

# Instructions for phase 4 Demo

Jeremy Case, Xinhua Fan, Teodor Georgiev, Julie Yu

## Part 1 - Floating Point and Vector Operations

### Design note:

- In our program, we intend to represent a floating point number as  $\text{mantissa}^{\text{exponent}}$ , such as  $100 = 10^2$ ,  $16 = 4^2$ ,  $160 = 12^2$  (take the closest one 144 for 160);
- In FR, the representation format is:



- So, E.g.,  $100 = 10^2$  will be populated as below in FR

```
[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0]
```

### Deliverable description:

Demonstrate floating point add/subtract, vector add/subtract, and floating point conversion.

### Demo Steps (Please reopen the GUI if the result is not expected):

#### 1. FADD

**Case 1** -  $\text{fr} \leftarrow c(\text{EA})$

s1. input 20 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory	<input type="text" value="20"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	---	----------------------------------	---------------------------------------

s3. input 81 to FR0 -> accept

FR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1]"/>	<input type="text" value="81.0"/>	<input type="button" value="Accept"/>
-----	---	-----------------------------------	---------------------------------------

s4. select relevant values for FADD (IX = 0; GPR = 0; I = 0; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions	FADD	No...	GPR0	IA ...	20
	[0, 1, 1, 0, 1, 1]	[0, 0]	[0, 0]	[0]	[1, 0, 1, 0, 0]
Accept Manual Word Instruction					

**[Expected Result]** -> FR0 is populated with 169.0 ( $13^2$ , the closest number for 181)

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1]	169.0
-----	--	-------

**Case 2** -  $fr \leftarrow c(c(EA))$ , if I bit set

s1. input 20 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory	20	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	---	-----	--------

s3. input 81 to FR0 -> accept

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1]	81.0	Accept
-----	--	------	--------

s4. input 5 to IX3 -> accept

IX3	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1]	5	Accept
-----	--	---	--------

s5. select relevant values for FADD (IX = 3; GPR = 0; I = 1; Address = 15)

in manual input part -> accept manual word instruction

Manual Instructions	FADD	IX...	GPR0	IA ...	15
	[0, 1, 1, 0, 1, 1]	[1, 1]	[0, 0]	[1]	[0, 1, 1, 1, 1]
Accept Manual Word Instruction					

**[Expected Result]** -> FR0 is populated with 169.0 ( $13^2$ , the closest number for 181)

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1]	169.0
-----	--	-------

## 2. FSUB

**Case 1** - fr <- c(EA)

s1. input 20 to memory location -> accept

s2. input 96 to memory location's value -> accept

Memory	20	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0]	96	Accept
--------	----	--------	--	----	--------

s3. input 196 to FR0 -> accept

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0]	196.0	Accept
-----	--	-------	--------

s4. select relevant values for FSUB (IX = 0; GPR = 0; I = 0; Address = 20)

in manual input part -> accept manual word instruction

Manual Instructions	FSUB	No...	GPR0	IA ...	20
	[0, 1, 1, 1, 0, 0]	[0, 0]	[0, 0]	[0]	[1, 0, 1, 0, 0]
Accept Manual Word Instruction					

**[Expected Result]** -> FR0 is populated with 100.0 ( $10^2$ )

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0]	100.0
-----	--	-------

**Case 2** - fr <- c(c(EA)), if I bit set

s1. input 20 to memory location -> accept

s2. input 96 to memory location's value -> accept

Memory	20	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0]	96	Accept
--------	----	--------	--	----	--------

s3. input 196 to FR0 -> accept

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0]	196.0	Accept
-----	--	-------	--------

s4. input 5 to IX3 -> accept

IX3	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1]	5	Accept
-----	--	---	--------

s5. select relevant values for FSUB (IX = 3; GPR = 0; I = 1; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions	FSUB	IX...	GPR0	IA ...	15
	[0, 1, 1, 1, 0, 0]	[1, 1]	[0, 0]	[1]	[0, 1, 1, 1, 1]
Accept Manual Word Instruction					

**[Expected Result]** -> FR0 is populated with 100.0 ( $10^2$ )

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0]	100.0
-----	--	-------

### 3. VADD (please reopen GUI before testing)

**Case 1** - I bit not set

s1. input 4 to FR0 -> accept

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0]	4.0	Accept
-----	--	-----	--------

s2. input 10 to memory location -> accept

s3. input 100 to memory location's value -> accept

Memory	10	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	---	-----	--------

s4. input 11 to memory location -> accept

s5. input 100 to memory location's value -> accept

Memory	11	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	---	-----	--------

s6. input 12 to memory location -> accept

s7. input 100 to memory location's value -> accept

Memory	12	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	---	-----	--------

s8. input 13 to memory location -> accept

s9. input 100 to memory location's value -> accept

Memory	13	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	---	-----	--------

s10. input 14 to memory location -> accept

s11. input 100 to memory location's value -> accept

Memory	14	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s12. input 15 to memory location -> accept

s13. input 100 to memory location's value -> accept

Memory	15	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s14. input 16 to memory location -> accept

s15. input 100 to memory location's value -> accept

Memory	16	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s16. input 17 to memory location -> accept

s17. input 100 to memory location's value -> accept

Memory	17	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s18. select relevant values for VADD (IX = 0; GPR = 0; I = 0; Address = 10)

in manual input part -> accept manual word instruction

Manual Instructions	VADD	No...	GPR0	IA ...	10
	[0, 1, 1, 1, 0, 1]	[0, 0]	[0, 0]	[0]	[0, 1, 0, 1, 0]
Accept Manual Word Instruction					

**[Expected Result] ->**

a.  $M(10) = M(10) + M(14) = 200 (100 + 100)$

Memory	10	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

b.  $M(11) = M(11) + M(15) = 200 (100 + 100)$

Memory	11	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

c.  $M(12) = M(12) + M(16) = 200 (100 + 100)$

Memory	12	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

d.  $M(13) = M(13) + M(17) = 200 (100 + 100)$

Memory	13	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

## Case 2 - I bit set

s1. input 4 to FR0 -> accept

FR0	<input type="text" value="0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0"/>	<input type="text" value="4.0"/>	<input type="button" value="Accept"/>
-----	--	----------------------------------	---------------------------------------

s2. input 10 to memory location -> accept

s3. input 100 to memory location's value -> accept

Memory	<input type="text" value="10"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s4. input 11 to memory location -> accept

s5. input 100 to memory location's value -> accept

Memory	<input type="text" value="11"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s6. input 12 to memory location -> accept

s7. input 100 to memory location's value -> accept

Memory	<input type="text" value="12"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s8. input 13 to memory location -> accept

s9. input 100 to memory location's value -> accept

Memory	<input type="text" value="13"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s10. input 14 to memory location -> accept

s11. input 100 to memory location's value -> accept

Memory	<input type="text" value="14"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s12. input 15 to memory location -> accept

s13. input 100 to memory location's value -> accept

Memory	<input type="text" value="15"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s14. input 16 to memory location -> accept

s15. input 100 to memory location's value -> accept

Memory	<input type="text" value="16"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s16. input 17 to memory location -> accept

s17. input 100 to memory location's value -> accept

Memory	17	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s18. input 5 to IX3 -> accept

