

# **ENGI E1102 Departmental Project:** Computer Science/Computer Engineering Fully-Functional Calculator -> An RPN Calculator

---

David Figueroa and Justin Zhao

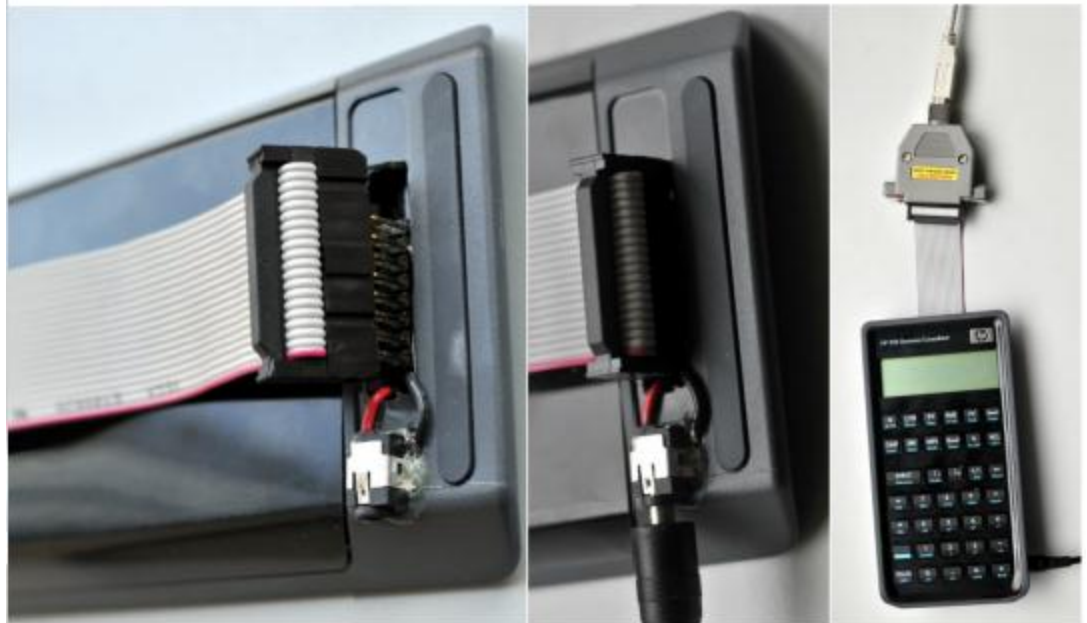
May 4, 2012

Professor Stephen Edwards

# HP -20b Financial Calculator Set-Up



Professor Edwards  
Xcode, C  
make flash



# The Processor

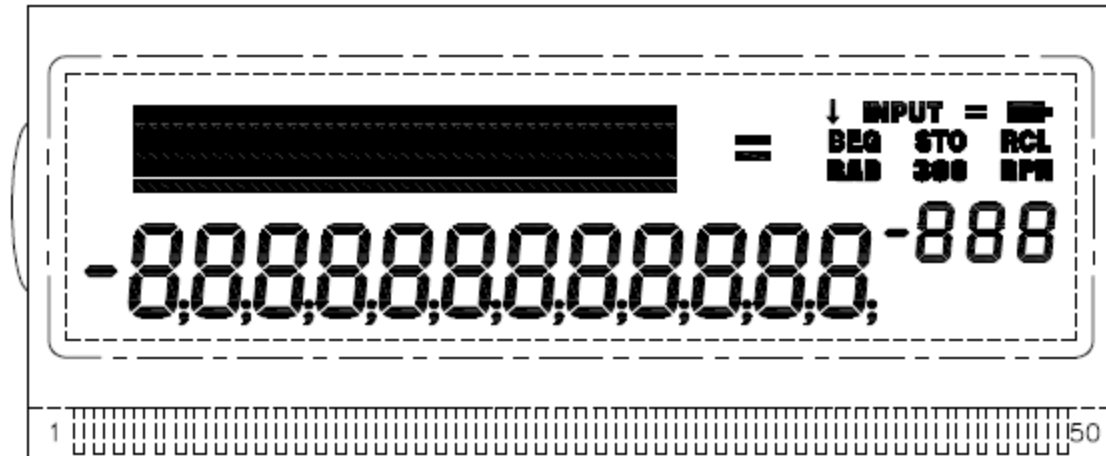


# LAB 1

---

Getting Started: Hello World

# LCD Display



# Software Details

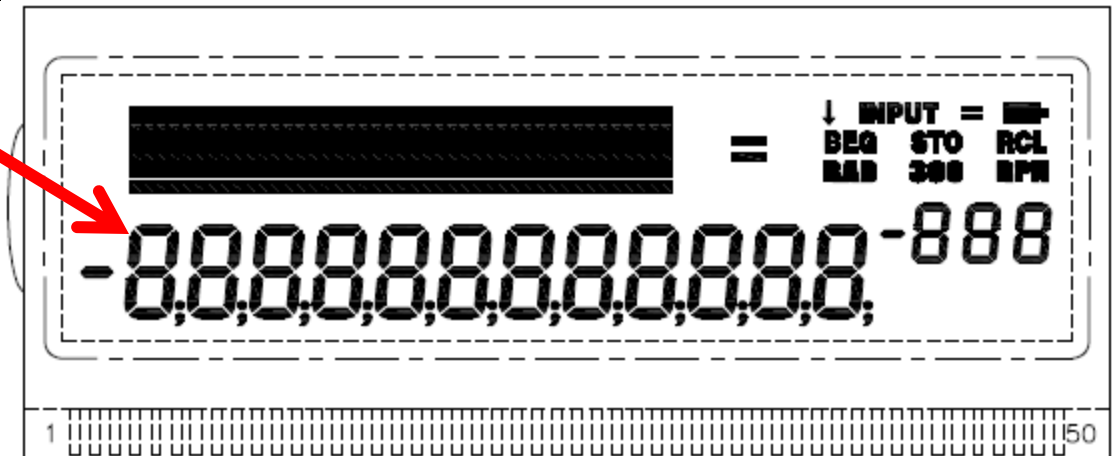
Library functions: lcd\_put\_char7, lcd\_print7

- int lcd\_display\_int(int toDisplay) in main.c

Steps:

- Count digits
- Print number left to right
- Check for negativity

```
for (i=1; i<=digits; i++) {  
    lcd_put_char7((abs(toDisplay)%10)+48, digits-i+1);  
    toDisplay = toDisplay/10;  
}
```

