

State Diagram

เรื่องที่จะศึกษา

- State และ Transition
- State Diagram
- เงื่อนไขของการเข้าและออกจาก State ต่างๆ ใน State Diagram

จุดประสงค์

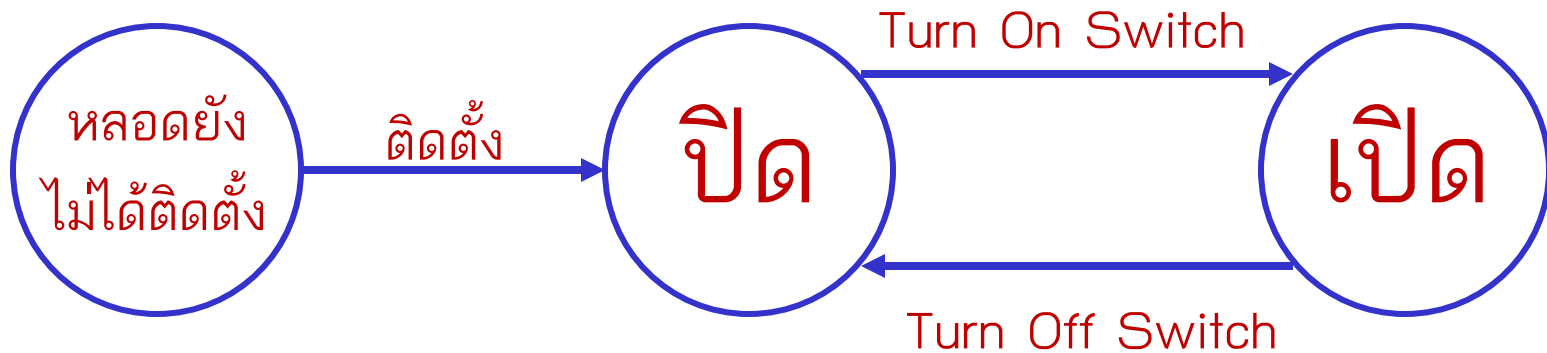
- เพื่อให้เข้าใจแนวคิด การอธิบายกิจกรรมต่างๆ ภายใน class ด้วยสถานะ (State) และการเปลี่ยนสถานะ (Transition)
- เพื่อให้สามารถเขียน State Diagram เพื่ออธิบายกิจกรรมภายในclass ได้

State และ Transition

- กิจกรรมโดยรวมของระบบเกิดจากกิจกรรมย่อยของ object แต่ละตัวรวมกัน
- กลไกที่ทำให้ระบบมีกิจกรรมก็คือการส่ง Message หรือในทาง OO ก็คือการเรียกใช้ Function ของ Object
- กิจกรรมที่เกิดขึ้นใน Object นั้น เกิดจาก 2 สิ่งประกอบกัน คือ สถานะ (state) และ การเปลี่ยนสถานะ (Transition)
- การที่ object ใดๆ เปลี่ยน จาก state ที่ 1 ไปยัง state ที่ 2 จะทำให้เกิดกิจกรรม หรือส่วนของกิจกรรมขึ้นในตัว object นั้น

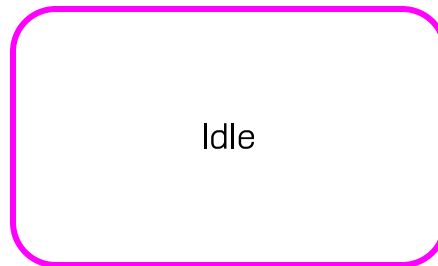
State & Transition

State และ Transition แสดงการทำงานของหลอดไฟ



สัญลักษณ์ State

- แทนด้วย สีเหลี่ยมมุมมน
- ระบุชื่อของ state ลงในสี่เหลี่ยม
- เช่น การเขียน State Idle



สัญลักษณ์ Transition

- แทนด้วย ลูกศร
- ลากจาก state เริ่มต้นไปยัง state ที่ต้องการ
- ใส่ชื่อ transition บนลูกศร มีรูปแบบคือ

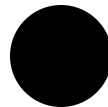
[Condition]/[Action]