IX3	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1]	5	Accept
-----	---	---	--------

s19. select relevant values for VADD (IX = 3; GPR = 0; I = 1; Address = 5)

in manual input part -> accept manual word instruction

Manual Instructions	VADD	IX...	GPR0	IA ...	5
	[0, 1, 1, 1, 0, 1]	[1, 1]	[0, 0]	[1]	[0, 0, 1, 0, 1]
Accept Manual Word Instruction					

**[Expected Result] ->**

a.  $M(10) = M(10) + M(14) = 200 (100 + 100)$

Memory	10	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

b.  $M(11) = M(11) + M(15) = 200 (100 + 100)$

Memory	11	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

c.  $M(12) = M(12) + M(16) = 200 (100 + 100)$

Memory	12	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

d.  $M(13) = M(13) + M(17) = 200 (100 + 100)$

Memory	13	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

#### 4. VSUB (please reopen GUI before testing)

s1. input 4 to FR0 -> accept

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0]	4.0	Accept
-----	--	-----	--------

s2. input 10 to memory location -> accept

s3. input 200 to memory location's value -> accept

Memory	10	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

s4. input 11 to memory location -> accept

s5. input 200 to memory location's value -> accept

Memory	11	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

s6. input 12 to memory location -> accept

s7. input 200 to memory location's value -> accept

Memory	12	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

s8. input 13 to memory location -> accept

s9. input 200 to memory location's value -> accept

Memory	13	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	200	Accept
--------	----	--------	--	-----	--------

s10. input 14 to memory location -> accept

s11. input 100 to memory location's value -> accept

Memory	14	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s12. input 15 to memory location -> accept

s13. input 100 to memory location's value -> accept

Memory	15	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s14. input 16 to memory location -> accept

s15. input 100 to memory location's value -> accept

Memory	16	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s16. input 17 to memory location -> accept

s17. input 100 to memory location's value -> accept

Memory	17	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0]	100	Accept
--------	----	--------	--	-----	--------

s18. select relevant values for VSUB (IX = 0; GPR = 0; I = 0; Address = 10)

in manual input part -> accept manual word instruction

Manual Instructions	VSUB	No...	GPR0	IA ...	10
	[0, 1, 1, 1, 1, 0]	[0, 0]	[0, 0]	[0]	[0, 1, 0, 1, 0]
Accept Manual Word Instruction					



**[Expected Result] ->**

a.  $M(10) = M(10) - M(14) = 100 (200 - 100)$

Memory	<input type="text" value="10"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	---	----------------------------------	---------------------------------------

b.  $M(11) = M(11) - M(15) = 100 (200 - 100)$

Memory	<input type="text" value="11"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	---	----------------------------------	---------------------------------------

c.  $M(12) = M(12) - M(16) = 100 (200 - 100)$

Memory	<input type="text" value="12"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	---	----------------------------------	---------------------------------------

d.  $M(13) = M(13) - M(17) = 100 (200 - 100)$

Memory	<input type="text" value="13"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	---	----------------------------------	---------------------------------------

## 5. CNVRT

**Case 1 - F = 0**

s1. input 10 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory	<input type="text" value="10"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	---	----------------------------------	---------------------------------------

s3. select relevant values for CNVRT (IX = 0; GPR = 0; I = 0; Address = 10)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="CNVRT"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input type="text" value="[0, 1, 1, 1, 1, 1]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result] ->** FR0 is populated with 100 as a floating point number  
(10<sup>2</sup>)

FR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="100.0"/>
-----	--	------------------------------------

## Case 2 - $F = 1$

s1. input 10 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory	<input type="text" value="10"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s3. select relevant values for CNVRT (IX = 0; GPR = 1; I = 0; Address = 10)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="CNVRT"/>	<input type="button" value="No..."/>	<input type="button" value="GPR1"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input type="text" value="[0, 1, 1, 1, 1, 1]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> FR0 is populated with 100 as a fixed number

FR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>
-----	--

## 6. LDFR

### Case 1 - $fr \leftarrow c(EA)$

s1. input 20 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory	<input type="text" value="20"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s3. select relevant values for LDFR (IX = 0; GPR = 0; I = 0; Address = 20)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="LDFR"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="20"/>
	<input type="text" value="[1, 0, 1, 0, 0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[1, 0, 1, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> FR0 is populated with 100.0 ( $10^2$ )

FR0	<input type="text" value="0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0"/>	<input type="text" value="100.0"/>	<input type="button" value="Accept"/>
-----	--	------------------------------------	---------------------------------------

**Case 2** -  $fr \leftarrow c(c(EA))$ , if I bit set

s1. input 5 to IX3

IX3	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1"/>	<input type="text" value="5"/>	<input type="button" value="Accept"/>
-----	--	--------------------------------	---------------------------------------

s2. input 20 to memory location -> accept

s3. input 100 to memory location's value -> accept

Memory	<input type="text" value="20"/>	<input type="button" value="Accept"/>	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s4. select relevant values for LDFR (**IX = 3**; GPR = 0; I = 1; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="text" value="LDFR"/>	<input type="text" value="IX..."/>	<input type="text" value="GPR0"/>	<input type="text" value="IA ..."/>	<input type="text" value="15"/>
	<input type="text" value="[1, 0, 1, 0, 0, 0]"/>	<input type="text" value="[1, 1]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[1]"/>	<input type="text" value="[0, 1, 1, 1, 1]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> FR0 is populated with 100.0 ( $10^2$ )

FR0	<input type="text" value="0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0"/>	<input type="text" value="100.0"/>	<input type="button" value="Accept"/>
-----	--	------------------------------------	---------------------------------------

## 7. STFR

**Case 1** -  $fr \leftarrow c(EA)$

s1. input 100 to FR0

FR0	<input type="text" value="0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0"/>	<input type="text" value="100.0"/>	<input type="button" value="Accept"/>
-----	--	------------------------------------	---------------------------------------

s2. select relevant values for STFR (IX = 0; GPR = 0; I = 0; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions	STFR	No...	GPR0	IA ...	20
	[1, 0, 1, 0, 0, 1]	[0, 0]	[0, 0]	[0]	[1, 0, 1, 0, 0]
Accept Manual Word Instruction					

**[Expected Result] ->**

Value for memory location 20 is 0,0000010,00001010 ( $10^2$ )

Memory	20	Accept	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0]
--------	----	--------	--

**Case 2 - fr <- c(c(EA)), if I bit set**

s1. input 5 to IX3

IX3	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1]	5	Accept
-----	---	---	--------

s2. input 100 to FR0

FR0	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0]	100.0	Accept
-----	--	-------	--------

s3. select relevant values for STFR (IX = 3; GPR = 0; I = 1; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions	STFR	IX...	GPR0	IA ...	15
	[1, 0, 1, 0, 0, 1]	[1, 1]	[0, 0]	[1]	[0, 1, 1, 1, 1]
Accept Manual Word Instruction					

**[Expected Result] ->**

Value for memory location 20 is 0,0000010,00001010 ( $10^2$ )

Memory	20	Accept	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0]
--------	----	--------	--

## Part 2 - Test Program 1

### Deliverable description:

Demonstration that Program 1 works.