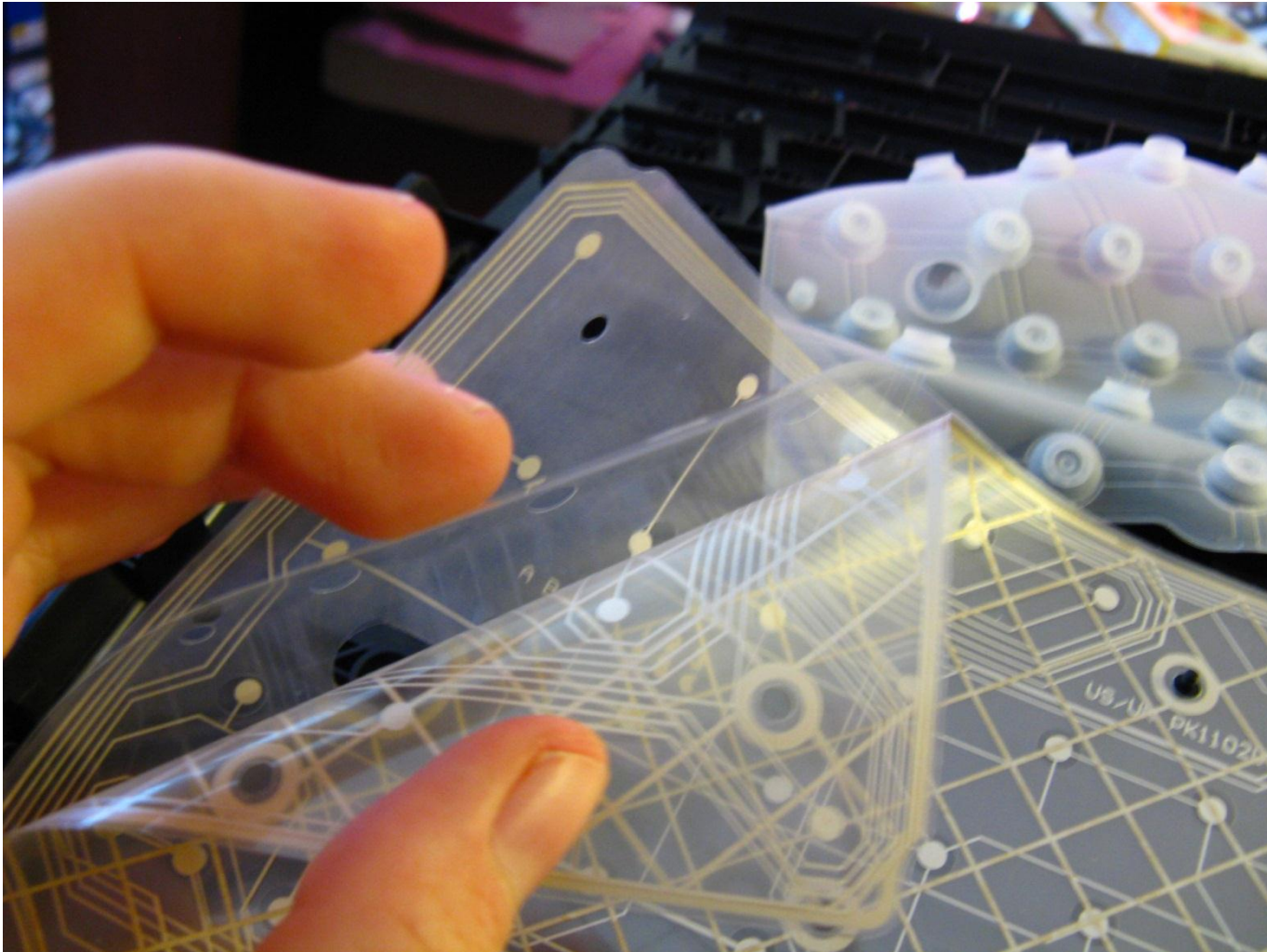


# LAB 2

---

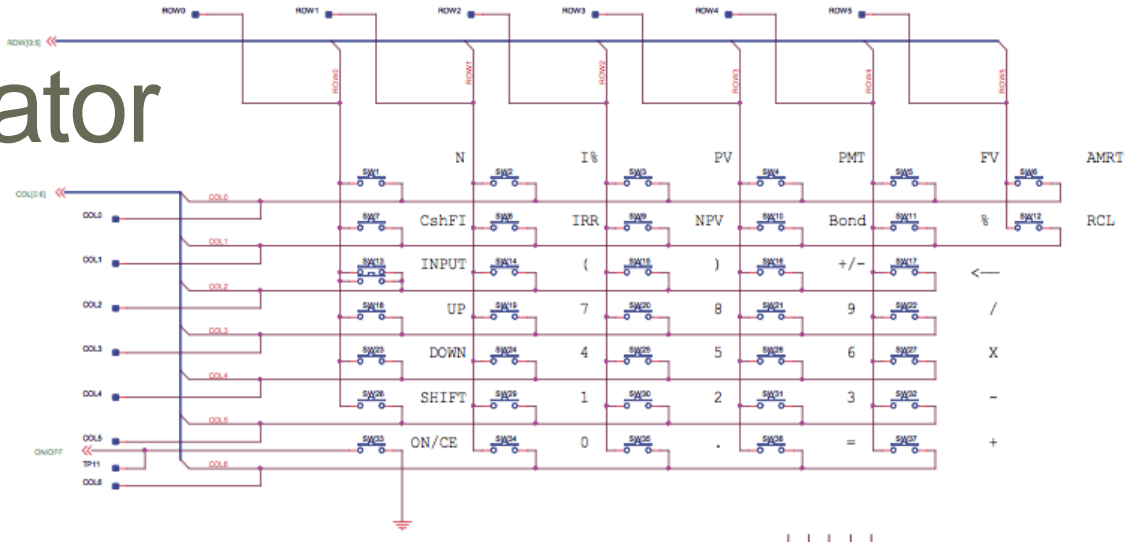
Listening to the Keyboard

# Keyboard





# On the Calculator



- Rows = Columns
- Unique correspondence



# LAB 3

---

Entering and Displaying Numbers

# Displaying Numbers

- Processor speed vs. finger speed
- Additional functionality possibilities
- Keys are assigned meaning

	Row0	Row1	Row2	Row3	Row4	Row5	
	PC11	PC12	PC13	PC14	PC15	PC26	
Col0	PC0	N	I/YR	PV	PMT	FV	Amort
Col1	PC1	CshFI	IRR	NPV	Bond	%	RCL
Col2	PC2	INPUT	(	)	+/-	<-	
Col3	PC3	UP	7	8	9	/	
Col4	PC4	DOWN	4	5	6	*	
Col5	PC5	SHIFT	1	2	3	-	
Col6	PC6		0	.	=	+	

# Software Details

- `int main()`
- `int keyboard_key()` – from lab 2
- `void keyboard_get_entry(struct entry *result)`
- `int get_number(int storedNumber[], int lcdPos, int negative)`

“\r”, “+”, “-”, “\*”, or “/”

- `result -> number`
- `result -> operation`

# LAB 4

---

An RPN Calculator

# An RPN Calculator

<b>J</b>
<b>I</b>
<b>H</b>
<b>G</b>
<b>F</b>
<b>E</b>
<b>D</b>
<b>C</b>
<b>B</b>
<b>A</b>

