Real Time Operating Systems

UNIVERSITY OF IDAHO – JOHN C. SHOVIC, PHD

COEUR D'ALENE

Counting Semaphores

- Binary semaphores can be thought of as queues of length one, counting semaphores can be thought of as queues of length greater than one
- users of the semaphore **are not interested** in the data that is stored in the queue - just whether or not the queue is empty or not.
- Counting semaphores are typically used for two things:
- Counting events.
  - An Event will 'give' a semaphore each time an event
  - Handler task will 'take' a semaphore each time it processes an event (decrementing the semaphore count value). The count value is therefore the difference between the number of events that have occurred and the number that have been processed
- Resource management. In this usage scenario the count value indicates the number of resources available.

Queues and Stream/Message Buffers

Stream and Message Buffers

- Stream buffers are an RTOS task to RTOS task
- Interrupt to task communication
- Optimized for single reader single writer scenarios
- Passing data from an interrupt service routine to a task,
- From one microcontroller core to another on dual core CPUs
- Data is passed by copy - the data is copied into the buffer by the sender and out of the buffer by the read.Stream
- buffers pass a continuous stream of bytes.
- Message buffers pass variable sized but discrete messages

Queues

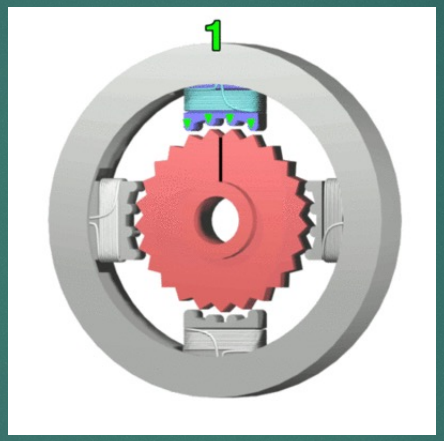
- Queues are the primary form of intertask communications
- Used to send messages between tasks
- Between interrupts and tasks.
- Used as thread safe FIFO (First In First Out) buffers with new data being sent to the back of the queue
- Although data can also be sent to the front.

Stepper Motors

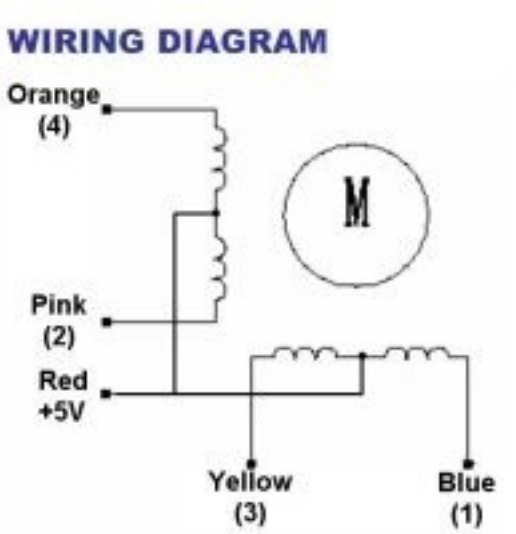
- Brushed DC motors rotate continuously when DC voltage is applied to their terminals
- The stepper motor is known by its property to convert a train of input pulses (typically square wave pulses) into a precisely defined increment in the shaft position.
- Each pulse moves the shaft through a fixed angle.
- Can be commanded to move and hold at one of these steps without any position sensor for feedback (an open-loop controller), as long as the motor is carefully sized to the application in respect to torque and speed.

- Each pulse moves the shaft through a fixed angle.

