



ข้อเสนอโครงร่างปริญญานิพนธ์/โครงการ  
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์  
มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย

---

ชื่อปริญญานิพนธ์/โครงการ

(ภาษาไทย)	การแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัว
(ภาษาอังกฤษ)	Cross Compiler for Embedded Systems

ผู้ทำปริญญานิพนธ์/โครงการ

นายกอมารุดดิน บูเก็ม  
นายชานน หลีนายน้ำ  
สาขาวิชาวิศวกรรมคอมพิวเตอร์ หลักสูตรวิศวกรรมศาสตรบัณฑิต  
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย

อาจารย์ที่ปรึกษาปริญญานิพนธ์/โครงการ

อาจารย์สิทธิโชค อุ่นแก้ว อาจารย์ที่ปรึกษาหลัก  
อาจารย์ณัฐพล หนูฤทธิ อาจารย์ที่ปรึกษาร่วม

สาขาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลศรีวิชัย

## 1. ความสำคัญและที่มา

เนื่องจากผู้จัดทำได้เข้าร่วมโครงการแพลตฟอร์มบริหารจัดการทรัพยากรผู้มีศักยภาพของกลุ่มมหาวิทยาลัยเทคโนโลยีราชมงคล เพื่อปฏิรูประบบการพัฒนากำลังคนของประเทศ (TALENT RESOURCE MANAGEMENT - RAJAMANGALA UNIVERSITY OF TECHNOLOGY: TRM-RMUT) ในหัวข้อโครงการ “การพัฒนาบอร์ดวงจรรวมอเนกประสงค์สำหรับการประยุกต์ใช้งานด้านระบบควบคุมอัตโนมัติ และปัญญาประดิษฐ์อัจฉริยะสมัยใหม่ (A Development of Multi- functional Integrated Circuit Boards and their applications to Automatic Control Systems and Modern Artificial Intelligence) โดยโครงการนี้เป็นการวิจัยและพัฒนา ร่วมกับสถานประกอบการด้านการพัฒนาผลิตภัณฑ์ เพื่อออกแบบและสร้างบอร์ดวงจรรวมอเนกประสงค์ ที่มีความยืดหยุ่นต่อการใช้งาน และครอบคลุมกับการใช้งานด้านระบบควบคุมอัตโนมัติ และปัญญาประดิษฐ์อัจฉริยะสมัยใหม่ โดยบอร์ดมีคุณลักษณะพิเศษที่สามารถประยุกต์ใช้งานได้หลากหลาย ทั้งการเชื่อมต่อที่รองรับกับการสื่อสารแบบลอรา (LoRa) อาร์เอสสี่แปดห้า (RS-485) และดิจิทัล (Digital) สามารถประมวลผลรับค่าและแสดงค่านำเข้า (Input) หรือ ส่งออก (Output) ทั้งรูปแบบอะนาล็อก (Analog) และดิจิทัล ที่รองรับกับงานอุตสาหกรรมสมัยใหม่ทั้งขนาดเล็กและขนาดใหญ่ ซึ่งสามารถเก็บรวบรวมข้อมูลที่ได้จากตัวรับรู้ (Sensor) อุตสาหกรรมชนิดต่าง ๆ เพื่อแสดงผลผ่านตัวควบคุมและเก็บผลในรูปแบบตัวลงบันทึกข้อมูล (Data logger) การเชื่อมต่อประยุกต์ใช้กับในรูปแบบอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things: IoTs) รวมทั้งการวิเคราะห์ค่าที่เกี่ยวข้องด้านกำลังไฟฟ้า แรงดันไฟฟ้า กระแสไฟฟ้า เพื่อการวิเคราะห์อุปกรณ์ไฟฟ้าในงานอุตสาหกรรม สามารถรายงานการใช้งานไฟฟ้าในรูปแบบเอเอ็มอาร์ (Automatic Meter Reading: AMR) ตลอดจนการใช้งานบอร์ดสำหรับตรวจสอบสถานการณ์ใช้งานและ ควบคุมการใช้งานของอุปกรณ์ไฟฟ้าในช่วงเวลาต่าง ๆ ได้

จากที่กล่าวมาข้างต้น ผู้จัดทำได้รับมอบหมายจากผู้วิจัยโครงการให้พัฒนาการแปลโปรแกรม (compiler) ข้ามเครื่องสำหรับระบบฝังตัว (embedded system) สำหรับผู้ใช้ที่มีความรู้ด้านการโปรแกรมคอมพิวเตอร์เบื้องต้นหรือผู้ใช้ที่ไม่มีความรู้ด้านการโปรแกรมคอมพิวเตอร์ เพื่อให้สามารถแปลชุดคำสั่งในโปรแกรมและบรรจุขึ้น(upload)ไปยังส่วนต่อประสาน (interface เพื่อควบคุมการใช้งานบอร์ดวงจรรวมอเนกประสงค์ สำหรับการประยุกต์ใช้งานด้านระบบควบคุมอัตโนมัติ ผู้จัดทำจึงได้พัฒนาการแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัวเพื่อแก้ปัญหาดังกล่าว

