

**IMPLEMENTATION OF ASSEMBLY SEQUENCE BY ARDUINO AND
MONITORING SYSTEM USING NI MYRIO**

WONG CHOON CHEN

Thesis submitted as partial fulfillment of
the requirements for the award of the
B.Engineering (Hons.) Mechatronics Engineering

FACULTY OF MANUFACTURING ENGINEERING

UNIVERSITI MALAYSIA PAHANG

JUNE 2015

Abstract

Nowadays, microcontroller is widely used in most of the industries especially in assembly line because of its advantage in assembly sequence and the easiness to control it. However, the user needs to control the assembly sequence of the production in order to have faster respond that control by the controller. Hence, the assembly sequence of the production can be implement by using the microcontroller that able to control the motor or even can control without workmanship. Besides that, the whole assembly process can be monitor by using the latest microprocessor, with higher processing rate that able to let the image captured from the camera more clear and precise. The user able to control the position of the camera, capture and record the data during the assembly process by using the high technology in this new globalization. To be more advance ,the user also able to save and play the recorded image/video if they choose the options given inside the interface (GUI) and can choose the program that help to analyze the products.

Abstrak

Pada masa kini, pengawal mikro digunakan secara meluas dalam kebanyakan industri terutama di bahagian pemasangan kerana kelebihan dalam urutan pemasangan dan kegampangan untuk mengawalinya. Walau bagaimanapun, pengguna perlu mengawal urutan jemaah pengeluaran untuk mempunyai lebih cepat bertindak balas kawalan yang oleh pengawal. Oleh itu, urutan jemaah pengeluaran boleh melaksanakan dengan menggunakan pengawal mikro yang dapat mengawal motor atau boleh mengawal tanpa mutu kerja. Selain itu, proses pemasangan keseluruhan boleh monitor dengan menggunakan mikropemproses terkini, dengan kadar pemprosesan yang lebih tinggi yang dapat membiarkan imej yang ditangkap dari kamera yang lebih jelas dan tepat. Pengguna dapat mengawal kedudukan kamera, menangkap dan merekodkan data semasa proses pemasangan dengan menggunakan teknologi tinggi dalam globalisasi baru ini. Untuk menjadi lebih maju, pengguna juga boleh menyimpan dan memainkan imej / video yang dirakam jika mereka memilih pilihan yang diberikan dalam antara muka (GUI) dan boleh memilih program yang membantu untuk menganalisis produk.

TABLE OF CONTENTS

	Page
EXAMINER'S DECLARATION	i
SUPERVISOR'S DECLARATION	ii
STUDENT'S DECLARATION	iii
ACKNOWLEDGEMENT	iv
ABSTRACT	v
ABSTRAK	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	x
LIST OF FIGURES	xi
LIST OF SYMBOLS	xii
LIST OF ABBREVIATIONS	xiii
CHAPTER 1 INTRODUCTION	
1.1 Background of study	1
1.2 Problem Statement	2
1.3 Objectives	2
1.4 Scope of project	2
CHAPTER 2 LITERATURE REVIEW	
2.1 Designing Real and Fast system by NI Myrio	4
2.2 Control Motor by Arduino	6
2.3 NI myrio applications	
2.4 Applying Automatically Control Sequence Using Programmable Logic Circuir(PLC)	8
CHAPTER 3 METHODOLOGY	10
3.1 Project Flowchart	10
3.2 Electrical circuit parts	13

3.2.1	Microprocessor- NI Myrio	13
3.2.2	Electrical Connection Using Arduino	14
3.2.3	L298 Motor Driver connection	18
3.2.4	HC-05 Bluetooth Module connection	19
3.2.5	YL-63 Infrared Module Connection	20
3.2.6	Liquid Crystal Display(LCD Display) connection	21
3.3	Software Designed Parts	21
3.3.1	Labview 2013	21
3.3.2	Arduino Software and Codings	24
3.3.3	Graphical User Interface (GUI)	25
CHAPTER 4	RESULTS AND DISCUSSION	28
4.1	Implementation of Object Detection	29
4.2	Implementation of Assembly Sequences by User	31
4.3	NI Myrio GUI Parts	32
4.4	Overall Hardware Designed	36
4.5	Discussion	37
CHAPTER 5	CONCLUSION AND RECOMMENDATION	
5.1	Conclusion	39
5.2	Recommendation	40
REFERENCES		41
APPENDICES		42
A	Arduino Coding for Assembly Sequences	42
B	Overall Datasheet	56
C	Gantt Chart	64

