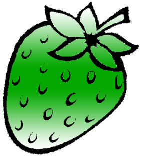
STRAWBERRY



/strawberrydevelopers

/strawberry_app

For more visit:

Strawberrydevelopers.weebly.com

INTERFACING

Richa Upadhyay Prabhu

NMIMS's MPSTME

richa.upadhyay@nmims.edu

February 25, 2016

8255: Programmable Peripheral Interface or Programmable Input output Device

Introduction

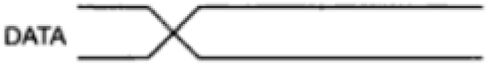
METHODS OF DATA TRANSFER

- Simple Input and Output
- Simple Strobe Input/ Output
- Single Handshake data transfer
- Double Handshake data transfer

Simple Input and Output

I/O operations are not directly dependent on any other signals.

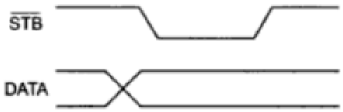
For Example: Reading data from thermostat; it is simple connected to input port line and processor simply reads the port, as data is always available and ready so it can be read any time.



The crossed lines on the waveform represents time at which data byte is available at the port

Strobe Input and Output

- This mode is used when valid data is present on the port only at a certain time, and needs to be read at that time.
- External device sends a strobe signal to indicate valid data is present on the data lines.
- Strobe can be connected to interrupt input
- Transfer is time dependent



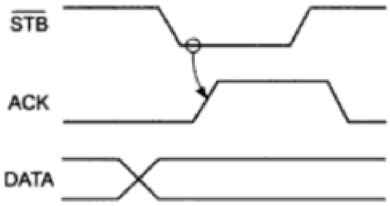
Strobe Input and Output

- High speed data transfer not possible
- data rates of sending system and receiving system can be different
- to avoid this HANDSHAKE data transfer is used

Single handshake I/O

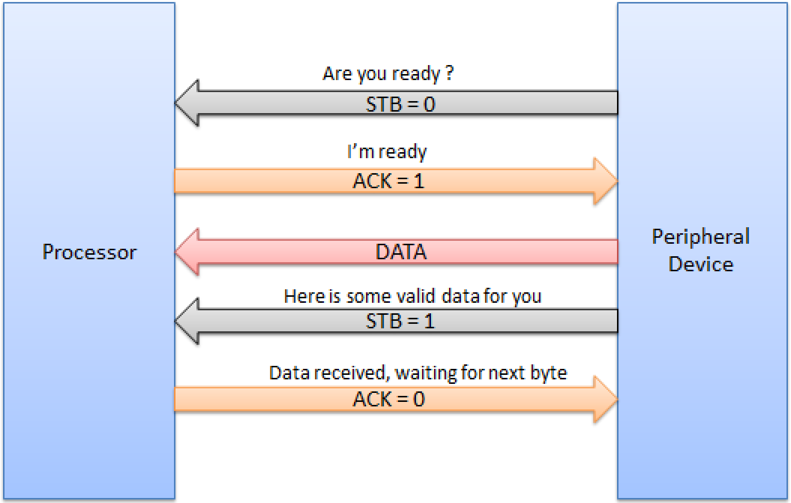
Handshake Data Transfer

- Peripheral device outputs some data and sends strobe signal to processor
- Processor detects strobe on poll or interrupt basis and reads the data
- Also it sends an Acknowledge signal to device indicating that data has been read and next byte of data can be sent.