## 2. วัตถุประสงค์

2.1 เพื่อสร้างการแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัว

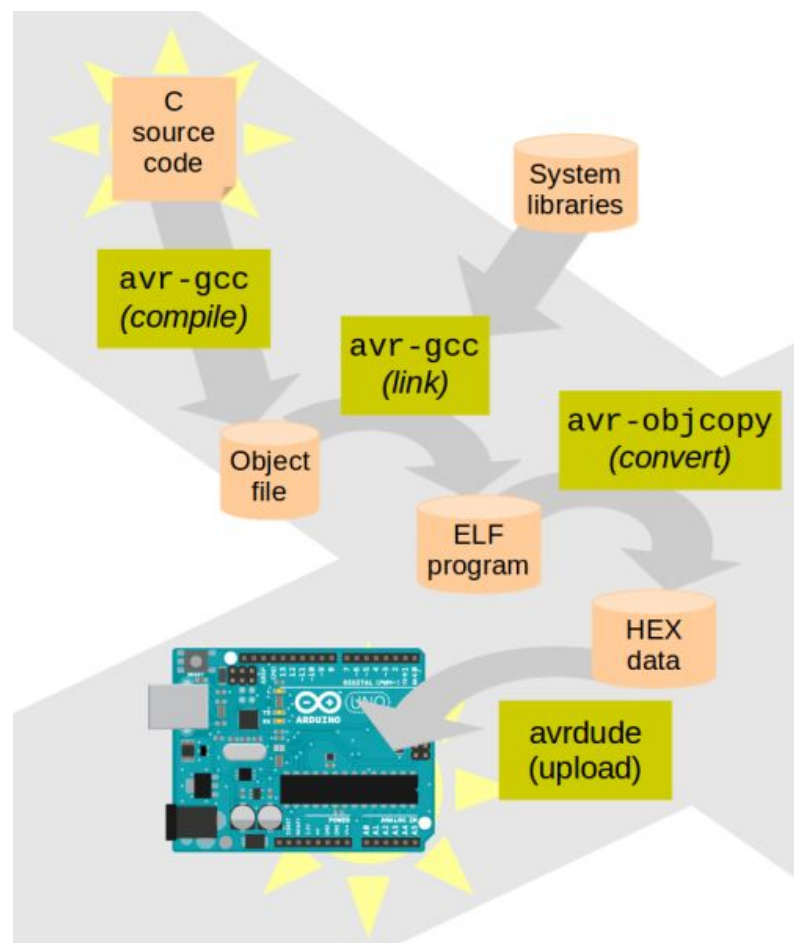
## 3. ขอบเขตงานวิจัย

3.1 สามารถแปลโปรแกรมชุดคำสั่งของโปรแกรม

3.2 สามารถบรรจุขึ้นไปยังส่วนต่อประสาน

#### 4. ทฤษฎีและหลักการที่เกี่ยวข้อง

ในการจัดทำโครงการนี้ผู้จัดทำได้พัฒนาการแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัว เนื่องจากในปัจจุบันนี้ การเขียนชุดคำสั่งของโปรแกรมมีความสำคัญและเป็นที่น่าสนใจมากยิ่งขึ้น ซึ่งมีวัตถุประสงค์การใช้งานที่แตกต่างกันไปมากมาย แต่หลักสำคัญคือการที่จะสื่อสารกับคอมพิวเตอร์ด้วยการเขียนคำสั่งให้คอมพิวเตอร์ทำตามในสิ่งที่ต้องการผ่านภาษาคอมพิวเตอร์ที่มีอยู่นั้น เช่น ภาษาซี ภาษาจาวา และ ภาษาไพธอน คอมพิวเตอร์ไม่สามารถที่จะรับรู้สิ่งที่ต้องทำหรือทำตามความต้องการได้เพราะภาษาที่ใช้กันในปัจจุบันจะเป็นภาษาระดับสูงซึ่งคอมพิวเตอร์ไม่สามารถที่จะทำความเข้าใจได้จึงต้องมีการแปลโปรแกรมให้เป็นภาษาที่เครื่องคอมพิวเตอร์สามารถเข้าใจ และสามารถติดต่อได้โดยตรงกับเครื่องคอมพิวเตอร์ ซึ่งอยู่ในรูปแบบของบิต รีจิสเตอร์ เรียกว่า ภาษาเครื่อง (Machine Language ซึ่งเป็นเรื่องยากสำหรับปัจจุบันที่จะทำความเข้าใจและเขียนคำสั่งดังกล่าวจึงจำเป็นต้องมีการแปลโปรแกรม [1]