LIST OF TABLES

Table No.	Title	Page
4.1	Data recorded for delay time of Arduino and NI Myrio	33

LIST OF FIGURES

Figure No.	Title	Page
2.3	AT-mega 328 schematic diagram	6
2.3.1	Sketch of Transmission of Assembly Line	8
3.1.1	Flow Chart of Project	11
3.1.2	Integration of electrical parts and mechanical parts	12
3.3	NI myRIO- I900 assembly drawing	13
3.4	Overall electrical connection for Arduino Mega	14
3.5	NI myRIO- I900 Hardware Block Diagram	15
3.6	Arduino Mega 2560 top view	17
3.7	L298 motor driver	19
3.8	HC-05 Bluetooth Module	20
3.9	YL-63 Infrared Module	21
3.10	LCD Display	22
3.11	“Getting Started “ window	23
3.12	Front Panel control by users	24
3.13	Block Diagram for the Processing purpose	24
3.14	Monitoring platform using NI Myrio	25
3.15	Arduino Software Icon	25
3.16	Arduino coding for implement assembly sequences	26
3.17	Visual Studio Icon	27
3.18	Control form of GUI	27

Figure No.	Title	Page
4.1	GUI for the initial status	29
4.2	GUI when object placed	29
4.3	GUI for user select assembly sequences	30
4.4	Assembly sequence for '1423'	30
4.5	Labview's Interface for control position of camera	31
4.6	Processing rate for Arduino Mega	32
4.7	Processing rate for NI Myrio	32
4.8	Graph of data comparison between arduino and NI Myrio for delay time	33
4.9	Monitoring system using NI Myrio (Normal situation)	34
4.10	Monitoring system using NI Myrio (Edges detection)	35
4.11	Electrical circuit for the Arduino Part(i)	35
4.12	Electrical circuit for the Arduino Part(ii)	36
4.13	Overall Integration of NI Myrio and Arduino	36

LIST OF SYMBOLS

A	Ampere
Hz	Hertz
k	Kilo
t	Time(s)
Ton	Time on
Toff	Time off
Ω	Ohm
°	Degree
%	Percent
n	nano
V	Voltage
Amp	Amplitude

LIST OF ABBREVIATIONS

AC	Alternating current
DC	Direct current
IC	Integrated circuit
GUI	Graphical user interface
LCD	Liquid crystal display
PIC	Peripheral interface controller
DIO	Digital Input/Output
GND	Ground

CHAPTER 1

Introduction

This chapter mainly focuses about the study background of project, problem statement, objectives and the scope of work to be done in this project.

1.1 STUDY BACKGROUND OF PROJECT

An important factor in industrial progress during the past five decades has been the increasing sophistication of factory automation which has improved productivity in manifold. Prior to the 1950s major of factory applications required the use of implement the correct assembly product sequence since the old technology that carry out the assembly sequence were not capable of smoothly varying speed since they inherently operated synchronously or nearly synchronously with the input from the worker. However, to a large extent, these applications are now serviced by what can be called intelligent system that can automatically recognize and implement the correct assembly sequence for the product. In general, such intelligent system often feature a cost advantage over their assembly line, in addition, offer lower maintenance, and improved reliability for implement the assembly sequence during assemble the products.

Nowadays, microcontroller is widely used in most of the industries especially in assembly line because of its advantage in assembly sequence and the easiness to control it. However, the user needs to control the assembly sequence of the production in order to have faster respond that control by the controller. Hence, the assembly sequence of the production can be implement by using the microcontroller that able to control the motor. This can be done by using Arduino Mega that able to control the motor so that it can help to reduce the waste of workmanship . Besides that, the whole assembly process can be monitor by using the latest microprocessor, National Instrument (NI) Myrio. The user able to control the position of the

camera, capture and record the data during the assembly process. It also able to save and play the recorded image/video if user choose the options given inside the interface (GUI).