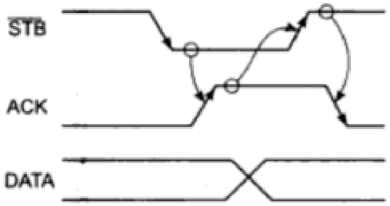


Double Handshake Data Transfer

- Used where more co-ordination is required



Double Handshake Data Transfer

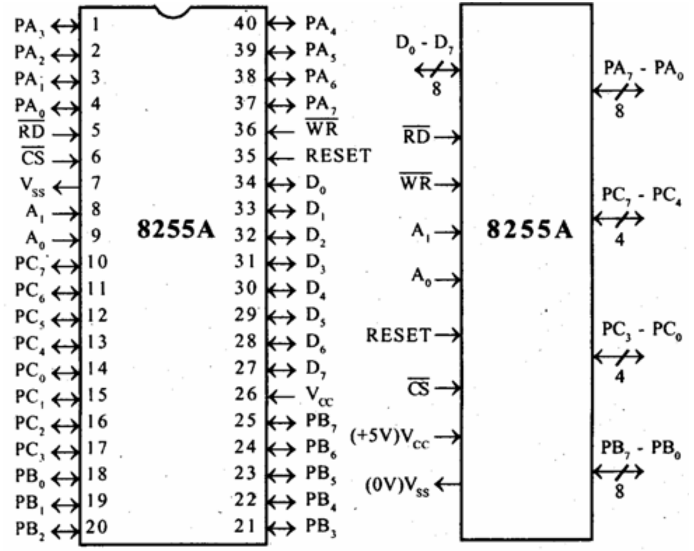


Handshaking is a good idea but uses too much processor time, so parallel port devices such as 8255 are used to automatically manage handshake operation.

8255 Programmable Peripheral Interface

- It is a programmable parallel I/O device
- 40 pin IC
- 24 programmable I/O pins arranged as two 8-bit ports and two 4-bit ports
- It has Three 8-bit ports: Port A,Port B and Port C, which are arranged in two groups of 12 pins
- Control register defines the functions of each I/O port and in which mode they should operate

Pin Diagram


PA3 1
PA2 2
PA1 3
PA0 4
RD 5
CS 6
Vss 7
A1 8
A0 9
PC7 10
PC6 11
PC5 12
PC4 13
PC0 14
PC1 15
PC2 16
PC3 17
PB0 18
PB1 19
PB2 20
40 PA4
39 PA5
38 PA6
37 PA7
36 WR
35 RESET
34 D0
33 D1
32 D2
31 D3
30 D4
29 D5
28 D6
27 D7
26 Vcc
25 PB7
24 PB6
23 PB5
22 PB4
21 PB3
8255A
D0 - D7
8
RD
WR
A1
A0
RESET
CS
(+5V)Vcc
(0V)Vss
PA7 - PA0
PC7 - PC4
4
PC3 - PC0
PB7 - PB0

Pin Diagram Description

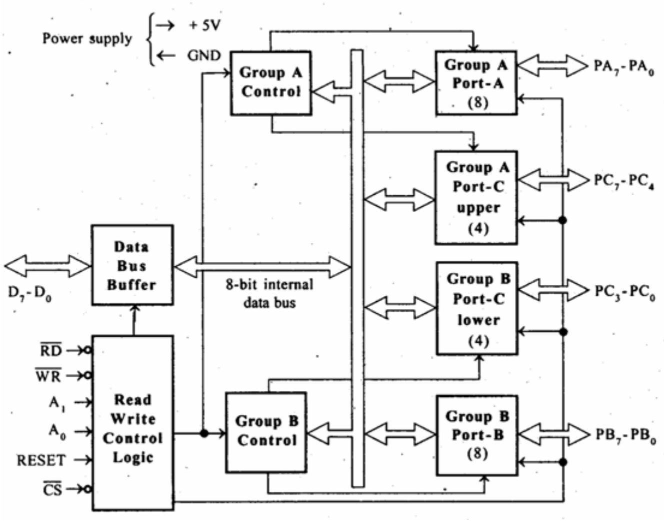
PA_0- PA_7	: Port A; 8-bit bidirectional I/O pins
PB_0- PB_7	: Port B; 8-bit bidirectional I/O pins
PC_0- PC_7	: Port C; 8-bit bidirectional I/O pins. These can even be used as two 4-bit ports ($PC_0 - PC_3$ and $PC_4 - PC_7$) , Also used to produce handshake signals for A and B
$D_0 - D_7$	: 8-Bit bi-directional data bus connected to processor
RESET	: When reset clears control word and all ports
$\overline{RD}$	: Read, active low input line used to read data from processor
$\overline{WR}$	: Write, active low input line used to send data to processor