รูปที่ 1 ภาพรวมระบบ

ที่มา : <https://pin.it/6oGn1uL>

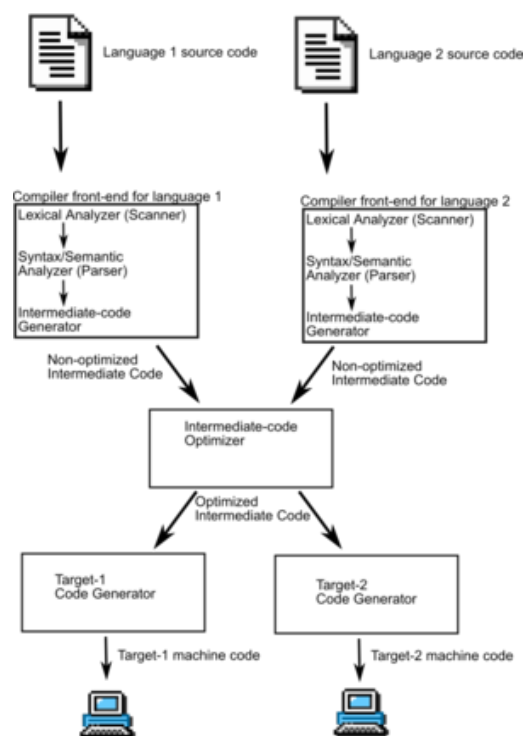
#### 4.1 การแปลโปรแกรม



รูปที่ 2 การแปลโปรแกรม

ที่มา : <https://www.mindphp.com/คู่มือ/73-คืออะไร/3596-compiler-คอมไพเลอร์คืออะไร.html>

การแปลโปรแกรม เป็น โปรแกรมคอมพิวเตอร์ที่ทำหน้าที่แปลงชุดคำสั่งภาษาคอมพิวเตอร์หนึ่ง ไปเป็นชุดคำสั่งที่มีความหมายเดียวกัน ในภาษาคอมพิวเตอร์อื่น การแปลโปรแกรมส่วนใหญ่ จะทำการแปลรหัสต้นฉบับ (source code) ที่เขียนในภาษาระดับสูง เป็น ภาษาระดับต่ำ หรือภาษาเครื่อง ซึ่งคอมพิวเตอร์สามารถที่จะทำงานได้โดยตรง อย่างไรก็ตาม การแปลจากภาษาระดับต่ำเป็นภาษาระดับสูงเป็นไปได้โดยใช้การแปลโปรแกรมย้อนกลับ (decompiler)



รูปที่ 3 ขั้นตอนการทำงานของ การแปลโปรแกรม

ที่มา : <https://th.wikipedia.org/wiki/คอมไพเลอร์>

ผลลัพธ์ของการแปลโปรแกรมโดยทั่วไป ที่เรียกว่า ออบเจกต์โค้ด (object code) จะประกอบด้วย ภาษาเครื่อง (Machine code) ที่เต็มไปด้วยข้อมูลเกี่ยวกับชื่อและสถานที่ของแต่ละจุด และการเรียกใช้วัตถุภายนอก (Link object) สำหรับเครื่องมือที่เราใช้รวม ออบเจกต์ เข้าด้วยกัน จะเรียกว่าโปรแกรมเชื่อมโยงเพื่อที่ผลลัพธ์ที่ออกมาในขั้นสุดท้าย เป็นไฟล์ที่ผู้ใช้งานทั่วไปสามารถใช้งานได้สะดวก

การแปลโปรแกรมที่สมบูรณ์ตัวแรก คือ ภาษาฟอร์แทรน (FORTRAN) ของ ไอบีเอ็ม ในปี ค.ศ. 1957 และ ภาษาโคบอล (COBOL) ก็เป็นการแปลโปรแกรมตัวแรก ๆ ที่สามารถทำงานได้บนหลาย ๆ สถาปัตยกรรมทางคอมพิวเตอร์ การพัฒนาตัวแปลภาษารุดหน้าอย่างรวดเร็ว และเริ่มมีรูปแบบที่ชัดเจนยิ่งขึ้นต่อมา ในช่วงทศวรรษ 1960

การทำงานของโปรแกรมการแปล เริ่มจากการแปลโปรแกรมจะอ่านรหัสต้นฉบับของภาษานั้น ๆ แล้วเริ่มตรวจสอบความผิดพลาด ถ้าพบก็จะแปลโปรแกรมไม่ผ่านและให้ผู้แก้ไขรหัสต้นฉบับก่อน เมื่อแปลผ่าน การแปลโปรแกรมจะสร้างไฟล์วัตถุ (.obj) บนระบบปฏิบัติการแบบใช้จาน (Disk Operating System : DOS และไฟล์ดอทโอ (.o) บนระบบปฏิบัติการลินุกซ์ (Linux ขึ้นมา แล้ว การแปลโปรแกรมจะเชื่อมโยงแฟ้มข้อมูลวัตถุเข้ากับรหัสต้นฉบับ และสร้างไฟล์เอกซ์คิวต์ (.exe) บนระบบปฏิบัติการแบบใช้จาน ขึ้นมา[2]

#### 4.1.1 ชุดแปลโปรแกรมของกนู (GNU Compiler Collection : GCC



รูปที่ 4 ชุดแปลโปรแกรมของกนู

ที่มา : <https://th.wikipedia.org/wiki/ชุดแปลโปรแกรมของกนู>.