Stepper Motors



Stepper Motor Sequence



Switching Sequence

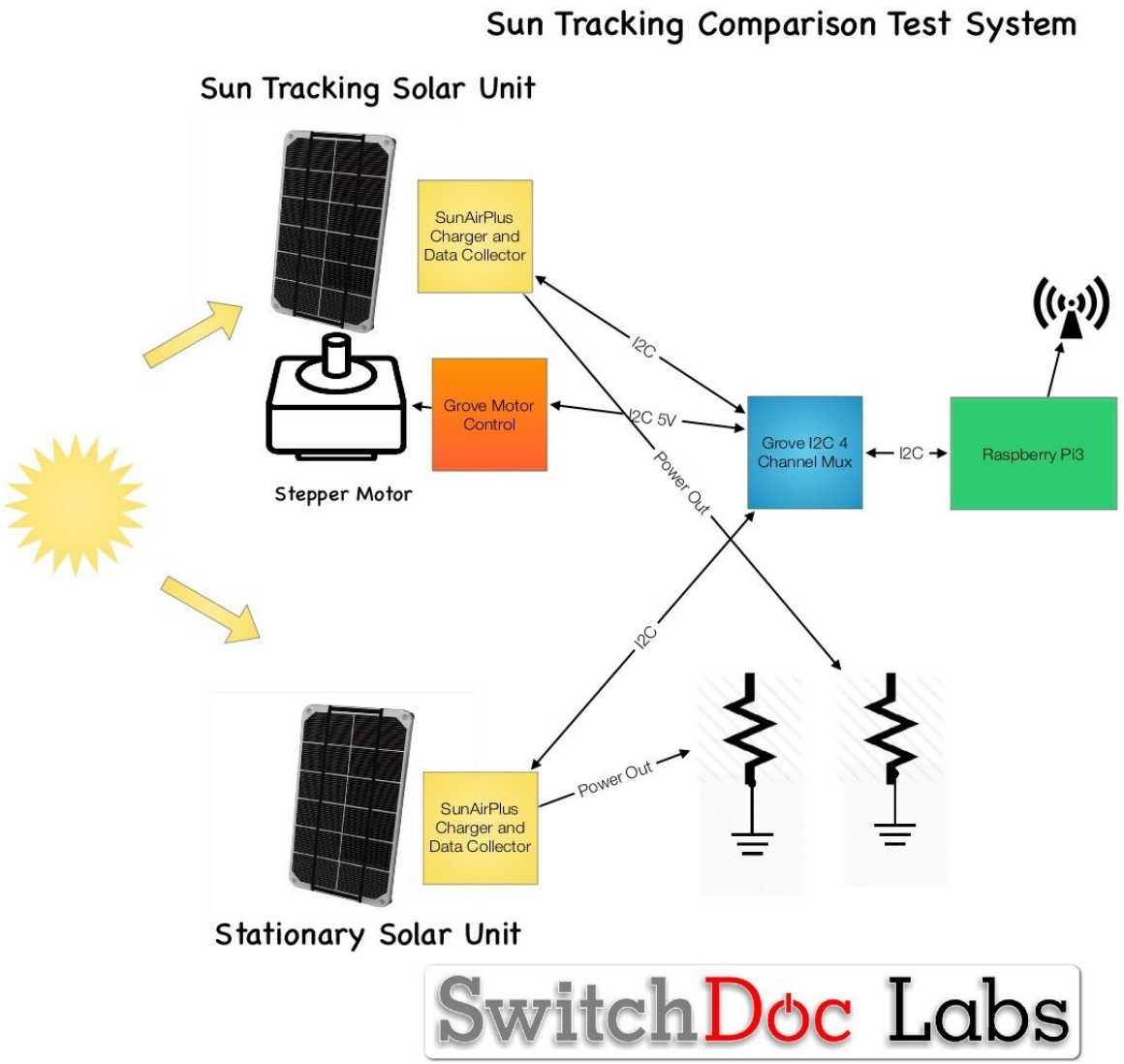
Lead Wire Color	---> CW Direction (1-2 Phase)							
	1	2	3	4	5	6	7	8
4 Orange	-	-						-
3 Yellow		-	-	-				
2 Pink				-	-	-		
1 Blue						-	-	-