Pin Diagram Description

$A_0 - A_1$: Address Lines; used to select ports of 8255

A_0	A_1	Selected Port
0	0	Port A
0	1	Port B
1	0	Port C
1	1	Control Register

8255 Functional Block Diagram



8255 Operational Modes and Initialization

MODE 0

- simple input or output without handshaking
- If Port A and B are initialized in mode 0, port C can be used as 8-bit port
- If port C lines are used as output, individual bits can be set and reset by special control word
- Two halves of port C (4-bits)are independent
- One can be used as input and other as output

8255 Operational Modes and Initialization

MODE 1

- Use port a or B as handshake(strobed) input/output
- PC_0, PC_1, PC_2 lines of port C are used as handshake lines for port B
- PC_3, PC_4, PC_5 lines of port C are used as handshake lines for port A(input port)
- PC_6, PC_7 lines are used as input output lines
- PC_3, PC_6, PC_7 lines of port C are used as handshake lines for port A(output port)
- PC_4, PC_5 lines are used as input output lines

8255 Operational Modes and Initialization

MODE 2

- Bi-directional handshake data transfer
- Data can be output or input on the same eight lines
- If port A is used in mode 2 :
  - PC_3 to PC_7 are used as handshake line for port A
  - PC_0 to PC_2 can be used as I/O pins if Port B is in Mode 0
  - Or PC_0 to PC_2 are used as port B handshake if Port B is in Mode 1

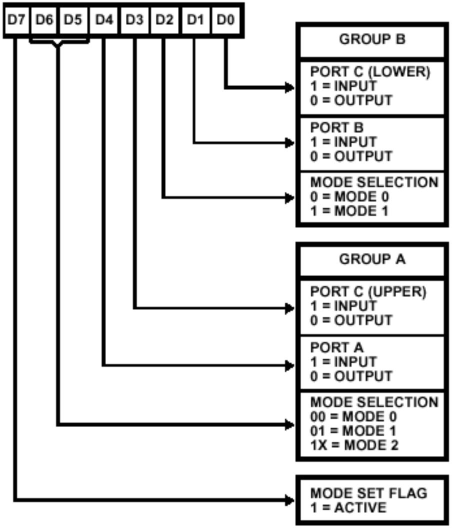
8255 Control word (8-bit)

There are two types of Control word:

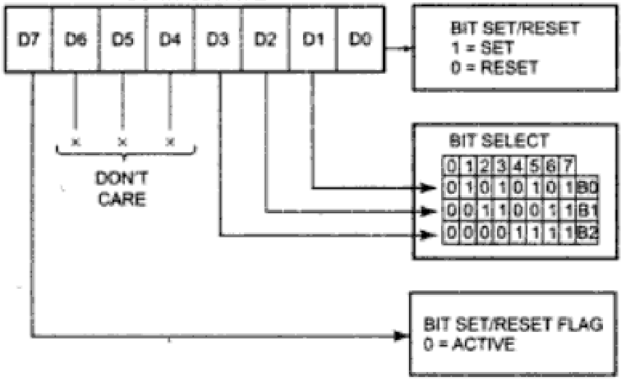
The MSB tells which control word is being sent

1. Mode definition control word (if MSB = 1)
2. Port C bit set/reset control word (if MSB = 0)

Mode definition control word (if MSB = 1)



Port C bit set/reset control word (if MSB = 0)



8254 Software Programmable Timer/Counter

Why 8253/8254?
Since, it is not possible to generate accurate time delays Using delay subroutines in 8086.

8254/8253 facilitates:

- Accurate time delays
- Minimizes load on processor
- real time clock
- event counter
- square wave generator
- complex waveform generator

8254/8253

Software Programmable Timer/Counter means, we can load count in them, start them and stop them with instructions in the program.
To program such device, COUNT BYTES and CONTROL BYTES have to be sent to the timer/counter peripheral device

8254 vs 8253

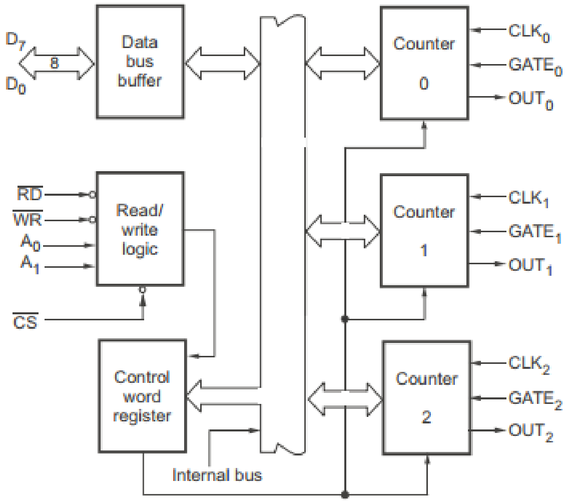
8253	8254
clock frequency 2.6MHz	clock frequency 8MHz
No read back feature	Read back feature

Read back feature allows to latch the count in all counters and the status of counters at any point.