ชุดแปลโปรแกรมของกนู เป็นชุดโปรแกรมแปลโปรแกรมสำหรับแปลภาษาโปรแกรมต่าง ๆ พัฒนาโดยโครงการกนู (GNU) และแจกจ่ายเป็นซอฟต์แวร์เสรีภายใต้สัญญาอนุญาตสาธารณะทั่วไปของกนู (GPL & LGPL) โดยชุดแปลโปรแกรมของกนู ถือว่าเป็นชุดการแปลโปรแกรมมาตรฐานในระบบปฏิบัติการแบบยูนิกซ์ (Unix) อย่างเช่น ลินุกซ์ หรือ ระบบปฏิบัติการแมค เทีน (Mac OS X) เป็นต้นชื่อเดิมของชุดแปลโปรแกรมของกนู

(GNU C Compiler) เนื่องจากในระยะแรกถูกพัฒนาขึ้นเป็นการแปลโปรแกรมของภาษาซีเท่านั้น แต่ในภายหลังเมื่อได้พัฒนาให้สนับสนุนภาษาโปรแกรมอื่น ๆ มากขึ้น จึงเปลี่ยนชื่อเพื่อให้สื่อครอบคลุมความหมายมากกว่าเดิม

ภาษาโปรแกรมที่สนับสนุนในชุดแปลโปรแกรมมาตรฐานเวอร์ชัน 4.3 สนับสนุนภาษาโปรแกรมดังต่อไปนี้ ซี, ซีพลัส, จาวา (java), เอดา (Ada), ออบเจกทีฟซี (Objective-C), ออบเจกทีฟซีพลัสพลัส (Objective-C++) และ ฟอรัทเทน (Fortran) นอกจากนี้ยังสนับสนุนภาษาโปรแกรมอื่น ๆ เพิ่มเติม แต่ไม่ได้รวมอยู่ในชุดการแปลโปรแกรมมาตรฐานได้แก่ โมดูลา-2 (Modula-2), โมดูลา-3 (Modula-3), ปาสคาล (Pascal), พีแอล/วัน (PL/I), ดี (D), เมอร์คิวรี (Mercury) และวีเอชดีแอล (VHDL)

โครงสร้างการทำงานชุดแปลโปรแกรมของกนู ทำงานโดยใช้บรรทัดคำสั่ง ผู้ใช้มักเรียกโปรแกรมโดยการเรียกโปรแกรมที่ชื่อ gcc พร้อมทั้งใส่ตัวเลือกต่าง ๆ ทางพารามิเตอร์ในบรรทัดคำสั่ง โปรแกรมนี้จะวิเคราะห์ภาษาที่ใช้จากนามสกุลของไฟล์ แล้วเรียกตัวประมวลผลขั้นต้น (Pre Processor), การแปลโปรแกรม, แอสเซมเบลเลอร์ (Assembler) และ โปรแกรมเชื่อมโยง (Linker) มาทำงานตามลำดับ แต่ผู้ใช้สามารถสั่งให้ทำงานเฉพาะบางขั้นตอนข้ามจากพารามิเตอร์ได้

ตัวประมวลผลขั้นต้น ที่ใช้เป็นโปรแกรมหนึ่งใน ชุดแปลโปรแกรมของกนู มีชื่อว่า ซีพรี เรียกใช้ก็ต่อเมื่อแปลโปรแกรมที่เขียนในบางภาษา เช่น ซี หรือ ซีพลัส เป็นต้น

ในชุดแปลโปรแกรมของกนู แยกการแปลโปรแกรมแต่ละภาษาออกเป็นคณะโปรแกรมตัวอย่างเช่น การแปลโปรแกรมภาษาซี จะเป็นโปรแกรมที่ชื่อ ซีซี1 โครงสร้างภายในการแปลโปรแกรมแต่ละภาษาแบ่งเป็นสามส่วนเหมือนกัน ได้แก่ ฟรอนต์เอนด์ (frontend) มิดเดิลเอนด์ (middle-end) และ แบ็คเอนด์ (backend)

1 ฟรอนต์เอนด์ ทำหน้าที่อ่านไฟล์ที่ผ่านการประมวลผลตัวประมวลผลขั้นต้น แล้วตีความหมาย ตรวจสอบหาข้อผิดพลาด และสร้างโครงสร้างข้อมูลที่ชื่อ เจเนอริก (GENERIC) ซึ่งมีรูปแบบแอบสแทรค ซินแทกซ์ ทรี (abstract syntax tree : AST) ฟรอนต์เอนด์นี้เป็นส่วนของการแปลโปรแกรมที่ขึ้นกับภาษา

