MAVinator: A High-Precision Millimeter-Wave 3D Scanner

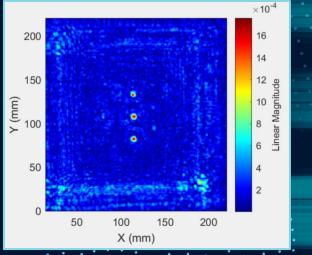


SDMay25-15 Client: Dr. Tayeb | CNDE Nathan Reff, Luke Post, Daniel Ripley-Betts, James Peterson

Background Information

- CNDE (Center for Non-Destructive Evaluation)
 - Millimeter wavelength imaging
 - Detect defects in metals and other materials
 - To be used in industry or governmental applications
 - Ensuring materials meet safety and reliability standards
- Limited millimeter wavelength scanner capacity
 - Pre-existing scanners are slow
 - Current user interfaces are bulky and hard to update





Technical Requirements

- 300 mm x 300 mm x 300 mm volume
 - positional accuracy of 0.5 mm
 - Operate within 2.2-2.3mm wavelength range
- Develop a Python based user interface.
 - Home and align the scanner
 - > Perform automated scans on a uniform cartesian or user-defined grid

3

- Configure scanner parameters
- > Perform data collection from a millimeter-wave device
- Process the data using SAR algorithm and display results

MAVinator Project Overview

Goal:

 Design the physical and digital interfaces for our 3D millimeter wavelength imaging system

Benefits:

- Compact, affordable, professional, and easy to maintain
- Python based, web-hosted intuitive, user interface

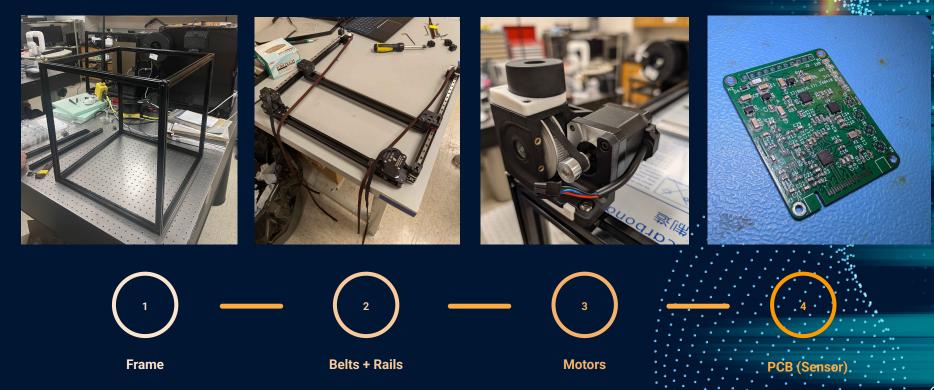




Overview

	Build + Electronics	Compute/Control Boards Sensor Housing	A	Radar + DAQ Setup Limit Switches
02	Frontend	Html, CSS, JS Movement	A	Status Box Libraries
03	Backend	External Connections Radar + DAQ Classes	A A A	G-Code Pattern .scan File System Endpoints
04	Scan, Files, and SAR Implementation	Performing a Scan Saving Files	A A	SAR Algorithm Viewing SAR Data

Build



Electronics



+ BTT Motion Controller

Internal Power Supply (PSU)

+

Automatic movement

ightarrow Connection from the computer to the raspberry pi and given firmware uploaded to BTT

 \rightarrow Successful wiring and utilizing previous User interface for automatic movement



Raspberry Pi





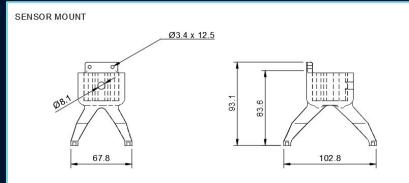
Sensor Housing

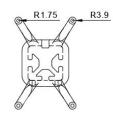
Sensor housing was redesigned to support new radar

Current approach: Modify existing model with minimal changes

- increased compatibility with existing equipment
- time constraints







External Connections

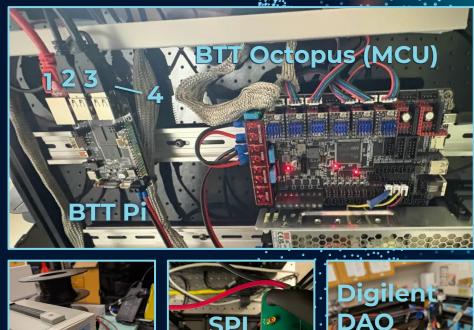
Ethernet (1) - Connecting to the MAVinator and controlling it via web UI.