### Demo Steps:

1. Click on “Test Program 1”

**Test Program 1**

2. Randomly enter 21 numbers between 0 and 36767, such as - 10, 20, 13, 5, 1, 2, 6, 9, 11, 34, 15, 7, 16, 18, 30, 100, 36, 19, 22, 50, 101

The closest number input to the 21st number was 100

## Part 3 - Test Program 2

### Deliverable description:

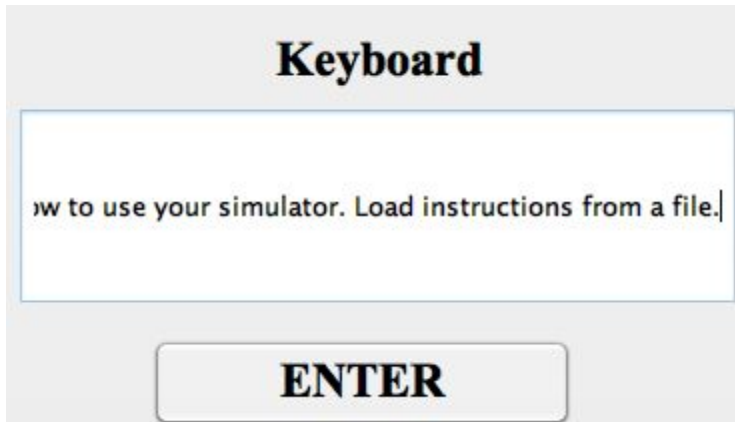
Demonstration that Program 2 works.

### Demo Steps:

1. Click on “Test Program 2”

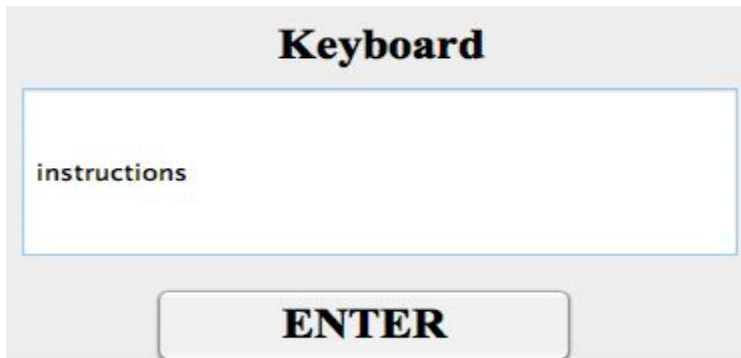
**Test Program 2**

2. (Please use only single-space and single periods) Randomly enter 6 sentences with **period(.)** separated, such as - “**Demonstration that Program 2 works. and how to operate it. Include source code for program 2. what the console layout is and how to operate it. Simple documentation describing how to use your simulator. Load instructions from a file.**” - Click on “ENTER”



The screenshot shows a window titled "Keyboard" with a text input field. The text inside the field is "ow to use your simulator. Load instructions from a file." followed by a cursor. Below the input field is a button labeled "ENTER".

3. Enter the **word** what you intend to search for - such as “**instructions**” - then click on “ENTER”



The screenshot shows a window titled "Keyboard" with a text input field. The text inside the field is "instructions". Below the input field is a button labeled "ENTER".

**[Expected Result]** - Output the first location and sentence number of the searched word in the six sentences. (our program will only find the first one if there are multiple matches to be found. And please reopen the GUI and get it reloaded if the result is not expected)

Match found at word 2 of sentence 6

## Part 4 - Demo for Trap/Machine Fault

### Deliverable description:

#### Machine Fault (Part III)

### Demo Steps (Please reopen the GUI if the result is not expected):

#### 1. TRAP code

s1. input 100 to PC

PC	<input type="text" value="[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept Input"/>
----	---	----------------------------------	---

s2. input 10 (between 0 - 15) to GPR0 as the code

GPR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

s3. select relevant values for TRAP (IX = 0; **GPR = 0**; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="TRAP"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="0"/>
	<input type="text" value="[0, 1, 1, 0, 0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 0, 0, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

### [Expected Result] ->

1. PC = 100 (actually it has changed from 100[current pc] to 20[the address of the user-specified instructions stored elsewhere in memory for trap] to 100[return previous pc after executing routine])

PC	<input type="text" value="[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>
----	---	----------------------------------

2. MFR = 10 (contains the ID code [trap code] if a machine fault after it occurs)

MFR	[1, 0, 1, 0]
-----	--------------

3. MSR = 110 (the routine instruction is defined as 100 + trap code to indicate the status of the health of the machine)

MSR	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0]
-----	--

4. Memory location 0 = 20 (contains the address of the specific executed routine) - Click “Accept” close to Memory [0] to get content (20) visible

Memory	0	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]	20	Accept
--------	---	--------	---	----	--------

## 2. Machine Fault

### 2.1. Illegal Memory Address to Reserved Locations

s1. input 100 to PC

PC	[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept Input
----	--------------------------------------	-----	--------------

s2. select relevant values for LDR (IX = 0; GPR = 0; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions	LDR	No...	GPR0	IA ...	0
	[0, 0, 0, 0, 0, 1]	[0, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
Accept Manual Word Instruction					

**[Expected Result] ->**

1. PC = 100 (actually it has changed from 100[current pc] to 11[the address of the user-specified instructions stored elsewhere in memory for trap] to 100[return previous pc after executing routine])

PC	[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
----	--------------------------------------	-----



2. MFR = 1 (contains the ID code [id for Illegal Memory Address to Reserved Locations] if a machine fault after it occurs)

MFR

3. MSR = 101 (the routine instruction is defined as 100 + ID code to indicate the status of the health of the machine)

MSR

4. Memory location 1 = 11 (contains the address of the specific executed routine) - Click “Accept” close to Memory [1] to get content (11) visible