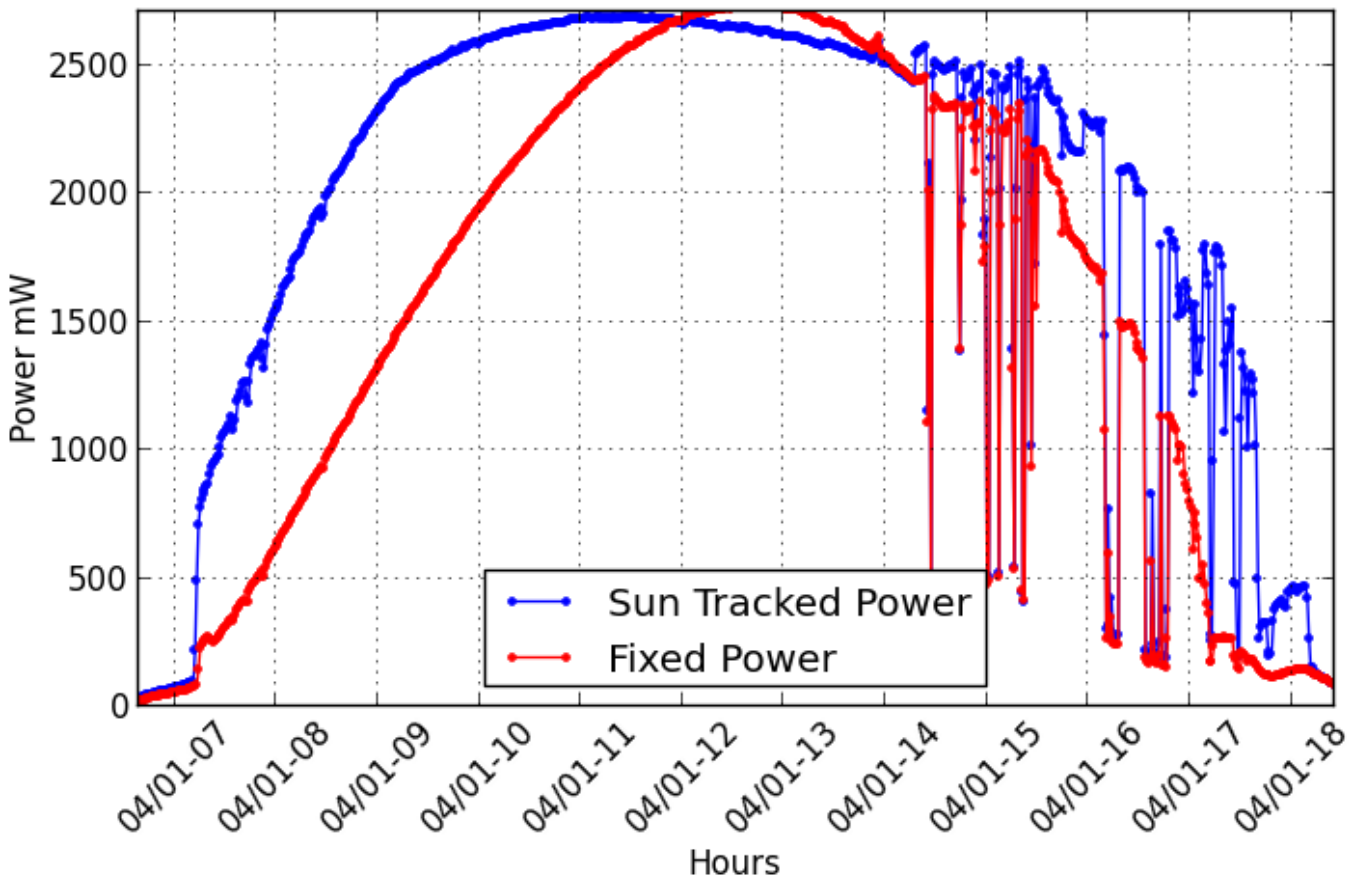
Stepper Motor Driver

- Each input is connected to a GPIO pin (through the motor driver)
- You set all four lines in sequence to go through a step
- Some windings are always powered which give more resistance to torque at the cost of power