Condition คือ เงื่อนไขในการเข้าหรือออกจาก state

Action คือ กิจกรรมที่ทำระหว่างการเปลี่ยน state

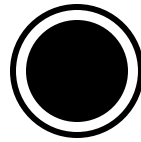
สัญลักษณ์จุดเริ่มต้น

- จุดเริ่มต้นของกิจกรรมต่าง ๆ ใน state diagram เรียกว่า initial state
- แทนด้วยวงกลมทึบ



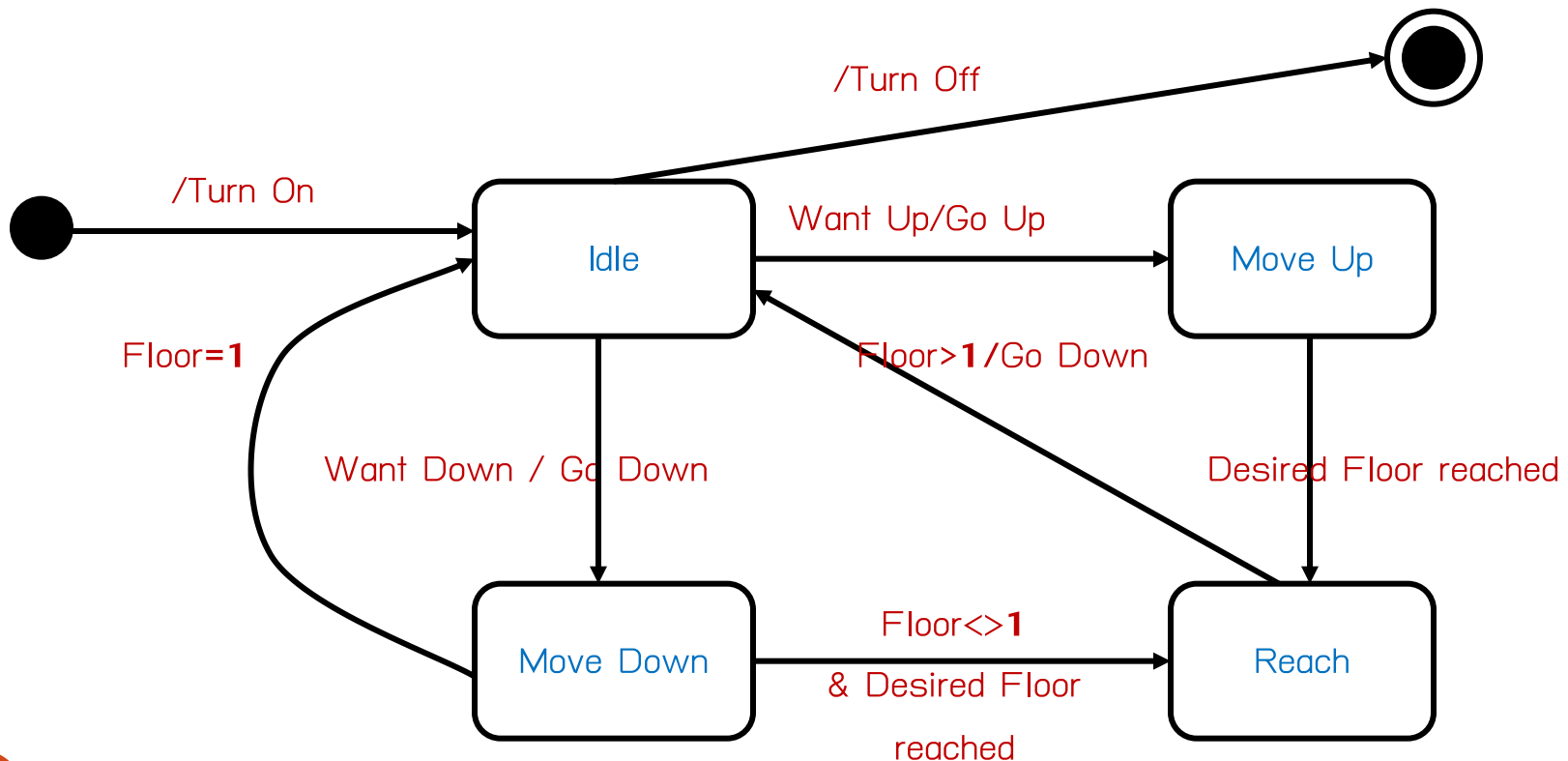
สัญลักษณ์จุดสิ้นสุด

- จุดสิ้นสุดของกิจกรรมทั้งหมดเรียกว่า End state
- แทนด้วยวงกลมใส ล้อมรอบวงกลมทึบ



ตัวอย่าง State Diagram

○ State Diagram การทำงานของลิฟต์



การจำลองกิจกรรมภายใน state

- ใช้เพื่อระบุรายละเอียดในการทำงานของ state ต่าง ๆ แบ่งได้เป็น
 - กิจกรรมที่ทำเมื่อเข้ามาใน state
 - กิจกรรมที่ทำระหว่างอยู่ใน state
 - กิจกรรมที่ทำก่อนที่จะออกจาก state
 - กิจกรรมที่ทำเมื่อเกิดเงื่อนไขต่าง ๆ ขึ้น

กิจกรรมที่ทำเมื่อเข้ามาใน state