Memory

## 2.2. Illegal TRAP code

s1. input 100 to PC

PC

s2. input 20 (out of 0 - 15) to GPR0 as the code

GPR0

s3. select relevant values for TRAP (IX = 0; **GPR = 0**; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions

**[Expected Result] ->**

1. PC = 100 (actually it has changed from 100[current pc] to 12[the address of the user-specified instructions stored elsewhere in memory for trap] to 100[return previous pc after executing routine])

PC	[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
----	--------------------------------------	-----

2. MFR = 2 (contains the ID code [id for Illegal TRAP code] if a machine fault after it occurs)

MFR	[0, 0, 1, 0]
-----	--------------

3. MSR = 102 (the routine instruction is defined as 100 + ID code to indicate the status of the health of the machine)

MSR	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0]
-----	--

4. Memory location 1 = 12 (contains the address of the specific executed routine) - Click "Accept" close to Memory [1] to get content (12) visible

Memory	1	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0]	12
--------	---	--------	--	----

## 2.3. Illegal Operation Code

- s1. input 100 to PC

PC	[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
----	--------------------------------------	-----

- s2. select relevant values for Invalid OpCode (IX = 0; GPR = 0; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions	Invalid OpCode	No...	GPR0	IA ...	0
	[1, 1, 0, 1, 1, 1]	[0, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
Accept Manual Word Instruction					

**[Expected Result] ->**

1. PC = 100 (actually it has changed from 100[current pc] to 13[the address of the user-specified instructions stored elsewhere in memory for trap] to 100[return previous pc after executing routine])

PC	[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
----	--------------------------------------	-----

2. MFR = 3 (contains the ID code [id for Illegal Operation Code] if a machine fault after it occurs)

MFR

3. MSR = 103 (the routine instruction is defined as 100 + ID code to indicate the status of the health of the machine)

MSR

4. Memory location 1 = 13 (contains the address of the specific executed routine) - Click "Accept" close to Memory [1] to get content (13) visible

Memory

## 2.4. Illegal Memory Address beyond 2048 (memory installed)

- s1. input 100 to PC

PC

- s2. input 2300 to IX1

IX1

- s3. select relevant values for LDR (IX = 1; GPR = 0; I = 1; Address = 10) in manual input part -> accept manual word instruction

Manual Instructions

### [Expected Result] ->

1. PC = 100 (actually it has changed from 100[current pc] to 14[the address of the user-specified instructions stored elsewhere in memory for trap] to 100[return previous pc after executing routine])

PC	[0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
----	--------------------------------------	-----

2. MFR = 4 (contains the ID code [id for Illegal Memory Address beyond 2048] if a machine fault after it occurs)

MFR	[0, 1, 0, 0]
-----	--------------

3. MSR = 104 (the routine instruction is defined as 100 + ID code to indicate the status of the health of the machine)

MSR	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0]
-----	--

4. Memory location 1 = 14 (contains the address of the specific executed routine) - Click “Accept” close to Memory [1] to get content (14) visible

Memory	1	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0]	14	Accept
--------	---	--------	--	----	--------

## Part 5 - Demonstrate instructions through GUI

(Except the floating point and vector operations)

### Deliverable description:

Load instructions from a file.

### Explanation:

We are not intending to make duplicate load work, so keep having a list of opcode/assembly instructions (as phase 2) that the GUI can “load” from to replace load instructions from a file.

Manual Instructions	HLT	No...	GPR0	IA ...	0
	[0, 0, 0, 0, 0, 0]	[0, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
Accept Manual Word Instruction					

### Demo Steps (Please reopen the GUI if the result is not expected):

1. HLT/RESUME
  - s1. Click on “HALT”

**HALT**

**[Expected Result]** -> ALU stops working, no more instructions will be executed.

Machine is stopping work now...please press down RESUME button if you want to get it back to work!

s2. Click on “RESUME”

**RESUME**

**[Expected Result]** -> ALU gets back to work.

Machine is getting back to work!

## 2. LDR

**Case 1** -  $IX = 0$ ,  $I = 0$  -> EA = address

s1. input 10 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory

10

Accept

[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]

100

Accept

s3. select relevant values for LDR (IX = 0; GPR = 3; I = 0; Address = 10) in manual input part -> accept manual word instruction



Manual Instructions

LDR [dropdown] No... [dropdown] GPR3 [dropdown] IA ... [dropdown] 10 [dropdown]

[0, 0, 0, 0, 0, 1] [0, 0] [1, 1] [0] [0, 1, 0, 1, 0]

Accept Manual Word Instruction

**[Expected Result]** -> GPR3 is populated with 100



GPR3 [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0] 100 Accept

**Case 2** - IX != 0, I = 1 -> EA = address + c(IR)

s1. input 5 to IX1



IX1 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1] 5 Accept

s2. input 20 to memory location -> accept

s3. input 100 to memory location's value -> accept



Memory 20 Accept [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0] 100 Accept

s4. select relevant values for LDR (IX = 1; GPR = 3; I = 1; Address = 15) in manual input part -> accept manual word instruction



Manual Instructions

LDR [dropdown] IX... [dropdown] GPR3 [dropdown] IA ... [dropdown] 15 [dropdown]

[0, 0, 0, 0, 0, 1] [0, 1] [1, 1] [1] [0, 1, 1, 1, 1]

Accept Manual Word Instruction

**[Expected Result]** -> GPR3 is populated with 100



GPR3 [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0] 100 Accept

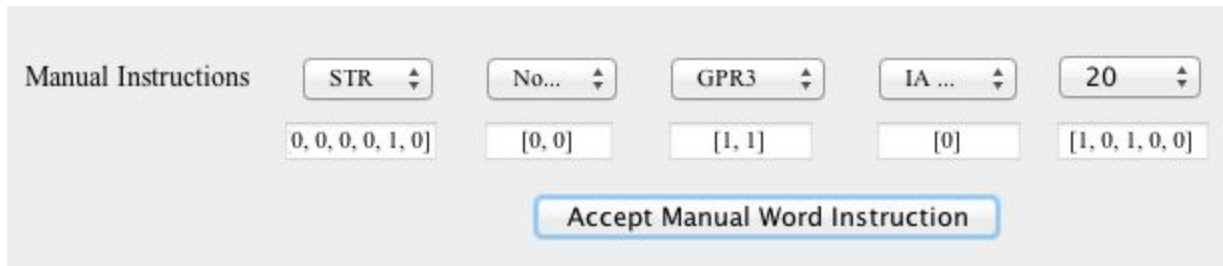
### 3. STR

**Case 1** -  $IX = 0, I = 0 \rightarrow EA = \text{address}$

s1. input 100 to GPR3 -> accept



s2. select relevant values for STR ( $IX = 0$ ;  $GPR = 3$ ;  $I = 0$ ; Address = 20)  
in manual input part -> accept manual word instruction