The purpose of this project is to use Arduino Mega as a microcontroller to control the assembly sequence when the object is detected in an assembly line. Then, the user able to monitor the whole assembly process by using NI Myrio as a processor to implement the integration of different electrical components. The user can monitor the image/video of the process inside the interface, and control the position of the camera manually. Which mean that the user can adjust the camera to the position that they want. Finally, the result of the project of the assembly sequence and the monitoring system will be discussed.

1.2 PROBLEM STATEMENT

1. Conventional assembly system in industries has less durability and waste power of workmanship
2. Conventional processing rate of monitoring system has less precise response
3. Conventional assembly system causes instability when the assembly sequence change at a fast rate

1.3 OBJECTIVES

The objectives of this project are:

1. To implement the assembly sequence by using Arduino
2. To develop basic assembly platform by using Arduino and GUI for simulation
3. To analyze and compare the processing rate of NI MyRio with the Arduino
4. To implement the assembly sequence into basic conveyor system

1.4 SCOPE OF PROJECT

1. Interfacing NI Myrio with Labview
2. Interfacing Arduino with Visual Studio
3. Integrated all the controller and sensors in conveyor system
4. Monitor the assembly process using camera that control by NI Myrio
5. Arduino control the stepper motor with L298 and send signal to GUI,GUI will respond to the signal and resend the signal to Arduino to implement the required sequences

Chapter 2

Literature Review

Introduction

This chapter present about the research and recommendation for this project. By doing some researches from the internet, by asking friends and lecturers about this project, it helps in understanding the concept and idea to construct the project and also improving the previous technologies used. The concept of this project which is about assembly using Myrio is being discussed in this chapter.

2.1 Case Study

Designing Real and Fast System

Over the winter holiday, University of Virginia students can enroll in two-week courses (called JTERM) that equate to a three-hour semester course. Students voluntarily enroll and can choose a topic that interests them. One option is a course on embedded programming with LabVIEW that provides a contrast to the traditional C embedded course that the students have previously taken. Using LabVIEW, students can appreciate the unique benefits of a graphical programming language and challenge themselves to see what they can accomplish in a short amount of time.

They designed the course as a pilot to test the viability of NI myRIO and LabVIEW for an experimental full-semester, model-based course. One goal was to see how well a small group of students that had never seen LabVIEW could learn the environment and develop a workable

project. They chose LabVIEW and NI myRIO because the flexible platform helped us get projects going in a short time frame. Also, they wanted to expose the students to a new programming paradigm that could broaden the undergraduate engineering experience.

In addition to the LabVIEW Core materials, they worked some of their own short answer quizzes into the class, which provided frequent breaks. This approach also helped in spot areas which the students lacked understanding.

After completing a basic LabVIEW introduction, students worked in teams to build a smart four-way stop sign in LabVIEW on their laptops. This project referenced an infamous four-way stop at the edge of their campus where nobody obeys the traffic rules. The program used parallel loops and a first in, first out (FIFO) method to feed simulated traffic data to a separate loop that ran a state machine to control the simulated lights.

Once the students had the program running on a laptop, they moved it to an NI myRIO device. The students were impressed with how well they could move between the two environments—desktop Windows and an embedded real-time OS. One of the more ambitious groups used the accelerometers on the NI myRIO to insert “cars” into the queues for each direction and appreciated how well the hardware could be merged into an existing program without requiring major changes.

After this, students took some of the projects that they had previously done in Introduction to Embedded course and repeated them on the NI myRIO device. They then chose several projects at random to implement from the high-level perspective of LabVIEW, after having seen the same ones at an extremely low “bit-banging” perspective on the Launchpad. One involved reading a rotary encoder and displaying its count on a seven-segment display. Students also did a control theory exercise based on pulse-width modulation control of a small DC motor and position feedback.