- entry/action

- หมายถึง เมื่อเข้ามายัง state นี้ให้ทำกิจกรรม action

เช่น `entry/count=0`

หมายถึง เมื่อเข้ามายัง state ให้ค่า count เป็น 0

กิจกรรมที่ทำระหว่างอยู่ใน state

- do/action

- หมายถึง หลังจากเข้ามายัง state นี้แล้ว หากไม่มีเงื่อนไขอื่นใด ให้ทำกิจกรรม action

เช่น `do/count:=count+1`

หมายถึง เมื่อเข้ามายัง state นี้ให้เพิ่มค่า count ที่ละ 1

กิจกรรมที่ทำก่อนที่จะออกจาก state

- Exit/action

- หมายถึง ขณะที่ออกจาก state นี้ให้ทำกิจกรรม action

เช่น Exit/show “Good Bye” message

หมายถึง หากออก state นี้ให้แสดงข้อความ “Good Bye”

กิจกรรมที่ทำเมื่อเกิดเงื่อนไขขึ้น

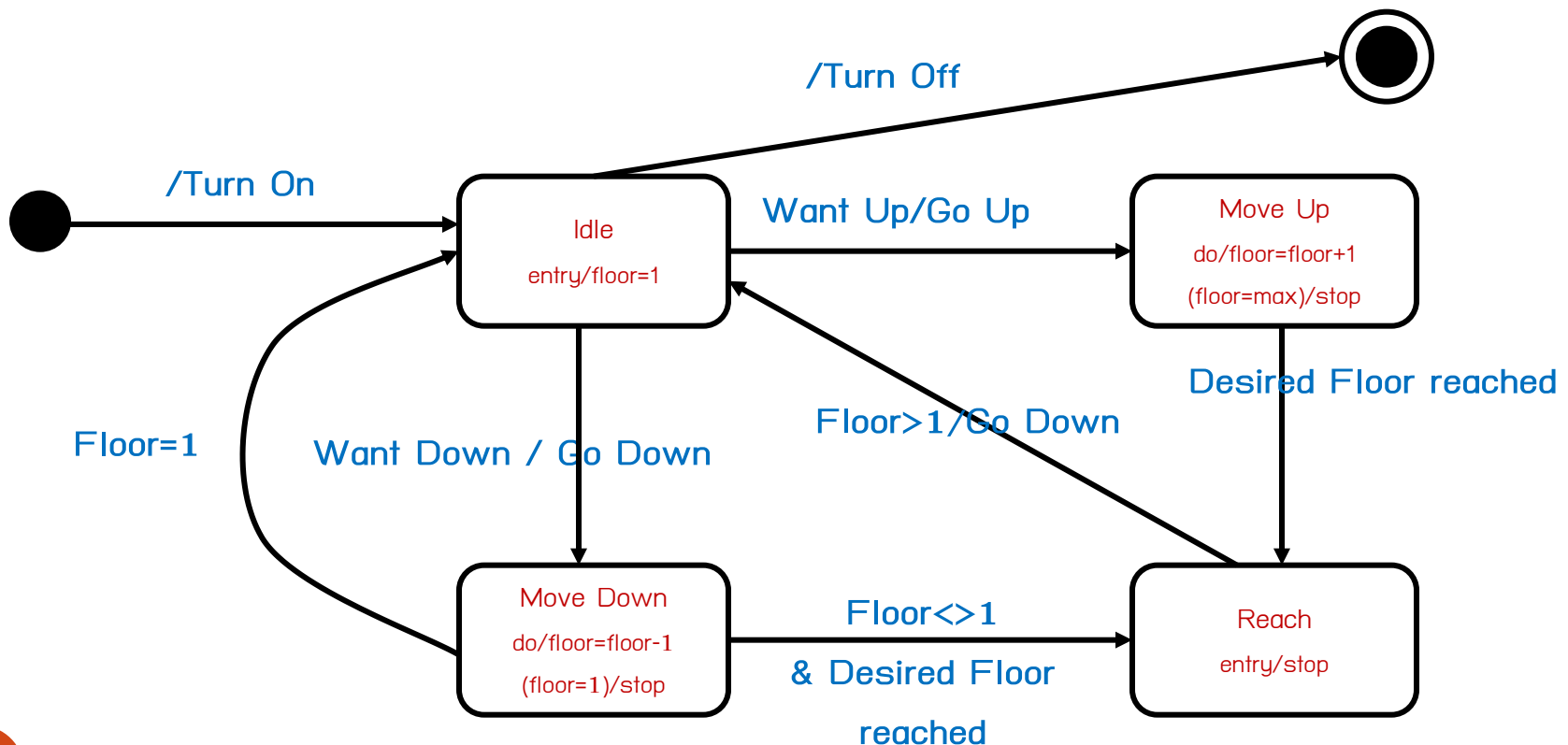
- condition/action
- หมายถึง ขณะที่ยังอยู่ใน state นี้เมื่อเกิดเงื่อนไขใด ๆ ที่กำหนดโดย condition ให้ทำกิจกรรม action