DAQ (2) - Connected to BTT Pi via USB for configuration and data acquisition from SPI

FTDI (3) - Programing radar's registers, triggers DAQ and radar

Serial (4) - G-code to Marlin firmware on MCU

External PSU - Supplies 5V to Radar







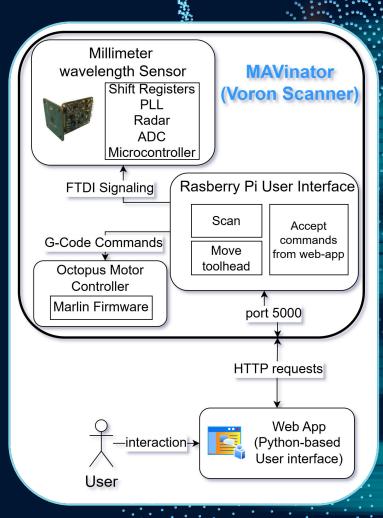


Software Overview

≻ Files

- HTML, CSS, JS,
- Python (Flask)
- > Connection
 - Frontend to Backend
 - Backend to Scanner
- > Pages
 - Move
 - Scan
 - SAR





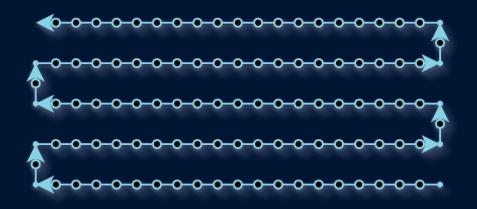
Endpoints

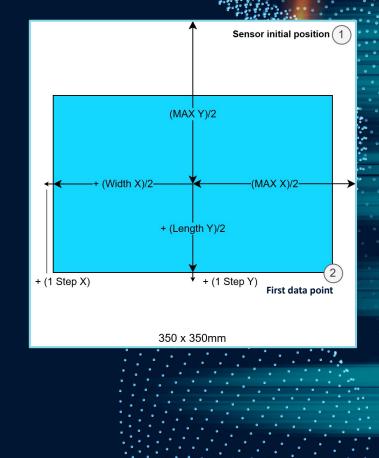
- **Core Role:** The main application file, connecting the web frontend to hardware control, data acquisition, and processing.
- Key Web Endpoints (HTTP):
 - Motion Control: /move, /goTo, /homeOne, /homeAll (POST requests to control sensor position), /emergencyStop(POST).
 - **Status:** /getCoordinates (GET request to retrieve current position).
 - Radar Control: /initRadar (POST to initialize radar hardware/simulation)...
 - **SAR Processing:** /SAR (GET/POST for upload, calculation & display), /plot/sar heatmap slice/<z index>(GET for SAR image slices).
 - Data: /download (GET to download scans).
 - Pages: /move, /SAR, /plot (Serve HTML pages).
- Real-time Communication (SocketIO):
 - Handles start_scan event from frontend.
 - Emits scan_data_point, scan_complete, scan_error during scans.
- Frameworks: Built using Flask and Flask-SocketIO.

G-code Scan Pattern Generation

Designed as an independent script for easy modification.

Based off of the idea objects will be centered on build plate and a given Z-height.