**[Expected Result]** -> 100 is stored into memory address 20 (please click Accept close to Memory[20] to get content[100] visible)



**Case 2** -  $IX \neq 0, I = 1 \rightarrow EA = \text{address} + c(IX)$

s1. input 5 to IX1



s2. input 100 to GPR3 -> accept



s3. select relevant values for STR ( $IX = 1$ ;  $GPR = 3$ ;  $I = 1$ ; Address = 15)  
in manual input part -> accept manual word instruction



**[Expected Result]** -> 100 is stored into memory address 20 (please click Accept close to Memory[20] to get content[100] visible)

Memory	<input type="text" value="20"/>	<input type="button" value="Accept"/>	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

#### 4. LDA

**Case 1** -  $IX = 0, I = 0 \rightarrow EA = \text{address}$

s1. select relevant values for LDA ( $IX = 0$ ;  $GPR = 3$ ;  $I = 0$ ; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="L..."/>	<input type="button" value="No..."/>	<input type="button" value="GPR3"/>	<input type="button" value="IA ..."/>	<input type="button" value="20"/>
	<input type="text" value="0, 0, 0, 0, 1, 1"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[1, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[1, 0, 1, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> GPR3 is populated with 20

GPR3	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0"/>	<input type="text" value="20"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

**Case 2** -  $IX \neq 0, I = 1 \rightarrow EA = \text{address} + c(IR)$

s1. input 5 to IX1

IX1	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1"/>	<input type="text" value="5"/>	<input type="button" value="Accept"/>
-----	--	--------------------------------	---------------------------------------

s2. select relevant values for LDA ( $IX = 1$ ;  $GPR = 3$ ;  $I = 1$ ; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="L..."/>	<input type="button" value="IX..."/>	<input type="button" value="GPR3"/>	<input type="button" value="IA ..."/>	<input type="button" value="20"/>
	<input type="text" value="0, 0, 0, 0, 1, 1"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[1, 1]"/>	<input type="text" value="[1]"/>	<input type="text" value="[1, 0, 1, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> GPR3 is populated with 25



GPR3	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1"/>	<input type="text" value="25"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

## 5. LDX

(Please note here the Index Register and GPR should be worked in pair,  
IX1 works with GPR1, IX2 works with GPR2 and IX3 works with GPR3)

s1. input 10 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory	<input type="text" value="10"/>	<input type="button" value="Accept"/>	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------	---------------------------------------

s3. select relevant values for LDX (IX = 1; **GPR = 1**; I = 0; Address = 10)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="LDX"/>	<input type="button" value="IX..."/>	<input type="button" value="GPR1"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input type="text" value="[1, 0, 0, 0, 0, 1]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> IX1 is populated with 100

IX1	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>
-----	--

## 6. STX

s1. repeat 5. LDX

IX1	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>
-----	--

s2. select relevant values for STX (IX = 1; GPR = 3; I = 0; Address = 31)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="STX"/>	<input type="button" value="IX..."/>	<input type="button" value="GPR3"/>	<input type="button" value="IA ..."/>	<input type="button" value="31"/>
	<input type="text" value="1, 0, 0, 0, 1, 0]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[1, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[1, 1, 1, 1, 1]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> 100 is stored into memory location 31 (please click Accept close to Memory[31] to get content[100] visible)

Memory	31	Accept	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
--------	----	--------	---	-----	--------

## 7. JZ

**Case 1** - If  $c(r) = 0$ , then  $PC \leftarrow EA(\text{address})$

s1. select relevant values for JZ (IX = 0; GPR = 2; I = 0; Address = 15)  
in manual input part -> accept manual word instruction

GPR2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	0	Accept
------	--	---	--------

Manual Instructions	JZ	No...	GPR2	IA ...	15
	[0, 0, 1, 0, 0, 0]	[0, 0]	[1, 0]	[0]	[0, 1, 1, 1, 1]
Accept Manual Word Instruction					

**[Expected Result]** -> PC is populated with address 15

PC	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]	15
----	--------------------------------------	----

**Case 2** - If  $c(r) \neq 0$ , then  $PC \leftarrow PC + 1$

s1. input 10 to GPR2

GPR2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]	10
------	--	----

s2. PC is populated with 15

PC	[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]	15
----	--------------------------------------	----

s3. select relevant values for JZ (IX = 0; GPR = 2; I = 0; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions

JZ	No...	GPR2	IA ...	15
[0, 0, 1, 0, 0, 0]	[0, 0]	[1, 0]	[0]	[0, 1, 1, 1, 1]

Accept Manual Word Instruction

**[Expected Result]** -> PC is populated with address 16

PC

[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	16
--------------------------------------	----

## 8. JNE

**Case 1** - If  $c(r) \neq 0$ , then  $PC \leftarrow EA(\text{address})$

s1. input 100 to GPR0

GPR0

[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
---	-----	--------

s2. select relevant values for JNE (IX = 0; GPR = 0; I = 0; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions

JNE	No...	GPR0	IA ...	15
[0, 0, 1, 0, 0, 1]	[0, 0]	[0, 0]	[0]	[0, 1, 1, 1, 1]

Accept Manual Word Instruction

**[Expected Result]** -> PC is populated with address 15

PC

[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]	15
--------------------------------------	----

**Case 2** - If  $c(r) \neq 0$ ,  $IX \neq 0$  and  $I = 1$  then  $PC \leftarrow \text{address} + c(IX)$

s1. input 100 to GPR0

GPR0

[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
---	-----	--------

s2. input 5 to IX1

IX1	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1"/>	<input type="text" value="5"/>	<input type="button" value="Accept"/>
-----	--	--------------------------------	---------------------------------------

s3. select relevant values for JNE (IX = 1; GPR = 0; I = 1; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="JNE"/>	<input type="button" value="IX..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="15"/>
	<input type="text" value="0, 0, 1, 0, 0, 1"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[1]"/>	<input type="text" value="[0, 1, 1, 1, 1]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> PC is populated with address 20

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]"/>	<input type="text" value="20"/>
----	---	---------------------------------

**Case 3** - If  $c(r) = 0$ , then  $PC \leftarrow PC + 1$

s1. input 0 to GPR1

GPR1	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0"/>	<input type="text" value="0"/>
------	---	--------------------------------

s2. PC is populated with 20

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]"/>	<input type="text" value="20"/>
----	---	---------------------------------

s3. select relevant values for JNE (IX = 0; GPR = 1; I = 0; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="JNE"/>	<input type="button" value="No..."/>	<input type="button" value="GPR1"/>	<input type="button" value="IA ..."/>	<input type="button" value="15"/>
	<input type="text" value="0, 0, 1, 0, 0, 1"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 1, 1, 1]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> PC is populated with 21