Operation	Standard Notation	Reverse Polish Notation	Key Presses	Result
7 added to 13	$7 + 13$	7 13 +	7 ^ 13 +	20
2 subtracted from 11	$11 - 2$	11 2 -	11 ^ 2 -	9
2 multiplied by the product of 3 and 5	$2 * (3 * 5)$	2 3 5 * *	2 ^ 3 ^ 5 * *	30

# Software Details

```
#define STACK_SIZE 10
```

```
int main() in main.c
```

Error considerations:

2 ^ 3 ^ +

12 \*

^

1 ^ 2 ^ 3 ^ 4 ^ 5 ^ 6 ^ 7 ^ 8 ^ 9 ^ 10 ^



```
if (index > 1) {  
    switch (entry.operation) {  
        case '\r':  
            break;  
        case '+':  
            index--;  
            stack[index] = stack[index] + stack[index+1];  
            break;  
        case '-':  
            index--;  
            stack[index] = stack[index] - stack[index+1];  
            break;  
        case '*':  
            index--;  
            stack[index] = stack[index] * stack[index+1];  
            break;  
    } //end switch  
} else {  
    lcd_print7("ERROR");  
}
```



# Lessons Learned

## Procrastination



the longer you wait the worse it gets

Copyright 2004 Health Charge 800-540-5131 www.pathtoby.com