2 มิดเดิลเอนด์ ทำหน้าที่แปลงเจเนอริก เป็นอีกโครงสร้างหนึ่งที่ชื่อ กิมเพล (GIMPLE) ซึ่งลดรูปโครงสร้างต่าง ๆ เช่น คำสั่งฟอร์ (for), คำสั่งไวน์ (while) ให้เหลือเพียงคำสั่งอิฟ (if) และ คำสั่งโกทู (goto) แล้วแปลงเข้ารูปแบบฟอร์มการกำหนดเดี่ยวแบบคงที่ (static single assignment : SSA) สุดท้ายสร้างโครงสร้างข้อมูลที่ชื่อการลงทะเบียนภาษาที่ถ่ายโอน (Register Transfer Language : RTL) ซึ่งในแต่ละขั้นตอนการแปลข้างบน ก็มีการทำ ออปติไมซ์ (optimize) เพื่อให้โปรแกรมที่ได้ทำงานเร็วขึ้น ส่วนนี้ของการแปลโปรแกรมไม่ขึ้นกับภาษาหรือสถาปัตยกรรมที่ใช้

3 แบ็คเอนด์ แปลงการลงทะเบียนภาษาที่ถ่ายโอน ไปเป็นโปรแกรมภาษาแอสเซมบลี ส่วนนี้ขึ้นกับสถาปัตยกรรมที่จะการแปลโปรแกรม

ส่วน แอสเซมเบลอร์ และ โปรแกรมเชื่อมโยง นั้นไม่รวมอยู่ในชุดโปรแกรมแปลโปรแกรมของกนู ในระบบกนูโปรแกรมทั้งสองจะอยู่ในอีกชุดโปรแกรมหนึ่ง คือ ไบนูทิลส์ (binutils) แอสเซมเบลอร์ ทำหน้าที่แปลภาษาแอสเซมบลี เป็นภาษาเครื่อง ในขณะที่ โปรแกรมเชื่อมโยง ทำหน้าที่รวมไฟล์ภาษาเครื่อง ซึ่งอาจมีหลายไฟล์ และไลบรารี (Library) เข้าด้วยกัน เป็นโปรแกรมที่สมบูรณ์สามารถทำงานได้[3]

#### 4.1.2 แฟ้มต้นฉบับเลขฐานสิบหก (HEX)

:10010000214601360121470136007EFE09D2190140
:100110002146017EB7C20001FF5F16002148011988
:10012000194E79234623965778239EDA3F01B2CAA7
:100130003F0156702B5E712B722B732146013421C7
:00000001FF

	Start code
	Byte count
	Address
	Record type
	Data
	Checksum

รูปที่ 5 รูปแบบแฟ้มต้นฉบับเลขฐานสิบหก

ที่มา : <https://www.flx.cat/format-intel-hex/>

แฟ้มต้นฉบับเลขฐานสิบหกเป็นรูปแบบไฟล์ที่สื่อถึงข้อมูลเลขฐานสอง (Binary ในรูปแบบข้อความรหัสมาตรฐานอเมริกันสำหรับการแลกเปลี่ยนข้อมูล (American Standard Code for Information Interchange : ASCII) มันมักจะใช้สำหรับการเขียนโปรแกรมไมโครคอนโทรลเลอร์ อีพรีม (Erasable Programmable Read-Only Memory : EPROM และอุปกรณ์ตรรกะที่ตั้งโปรแกรมได้ประเภทอื่น ๆ และอิมูเลเตอร์ฮาร์ดแวร์ ในโปรแกรมประยุกต์ทั่วไป การแปลโปรแกรมหรือแอสเซมเบลอร์จะแปลงรหัสต้นฉบับของโปรแกรม เช่น ในภาษาซี หรือภาษาแอสเซมบลี เป็นรหัสเครื่องและแสดงผลเป็นแฟ้มต้นฉบับเลขฐานสิบหก บางคนยังใช้เป็นรูปแบบคอนเทนเนอร์ที่เก็บข้อมูลสตรีมแพ็คเก็ต นามสกุลไฟล์ทั่วไปที่ใช้สำหรับไฟล์ผลลัพธ์คือแฟ้มต้นฉบับเลขฐานสิบหก จากนั้นโปรแกรมจะอ่านแฟ้มต้นฉบับเลขฐานสิบหกเพื่อเขียนรหัสเครื่องลงในพรีม (programmable read-only memory : PROM) หรือถูกถ่ายโอนไปยังระบบเป้าหมายเพื่อโหลดและดำเนินการ [4]

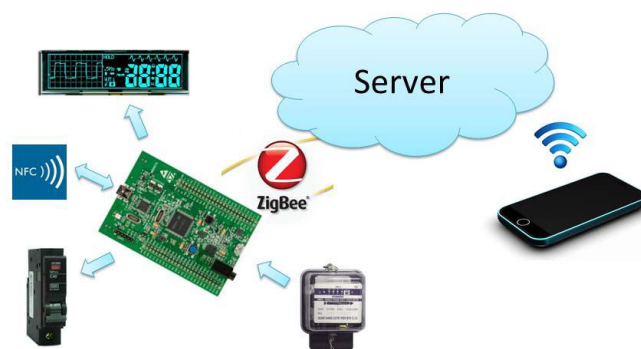
#### 4.2 บรรจุนขึ้น

บรรจุนขึ้น คือ การถ่ายโอนข้อมูลจากคอมพิวเตอร์เครื่องหนึ่ง ไปยังคอมพิวเตอร์อีกเครื่องหนึ่งคอมพิวเตอร์ตัวส่งข้อมูลเรียกว่าเป็นตัวที่ทำการบรรจุนขึ้น ส่วนคอมพิวเตอร์ตัวรับข้อมูลเรียกว่าเป็นตัวที่ทำการบรรจุนลง (download) ส่วนมากใช้ในการถ่ายโอนจากคอมพิวเตอร์ในเครือข่ายไปไว้ในแม่ข่าย ซึ่งถือว่าสูงกว่า ถ้าเป็นการถ่ายโอนจากแม่ข่าย จะเรียกว่าบรรจุนลง [5]