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1]"/>	<input type="text" value="21"/>
----	---	---------------------------------

## 9. JCC

**Case 1** - If cc bit = 1,  $PC \leftarrow EA(\text{address})$

s1. input 15 to Condition Code

CC	<input type="text" value="[1, 1, 1, 1]"/>	<input type="text" value="15"/>	<input type="button" value="Accept Input"/>
----	---	---------------------------------	---

s2. select relevant values for JCC (IX = 0; GPR = 1; I = 0; Address = 15)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="JCC"/>	<input type="button" value="No..."/>	<input type="button" value="GPR1"/>	<input type="button" value="IA ..."/>	<input type="button" value="15"/>
	<input ]"="" type="text" value="0, 0, 1, 0, 1, 0"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 1, 1, 1]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> PC is populated with 15

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]"/>	<input type="text" value="15"/>
----	---	---------------------------------

**Case 2** - If cc bit = 0,  $PC \leftarrow PC + 1$

s1. input 15 to PC

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]"/>	<input type="text" value="15"/>
----	---	---------------------------------

s2. input 0 to Condition Code

CC	<input type="text" value="[0, 0, 0, 0]"/>	<input type="text" value="0"/>
----	---	--------------------------------

s3. select relevant values for JCC (IX = 0; GPR = 1; I = 0; Address = 15)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="JCC"/>	<input type="button" value="No..."/>	<input type="button" value="GPR1"/>	<input type="button" value="IA ..."/>	<input type="button" value="15"/>
	<input ]"="" type="text" value="0, 0, 1, 0, 1, 0"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 1, 1, 1]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> PC is populated with 16

PC	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	16
----	--------------------------------------	----

## 10. JMA

**Case 1** -  $PC \leftarrow EA(\text{address})$ , if I bit not set

s1. select relevant values for JMA (IX = 0; GPR = 1; I = 0; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions	J...	No...	GPR1	IA ...	15
	[0, 0, 1, 0, 1, 1]	[0, 0]	[0, 1]	[0]	[0, 1, 1, 1, 1]
Accept Manual Word Instruction					

**[Expected Result]** -> PC is populated with 15

PC	[0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1]	15
----	--------------------------------------	----

**Case 2** -  $PC \leftarrow \text{Address} + c(\text{IX})$ , if I bit set

s1. input 5 to IX1

IX1	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1]	5	Accept
-----	---	---	--------

s2. select relevant values for JMA (IX = 1; GPR = 1; I = 1; Address = 15)  
in manual input part -> accept manual word instruction

Manual Instructions	J...	IX...	GPR1	IA ...	15
	[0, 0, 1, 0, 1, 1]	[0, 1]	[0, 1]	[1]	[0, 1, 1, 1, 1]
Accept Manual Word Instruction					

**[Expected Result]** -> PC is populated with 20

PC	[0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]	20
----	--------------------------------------	----

11. RFS -  $R0 \leftarrow \text{Immed}; PC \leftarrow c(R3)$

s1. input 20 to GPR3

GPR3	<input ]"="" type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0"/>	<input type="text" value="20"/>	<input type="button" value="Accept"/>
------	---	---------------------------------	---------------------------------------

s2. select relevant values for RFS ( $IX = 0$ ;  $GPR = 3$ ;  $I = 0$ ; Address = 10)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="RFS"/>	<input type="button" value="No..."/>	<input type="button" value="GPR3"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input ]"="" type="text" value="0, 0, 1, 1, 0, 1"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[1, 1]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

[Expected Result] -> GPR0 is populated with 10, PC is populated with 20

GPR0	<input ]"="" type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]"/>	<input type="text" value="20"/>	

12. SOB

**Case 1** -  $r \leftarrow c(r) - 1$  If  $c(r) > 0$ ,  $PC \leftarrow EA(\text{address})$

s1. input 10 to GPR0

GPR0	<input ]"="" type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
------	---	---------------------------------	---------------------------------------

s2. select relevant values for SOB ( $IX = 0$ ;  $GPR = 0$ ;  $I = 0$ ; Address = 10)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="SOB"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input ]"="" type="text" value="0, 0, 1, 1, 1, 0"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

[Expected Result] ->  $GPR0 = 9$ ;  $PC = 10$

GPR0	<input ]"="" type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1"/>	<input type="text" value="9"/>
------	---	--------------------------------

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>
----	--	---------------------------------

**Case 2** -  $r \leftarrow c(r) - 1$  If  $c(r) > 0$ ,  $PC \leftarrow \text{Address} + c(IX)$  if  $IX \neq 0$  and  $I = 1$

s1. input 10 to GPR0

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

s2. input 5 to IX1

IX1	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1]"/>	<input type="text" value="5"/>	<input type="button" value="Accept"/>
-----	--	--------------------------------	---------------------------------------

s3. select relevant values for SOB ( $IX = 1$ ;  $GPR = 0$ ;  $I = 1$ ;  $\text{Address} = 10$ )  
in manual input part -> accept manual word instruction

**[Expected Result]** ->  $GPR0 = 9$ ;  $PC = 15$

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1]"/>	<input type="text" value="9"/>
------	--	--------------------------------

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]"/>	<input type="text" value="15"/>
----	---	---------------------------------

**Case 3** - If  $c(r) = 0$ ,  $PC \leftarrow PC + 1$

s1. input 0 to GPR0

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]"/>	<input type="text" value="0"/>	<input type="button" value="Accept"/>
------	--	--------------------------------	---------------------------------------

s2. input 20 to PC

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]"/>	<input type="text" value="20"/>
----	--	---------------------------------

s3. select relevant values for SOB ( $IX = 0$ ;  $GPR = 0$ ;  $I = 0$ ;  $\text{Address} = 10$ )  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="SOB"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input type="text" value="0, 0, 1, 1, 1, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					



**[Expected Result]** -> PC = 21

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1]"/>	<input type="text" value="21"/>
----	---	---------------------------------

### 13. JGE

**Case 1** - If  $c(r) \geq 0$ , then  $PC \leftarrow EA(\text{address})$  if  $IX = 0$  and  $I = 0$

s1. input 10 to GPR0

GPR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

s2. select relevant values for JGE ( $IX = 0$ ;  $GPR = 0$ ;  $I = 0$ ; Address = 10)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="JGE"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input type="text" value="[0, 0, 1, 1, 1, 1]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> PC = 10

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>
----	--	---------------------------------

