# Run all binaries in **m5n.4xlarge**

- Log into an **m5n.4xlarge** instance, previously set up following the instructions above
- **Initiate `tmux` terminal session.** To ensure your terminal session remains active throughout the potentially lengthy execution, initiate a terminal multiplexer session

```
tmux
```

- Execute all exercises in a single **m5n.4xlarge** instance

```
make cpu_results M5N=4x USE_AWS_S3_EXE=yes
```

m5n.24xlarge



# Run all binaries in **m5n.24xlarge**

- Log into an **m5n.24xlarge** instance, previously set up following the instructions above
- **Initiate `tmux` terminal session.** To ensure your terminal session remains active throughout the potentially lengthy execution, initiate a terminal multiplexer session

```
tmux
```

- Execute all exercises in a single **m5n.24xlarge** instance

```
make cpu_results M5N=24x USE_AWS_S3_EXE=yes
```