Features of 8254

- Three independent 16-bit down counters.
- 8254 can handle inputs from DC to 10 MHz (5MHz 8254-5, 8MHz 8254, 10MHz 8254-2) where as 8253 can operate upto 2.6 MHz
- Three counters are identical presettable, and can be programmed for either binary or BCD count.
- Counter can be programmed in six different modes.
- Compatible with all Intel and most other microprocessors.
- 8254 has powerful command called READ BACK command which allows the user to check the count value, programmed mode and current mode and current status of the counter.

8254 block diagram



8254 block diagram

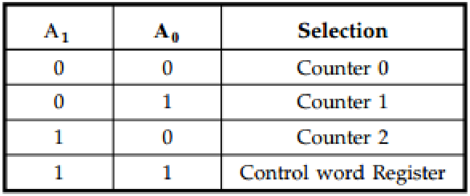
Data Bus Buffer :

- Bi-directional, 8-bit buffer is used to interface the 8253/54 to the system data bus.
- The Data bus buffer has three basic functions:
  - Programming the modes of 8253/54.
  - Loading the count registers.
  - Reading the count values.

8254 block diagram

Read/Write Logic : The Read/Write logic has five signals : $\overline{RD}$,$\overline{WR}$, $\overline{CS}$ and the address lines A0 and A1

- In the peripheral I/O mode, the $\overline{RD}$, and $\overline{WR}$ signals are connected to $\overline{IOR}$ and $\overline{IOW}$, respectively. In memory-mapped I/O, these are connected to $\overline{MEMR}$ and $\overline{MEMW}$



A_1	A_0	Selection
0	0	Counter 0
0	1	Counter 1
1	0	Counter 2
1	1	Control word Register

8254 block diagram

Control Word Register :

- This register is accessed when lines A0 and A1 are at logic 1
- used to write a command word which specifies the counter to be used (binary or BCD), its mode, and either a read or write operation.

8254 block diagram

Counters :

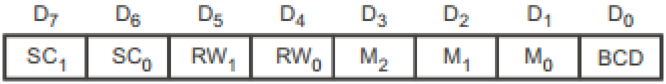
- These three functional blocks are identical in operation.
- Each counter consists of a single, 16 bit, pre-settable, down counter.
- Can operate in either binary or BCD
- Its input, gate and output are configured by the selection of modes stored in the control word register
- Programmer can read the contents of any of the three counters without disturbing the actual count in process

8254 block diagram

Counters :

CLK	: Signals of frequency between 0 to 8 MHz can be applied as input to these 3 pins
GATE	: Allows to start or stop the counter with an external hardware signal. If GATE $=$ 1, counter is enabled for countig and if GATE $=$ 0 it is disabled.
OUT	: Output signal of each counter appears on this pin.

8254 Control Word Format



D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	SC_0	RW_1	RW_0	M_2	M_1	M_0	BCD

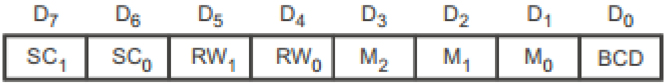
8254 Control Word Format

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	SC_0	RW_1	RW_0	M_2	M_1	M_0	BCD

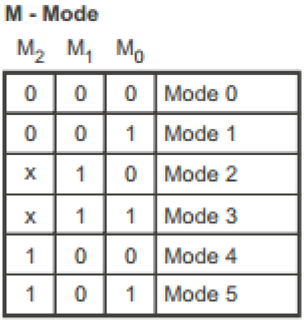
SC - Select counter

SC_1	SC_0	
0	0	Select counter 0
0	1	Select counter 1
1	0	Select counter 2
1	1	Illegal for 8253 Read -Back command for 8254 (See Read operations)