**Case 2** - If  $c(r) \geq 0$ ,  $IX \neq 0$  and  $I = 1$ , then  $PC \leftarrow \text{Address} + c(IX)$

s1. input 5 to IX1

IX1	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1]"/>	<input type="text" value="5"/>	<input type="button" value="Accept"/>
-----	--	--------------------------------	---------------------------------------

s2. input 10 to GPR0

GPR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

s3. select relevant values for JGE ( $IX = 1$ ;  $GPR = 0$ ;  $I = 1$ ; Address = 10)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="JGE"/>	<input type="button" value="IX..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="10"/>
	<input type="text" value="[0, 0, 1, 1, 1, 1]"/>	<input type="text" value="[0, 1]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[1]"/>	<input type="text" value="[0, 1, 0, 1, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> PC = 15

PC	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1]"/>	<input type="text" value="15"/>
----	---	---------------------------------

#### 14. AMR

s1. input 20 to memory location -> accept

s2. input 100 to memory location's value -> accept

Memory	<input type="text" value="20"/>	<input type="button" value="Accept"/>	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>
--------	---------------------------------	---------------------------------------	--	----------------------------------

s3. input 100 to GPR0

GPR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>
------	--	----------------------------------

s4. select relevant values for AMR (IX = 0; GPR = 0; I = 0; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="A..."/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="20"/>
	<input type="text" value="[0, 0, 0, 1, 0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[1, 0, 1, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> GPR0 = 200

GPR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="200"/>
------	--	----------------------------------

#### 15. SMR

s1. input 200 to GPR0

GPR0	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="200"/>
------	--	----------------------------------

s2. input 20 to memory location -> accept

s3. input 100 to memory location's value -> accept

Memory	<input type="text" value="20"/>	<input type="button" value="Accept"/>	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>	<input type="text" value="100"/>
--------	---------------------------------	---------------------------------------	---	----------------------------------

s4. select relevant values for SMR (IX = 0; GPR = 0; I = 0; Address = 20)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="S..."/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="text" value="20"/>
	<input type="text" value="0, 0, 0, 1, 0, 1"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[1, 0, 1, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> GPR0 = 100

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>	<input type="text" value="100"/>
------	---	----------------------------------

## 16. AIR

**Case 1** -  $r \leftarrow c(r) + \text{Immed}$ , do nothing if Immed = 0

s1. input 100 to GPR0

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>	<input type="text" value="100"/>
------	---	----------------------------------

s2. select relevant values for AIR (IX = 0; GPR = 0; I = 0; Address = 0)

in manual input part -> accept manual word instruction

Manual Instructions	<input type="button" value="AIR"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="text" value="0"/>
	<input type="text" value="0, 0, 0, 1, 1, 0"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 0, 0, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

**[Expected Result]** -> GPR0 = 100

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0"/>	<input type="text" value="100"/>
------	---	----------------------------------

**Case 2** -  $r \leftarrow c(r) + \text{Immed}$ , loads r with Immed, if  $c(r) = 0$

s1. input 0 to GPR0

GPR0

s2. select relevant values for AIR (IX = 0; GPR = 0; I = 0; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions

<input type="button" value="AIR"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="20"/>
<input type="text" value="0, 0, 0, 1, 1, 0"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[1, 0, 1, 0, 0]"/>

**[Expected Result]** -> GPR0 = 20

GPR0

**Case 3** -  $r \leftarrow c(r) + \text{Immed}$

s1. input 100 to GPR0

GPR0

s2. select relevant values for AIR (IX = 0; GPR = 0; I = 0; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions

<input type="button" value="AIR"/>	<input type="button" value="No..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="20"/>
<input type="text" value="0, 0, 0, 1, 1, 0"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[1, 0, 1, 0, 0]"/>

**[Expected Result]** -> GPR0 = 120

GPR0

## 17. SIR

**Case 1** -  $r \leftarrow c(r) - \text{Immed}$ , do nothing if Immed = 0

s1. input 100 to GPR0

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
------	---	-----

s2. select relevant values for SIR (IX = 0; GPR = 0; I = 0; Address = 0)  
in manual input part -> accept manual word instruction

Manual Instructions	SIR	No...	GPR0	IA ...	0
	0, 0, 0, 1, 1, 1]	[0, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
Accept Manual Word Instruction					

**[Expected Result]** -> GPR0 = 100

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
------	---	-----

**Case 2** -  $r \leftarrow c(r)$  - Immed

s1. input 100 to GPR0

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100
------	---	-----

s2. select relevant values for SIR (IX = 0; GPR = 0; I = 0; Address = 20)  
in manual input part -> accept manual word instruction

Manual Instructions	SIR	No...	GPR0	IA ...	20
	0, 0, 0, 1, 1, 1]	[0, 0]	[0, 0]	[0]	[1, 0, 1, 0, 0]
Accept Manual Word Instruction					

**[Expected Result]** -> GPR0 = 80

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0]	80
------	---	----

18. NOT

s1. input 8 to GPR0

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	8
------	---	---

s2. select relevant values for NOT (IX = 0; GPR = 0; I = 0; Address = 0)  
in manual input part -> accept manual word instruction



Manual Instructions

**[Expected Result]** -> GPR0 = ~8



GPR0

## 19. SRC

(Index Register is used to decide left/right shift, address value is used to decide the bits to shift)

**Case 1** - left shift

s1. input 10 to GPR0



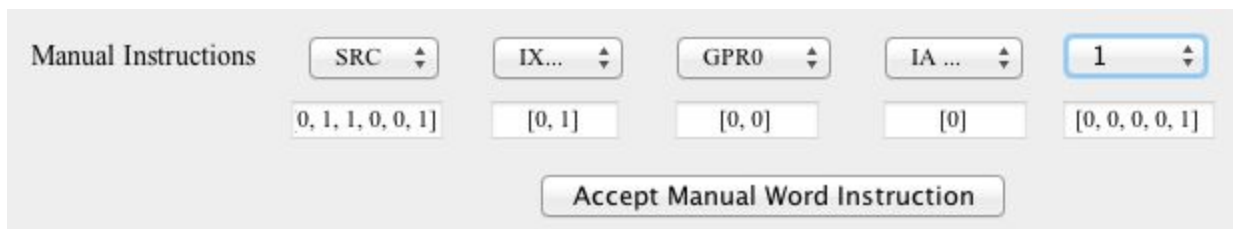
GPR0

s2. input 1 to IX1



IX1

s3. select relevant values for SRC (IX = 1; GPR = 0; I = 0; Address = 1)  
in manual input part -> accept manual word instruction



Manual Instructions

**[Expected Result]** -> GPR0 = 20

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]	20
------	---	----

## Case 2 - right shift

s1. input 20 to GPR0

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0]	20
------	---	----

s2. select relevant values for SRC (IX = 0; GPR = 0; I = 0; Address = 1)  
in manual input part -> accept manual word instruction