เช่น every 2 seconds/phone ring

หมายถึง ทุก ๆ 2 วินาทีให้เสียงโทรศัพท์ดัง 1 ครั้ง

State Diagram ที่มี Internal Activity

○ State Diagram การทำงานของลิฟต์



State Diagram & Class Diagram

- วัตถุดิบที่นำมาใช้สร้าง State Diagram คือ Class Diagram และ Sequence Diagram
 - Class diagram จะทำให้เห็นภาพของ class แต่ละ class
 - Function ของ class จะหมายถึง state diagram หนึ่งชุด
 - Attribute จะเป็นตัวที่มีการเปลี่ยนแปลงไปในกรณีใดกรณีหนึ่ง
 - Sequence diagram จะทำให้เห็นภาพกิจกรรมของ class ซึ่งจะใช้เพื่อโต้ตอบกับ class อื่น ๆ ใน problem domain
 - มีส่วนช่วยในการพิจารณาแนวการดำเนินไปของการเปลี่ยน state ของ class หนึ่ง ๆ

หลักการเขียน State Diagram

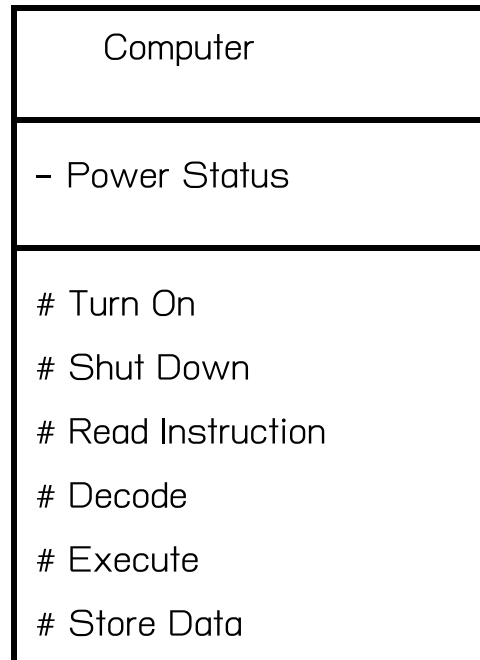
- จำนวนของ state diagram มักจะได้จากจำนวน function ของแต่ละ class รวมกัน
- ไม่จำเป็นต้องเขียน state diagram ของทุก function
- ถ้า function ใดไม่มีกิจกรรมซับซ้อนไม่จำเป็นต้องมี state diagram
- ให้พิจารณาว่าใน class นั้นจะมี state ใดบ้าง โดยไม่คำนึงถึง function ที่มี

หลักการเขียน State Diagram ...