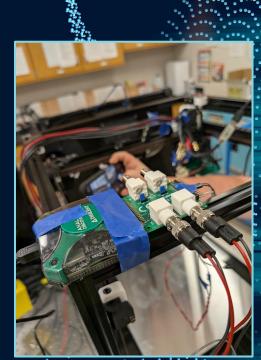




12

Radar and DAQ classes

- **Purpose:** Control custom radar hardware & acquire data.
- Two Main Modules:
 - DigilentDaq.py: High-level classes (Initialize, Measure, Close, Parameters, Simulation mode).
 - radarControl.py: Low-level hardware communication functions (FTDI, Triggering, Data Acquisition).
- **Benefit:** Flexible design supports different DAQ systems & simulation provided you know how to write a class for them.



Scan file handling

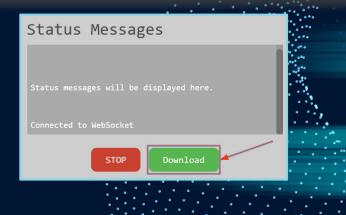
Proprietary .scan format support

Saved as soon as scan finishes

Import and export

Flattens data into point cloud of individual data points

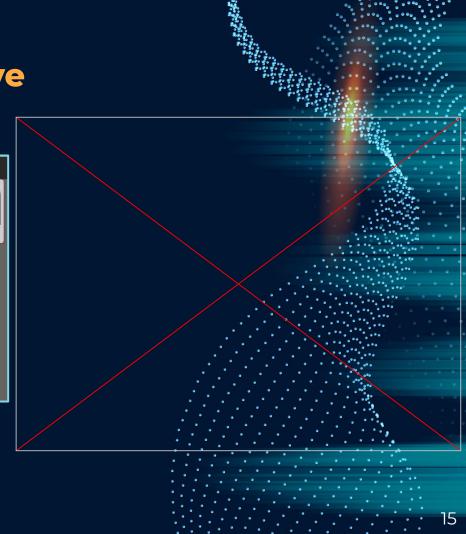
											Page 1						•		• •	5 a
📃 test	t.scan																			
File Ed	dit Vie	w																•		۽ چ
訳N#D쉉 -	+引什ジ究祥	#制宧敲€卝	ナロロノ相相	(4)易謐	は縄₽쀻□⋒≗	高の相戦쇈챌	はっています。 しょう しょう しょう しょう しょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひ	1947年2月	2090)¥	颧野h潊!	臣歙W願	∩ 閉回回り魚	意品の離	◎安る細き	当証ご爾-	~~ 년 표 皷 딧 []「壁蛾念四	擅Sinau	₩甄突	Mone.
■■善邦款	和日本	□₽₩듳愛齘	蒐ू> 野玩	當前開四伏	畝맨•炊∞≀	븷흿。껶븷흿	⊐⊯ौ∧Č畝땐	卝縮編□下報紙	お新音愛前	癍診□⊮∎	₩◎□俳剛	◎善邦轅◎	삙쬅趪ኛ≣	国際部舗	輪橫部蛾	≫@邮只□C	₩~~世務創	길똹?æ	⊧⊃安&â	€ U ⊡
ŋ苦013リ属	家口顧∩確	∧潊臣浅⊉≧續	₩舌殿 ² 01	ff ≄≂i	₹8~ ×4	1つ 州回迷顧醒	W夏∩0品®	カ呀굉똹ブ址	止덧O다垢	i RS≪saranz	29月1211001	非區美發的	们目期期回	u\$P\$(新闻)	붫뵢ĸŏČa	(巴東·朝氏開診(18净0营邦营	受486.交错	18.32U)	E#
壀∋安&囀	胞の動べ	¥臣獻最ª∞	家。蔽回。回	ōo. 蜡含o)	≋₽潊臣缺□□	1リ剀⊚安4曜	~ 医赛癖蛾》	念回塀1甌突は	煽口苦邦啊	匡鍁口⊮刀	LD 鹅呒黄	。敵型物間	∦휫₩@11	月識脯の苦	習餐節化	gulft 要回界	쬅重 ² maan	둪ㅁ다ㅁ	깉똹ゔ	重샎回
ຐឨ៝៝៝៝៝៝៝៝៝៝ឤ៝	儒。迷颧	整命かっ頭 図録		◉醋鱧⊗■	コ蹜賈・床禰	Xiless 烈 B 硬度	퇳읻粍曻미	·鉅兴敏。控计	出境感候	攀口眾鬥	能+0ά黄¢	11100来高	Ă릴~멼셓	☑■鷂縲⑴	kΩ□ 7厦7	蘭≍⇒建飯加	宙竊或響臺	制證鞫>	5 蝮晬排	塘越