Manual Instructions	SRC	No...	GPR0	IA ...	1
	0, 1, 1, 0, 0, 1]	[0, 0]	[0, 0]	[0]	[0, 0, 0, 0, 1]
Accept Manual Word Instruction					

[Expected Result] -> GPR0 = 10

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]	10
------	--	----

## 20. RRC

(Index Register is used to decide left/right shift, address value is used to decide the bits to shift)

### Case 1 - left shift

s1. input 10 to GPR0

GPR0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]	10
------	--	----

s2. input 1 to IX1

IX1	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]	1	Accept
-----	--	---	--------

s3. select relevant values for RRC (IX = 1; GPR = 0; I = 0; Address = 1)  
in manual input part -> accept manual word instruction

Manual Instructions

RRC IX... GPR0 IA ... 1

0, 1, 1, 0, 1, 0 [0, 1] [0, 0] [0] [0, 0, 0, 0, 1]

Accept Manual Word Instruction

**[Expected Result]** -> GPR0 = 20

GPR0 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0 20

**Case 2** - right shift

s1. input 10 to GPR0

GPR0 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0 10

s2. select relevant values for RRC (IX = 0; GPR = 0; I = 0; Address = 1)  
in manual input part -> accept manual word instruction

Manual Instructions

RRC No... GPR0 IA ... 1

0, 1, 1, 0, 1, 0 [0, 0] [0, 0] [0] [0, 0, 0, 0, 1]

Accept Manual Word Instruction

**[Expected Result]** -> GPR0 = 5

GPR0 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0 5

21. MLT

**Case 1** - positive \* positive

s1. input 10 to GPR0

GPR0 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0 10 Accept

s2. input 10 to GPR2



GPR2

s3. input 0 to IX2

IX2

s4. select relevant values for MLT (**IX = 2**[have to use this]; GPR = 0; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions

<input type="button" value="MLT"/>	<input type="button" value="IX..."/>	<input type="button" value="GPR0"/>	<input type="button" value="IA ..."/>	<input type="button" value="0"/>
<input type="text" value="0, 1, 0, 0, 0, 0"/>	<input type="text" value="1, 0"/>	<input type="text" value="0, 0"/>	<input type="text" value="0"/>	<input type="text" value="0, 0, 0, 0, 0"/>

**[Expected Result]** -> GPR1 is populated with 100 (10\*10)

GPR1

**Case 2** - positive \* negative

s1. input 10 to GPR0

GPR0

s2. input -10 to GPR2

GPR2

s3. input 0 to IX2

IX2

s4. select relevant values for MLT (**IX = 2**[have to use this]; GPR = 0; I = 0; Address = 0) in manual input part -> accept manual word

instruction

Manual Instructions	MLT	IX...	GPR0	IA ...	0
	[0, 1, 0, 0, 0, 0]	[1, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
Accept Manual Word Instruction					

**[Expected Result]** -> GPR1 is populated with -100 ( $10 \times -10$ )

GPR1	[1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0]	-100	Accept
------	--	------	--------

**Case 3** - overflow (the result should be less than 65536)

s1. input 32700 to GPR0

GPR0	[0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0]	32700	Accept
------	---	-------	--------

s2. input 2 to GPR2

GPR2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0]	2	Accept
------	--	---	--------

s3. input 0 to IX2

IX2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	0	Accept
-----	--	---	--------

s4. select relevant values for MLT (**IX = 2** [have to use this]; GPR = 0; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions	MLT	IX...	GPR0	IA ...	0
	[0, 1, 0, 0, 0, 0]	[1, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
Accept Manual Word Instruction					

**[Expected Result]** -> CC is populated with 1 (0001)

CC	[0, 0, 0, 1]
----	--------------

### Case 1 - positive / positive

s1. input 100 to GPR0

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
------	--	----------------------------------	---------------------------------------

s2. input 10 to GPR2

GPR2	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

s3. input 0 to IX2

IX2	<input type="text" value="[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]"/>	<input type="text" value="0"/>	<input type="button" value="Accept"/>
-----	---	--------------------------------	---------------------------------------

s4. select relevant values for DVD (**IX = 2**[have to use this]; GPR = 0; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions	<input type="text" value="DVD"/>	<input type="text" value="IX..."/>	<input type="text" value="GPR0"/>	<input type="text" value="IA ..."/>	<input type="text" value="0"/>
	<input type="text" value="[0, 1, 0, 0, 0, 1]"/>	<input type="text" value="[1, 0]"/>	<input type="text" value="[0, 0]"/>	<input type="text" value="[0]"/>	<input type="text" value="[0, 0, 0, 0, 0]"/>
<input type="button" value="Accept Manual Word Instruction"/>					

[Expected Result] -> GPR0 is populated with 10 (100/10)

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0]"/>	<input type="text" value="10"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

### Case 2 - positive / negative

s1. input 100 to GPR0

GPR0	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]"/>	<input type="text" value="100"/>	<input type="button" value="Accept"/>
------	---	----------------------------------	---------------------------------------

s2. input -2 to GPR2

GPR2	<input type="text" value="1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0]"/>	<input type="text" value="-2"/>	<input type="button" value="Accept"/>
------	--	---------------------------------	---------------------------------------

s3. input 0 to IX2

IX2	<input type="text" value="0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]"/>	<input type="text" value="0"/>	<input type="button" value="Accept"/>
-----	--	--------------------------------	---------------------------------------

s4. select relevant values for DVD (**IX = 2**[have to use this]; GPR =

0; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions	DVD	IX...	GPR0	IA ...	0
	[0, 1, 0, 0, 0, 1]	[1, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
<div>Accept Manual Word Instruction</div>					

[Expected Result] -> GPR0 is populated with -50 (100/-2)

GPR0	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0]	-50
------	--	-----

### Case 3 - division by zero

s1. input 100 to GPR0

GPR0	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0]	100	Accept
------	--	-----	--------

s2. input 0 to GPR2

GPR2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	0
------	--	---

s3. input 0 to IX2

IX2	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]	0	Accept
-----	--	---	--------

s4. select relevant values for DVD (**IX = 2**[have to use this]; GPR = 0; I = 0; Address = 0) in manual input part -> accept manual word instruction

Manual Instructions	DVD	IX...	GPR0	IA ...	0
	[0, 1, 0, 0, 0, 1]	[1, 0]	[0, 0]	[0]	[0, 0, 0, 0, 0]
<div>Accept Manual Word Instruction</div>					

[Expected Result] -> CC is populated with 4 (0100)

CC	[0, 1, 0, 0]
----	--------------

## 23. JSR/TRR/AND/ORR/IN/OUT

Our test program 1 and 2 have covered these SIX instructions, so not intended to demonstrate them again through this step.

**End of Document**