

Parallel Calculations on Raspberry Pi

Orest Babych

The faculty of electronics and computer technologies
Ivan Franko National University of Lviv
Lviv, Ukraine
e-mail: orestbabych@gmail.com

Oleh Chuprynskyi

The faculty of electronics and computer technologies
Ivan Franko National University of Lviv
Lviv, Ukraine

Vladyslav Galin

The faculty of electronics and computer technologies
Ivan Franko National University of Lviv
Lviv, Ukraine

Abstract—The article deals with the usage of Raspberry Pi as a low-cost computing cluster in order to decrease the cost of the energy consumption and the power of a computer. The example of the cluster is built on two processors from Raspberry Pi 3 and the Python 2 programming environment is used.

Index Terms—Raspberry Pi, parallel programming, cluster, cluster calculations

I. INTRODUCTION

The main costs in the field of computer computing are the power of the device and its energy consumption. Modern supercomputers are considerably large and consume hundreds of thousands of watts.

A major problem in this case is the process of learning supercomputers for parallel programming and execution of calculations on them. Moreover, the management of this type of computers becomes more complex as there is usually no direct access to them.

In order to solve this problem, in this paper, it is proposed to use the Raspberry Pi single-circuit microcomputers and their analogues. Based on these devices, you can build a low-cost computing cluster and perform the process of learning the basics of parallel programming. Thus, the purpose of this work is to create a low-cost training cluster from microcomputers for the development and implementation of parallel programming algorithms in the learning process. In this scientific article, an example of parallel computations is presented in the developed learning cluster.

Raspberry Pi – a microcomputer developed by Raspberry Pi Foundation. Small, with the size of a bank card, it is a full-fledged single-payment computer (System-on-a-Chip). Processor (in model PI 3): 4 cores ARM Cortex-A53 x64. Operating system (default): Raspberian (based on Linux kernel). With a total of \$ 35, the board contains all the necessary interfaces (Wi-Fi, Bluetooth, Usb, Ethernet), as well as a large set of ready-made programs for any kind of activity [1].

The above arguments were the reason for the selection of microcomputer data to create a small learning computing cluster.

II. THE CONCEPT OF A CLUSTER AND CLUSTER CALCULATIONS

Many people are familiar with the term “Cloud Computing”. Cloud computing generally refers to a service provided by an individual or a company by connecting through the Internet to a cluster of computers.

A cluster is a group of computers that are united by high-speed communication channels, which appears to be the only hardware resource [2]. On the other hand, the cluster is a weakly connected set of multiple computing systems that work together to execute related software applications [3]. Amazon’s High Performance Computing (HPC) services are an example of the cluster computing concept.

Using cluster computers can exponentially increase processing speed and processing power. For example, Figure 1 illustrates a typical cluster computer configuration consists of 6 computers, where the server communicates and schedules the workload to the five compute nodes for processing. With the use of specially designed software, all the nodes in the cluster act as a single computer to the outside world [3].

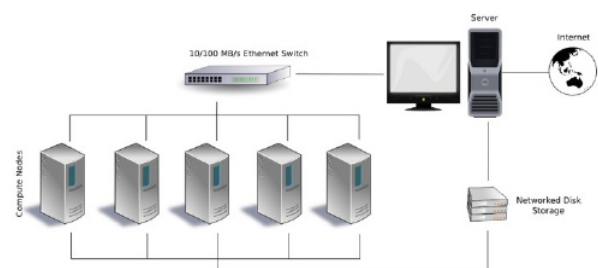


Figure 1. A typical cluster computer configuration

In order to bind a few Raspberry PI into a cluster, a typical cluster computing system (router, Ethernet, USB cables, etc.) was collected based on the PI 3 processor (Figure 2).

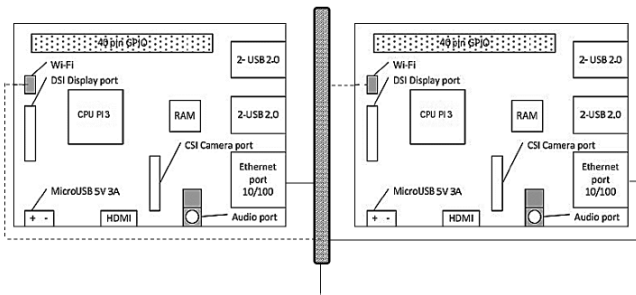


Figure 2. Cluster computing system based on two processors from Raspberry PI 3

III. DEMONSTRATION OF PARALLEL COMPUTING

For a clear demonstration of the cluster capabilities of the two Raspberry PI 3, the Python 2 programming environment was implemented and the algorithm for sorting the array by merging was implemented. The computers were connected to the local network. To simplify the compilation of a cluster from several Raspberry PI there is plenty of implemented applications, one of which is "mpi4py" [2].

The program code for sorting an array by merging in Python looks like this:

```
import time
def merge(left,right): #merges 2 sorted lists
together
    result = []
    i, j = 0, 0
    #Goes through both lists
    while i < len(left) and j < len(right):
    #Adds smaller element of the lists to the final list
    if left[i] <= right[j]:
    result.append(left[i])
    i += 1
    else:
    result.append(right[j])
    j += 1
    result += left[i:]
    result += right[j:]
    return result
def mergesort(lst):
    #if there's only 1 element, no need to sort
    if len(lst) < 2:
    return lst
    #breaks down list into 2 halves
    middle = len(lst)/ 2
    #recursively splits and sorts each half
    left = mergesort(lst[:middle])
    right = mergesort(lst[middle:])
    #merges both sorted lists together
    return merge(left, right)
```

The algorithm of the program consists of the following sequence of actions:

1. PI 3 (server) generates a random array of numbers.
2. This array is divided into n parts, by the number of processors in the local network.
3. Using the socket module and the local network PI 3 (server) transfers part of the array PI 3 (client).
4. PI 3 (server) sorts its part of the array and waits for a response from PI 3 (client).
5. PI 3 (client) sorts its part of the array and passes it to PI3 (server).
6. PI 3 (server) receives the sorted part of the array and performs a final sort.

Calculations have shown that for sorting an array of 500 thousand items for one PI 3 it takes about 23 seconds. After adding the second PI 3, this time decreased to 16 seconds. The increase in speed is nonlinear, but the more in the cluster of computers, the less time will be spent.

IV. CONCLUSION

Single-payment computers only recently went beyond the segment of devices to automate production and began to gain mass market. Their small size, low energy consumption and high computing capabilities can make them the basis for implementing a variety of projects, for example, parallel programming. The feature presented by the cluster computing system based on Raspberry PI 3 is a good scale, which is determined by the possibilities of switching equipment, low cost, the possibility of using free software, which is important in application in the learning process. The demonstration work performed shows that the cluster, even with two PI 3s, can accelerate the calculation of a simple but simultaneously bulk task, such as sorting a large data array.

In the future, it is planned to increase the number of micro- computers in the computer system and compare the performance of cryptographic algorithms, in particular those that will be used to encrypt / decrypt large images and transfer them over the Internet.

REFERENCES:

- [1] Robert Mullins/ Distributed computed //University Cambridge. – 2012. –URL: <http://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/distributed-computing/>.
- [2] Cluster. – Access mode. – URL: <https://ru.wikipedia.org/wiki/Cluster>
- [3] Lukin V.V., Marchevsky I.K. Educational-experimental computing cluster. Ch. 1. Instrumentation and capabilities. - Access mode. – URL: https://elibrary.ru/download/elibrary_17091004_33209664.pdf
- [4] Raspberry Pi. <https://www.raspberrypi.org>