Conclusion

In just two weeks, students learned a new programming language, paired it with powerful embedded hardware, and completed meaningful, sophisticated projects. We successfully evaluated and validated this approach for use in future courses, and most importantly, equipped

our students with knowledge that is already benefitting them in subsequent classes. We are excited about the possibilities with LabVIEW and NI myRIO and look forward to continuing to use these tools in our undergraduate and graduate curriculum.

2.2 Micro-controller

Control Motor by Arduino

The previous project used Arduino to control the system, however, it only applied the use of controller, but they use workmanship to implement the assembly product sequence. Arduino is a microcontroller are integrated circuit which include SRAM and EEPROM which all of these are the main components of a microprocessor. The entire component is integrated in a single microcontroller so that it can save space and easy construct the circuit. The microcontroller is low in cost, has many manuals or references to refer and easy to be programmed.

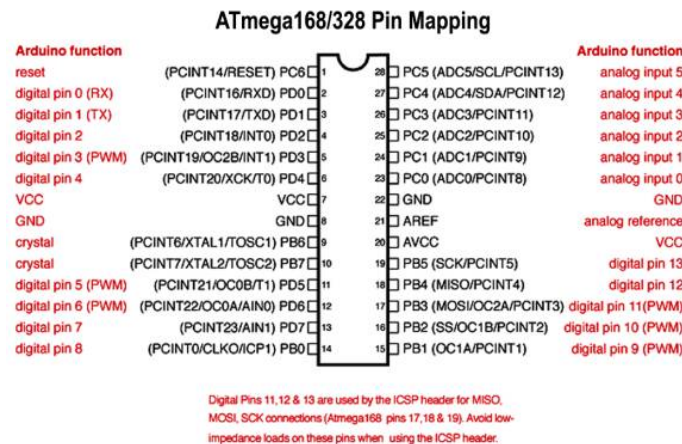


Figure 2.3 ATmega 328 schematic diagram

2.3 NI Myrio applications

The NI Myrio embedded student design device was created for students to ‘do real-world engineering’. It features a 667 MHz dual-core ARM Cortex-A9 programmable processor

and a customizable Xilinx field-programmable gate array (FPGA) that students can use to start developing systems and solve complicated design problems faster. The NI Myrio device features the Zynq-7010 All Programmable system on a chip (Soc) to unleash the power of NI LabVIEW system software both in a real-time (RT) application and on the FPGA level. Students can use the LabVIEW graphical programming paradigm to focus on constructing their systems and solving their design problems without the added pressure of a burdensome tool.

NI myRIO is a reconfigurable and reusable teaching tool that helps students learn a wide variety of engineering concepts as well as complete design projects. The RT and FPGA capabilities along with onboard memory and built-in WiFi allow students to deploy applications remotely and run them “heedlessly” (without a remote computer connection). Three connectors (two NI myRIO expansion ports [MXP] and one NI miniSystems port [MSP] that is identical to the NI myDAQ connector) send and receive signals from sensors and circuitry that students need in their systems. Forty digital I/O lines overall with support for SPI, PWM out, quadrature encoder input, UART, and I²C; eight single-ended analog inputs; two differential analog inputs; four single-ended analog outputs; and two ground-referenced analog outputs allow for connectivity to countless sensors and devices and programmatic control of systems. All of this functionality is built in and preconfigured in the default FPGA functionality, which eliminates the need for expansion boards or “shields” to add utility. Ultimately, these features allow students to do real-world engineering right now – from radio-controlling vehicles to creating stand-alone medical devices.

Introduction to LabVIEW

LabVIEW is a graphical programming environment that students can use to quickly develop applications that scale across multiple platforms and OSs. The power of LabVIEW is in its ability to interface with thousands of devices and instruments using hundreds of built-in libraries and prebuilt VIs to help you accelerate development time and quickly acquire, analyze, and present data.