- จาก state ที่มีอยู่ให้เขียน state diagram ของแต่ละ function
- เพิ่ม state ที่จำเป็นเพิ่มเติม เพื่อให้ state diagram สมบูรณ์ขึ้น
- ทำซ้ำจนครบทุก class ใน class diagram

ตัวอย่างการเขียน State Diagram

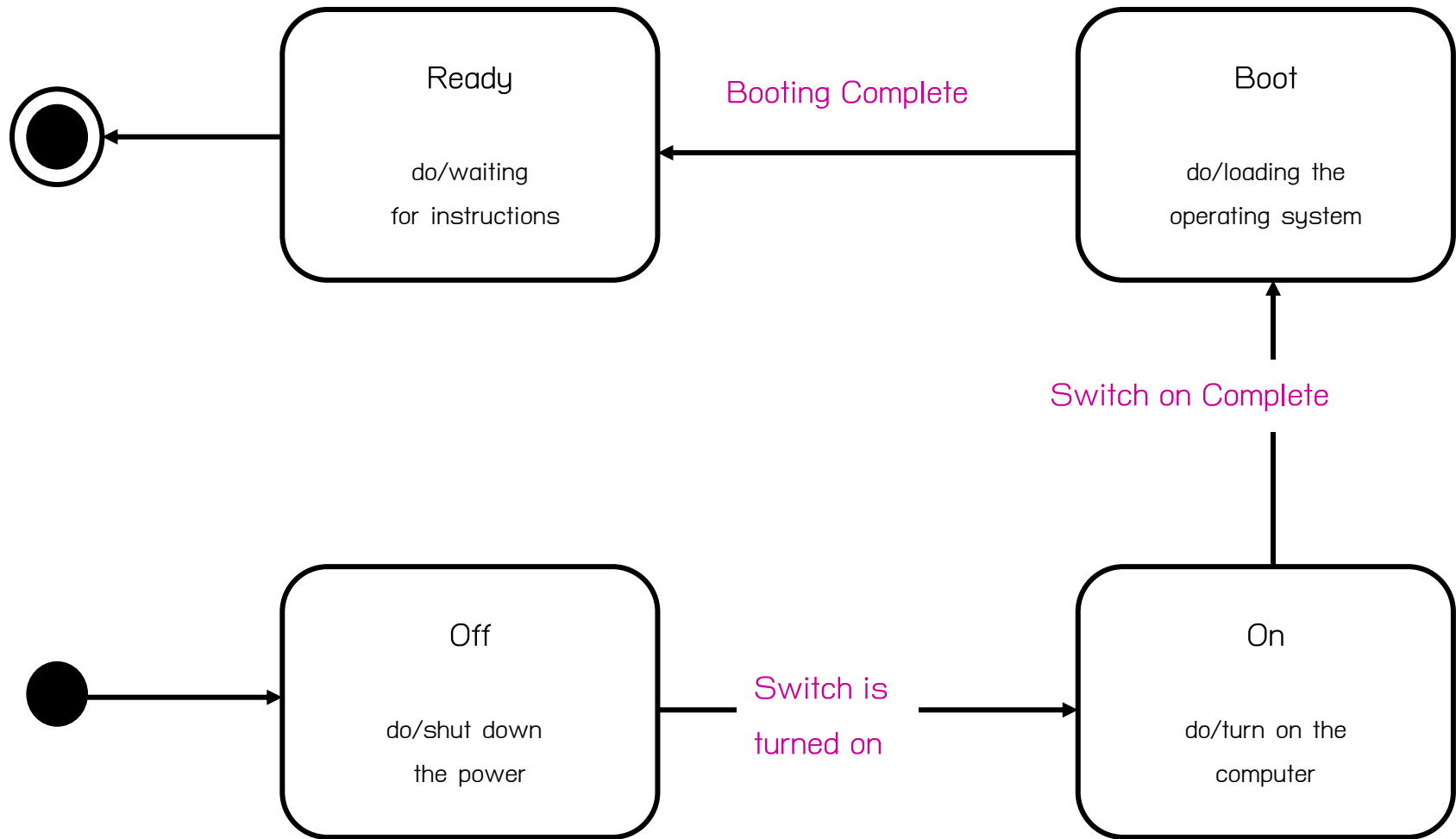
○ State Diagram ของ Class Computer



State ของ Class Computer

- Off
- On
- Boot
- Ready
- Reading
- Sending
- Decoding
- Executing
- Buffering
- Output
- Storing Data

State Diagram : Turn On Function



State Diagram ของ processes บน OS