■□瑠花ⓒ ■□址읹☞	● 離 軆 役 B 蕎 奀 斎 B 薇	■編 回天叠⊙ 天代■□址厳	舗塀買□床 記∈呀獎?	喫殻α亟割 発偀≤増	● 鉅 _{羽缺惑挟} 土棚嘗彎能自	竣式警훈붸航 ■acc黄(MA월 R 구888 입 성 88 [刘鼓 및 J 810 #	###日朝硼米: 28 7防航788	■什 成 書 書 表 的 当净障碍 今 散日 身	序推察重度战 황칌 4월10년	ioxia,鹬 鹤妖犀友	3妹查對칌 边口妹敬握	「期證範安)・ 創建型個響	設新7月宙 音叀價%	净 8陈 ++保鏍미	쁼셓四機 빩rĂ뮖兩	諮⊡⊡〕斄백 쑽à贲≀娟殠	影問 h 控出機 酵 耕 쭢 鉅 列 :	벯읻뚔(诀졕ː8쨷	英‡怩∈ 創覧 +	呀∺ 赤噀♀
y¥¥≇¥€	\帑覆◎++i	朝間回帰無来ら	馸 瓶y瘫	쎽셮蝎廩심	· 锭瑙苦B全	₩版및 K D 越回	雉첌ø굲ь	温陈裕00族0	◎耀♥眩号	1941日1953	菱遺of部 명	ノビ版印定会	對蚊B*敝!	制扳四輪	⊕ ⁴ स्रक्ष	_躸 딠骥僗贆	乾1月₩+10_∿	o∰tβj	l+0°∦	。婿
<u>気軆平□∞</u>)	1兀四草 俳明	E苚讀@塀@蟠	¥چ⊴و	奕뀠 K 山	:決諚瑙呀ロ#	Āŷ侣献膾□获♯	皆是大熊稚子	≣œ€j帑戴œ∋	现19年1月	쎽셮鷌힜	为8 <u>2</u> 梯喇	间增加增长	ᡂ妹댷벞	100年19月1日 19月1日	囀⊡*敝县	夏⊕碧桃□膂勝	獣家い□策	©尟□₽	その鏡展	桃蘭
현桃顔 :	■敝區型丿	は過ぎた回進が	200天-唯言	*:之椕噀	[셮蝎剀D末(質助党回转横	<u>₹</u> 06758×1	1月鐵加加速車	학瓶 ず俀呆	靛瑙哀望	꼠 K D重著	ья处回避	「坏蓚讀」	a\$\$0更好	設彳28時	砂焙豆が習る	1914 費請課	跍鐡쳐	碱D建e	ホ標驙
博me補maa	七騮款四、	≹‱்若県9	小歌ロマ茶野	涎纖么改能	홍혠섿오 thi	9妗矝∞粟⊞五	첮~≵@v₽氶:	○푄下□ 處	(4708) <u>:</u>	胎房凹凹幕	稻苦′□&設	5 <mark>虮炮</mark> 體趪	鎌□죆郎	・光口院の语	想陈oā	9款 • 并口险旗	□₩☎凱炮牒	É′□&\$6	io⊎#12	蒲口日
尾云7蓮魚	ŧ_ 新永·	⊃ଅ下¦⊧⊠v₽	군도첯~烋;	令い <i>乗</i> 回54	θ萍헨섿♀檎編	₩≾≈□२४₩[┧数∞。若昇	,亲细祝	ビ翩款悳□	io궤울+a	◎檀 辛碱23	遺動標品的	豊 綱쳐る	四天间20月	10, 229	■ 缺涎線么種	에 10 48표상	「雨永⊃	푄쑴갴	7蓮俀
◎♡轁哀虮	U包ssi · *	口姀o谱想姀	鍵□╕罅□≈	藤蒲ロ胎殻	L_ 頤}ev泼	烋稔ᅇᆡ萍혠	^৶ بto ₂ ⊁瑜	ॼ॑若偀四卍篇	離啀を+ネ	思ィの言	站突線霈	點좆잷+#	ŧ₩a卍騮	哐ы若偀	no¥暗冥	현섿城口种	∞ॊoडv没ฮ	陇蒲	胎动)	
編uw路oi	普想妖 🌑	杀口坏腐地嗜重	回じ昭哀日	遠侯海つ	픤쑴넞뚀쳧류	ic tate	列aid缺日,音	區徳⊵06曜짐	目朝的天	回潤額6:	ッ運回O珅	00001=2	∦#理≻句\$	穿と 駅 前例	酸戰組	≧→4成数回来(現線域感到的	回鐘びい	埃錫鐵	€叛□
						臺鴨四世日34.20 □葉日□1月沖運														
Ln 1, Col 1	108,72	na eas 소l 그 entr 24 characters	2T Minh I H	**********	175524	- 46k. AFRI/ 1664 m	- ++ - 328877	一方川2回の茶本3				UR 100 /17 4.L 100	Imm 75211 #	100	алын жыла 96 - 1	Windows (CR		UTF-	16 BE	Des till an