#### 4.3 ระบบฝังตัว

ระบบฝังตัว หรือ สมองกลฝังตัว คือระบบประมวลผล ที่ใช้ชิปหรือไมโครโพรเซสเซอร์ที่ออกแบบมาโดยเฉพาะ เปรียบเสมือนระบบคอมพิวเตอร์ขนาดเล็กที่ฝังไว้ในอุปกรณ์ เครื่องใช้ไฟฟ้า และเครื่องเล่นอิเล็กทรอนิกส์ต่าง ๆ เพื่อเพิ่มความฉลาด ความสามารถให้กับอุปกรณ์เหล่านั้นผ่านซอฟต์แวร์ซึ่งต่างจากระบบประมวลผลที่เครื่อง

คอมพิวเตอร์ทั่วไป ระบบฝังตัวถูกนำมาใช้กันอย่างแพร่หลายในยานพาหนะ เครื่องใช้ไฟฟ้าในบ้านและสำนักงาน อุปกรณ์อิเล็กทรอนิกส์ เทคโนโลยีซอฟต์แวร์ เทคโนโลยีฮาร์ดแวร์ เทคโนโลยีเครือข่ายเน็ตเวิร์ค เทคโนโลยีด้านการสื่อสาร เทคโนโลยีเครื่องกลและของเล่นต่าง ๆ คำว่าระบบฝังตัวเกิดจากการที่ระบบนี้เป็นระบบประมวลผลเช่นเดียวกับระบบคอมพิวเตอร์ แต่ว่าระบบนี้จะฝังตัวลงในอุปกรณ์อื่น ๆ ที่ไม่ใช่เครื่องคอมพิวเตอร์ ในปัจจุบันระบบสมองกลฝังตัวได้มีการพัฒนามากขึ้น โดยในระบบสมองกลฝังตัวอาจจะประกอบไปด้วยไมโครคอนโทรลเลอร์ หรือไมโครโพรเซสเซอร์ อุปกรณ์ที่ใช้ระบบสมองกลฝังตัวที่เห็นได้ชัดเช่นโทรศัพท์มือถือ และในระบบสมองกลฝังตัวยังมีการใส่ระบบปฏิบัติการต่าง ๆ แตกต่างกันไปอีกด้วย ดังนั้น ระบบสมองกลฝังตัวอาจจะทำงานได้ตั้งแต่ควบคุมหลอดไฟจนถึงใช้ในยานอวกาศ [6]



รูปที่ 6 ระบบฝังตัว

ที่มา : <https://embedded523372.wordpress.com/523372embedded/>

#### 4.4 อีเอสพี 32 (ESP32)

อีเอสพี 32 เป็นชิปไมโครคอนโทรลเลอร์ที่มาพร้อมไวไฟ มาตรฐาน 802.11 บี/จี/เอ็น และบลูทูธเวอร์ชัน 4.2 เป็นรุ่นต่อ ยอดความสำเร็จของ อีเอสพี8266 โดยในรุ่นนี้ได้ออกมาแก้ไขข้อเสียของ อีเอสพี8266 ทั้งหมด

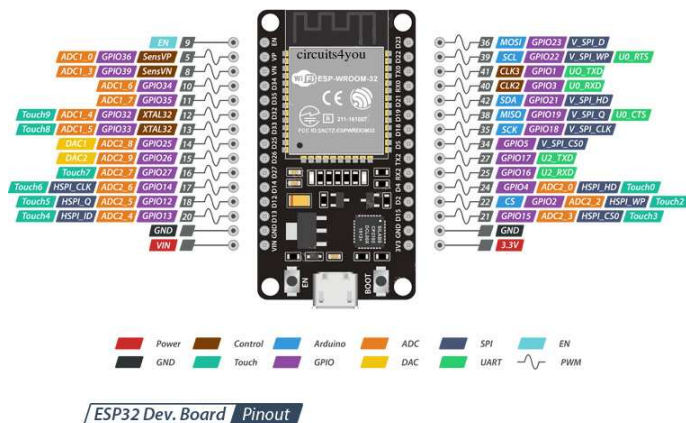


รูปที่ 7 อีเอสพี 32

ที่มา : <https://www.robotsiam.com/article/42/esp32-1-รู้จัก-ไมโครคอนโทรลเลอร์-ESP32>