Applications in LabVIEW mimic the appearance of real instruments (like multimeters, signal generators, or oscilloscopes), so they are called virtual instruments or Vis. Every LabVIEW application has a front panel, an icon/connector pane, and a block diagram. The front panel serves as the imitation of the real-world user interface of the device that the VI is defining. Programmers can leverage the flexibility of using multiple forms of representation for the data the instrument is analyzing. The icon/connector pane is analogous to terminals or plugs on a real-world instrument that allow it to be connected to other devices. Therefore, Vis can contain other Vis (called subVIs) that are all connected, and in turn, each of those subVIs can contain more Vis similar to function calls in a text-based programming language such as C, Java, C++, and Visual Basic, LabVIEW uses icons instead of lines of text to create applications. Due to this key difference, execution control is handled by a set of rules for data flow rather than sequentially. Wires connecting the nodes and Vis on the block diagram determine code execution order.

In summary, LabVIEW Vis are graphical, driven by dataflow and event-based programming, and are multitarget and multiplatform capable. They also have object-oriented flexibility and multithreading and parallelism features. LabVIEW VIs can be deployed to real-time and FPGA targets.

2.4 Applying Automatically Control Sequence Using Programmable Logic Circuit(PLC)



Figure 2.4.1 Sketch of Transmission of Assembly Line

The assembly line installed in major industries is suitable for completely assembling and measuring time taken for assembly the complete product in a short time. There are many operators working simultaneously at the stations of the assembly line because they need to monitor the assembly sequences for assembly the product and make sure there is no error occur during the assembly time. The assembly line consists of two main units: from the pre-assembly stations and the final assembly stations that are fully control by the operators. The assembly of the synchronous units is supervised by an intelligent industrial camera system (Machine Vision System). At each assembly station there is a touch-screen industrial PC terminal, on which the assembly instructions, the measurement results and the statistics can be read and be analyze by using image processing technique and decide the correct the correct assembly sequence and make sure the product assembled is in good condition. The controlled line supervises all the whole assembly line processes, and it will informs the operator about the assembly task that need to be done. About the measurement values that analyses and created by the PC, the appropriate quality of the execution will be determine based on the algorithm that had been set inside the operation. The PLC for sure is for controlling the system collects the

data, and the supervising IPC stores all the data and keep it for the reference if there is any error occur.

Conclusions From Review

It can be concluded from the review that the suitable components for the project is chosen which is using Arduino as a microcontroller to control the whole assembly sequence whereas NI Myrio will us as a processor to improve the processing rate of the camera and help in monitor the assembly system . The component are decided based on the capability, price ,ability ,accuracy and compatibility of the components to be applied in the system. Although the microcontroller and the software used are different compared to the PLC ,but the purpose is for improving the assembly system, especially the assembly sequence that need to decide by the worker in last few years. It is decided to use the Arduino and NI Myrio based on its advantages on the system , its faster processing rate and it is latest technology in 2014.

Chapter 3

Methodology

Introduction

In this chapter, I will describes the methodology on the way to develop a system that can used to implement the assembly sequence in a assembly system using Arduino and monitor the system using NI Myrio . This project consists of three important parts, which is input from the objects when it is detected in a assembly system, process and response from the microcontroller (Arduino) and lastly is the output that come from the motor that control by the Arduino .