8254 Control Word Format



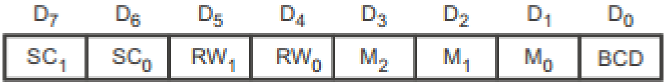
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	SC_0	RW_1	RW_0	M_2	M_1	M_0	BCD



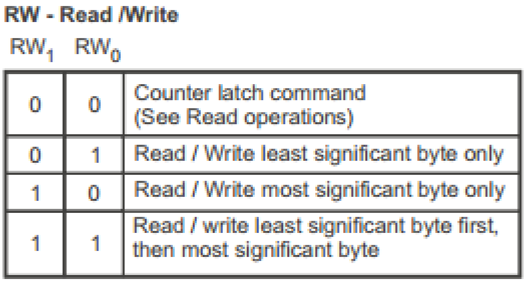
M - Mode

M_2	M_1	M_0	
0	0	0	Mode 0
0	0	1	Mode 1
x	1	0	Mode 2
x	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

8254 Control Word Format



D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	SC_0	RW_1	RW_0	M_2	M_1	M_0	BCD



RW - Read /Write

RW_1	RW_0	
0	0	Counter latch command (See Read operations)
0	1	Read / Write least significant byte only
1	0	Read / Write most significant byte only
1	1	Read / write least significant byte first, then most significant byte

8254 Control Word Format

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	SC_0	RW_1	RW_0	M_2	M_1	M_0	BCD

BCD :

0	Binary counter 16 - bits
1	Binary coded decimal (BCD) Counter (4 Decades)

8254 Counter Modes

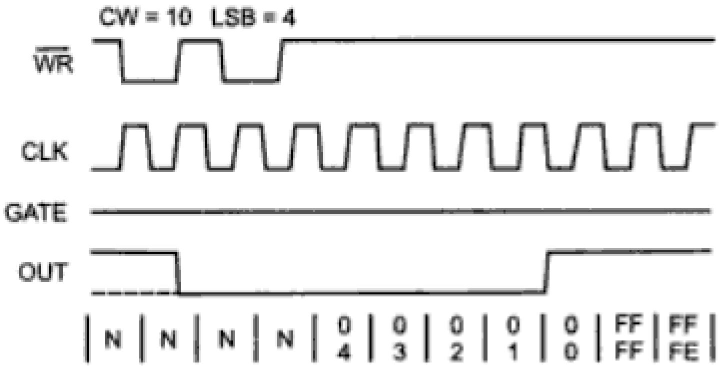
1. Mode 0 (Interrupt On Terminal Count)
2. Mode 1 (Hardware retriggerable Monoshot)
3. Mode 2 (Timed interrupt Generator)
4. Mode 3 (Square Wave Generator)
5. Mode 4 (Software Triggered Strobe)
6. Mode 5 (Hardware Triggered Strobe)

Mode 0 (Interrupt On Terminal Count)

- The output becomes a logic 0 when the control word is written remains low even after count value loaded in counter.
- Counter starts decrementing after falling edge of clock The OUT goes high upon reaching the terminal count and remains high till reloading OUT can be used as interrupt
- Writing a count register , when previous counting is in process
  - first byte when loaded stops the previous count,
  - second byte when loaded starts new count
- Gate high = normal counting
- Gate low = counting terminated and current count latched till GATE goes high again

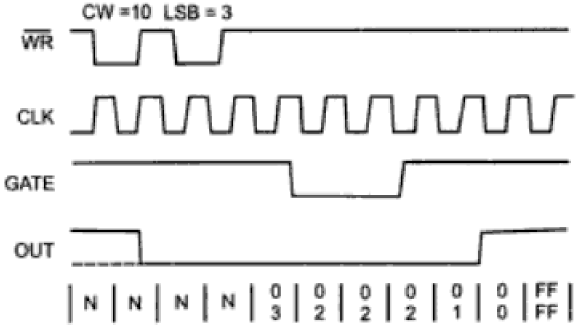
Mode 0 (Interrupt On Terminal Count)

Assuming, CW = 10 ; counter 0 initialized for binary counting, mode 0 and read/write of only LSB