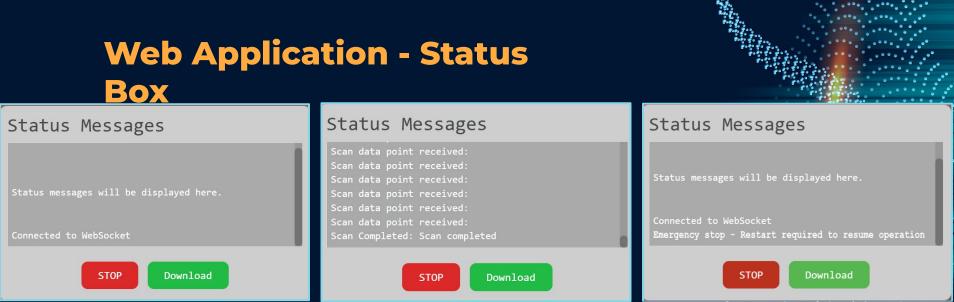


4

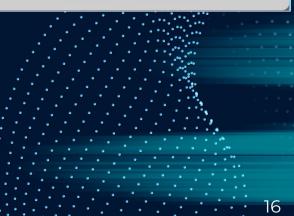
Web Application - Move

MAVinator Move Scen SAR	Iowa State University sdmay25-15
Aovement Controls	Status Messages status micrope stil in displayed here. connected to unidented STOP (Monitoria)
Homing Home All ++ Home X E Home Y P Home Z Go to Position X Y Z Co	

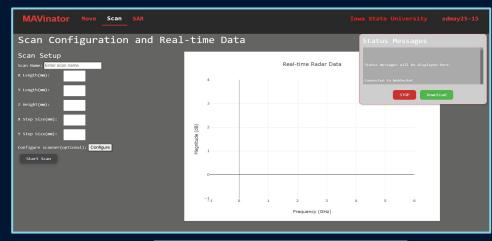


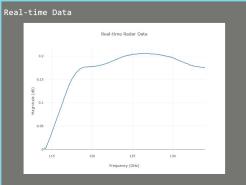


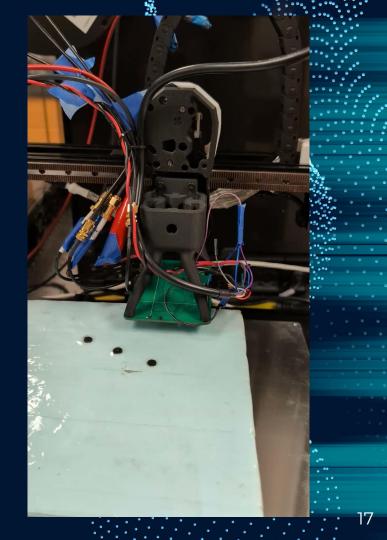
- Uses websockets for real time status updates
- Outputs messages to inform user on scanner operation
- Allows for file saving and emergency stop
- Helps detect errors for debugging
- Persists on all web pages