3.1 PROJECT FLOWCHART

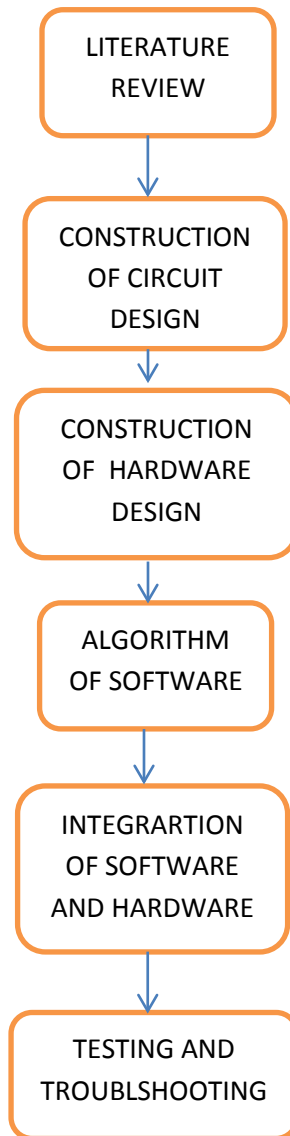


Figure 3.1.1 Flow Chart of Project

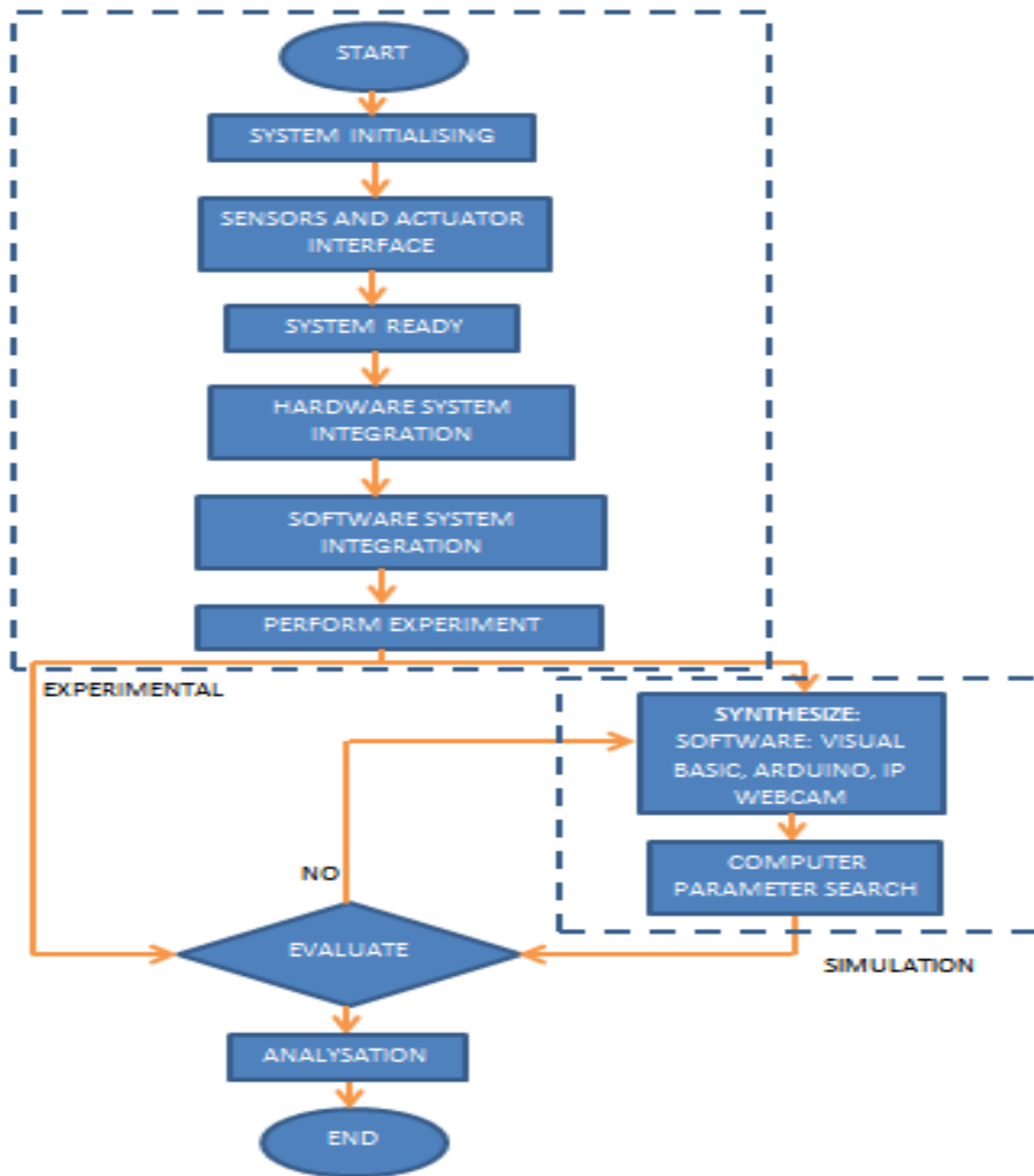


Figure 3.1.2 Integration of electrical parts and mechanical parts

From Figure 3.1.2, the project start with gathering information that related to this assembly sequence for assembly the product using Arduino through literature review. The electrical circuits and the mechanical parts were designed for this assembly purpose and the conveyor system is fabricated to control the stepper motor. After the fabrication, the hardware and the microcontroller is assembly together by using the specific program algorithm so that it can integrated together. Result in the control of motor using L298 motor driver. The user can