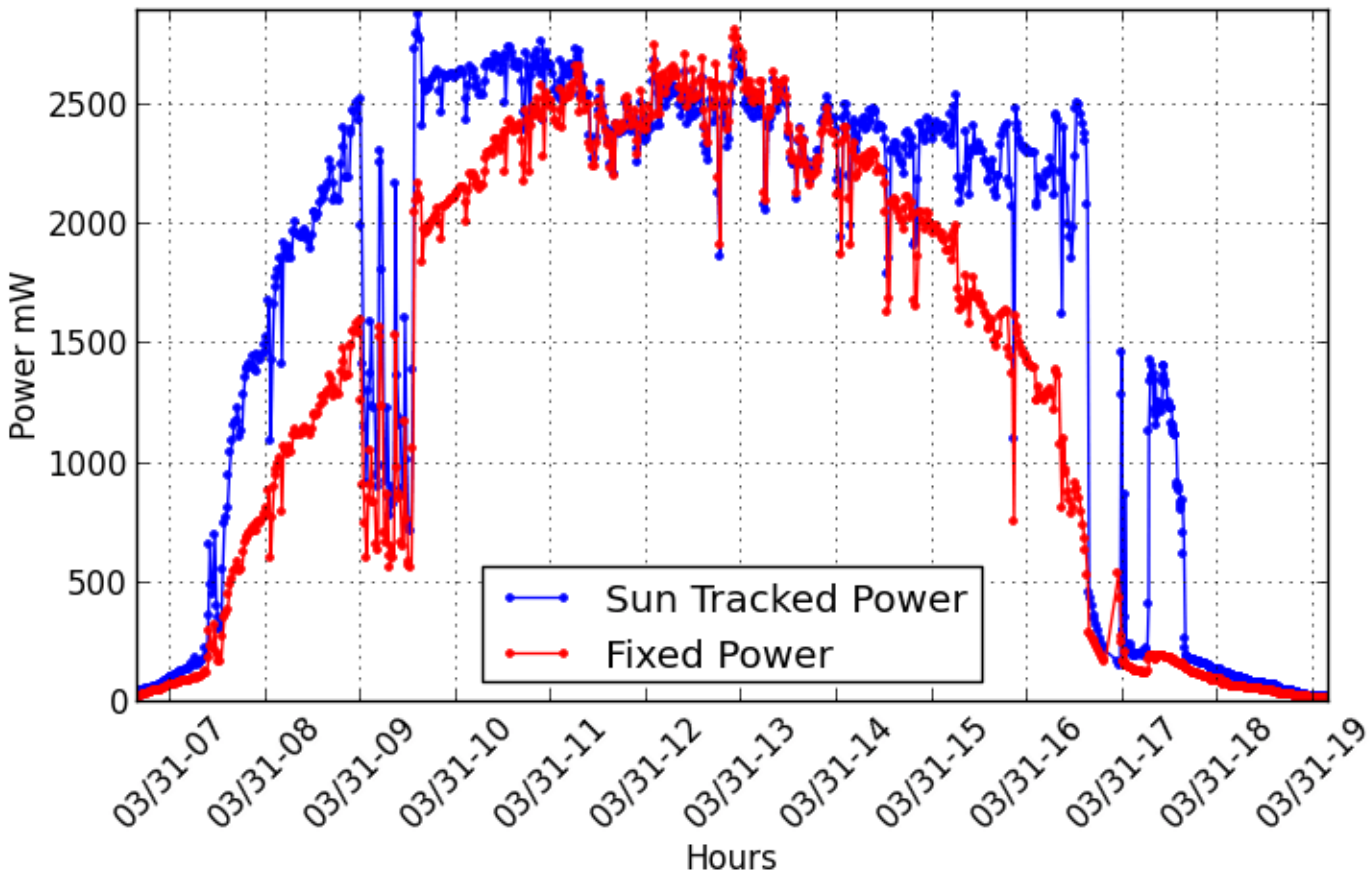
Solar Tracker Project

- Solar Project to empirically determine the amount of increase in solar power afforded by tracking the sun
- Two solar panels.
  - One faced south
  - One tracking the Sun across the sky (using a stepper motor)


Sun Tracking Comparison Test System
Sun Tracking Solar Unit
SunAirPlus Charger and Data Collector
Grove Motor Control
Stepper Motor
I2C
I2C 5V
Power Out
Grove I2C 4 Channel Mux
I2C
Raspberry Pi3
I2C
Power Out
SunAirPlus Charger and Data Collector
Stationary Solar Unit
SwitchDoc Labs



SunTracker Apr 1 (TP= 22.65Wh, UP= 18.30Wh +23.8%)



SunTracker Mar 31 (TP= 21.88Wh, UP= 17.52Wh +24.9%)

Results

- ~23% increase
- With power down on Stepper motors – we only burned about 2% of the savings.