โดยหน่วยประมวลผลกลางใช้สถาปัตยกรรมเท็นซิริกา แอลเอ็กซ์ 6 จำนวน 2 คอร์ สัญญาณนาฬิกา 240 เมกะเฮิร์ตซ์สามารถแยกการทำงานระหว่างโปรแกรมจัดการไอพี และแอปพลิเคชันออกจากกันได้ ทำให้มีเสถียรภาพเพิ่มขึ้นมาก มีแรม 520 กิโลไบต์ มาในตัว นอกจากนี้ยังมีพอร์ตอเนกประสงค์เพิ่มขึ้นมาก ใช้แรงดันไฟฟ้า 3.3 โวลต์ ในโหมดหลักใช้กระแสไฟฟ้าเพียง 2.5 ไมโครแอมป์[7]



รูปที่ 8 การใช้ขาต่าง ๆ ของอีเอสพี 32

ที่มา : <https://circuits4you.com/2018/12/31/esp32-devkit-esp32-wroom-gpio-pinout/>

#### 4.5 เซิร์ฟเวอร์ (Server)

เซิร์ฟเวอร์ หรือ เครื่องบริการ หรือ เครื่องแม่ข่าย คือ เครื่องหรือโปรแกรมคอมพิวเตอร์ซึ่งทำงานให้บริการ ในระบบเครือข่ายแก่ลูกข่าย (ซึ่งให้บริการผู้ใช้คนที่หนึ่ง เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเซิร์ฟเวอร์นี้ควรมีประสิทธิภาพสูง มีความเสถียร สามารถให้บริการแก่ผู้ใช้ได้เป็นจำนวนมาก ภายในเซิร์ฟเวอร์ให้บริการได้ด้วยโปรแกรมบริการ ซึ่งทำงานอยู่บนระบบปฏิบัติการอีกชั้นหนึ่ง[8]

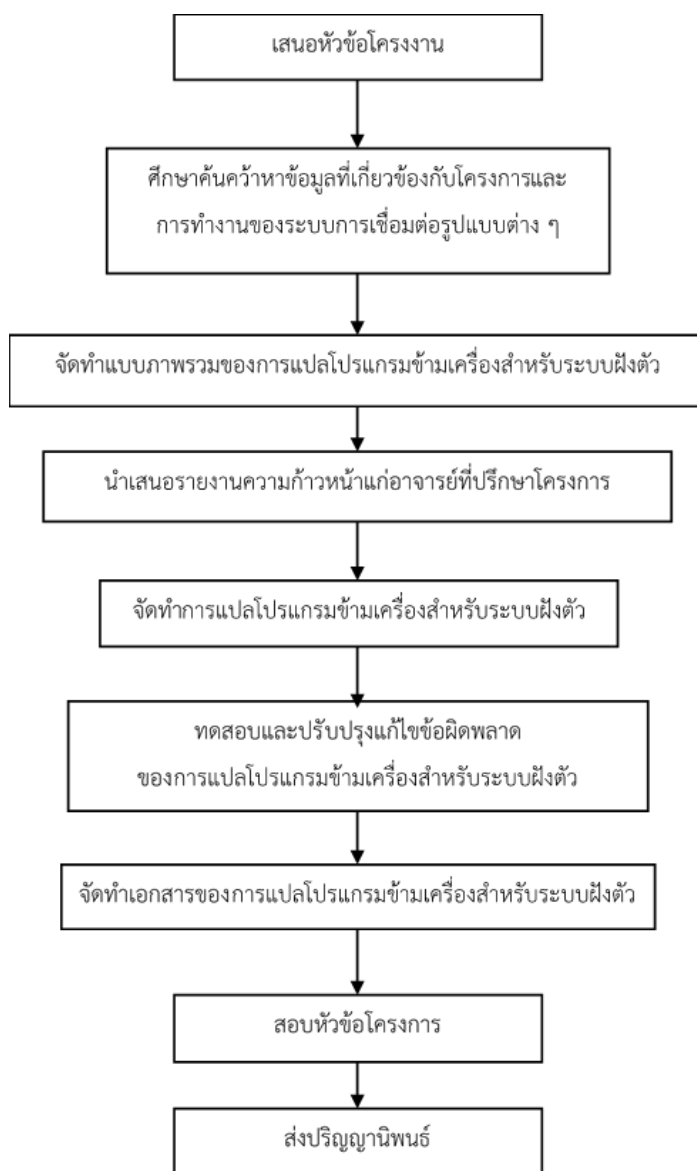


รูปที่ 9 เครื่องเซิร์ฟเวอร์

ที่มา : <https://th.wikipedia.org/wiki/เซิร์ฟเวอร์>

## 5. วิธีดำเนินการ

วิธีดำเนินการมีรายละเอียดดังแผนภาพนี้



รูปที่ 10 แผนภาพวิธีดำเนินการ

- 5.1 เสนอหัวข้อโครงการ
- 5.2 ศึกษาค้นคว้าหาข้อมูลที่เกี่ยวข้องกับโครงการและการทำงานของระบบการเชื่อมต่อรูปแบบต่าง ๆ
- 5.3 จัดทำแบบภาพรวมของการแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัว
- 5.4 นำเสนอรายงานความก้าวหน้าเป็นระยะแก่อาจารย์ที่ปรึกษาโครงการ โดยอาจารย์ที่ปรึกษาโครงการจะให้ข้อเสนอแนะต่าง ๆ ในการจัดทำเนื้อหาและการจัดทำระบบแล้วนำข้อเสนอแนะมาปรับปรุงแก้ไขให้ดียิ่งขึ้น
- 5.5 จัดทำการแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัว

