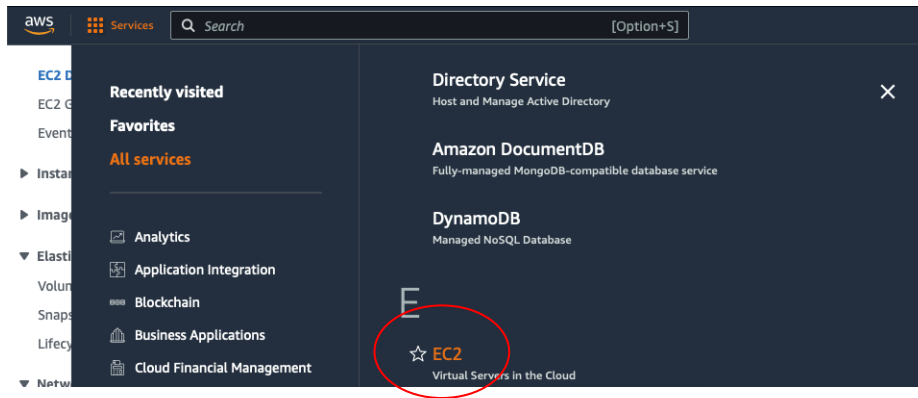


Run FPGA-design on AWS

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 FPGA Developer AMI: Browse more AMI

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

Launch an instance

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

Recents

Quick Start

Amazon Linux
aws


macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Li
SUS


Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Select FPGA Developer AMI

Quickstart AMIs (0)
Commonly used AMIs

My AMIs (0)
Created by me

AWS Marketplace AMIs (12)
AWS & trusted third-party AMIs

Community AMIs (1)
Published by anyone

▼ Refine results

Categories

[DevOps \(9\)](#)

[Infrastructure Software \(3\)](#)

▼ Publisher


☐ AMD Xilinx (9)

☐ Amazon Web Services (2)

☐ terracloudx (1)

FPGA developer AMI (12 results) showing 1 - 12

Sort By: Relevance ▼



FPGA Developer AMI

By [Amazon Web Services](#) | Ver 1.12.2

★★★★☆ 9 AWS reviews | [3 external reviews](#)

The FPGA (field programmable gate array) AMI is a supported and maintained CentOS Linux image provided by Amazon Web Services. The AMI is pre-built with FPGA development tools and run time tools required to develop and use custom FPGAs for hardware acceleration. The FPGA Developer AMI along with...

Select

Select: Subscribe on instance Launch.

Cancel

Subscribe on instance launch

Select FPGA Instance

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

Instance type

Launch an instance

f1.2xlarge

Family: f1 8 vCPU 122 GiB Memory Current generation: true

☒ All generations

[Compare instance types](#)

The AMI vendor recommends using a z1d.2xlarge instance (or larger) for the best experience with this product.

Repeat the same for all other FPGA instances: **f1.4xlarge**, **f1.16xlarge**

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 f1.2xlarge Instance

Repeat the same for all other FPGA 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

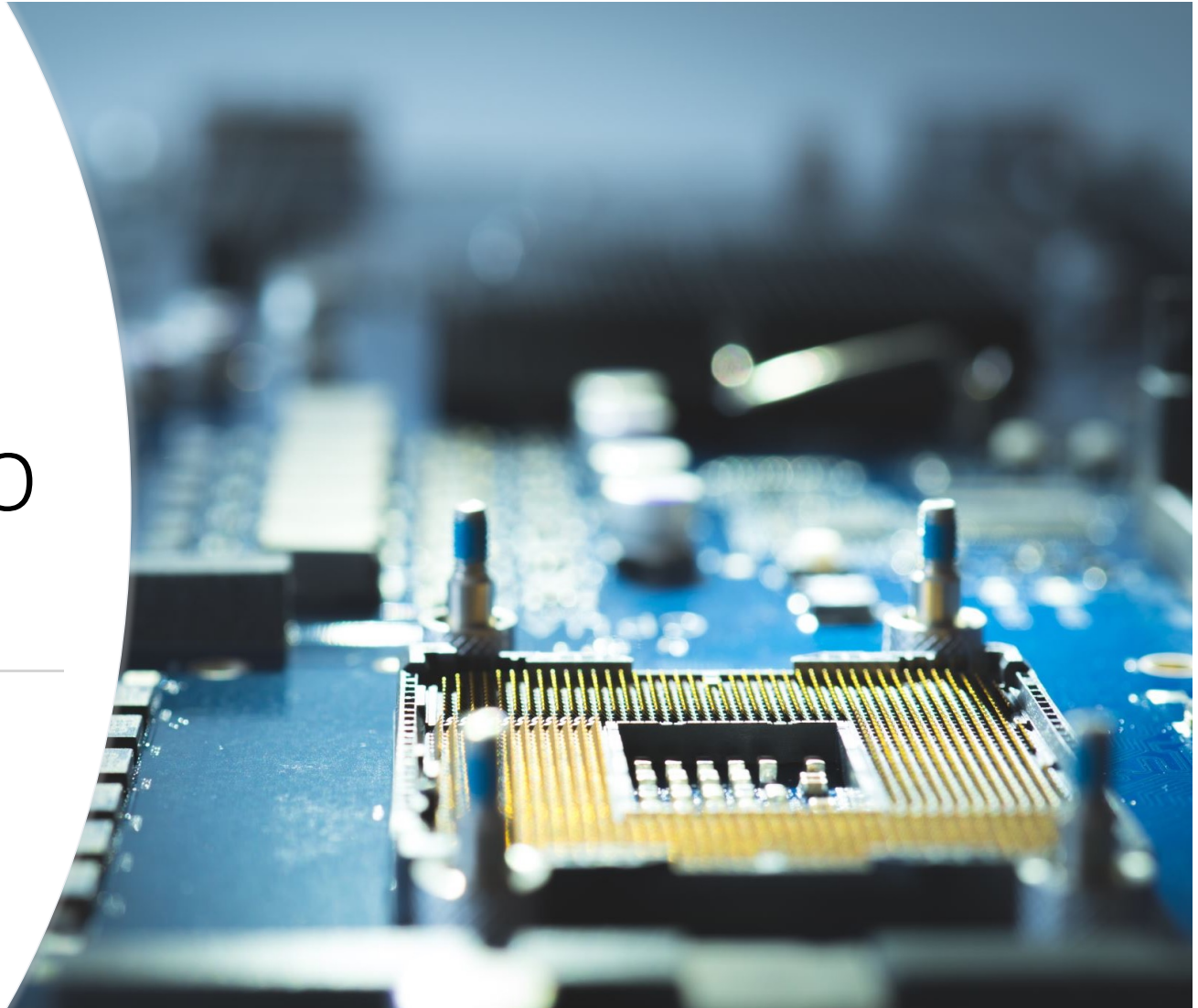
The screenshot displays the AWS Management Console interface for EC2 Instances. The left-hand navigation menu is visible, with the 'Instances' option highlighted by a red circle. The main panel shows the 'Instances (1)' view, featuring a search bar and a filter for 'Instance state = running'. Below this, a table lists the instance details:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
single-kernel-...	i-01a0f65274580349b	Running	z1d.2xlarge	Initializing	View alarms	us-west-2a	ec2-54-188-159-145.us-...