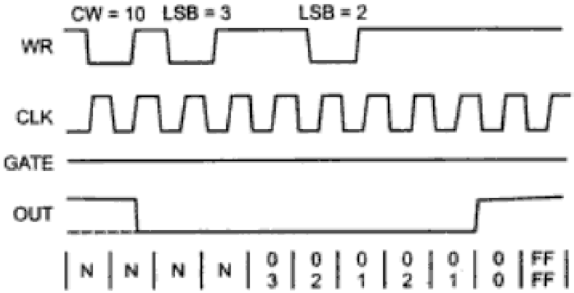
Mode 0 (Interrupt On Terminal Count)

When GATE = 0, counter is held and when GATE = 1, counter continues



Mode 0 (Interrupt On Terminal Count)

If new count is written, counter is loaded with this value in the next clock pulse

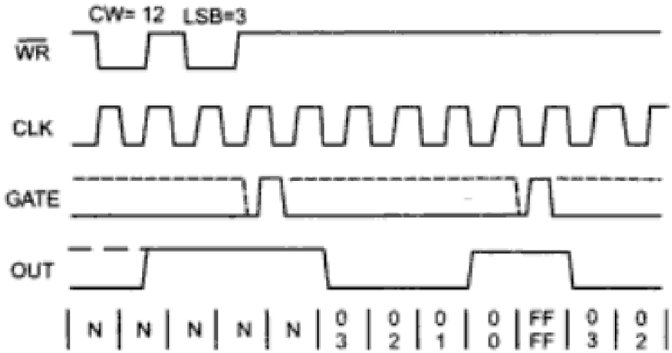


Mode 1 (Hardware retriggerable Monoshot)

- Gate input is used as trigger input
- Output remains high till the count is loaded after application of trigger, output goes low and remains low till count becomes zero
- Another count loaded, when output already low, it does not disturb counting until a new trigger is applied at the gate
- New counting starts after new trigger pulse

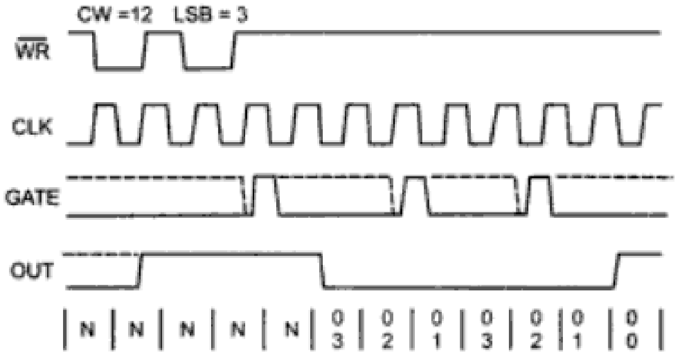
Mode 1 (Hardware retriggerable Monoshot)

For mode 1 GATE act as trigger input, When GATE =1 count is transferred from count register to actual counter



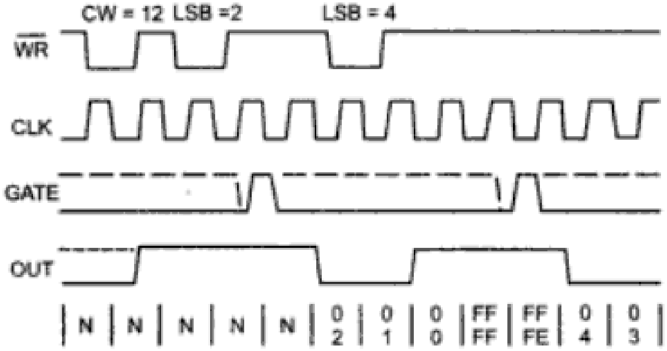
Mode 1 (Hardware retriggerable Monoshot)

This waveform demonstrates retriggerable behavior
If another trigger pulse comes,the count is reloaded and output will remain low



Mode 1 (Hardware retriggerable Monoshot)

New count will not be loaded

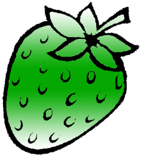


Do it yourself

Assignment - 2

Study all the other 5 modes and prepare a document which will contain theory and related figures.

STRAWBERRY



/strawberrydevelopers

/strawberry_app

For more visit:

Strawberrydevelopers.weebly.com