- 5.6 ทดสอบและปรับปรุงแก้ไขข้อผิดพลาดของการแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัว
- 5.7 จัดทำเอกสารของการแปลโปรแกรมข้ามเครื่องสำหรับระบบฝังตัว
- 5.8 สอบหัวข้อโครงการ
- 5.9 ส่งปริญญานิพนธ์

## 6. แผนการดำเนินการ

เริ่มดำเนินการเดือน...ธันวาคม...พ.ศ. 2564..... ถึง.....ตุลาคม.....พ.ศ. 2565.....

ขั้นตอนที่	เดือน									
	1	2	3	4	5	6	7	8	9	10
1.	←→		→							
2.	←→		→							
3.			←→	→						
4.				←→	→					
5.					←→	→				
6.						←→	→			
7.							←→	→		
8.								←→	→	
9.										←→

- ขั้นตอนที่ 1    เสนอหัวข้อโครงการ
- ขั้นตอนที่ 2    ศึกษาค้นคว้าหาข้อมูล
- ขั้นตอนที่ 3    จัดทำแบบภาพรวมของระบบ
- ขั้นตอนที่ 4    นำเสนอรายงานความก้าวหน้า
- ขั้นตอนที่ 5    ลงมือจัดทำระบบ
- ขั้นตอนที่ 6    ทดสอบและปรับปรุงแก้ไขข้อผิดพลาด
- ขั้นตอนที่ 7    จัดทำเอกสารของระบบ
- ขั้นตอนที่ 8    สอบหัวข้อโครงการ
- ขั้นตอนที่ 9    ส่งปริญญานิพนธ์

## 7. ประโยชน์ที่คาดว่าจะได้รับ

- 7.1 ตรวจสอบข้อผิดพลาดของชุดโปรแกรมต้นฉบับ
- 7.2 แปลแฟ้มชุดโปรแกรมต้นฉบับให้เป็นแฟ้มเลขฐานสิบหก
- 7.3 บรรจุขึ้นแฟ้มเลขฐานสิบหกไปยังส่วนต่อประสาน

## 8. เอกสารอ้างอิง

- [1] มายด์พีเอชพี. (2559). Compiler (คอมไพเลอร์ คืออะไร .  
<https://www.mindphp.com/คู่มือ/73-คืออะไร/3596-compiler-คอมไพเลอร์-คืออะไร.html>
- [2] วิกิพีเดีย สารานุกรมเสรี. (2564). คอมไพเลอร์.  
<https://th.wikipedia.org/wiki/คอมไพเลอร์>
- [3] วิกิพีเดีย สารานุกรมเสรี. (2564). ชุดแปลโปรแกรมของกนู.  
<https://th.wikipedia.org/wiki/ชุดแปลโปรแกรมของกนู>.
- [4] วิกิพีเดีย สารานุกรมเสรี. (2564). Intel HEX.  
[https://th.wikipedia.org/wiki/Intel\\_HEX](https://th.wikipedia.org/wiki/Intel_HEX)
- [5] ลองดู ดิกซ์. (2549). Upload.  
<https://dict.longdo.com/search/upload>.
- [6] วิกิพีเดีย สารานุกรมเสรี. (2564). ระบบฝังตัว.  
<https://th.wikipedia.org/wiki/ระบบฝังตัว>
- [7] โรบอทสยามดอทคอม. (2563). ESP32 #1: รู้จัก ไมโครคอนโทรลเลอร์ ESP32.  
<https://www.robotsiam.com/article/42/esp32-1-รู้จัก-ไมโครคอนโทรลเลอร์-ESP32>
- [8] วิกิพีเดีย สารานุกรมเสรี. (2564). เซิร์ฟเวอร์.  
<https://th.wikipedia.org/wiki/เซิร์ฟเวอร์>

## หมายเหตุ

- 1 โครงร่างปริญญานิพนธ์/โครงการ ควรมีจำนวนหน้าไม่ต่ำกว่า 12 หน้า และไม่เกิน 15 หน้า
- 2 ควรตรวจสอบชื่อปริญญานิพนธ์/โครงการให้ถูกต้องและตรงตามแบบขออนุมัติโครงร่าง (ปสศ. 4 และ ปสศ. 5
- 3 เอกสารโครงร่างปริญญานิพนธ์/โครงการให้เย็บมุมซ้ายบนเพียงแห่งเดียว ไม่ต้องเข้าเล่มใส่ปก
- 4 ส่งเอกสารโครงร่างปริญญานิพนธ์/โครงการพร้อมแบบขออนุมัติปริญญานิพนธ์/โครงการ (ปสศ. 5