Web Application - Scan

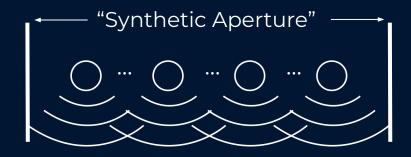






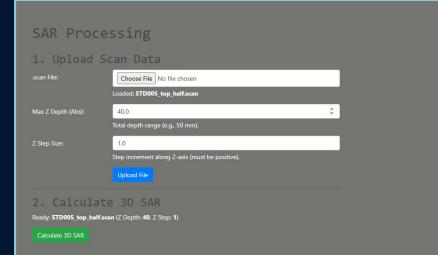
Synthetic Aperture Radar (SAR)

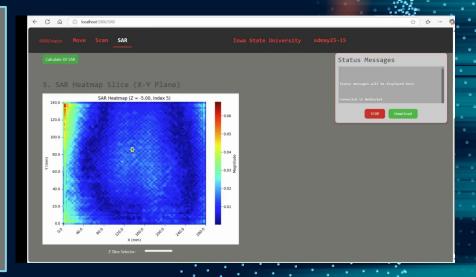
- Angular Resolution is dependent on the wavelength/diameter of the aperture
- Can create an "synthetic aperture" by moving the antenna, taking multiple images, and combining those multiple viewpoints
 - Allows for much better resolution
- Apply processing to turn data into a heatmap



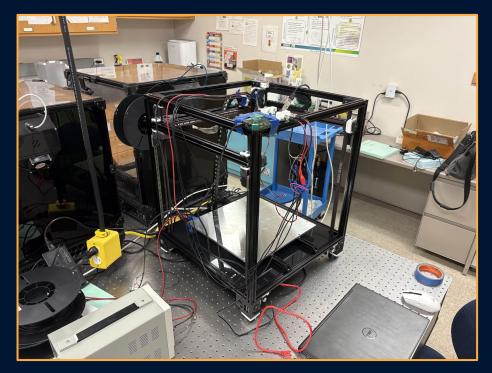


Web Application - SAR

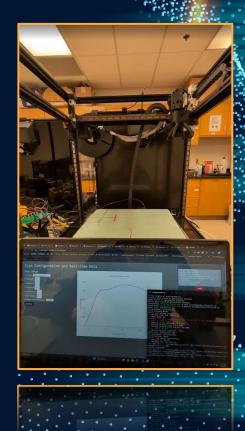




Final Product



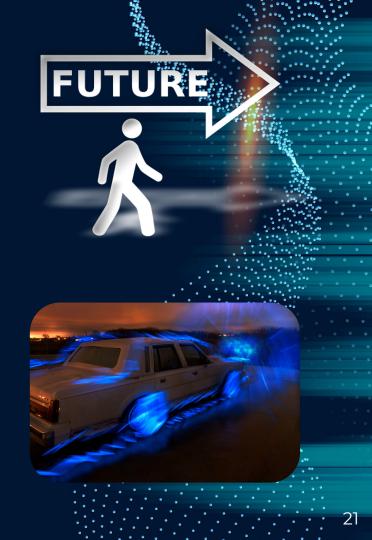




20

Future Work

- \rightarrow Modular classes that inherit from base class
 - DAQ
 - Radar
 - Scan pattern
- \rightarrow Further refining the user interface
- → Updating file management system
- \rightarrow Adding final physical polish
- → Update the BTT Pi access method from IP to URL
- → System testing for exporting data collected by comparing to known good scans of same object
- → Relative scan pattern generation
- → Final testing



Questions?

Appendi X

А



First Semester PCB

Supplied with a PCB and all the components

× 10⁻³

- ➤ Reflow oven
- Hand soldering
- ✤ Testing
 - > Visual
 - > Continuity
 - > Voltage Level
 - > SPI