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

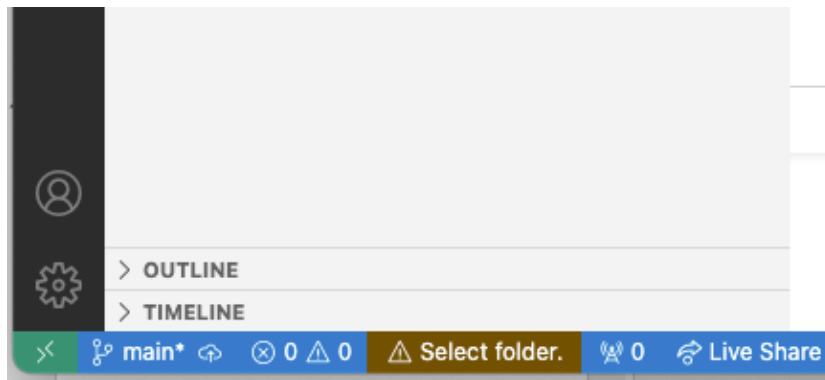


VISUAL STUDIO CODE

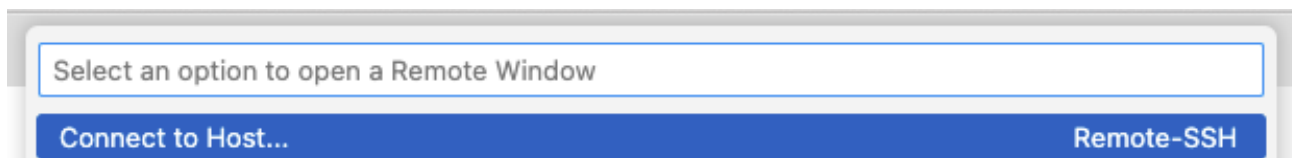


Open a Remote Window

- 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
- Copy the public IP address
- Connect to aws-ec2.
- If you receive an error try to set: User root (in place of User Centos)



```
# ----- CONNECT TO AMAZON AWS
Host aws-ec2
HostName ec2-54-188-159-145.us-west-2.compute.amazonaws.com
User centos
IdentityFile ~/[YOUR KEY PATH].pem
# -----
```



Execute on f1 Instance



Steps

1. Log into your instance. Clone GitHub repository from the terminal of your instance. Alternatively, you can drag and drop the folder code from your local machine to the visual studio code left-panel on your instance.

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

2. Set the AWS credentials

```
aws configure
```

```
$ aws configure
AWS Access Key ID [*****]: <Your AWS Access Key ID>
AWS Secret Access Key [*****]: <Your AWS Secret Access Key>
Default region name: us-west-2
Default output format: json
```

Steps

3. Modify `code/Makefile` to select the AWS region of the S3-bucket (default is us-west-2)

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

4. Modify `code/common/util/generate_fpga_results.sh` to select the AWS region of the S3-bucket (default is us-west-2):

```
AWS_REGION="us-west-2"
```

Steps

5. Initiate tmux terminal session

```
tmux
```

6.1. On an **f1.2xlarge** instance, execute:

```
make fpga_results TABLE=all USE_AWS_S3_EXE=yes
```

6.2. On an **f1.4xlarge** or **f1.16xlarge** instance, execute:

```
make fpga_results TABLE=3 USE_AWS_S3_EXE=yes
```