Video

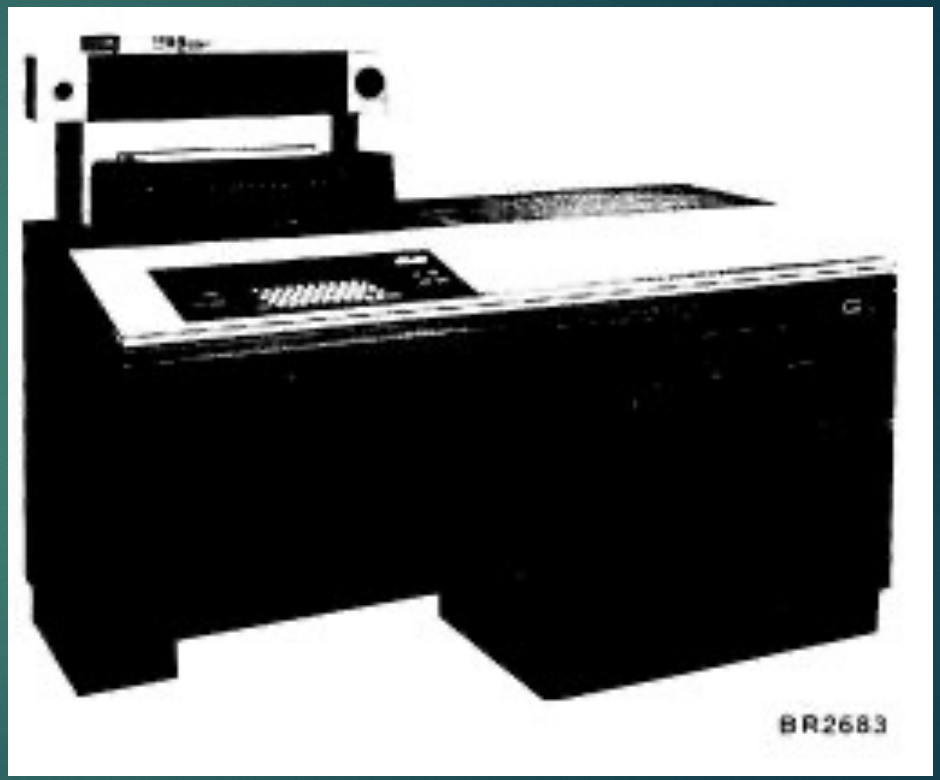
Assignment #6 – The Simple Temperature Test Project

- Due Thursday, February 9th midnight. 10 points
- HDC1080 I2C device
- Connect the HDC1080 (HDC1000 compatible) to the first of the I2C Grove Plugs
- Compile the V3-I2CHDC1080 program (on canvas) and capture the results and upload to Canvas
- Unplug your pico, move the HDC1080 to the second I2C Grove plug
- Now capture the results from the second plug
- Repeat for third I2C plug

- If you get something like this (the "-2" is definitive):
- i2c_write_blocking return=-2
  registeraddress=0XFE
  dev[0]=0X64, dev[1]=0X2
  deviceID=0x6402
- Your port is bad (probably a cold solder joint) and needs to be resoldered.
- If you get this (all values are definitive):
- i2c_write_blocking return=1
  registeraddress=0XFE
  dev[0]=0X54, dev[1]=0X49
  deviceID=0x5449
- Your port is good!


SSD1306

- IBM 1130
- Released in 1965
- 512Kbyte Hard Disk
- 8Kbytes Memory
- $895 a month rental
- $41,280 for purchase
- No Screen - Printer I/O
- Fortran IV
- Processor ~ 277KHz



Dates to Be Remembered

- Everyone: Start thinking about your final project - IOT oriented. Must use WiFi.
- First Test March 9th in class – Hand Written – closed book - essay timed test during class – Emailed to you and then must be handed to TA for scanning and sending to me.