select the required assembly sequence using the GUI that created . Finally, the final assembly prototype will be tested and troubleshooted if there is any problem occur.

3.2 ELECTRICAL PARTS

3.2.1 Micro-processor (NI Myrio)

The NI Myrio embedded student design device was created for students to ‘do real-world engineering’. It features a 667 MHz dual-core ARM Cortex-A9 programmable processor and a customizable Xilinx field-programmable gate array (FPGA) that students can use to start developing systems and solve complicated design problems faster. The NI Myrio device features the Zynq-7010 All Programmable system on a chip (SoC) to unleash the power of NI LabVIEW system software both in a real-time (RT) application and on the FPGA level.

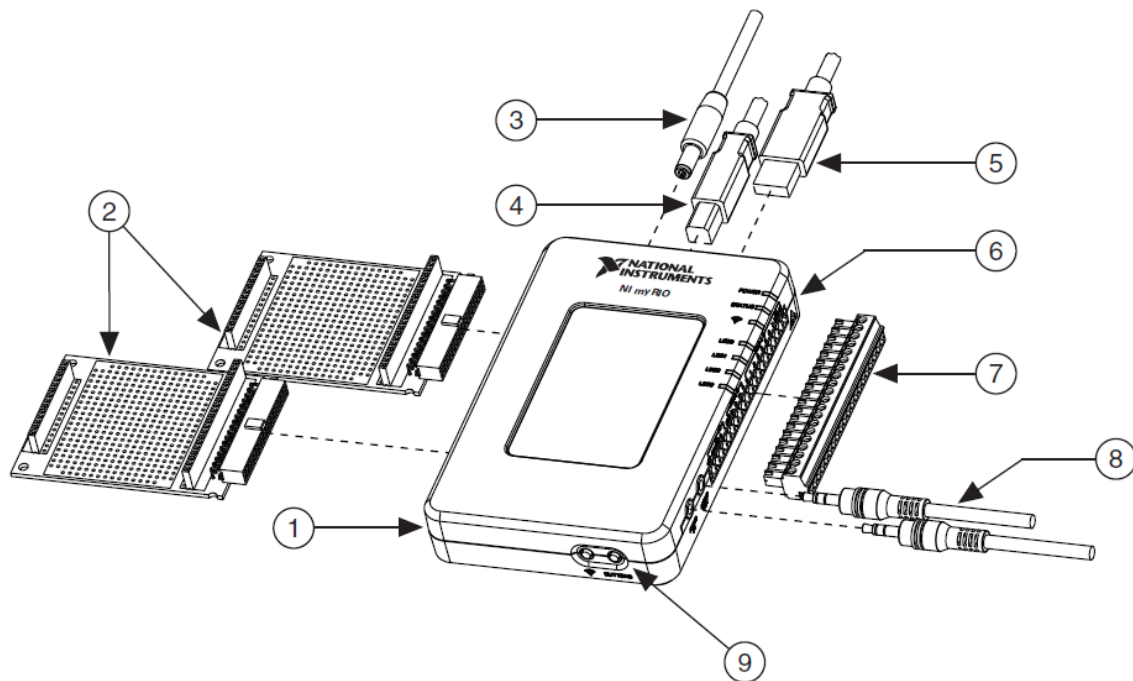


Figure 3.3 NI myRIO- I900 assembly drawing

1	NI Myrio –I900
2	Myrio Expansion Port (MXP) Breakouts